# Indoor Localization of Sensors : Application to Dependent Elderly People

Daniel Alshamaa

Thèse
de doctorat
de l'UTT

**Daniel ALSHAMAA**

# Indoor Localization of Sensors: Application to Dependant Elderly People

**Champ disciplinaire :**
Sciences pour l'Ingénieur

**2018TROY0038** **Année 2018**

utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

# THESE

*pour l'obtention du grade de*

# DOCTEUR
# de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

# en SCIENCES POUR L'INGENIEUR

**Spécialité : OPTIMISATION ET SURETE DES SYSTEMES**

*présentée et soutenue par*

**Daniel ALSHAMAA**

*le 13 novembre 2018*

## Indoor Localization of Sensors: Application to Dependent Elderly People

## JURY

| | | |
|---|---|---|
| M. O. BERDER | PROFESSEUR DES UNIVERSITES | Président |
| Mme V. CHERFAOUI | PROFESSEURE DES UNIVERSITES | Rapporteure |
| M. F. ABDALLAH | PROFESSEUR | Examinateur |
| M. F. VALOIS | PROFESSEUR DES UNIVERSITES | Examinateur |
| Mme F. CHEHADE | MAITRE DE CONFERENCES - HDR | Directrice de thèse |
| M. P. HONEINE | PROFESSEUR DES UNIVERSITES | Directeur de thèse |

# *Acknowledgments*

My advisor **Farah Chehade** does not deserve only my thanks, but also direct credit for some of the work this thesis represents. In fact, much of the intellectual work of this thesis took place in conversations with her. Farah has not just been a great advisor, but a great friend as well. I am very grateful for her patience, motivation, and immense knowledge in the field, which made this thesis as successful as it is. I would also like to thank my second advisor **Paul Honeine**, who has always been guiding me in the right direction. His excellent research skills are one of a kind, allowing me to work on a high professional level. I simply could not wish for better advisors.

I would like to thank the reviewers, **Veronique Cherfaoui** and **Olivier Berder**, and the committee members, **Fahed Abdallah** and **Fabrice Valois**, for their careful reading of my thesis and the comments they addressed me during the defense in order to improve my work.

A special thanks to **Aly Chkeir** and the Living Lab of the University of Technology of Troyes, for allowing me to realize the experiments and providing me with the necessary information. This extends to my colleagues in the M2S laboratory who were kind and allowed me to acquire data in their offices.

I also thank the doctoral school of the University of Technology of Troyes, in particular the director **Khemais Saanouni**, and the secretaries **Pascale Denis**, **Isabelle Leclercq** and **Thérèse Kazarian**. My gratitude extends to **Bernadette André** and **Véronique Banse** for their availability and all the help they provided me with.

Finally, I would like to take the opportunity to thank my friends and colleagues for their support during the last three years.

# *Abstract*

This thesis deals with the problem of localization of dependent elderly people using wireless sensor networks. Each person is equipped with a bracelet or a medallion that consists of a sensor capable of measuring the WiFi signals and communicating with the environment. We tackle the problem of localization by zoning, where the objective is to determine the zone where the person resides, instead of its exact position. It is formulated as a problem of multi-class classification, which we treat by associating the flexibility of statistical learning methods to the theory of Dempster-Shafer for fusion of information with uncertainty. The interest of this theory is in its ability to model at the same time the uncertainty and ambiguity of data, and the reliability and conflict of the sources. First, we propose an observation model exploiting the power of the exchanged signals between the sensors with the fingerprinting technique. Afterwards, we extend the proposed model, through hierarchical clustering, in order to cover larger surface areas with higher number of zones. We then develop a method for selecting the WiFi terminals in favor. This method helps enhancing the overall accuracy and reducing the complexity of the localization method, by selecting the best subset in terms of discriminative capacity and redundancy. We also propose mobility models, which we combine with the observation model, in order to correct the estimations by aggregating all available evidence. In addition, we present a decentralized approach of the localization method, to increase the robustness and reduce the complexity. The performance of the proposed methods is validated through experiments on real data, and evaluated in comparison with other well-known methods in the domain.

**Keywords :**

- Sensor networks
- Location-based services
- Dempster-Shafer theory
- Signal processing
- Multisensor data fusion

# Résumé

Cette thèse porte sur le problème de localisation des personnes âgées dépendantes à l'aide de réseaux de capteurs sans fil. Chaque personne est équipée d'un bracelet ou médaillon, comprenant un capteur intelligent capable de mesurer les signaux WiFi et de communiquer avec l'environnement. Nous abordons le problème de localisation par zonage, où le but est de déterminer la zone où se trouve la personne, au lieu de sa position exacte. Il s'agit d'un problème de classification multi-classes, que nous traitons en associant la flexibilité des méthodes d'apprentissage statistique à la théorie de Dempster-Shafer pour la fusion de l'information avec incertitude. L'intérêt de l'utilisation de cette théorie réside dans sa capacité à modéliser à la fois l'incertitude et l'ambiguïté des données, ainsi que la fiabilité et le conflit des sources. Nous proposons en premier un modèle d'observation exploitant la puissance des signaux échangés entre les capteurs avec la technique de fingerprinting. Nous développons ensuite le modèle proposé, au travers du regroupement hiérarchique, dans le but de couvrir des surfaces plus grandes avec un nombre de zones plus élevé. Nous élaborons ensuite une méthode de sélection de paramètres, qui sont les bornes WiFi à privilégier. Cette méthode permet d'améliorer la précision et de réduire la complexité de la méthode de localisation, en choisissant le meilleur ensemble de paramètres en termes de la capacité de discrimination et de la redondance. Nous proposons également des modèles de mobilité que nous combinons au modèle d'observation, afin de corriger les estimations en fusionnant toutes les preuves disponibles. D'autre part, nous présentons une approche décentralisée de la méthode de localisation, pour accroître la robustesse et diminuer la complexité. Les performances des méthodes proposées sont validées par des expérimentations sur des données réelles, et évaluées en comparaison avec d'autres méthodes très connues dans le domaine.

**Mots-clés :**

- Réseaux de capteurs (technologie)
- Services basés sur la localisation
- Théorie de Dempster-Shafer
- Traitement du signal
- Fusion multicapteurs

# Contents

# List of Figures

## LIST OF FIGURES

# List of Tables

## LIST OF TABLES

# Chapter 1

# Introduction

## Contents

*Wireless Sensor Networks (WSNs) have attracted intensive interest from both academia and industry due to their wide applications. These networks consist of spatially distributed autonomous devices that use sensors to monitor a certain area. They are tremendously being used in various fields to perform several tasks ranging from medical and military applications, to monitoring homes, hospitals and forests. This chapter introduces first the wireless sensor networks, their topologies, and their applications. Next, the localization and tracking problems in these networks are described, discussing some of the state-of-the-art techniques. Finally, the organization of the manuscript is outlined.*

## 1.1 Wireless Sensor Networks

With the recent technological advances in wireless communications, processor, memory, radio, low power, highly integrated digital electronics, and micro electro-mechanical systems, it has become possible to significantly develop tiny and small size, low power, and low cost multi-functional sensor nodes. A wireless sensor network (WSN) is a network that is made of tens to thousands of these sensor nodes that are densely deployed in an unattended environment with the capabilities of sensing, wireless communications and computations [Akyildiz *et al.*, 2002]. As a result, a WSN is the combination of sensor techniques, embedded techniques, distributed information processing, and communication mechanisms [Zhou *et al.*, 2015]. Several types of WSNs exist and they are used for a wide range of applications, making them an interesting solution for various problems. Functionally, smart sensor nodes are low power devices equipped with one or more sensors, a processor, memory, power supply, and a radio interface. A key role is played by the environment in determining the size, the topology, and the deployment scheme of the WSN network. In indoor environments for example, fewer nodes are required to construct a network in a limited space whereas in outdoor environments, more nodes are needed to cover a larger area. In this section, we list the types of the WSNs, an overview on their topologies, and a survey of some of their applications.

### 1.1.1 Types

Different types of WSNs can be deployed depending on the nature of the environment. These types include terrestrial, underground, underwater, multimedia, and mobile WSNs, presented in Figure 1.1.

- **Terrestrial WSNs:** In a terrestrial WSN, reliable communication in a dense environment is a must. Sensor nodes must be able to effectively communicate data back to the basestation. While battery power is limited and may not be rechargeable, terrestrial sensor nodes however can be equipped with a secondary power source such as solar cells, it is important for sensor nodes to conserve energy [Hancke, 2012].
- **Underground WSNs:** This type consists of a number of sensor nodes deployed in caves, mines or underground. To enable information retrieval from the underground sensor nodes to the base station, additional sink nodes are located above ground [Akyildiz & Stuntebeck, 2006]. Wireless communication is a challenge in such environment due to high attenuation and signal loss. Moreover, it is difficult to recharge or replace the batteries buried underground, motivating the design of energy efficient communication protocols for prolonged lifetime. Underground WSNs are used for agriculture monitoring and landscape management [Stuntebeck *et al.*, 2006].
- **Underwater WSNs:** This type consists of sensors deployed underwater

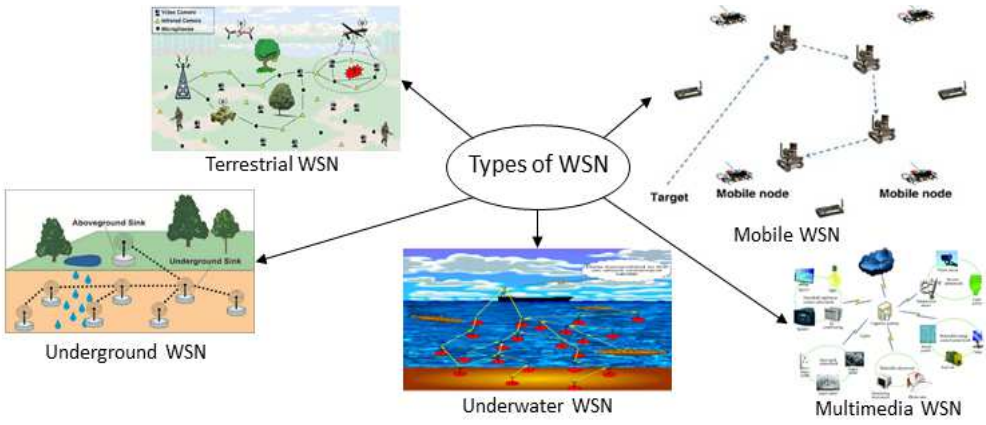



**Figure 1.1:** The five main types of wireless sensor networks.

such as ocean environments. Only a few nodes are deployed due to their high cost, and autonomous underwater vehicles are used to explore or gather data from them. Such communication uses acoustic waves, presenting various challenges such as limited bandwidth, long propagation delay, high latency, and signal fading problems [Murad *et al.*, 2015]. The nodes must be able to adapt to extreme conditions, and are equipped with limited batteries that cannot be replaced or recharged. Applications of underwater WSNs include pollution monitoring and under-sea surveillance.

- **Multimedia WSNs:** Multimedia WSNs have been proposed in a variety of areas including digital signal processing, communication, networking and control systems. [Misra *et al.*, 2008]. The deployed sensor nodes communicate with each other via a wireless connection for data retrieval, processing, correlation, and compression. They face various challenges such as high-energy consumption, high bandwidth demand, data processing, and compressing techniques. It is required to develop transmission techniques that support high bandwidth and low energy consumption in order to deliver multimedia content such as a video stream [Akyildiz *et al.*, 2007].

- **Mobile WSNs:** This type consists of mobile sensor nodes that can move on their own and interact with the physical environment. The mobile nodes have the ability to reposition and organize themselves in the network. A mobile WSN can start off with some initial deployment and nodes can then spread out to gather information. Information gathered by a mobile node can be communicated to another mobile node when they are within range of each other. Mobile WSNs introduce significant challenges such as reliable data transfer, mobility management, localization with mobility, contact detection, minimizing energy consumption, and maintaining network connectivity [Di Francesco *et al.*, 2011]. Mobile WSN applications include environment monitoring and target tracking. A higher degree of coverage and connectivity can be achieved with mobile sensor nodes as compared to static nodes [Amundson & Koutsoukos, 2009].

### 1.1.2 Topologies

Various resource constraints of WSNs depend on the network topology [Chen *et al.*, 2011]. The amount of communication required by the sensors to exchange information is reduced in an efficient topology, thus saving energy. A topology based on minimizing the distance between neighbor nodes for instance, reduces the probability of losing a message during communication. Moreover, a well-designed topology can also reduce radio interference and facilitate data aggregation, thus reducing the amount of processing cycles and elongating the network lifetime [Velmani & Kaarthick, 2015]. Three main topologies have been proposed in literature [Cota-Ruiz *et al.*, 2016; Üney *et al.*, 2016; Yan *et al.*, 2017]:

- **Centralized topology:** In the centralized topology, sensors acquire data measurements and transmit them to the fusion center for processing [Talebi & Hemmatyar, 2014]. In such topology, the sensors are not required to carry out complex computations. Although it can achieve high quality processing, the centralized topology results in unnecessary energy costs due to the transmission of all measurements even if many are not needed [Mamun, 2012].

- **Distributed topology:** The distributed topology treats equally all the sensors. In such topology, the sensors perform computations and exchange data with their neighboring sensors, located within their communication range [Wang, 2008]. Since information processing is no longer limited to a single fusion center, the network is more robust to failures. However, developing relevant distributed algorithms remains a challenging issue.

- **Decentralized topology:** A combination between the two preceding topologies, taking advantages from both of them, is the decentralized topology, also called the clusterized topology, where the sensors are partitioned into several sectors or clusters, each having its own fusion center [Mahfouz *et al.*, 2013]. Information is exchanged between the sensors of each sector and transmitted to the local fusion center. The outputs of all fusion centers are combined to yield a final decision. Such topology increases the scalability of the network, and reduces the energy consumption leading to a

prolonged network lifetime [Üney *et al.*, 2016].

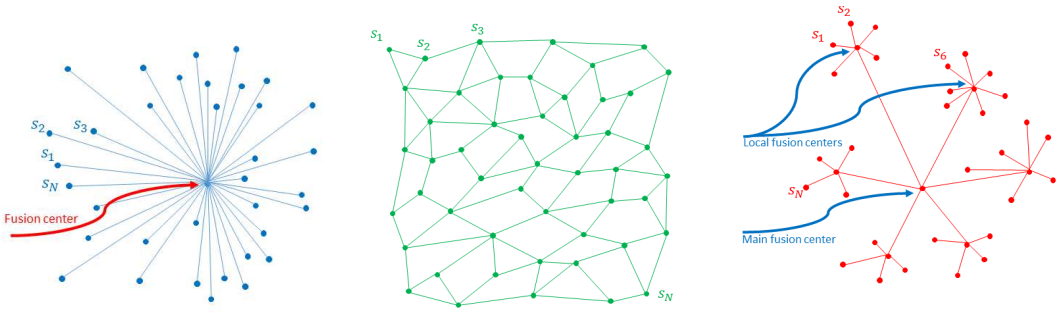
The general structure of the three topologies, considering a set of sensors $\{s_1, \ldots, s_N\}$, is shown in Figure 1.2.

### 1.1.3 Applications

The number of potential applications for wireless sensor networks is huge. We mention here some of the applications in the military, environmental, industrial, and health-care domains.

- **Military applications:** Several areas of research are encompassed in the use of WSNs in military applications. Acoustic detection and recognition has been under research since the early fifties. An analysis of the complex near-field pressure waves that occur within a foot of the blast is possible. Another area of research is the signal processing of gunfire acoustics. The focus is on the robust detection and length estimation of small caliber acoustic shockwaves and blasts.[Hussain & kyung Sup, 2009; Lee *et al.*, 2009].

- **Environmental applications:** Some environmental applications of wireless sensor networks comprise agriculture, farming, mining, seismology, climatology, volcanology, wildlife surveillance, and many others. WSNs facilitate the study of fundamental processes and the development of hazard response systems. Indoor monitoring applications typically include build-




(a) Centralized topology. (b) Distributed topology. (c) Decentralized topology.

**Figure 1.2:** The three main topologies in wireless sensor networks.

ings and offices monitoring. These applications involve sensing temperature, light, humidity, and air quality. Other important applications may include detection of fire and civil structures. Outdoor monitoring applications include chemical hazardous detection, habitat monitoring, traffic monitoring, earthquake detection, volcano eruption, flooding detection and weather forecasting [Corke *et al.*, 2010; Oliveira & Rodrigues, 2011].

- **Home applications:** As technology advances, smart sensor nodes and actuators can be incorporated into appliances, such as vacuum cleaners, micro-wave ovens, and refrigerators. These sensor nodes inside the devices can communicate with each other and with the external network via the Internet. They allow end users to control home devices locally and remotely in an easier way and can be used as alarms for accidents at homes [Hussain *et al.*, 2009; Robles & Kim, 2010].

- **Industrial applications:** The application of WSN technology to the design of field-area networks for industrial communication and control systems has the potential to provide major benefits in terms of flexible installation and maintenance of field devices, support for monitoring the operations of mobile robots, and reduction in costs and problems due to wire cabling. Of these applications, we mention energy evaluation and condition monitoring, motor condition monitoring, and low-cost measurements in the oil and gas industry [Lin *et al.*, 2008; Neuzil *et al.*, 2014; Zhuang *et al.*, 2007].

- **Health-care applications:** The medical applications of WSNs aim to improve the existing healthcare and monitoring services especially for the elderly, children and chronically ill. Numerous benefits are achieved with these systems. Some of the applications involve providing interfaces for the disabled, integrated patient monitoring, diagnosis, drug administration in hospitals, telemonitoring of human physiological data, and tracking and monitoring doctors and patients inside a hospital or a nursing home [Alemdar & Ersoy, 2010; Ko *et al.*, 2010].

### 1.1.4 WSNs for dependent elderly people

A key aspect when developing a new system is to find out the determinants of its acceptance upon the target group. This is especially important for systems designed for the elderly people, as most of them are not accustomed to modern technology. Hence, any biases they may hold towards a technology are likely to hinder its acceptance and adoption. For that reason, Steele *et al.* [2009] present an exploratory study carrying out qualitative research into the perceptions, attitudes and concerns of elderly people toward WSN technologies in terms of their application to health-care. They have identified several concepts, such as independence, privacy, and cost, which have been further classified into themes describing the determinants that may affect an elderly person's acceptance of WSNs for assisting health-care. They have found that the participants' attitudes towards the idea of WSNs for health monitoring are generally positive. Here, we briefly explore the capabilities of WSNs to help elderly people.

- **Monitoring of physiological data:** The physiological data such as intra-body temperature, heartbeat rate, and arterial blood pressure, collected by sensor networks may be stored for a long period of time, and can be used for medical investigations when needed [Mukhopadhyay, 2015].

- **Drug administration in hospitals:** WSNs can help minimize the side effects of drugs by attaching sensor nodes to medication, thus identifying possible allergies and prescribing adequate medication [Gara *et al.*, 2015].

- **Tracking and Monitoring:** Each elderly person is equipped with a small sensor in the form of a bracelet or a watch. The sensor runs a localization algorithm to determine its location using measured data. Tsirmpas *et al.* [2015] propose a navigation system considering scenarios that the user might follow and suggest guidelines for elderly people in an unknown built environment. The installed sensors may have more than one function, and thus can also monitor and detect the behavior of the elderly people. A health smart home was designed in Grenoble, France, for instance, to study the feasibility of such systems [Noury *et al.*, 2000].

## 1.2 Position-based Localization

The localization of sensors from measurements collected by the nodes themselves is one of the important topics when dealing with WSNs.

### 1.2.1 Problem description

Localization is an essential aspect in WSNs, since the knowledge of the sensor's location is critical to process the information originating from this sensor. Many existing works have been proposed to tackle the position-based localization problem. The objective is to determine the position of the sensor node according to some measured observations. A solution is to integrate a Global Positioning System or Global System for Mobile Communications (GPS-GSM) into sensor nodes. This is widely used in vehicle tracking systems [Xiong *et al.*, 2018]. However, it is not always the optimal solution because of the costs of having a GPS receiver at each node, especially when multiple objects are to be localized, as well as for the limited spatial resolution. Moreover, this technology can not be efficiently used for indoor applications due to the large attenuation caused by buildings' walls and ceilings. Its robustness against interference is also questionable [Oshin *et al.*, 2012].

For that reason, alternative solutions have been proposed. We consider the following approach. At first, two types of sensors are defined; anchor nodes (ANs), also called beacon nodes, of known positions, and non-anchor nodes of unknown position, to be localized. Since we consider, in the general setting, the case of moving sensors, we will refer to the non-anchor nodes by mobile nodes (MNs). The objective becomes to determine the position of any MN using collected measurements and information exchanged with the ANs. The remaining issue is to choose the appropriate enabling technology and the measurement technique. This is extremely important as it plays a vital role in the accuracy of the localization algorithm. The enabling technologies and techniques are discussed up next.

### 1.2.2 Enabling technologies

To extend the capability of mobile localization applications in indoor environments, researchers have been using alternative technologies to GPS such as vision, infrared, ultrasound, Ultra-wideband, Bluetooth, WiFi, etc. A brief description of these enabling technologies is presented in the following:

- **Vision:** This technology is based on the processing and evaluation of video data. The video-based localization can be performed in two different ways:
  - *Fixed camera systems:* The ANs in the environment are equipped with cameras, and the objective is to locate the MN using images captured by those cameras. Features of the object carrying the MN to be localized are extracted. Once the features appear in the field of view of the camera, its location is estimated based on its position within the captured image, and the spatial distribution of its salient features [Kim *et al.*, 2001].
  - *Mobile camera systems:* In such systems, either the MN is directly equipped with a camera, or a drone with an equipped camera is used to monitor its movement. The localization is performed by extracting environment features, and involves two phases; an offline phase where images of the environment are captured at predefined locations and are processed to extract unique features, and an online phase where the features of the new captured image are matched to the stored ones, to estimate the location of the camera, and thus the position of the MN [Ravi *et al.*, 2006].

  Nowadays, the increase in data transmission rate and computational capabilities, as well as the development of high performance image processing algorithms make this technology more efficient. Nevertheless, a major drawback of this technology is its high cost.

- **Infrared:** This technology uses the infrared radiation to localize sensors through infrared emitters and receivers. The MN is equipped with a badge,

carrying a unique identifier code (ID), that emits infrared signals at regular intervals via an infrared transmitter. Infrared receivers, placed at the ANs, detect the ID and communicate it to a localization software to determine the MN position based on the proximity between the transmitter and the receiver. The infrared technology is characterized by the absence of radio electromagnetic interference and the power of transmitted signal can be easily adjusted to cover only the area of interest. However, multipath errors, expensive system hardware and maintenance costs, and the necessity of a line-of-sight (LoS) for proper functioning are all drawbacks of this technology [Hauschildt & Kirchhof, 2010].

- **Ultrasound:** This technology uses the ultrasonic waves to measure the distance between the ANs and the MN. The transmitter sends a radio signal and an ultrasonic wave at the same time. The radio signal reaches the multiple receivers almost instantaneously, providing them with the synchronization signal. The receivers then measure the time between the synchronization signal and the detection of ultrasonic waves to calculate the distance between the emitters and receivers. The advantages of this technology are the relative low cost and the capability to reflect most of the indoor obstructions. However, disadvantages arise from the multipath reception that might disturb measurements of the distance between emitter and receivers, and the complexity of a large-scale implementation. In addition, the temperature has a significant influence on the sound speed, which affects the accuracy of the localization [Moreno *et al.*, 2002].

- **Ultra-wideband:** This technology is defined as a transmission from an antenna for which the emitted signal bandwidth exceeds the lesser of 500 MHz. Unlike other radio systems operating on specific radio frequency, Ultra-wideband (UWB) transmits a signal over an ultra-wide band of frequencies. The signals are transmitted for a much shorter duration with very low-power spectral density, thus consuming less power than the other systems. UWB can be used in close proximity to other radio frequency signals without suffering from interference. This technology is convenient for indoor environments where the UWB signal can be easily transmitted,

achieving interesting results in indoor localization applications [Shi & Ming, 2016]. However, A major drawback of this technology is the high cost of the UWB equipment.

- **Bluetooth:** This technology is a standard for Wireless Personal Area Networks (WPANs) and operates in the 2.4 GHz band. Bluetooth has a short range and is embedded in most devices such as mobile phones, laptops, desktop, etc. For that reason, adding a new user to such network does not require any additional hardware. Bluetooth is a low-cost technology, and its tags are small in size, making it an efficient technology for indoor localization. However, one of its drawbacks is that it runs the device discovery procedure at each location estimation, significantly increasing the localization latency and power consumption. This latency is unsuitable for real-time localization applications [Zuo *et al.*, 2018].

- **WiFi:** This technology uses the Wireless Local Area Network (WLAN) to estimate the location of any MN within this network. Since WLAN infrastructures are widespread in almost all indoor environments, due to the increase in demand for wireless communications, this approach is widely used for indoor localization. One of the main advantages of using WiFi over other technologies is its cost effectiveness due to the possibility to localize the position of almost every WiFi compatible device without installing any additional software. Another advantage of using WLAN is that no LoS is required. Nevertheless, it was found that WiFi signal strengths are unstable and vary widely even at the same position with time, temperature, and moving objects. Another limitation of the WiFi technology is the signal attenuation of the static environment like walls, doors, and furniture [Zou *et al.*, 2017].

### 1.2.3 Position-based localization techniques

Generally, localization techniques are split into two categories, geometrical and non-geometrical techniques.

- **Geometrical techniques:** In these techniques, the position of the MN is estimated by compiling one or more channel characteristics, such as Angle of Arrival (AoA), Time Difference of Arrival (TDoA), Time of Arrival (ToA), or Received Signal Strength Indicator (RSSI) into a geometric output. Equations relating the unknown position of the MN with the known positions of the ANs are derived and solved to estimate the MN position. Optimization routines such as the least squares algorithm are often used as a metric to minimize the estimation error.

  - *Time of arrival (ToA):* The ToA method requires a perfect synchronization between the MN and the ANs, and a large bandwidth to obtain a time delay resolution small enough for the desired localization accuracy. Also, the algorithm implicitly requires a LoS for proper functioning. This, however, is not always possible in indoor environments where the MN and the ANs are in Non-line-of-sight (NLoS). The NLoS condition adds a positive delay bias to the real ToA and can introduce severe localization errors if it is not corrected or removed with specific treatments [Gezici *et al.*, 2005].

  - *Time difference of arrival (TDoA):* The TDoA techniques rely on the time difference of arrival for signals arriving between all possible receivers. Its main advantage is that the synchronization between the transmitters and receivers is not required. Nevertheless, such techniques do not perform well in NLoS scenarios [Okello *et al.*, 2011].

  - *Angle of arrival (AoA):* The AoA techniques use the triangulation approach and require at least two ANs. Each one of them must be equipped with an antenna array to extract AoA of the impinging signals by an estimation algorithm. Information on the AoAs is then sent through the network. There is no need for clock synchronization but the technique suffers from a low accuracy due to the performance of the angular estimator. Also, the triangulation technique fails in NLoS scenarios [Rong & Sichitiu, 2006].

- *Received Signal Strength Indicator (RSSI):* The RSSI-based techniques exploit the attenuation of the signal strength with the traveled distance to estimate the distances separating the ANs from the MNs. Typically, ANs broadcast signals in the network, while MNs detect the broadcasted signals and measure their RSSIs. The distances separating the MNs from the ANs are then estimated using the measured RSSIs and the path-loss model [Patwari *et al.*, 2005; Zanella & Bardella, 2014]. RSSI-based techniques exhibit favorable properties with respect to power consumption, size, and cost, since no additional hardware is needed. However, distance estimation using RSSI is really challenging, since the measurements of signals' powers can be significantly altered by the presence of additive noise, multipath fading, shadowing, and other interferences.

- **Non-geometrical techniques:** These techniques do not use lines of position deduced from the estimated geometrical characteristics of the multipaths to compute the MN location. We mention here the approach using cooperative mobiles and the fingerprinting method.

  - *Cooperative Mobiles:* The precision and coverage of the localization technique can be improved by enabling the cooperation of the sensors, known as cooperative mobiles, around the MN to be positioned. In cooperative localization, the MNs can intercommunicate, removing the need to have all MNs within the communication range of multiple ANs. An advantage of this approach is that high ANs density or long-range transmissions are no longer required. Since MNs can obtain information from both ANs and other surrounding MNs, which are within communication range, cooperative localization can offer increased accuracy and coverage [Bahr *et al.*, 2009].

  - *Fingerprinting:* The fingerprinting techniques are alternatives to the previous methods that are sensitive to the propagation conditions, such as NLoS and multipath. Indeed, fingerprinting techniques can be applied to any scenario and environment. In a preliminary step,

often called offline, the area of interest is discretized into cells and a database is built from the signal signatures. An example of such signal signatures that can be used is the RSSIs. The database can be assembled using measured data or simulated using a propagation model. In the online step, the estimated signatures at each AN are compared with the database fingerprints. This can be formulated as a regression problem [Lv *et al.*, 2015], where the idea is to construct a model that takes the signature as an input and outputs the position of the MN [Mahfouz *et al.*, 2015]. The advantage of such approach is that there is no need for a geometrical model that relates the signal strengths to traveled distances. Instead, a radio-cartography is constructed by collecting measurements to cover the targeted area. It must be emphasized that major drawbacks of fingerprinting are associated with database maintenance, sensitivity to environmental changes and cumbersome learning. Furthermore, the offline step is often time consuming, especially if it is based on measurements [Yiu *et al.*, 2017].

## 1.3 Zoning-based Localization

The disadvantage of the described position-based localization algorithms using fingerprinting is their need for a database with exact locations, whose construction is time-costly and complex. For that reason, researchers started to tackle the localization problem by zoning, though very few approaches have yet been proposed. Although position-based techniques can be used for zoning by selecting the zone constituting the location obtained by the positioning technique, it is worthy searching for zoning-based techniques to avoid complex databases and algorithms, and aim at a better accuracy for the specific zoning application.

### 1.3.1 Problem description

We focus in this section on zoning-based localization methods, where the zone of the MN is of interest and not its exact position. This issue is important for health-care applications for instance, where Alzheimer's patients might be lost in

their nursing home [Liu *et al.*, 2013b]. Other applications include museums for supporting guides and emergency management [Smirnov *et al.*, 2012], large malls to facilitate shopping [Wang *et al.*, 2015], etc. Locating the sensor in a specific zone of such environments is completely sufficient. To tackle the problem, the targeted area is partitioned into several zones. These zones might be preexisting such as rooms, corridors, and offices, or manually created by partitioning an area into cells. A fingerprinting database is then constructed in an offline phase in each zone by collecting measurements using one of the enabling technologies described above. An advantage here as compared to position-based techniques is that no exact reference positions are needed, and a random collection of measurements inside each zone is sufficient. In an online phase, once a new measurement is carried for localization by zoning, machine learning algorithms are used to match the new measurement with the corresponding zone.

### 1.3.2 Zoning-based localization techniques

The issue of zoning-based localization has been scarcely studied in literature. Lee & Chen [2007] propose a method called WHAM! (Where Am I) that aims at determining the zone of the sensor. The signal strengths of WiFi Access Points are periodically collected while the user is moving, thus generating a long sequence of signal strength data. The latter is partitioned in order to separate the collected data into different segments such that the data of each segment are more likely to belong to the same zone. The maximum and minimum signal strengths of each zone are considered as inputs to the system, which determines the possible zones of each segment by calculating the mean value of the signal strengths of each AN, and then assigning each zone with a score showing how many mean values of the segment fall into the range of values of this zone. Khedo *et al.* [2010] present an overlapping zone partitioning localization technique. A set of RFID readers is installed in the targeted area, and the reading range is used to estimate the MN zone. The coverage areas are partly overlapped by the readers to reduce the approximated location area, and avoid blind regions that are outside the range of the readers. Chriki *et al.* [2017] present a Support Vector Machines (SVM) based technique for zoning localization using WiFi technology.

The idea is to construct an SVM model, by learning the type and the parameters of a polynomial or Gaussian kernel, to represent the RSSIs of each zone. Then, a one-against-one or one-against-all approaches are used to distinguish between the zones once a new measurement is carried for localization.

These methods match measurements to constructed fingerprint databases and thus motivate the implementation of machine learning algorithms such as classification for this purpose. In what follows, the problem is formulated as a multi-class classification, where the aim is to classify the zone of the mobile sensor according to the measured observation. We provide here a succinct survey of the classification methods that exist in literature, and can be used to solve the problem. Researchers have proposed techniques that are based on the concept of a perceptron, where a sum of weighted inputs is computed and the output is compared to a threshold in order to choose a class [Kotsiantis *et al.*, 2007]. However, perceptrons work only for instances that are linearly separable. When this is not the case, artificial neural networks are used to solve the problem [Rojas, 2013]. Another well-known non-linear method is the SVM that classifies the instances using a decision surface or hyperplane that maximizes the margin between the classes [Honeine *et al.*, 2013]. The k-nearest neighbors algorithm determines the class of an instance by examining the labels of its nearest neighbors and voting for the most frequent one [Souza *et al.*, 2014]. In addition, naive Bayes classifiers assume independency between features to release probabilistic output [Narayanan *et al.*, 2013]. Logistic regression fits data into a logistic function and distributes probabilities on classes according to the generated function [Liu *et al.*, 2014]. In another category, logic-based methods that use decision trees to create hierarchical partitions of the data, and then combine a sequence of logical tests to classify a data intance at each node has been also proposed [Kotsiantis, 2013]. A related approach is to use various rule-based algorithms induced from the training data to create a set of rules [Polkowski, 2013]. A disadvantage of such propositional classifiers is that they do not make use of any available background knowledge. Inductive logic programming methods solve this issue by using a framework of first order predicate logic [Sammut & Webb, 2011].

Alternatives to the classical flat classification techniques, are the hierarchical approaches. Random forests is an ensemble of trees, obtained by bootstrap sampling and by randomly changing the feature set during learning [Breiman, 2001]. More precisely, at each node in the decision tree, a random subset of the input attributes is taken, and the best feature is selected from this subset instead of all attributes. Hierarchical methods can also be derived from classical techniques. Hierarchical Support Vector Machines (HSVM) solves a series of max-cut problems to recursively partition the classes into two-subsets, till pure leaf nodes having only one class are obtained. Then, the classical SVM is applied to solve the binary classification problem at each internal node [Chen *et al.*, 2004].

## 1.4 Mobility-based Tracking

Tracking of sensors is another important research field in WSNs. The aim is to determine the trajectory of the MN using a set of collected measurements.

### 1.4.1 Problem description

Although localization and tracking are usually treated as distinct problems, the definition of a trajectory can be described as the solution of a set of localization problems at successive time instants where the position of the sensor is expressed as a function of time. The motion characteristics such as position, velocity, and acceleration, can be either measured using corresponding sensors, or computed from the knowledge of the time interval between two consecutive localizations of the MN. From an algorithmic point of view, the main difference between localization and tracking is that localization is a *one-time* detection procedure where the accuracy of estimation of the MN location is the only issue, while tracking is an *on-time* procedure, where the fast processing is an additional constraint that is mandatory in real-time applications. The measurements to be collected at the nodes to solve the two problems, as well as the underlying models relating the solutions to those measurements are different. Consequently, not all position-based localization approaches can be used for tracking purposes and vice versa.

### 1.4.2 Mobility-based tracking techniques

The primary objective of target tracking is to estimate the trajectory of a moving target. Tracking methods assume that the target motion can be represented by some known mathematical models that are sufficiently accurate. The most commonly used models are known as state-space models. We will call them *mobility models* hereafter, considering that they depend on the previous state of the MN and the mobility to estimate the next state. It is important to note that standalone mobility models are not recommended in practical applications, since uncertainty increases with time, and it becomes nearly impossible to have a confident estimation after a certain number of tracking trials. For that reason, the estimation should be updated at each time using additional information. This information is obtained by acquiring measurements from the network such as WiFi RSSIs for example, leading to an *observation model*. The problem of tracking becomes a matter of an estimation using a mobility model, and an estimation update using an observation model. Two types of mobility models exist; random process mobility models, and kinematics mobility models.

- **Random process mobility models:** The simplest model for target tracking is the white-noise acceleration model. It assumes that the target acceleration is an independent process. The main attractive feature of this model is its simplicity. However, in practical applications, the acceleration is rarely independent with respect to time, and hence it is only applied when the movement is quiet small or random. Another simple model is the Wiener-process acceleration model that assumes that the acceleration is a process with independent increments, which means the acceleration is supposed to be nearly constant. However, the assumption that the acceleration increment is independent with respect to time is hardly justifiable, except for its simplicity and mathematical tractability. A more general model is the polynomial one. It is well known that any continuous target trajectory can be approximated by an $n$-th degree polynomial to an arbitrary accuracy. Such model amounts to assuming that the $n$-th derivative of the position is nearly constant. The two previously described models thus become special cases of the $n$-th degree polynomial model for $n = 1$

and 2 respectively. This model in its general setting does not appear very attractive for tracking, as it is difficult to develop an efficient method to determine systematically the coefficients of the polynomial. These three models have been thoroughly studied by Bar-Shalom *et al.* [2004]. Whenever white-noise models are not good enough, it is natural to consider a Markov process model. The Singer model assumes that the target acceleration is a zero-mean first-order stationary Markov process [Singer, 1970]. This model corresponds to a motion in between the nearly constant velocity and the nearly constant acceleration models mentioned above, providing a wider coverage than both of them. One disadvantage of the Singer model is that the target acceleration has a zero mean at any moment. To solve the problem, other sophisticated acceleration models are proposed such as the mean-adaptive acceleration model or the asymmetrically distributed normal acceleration model [Kendrick *et al.*, 1981; Kumar & Zhou, 1984].

- **Kinematics mobility models:** The target tracking models are highly dependent on the choice of the state components, and thus the respective kinematic model. This is not a trivial problem, as the target dynamics, accuracy of approximations, and sensor coordinate system, must all be taken into account. The Kalman filter assumes that the posterior density of the states at every time step is Gaussian, and thus it recursively computes the mean and the covariance of the Gaussian posterior [Mahfouz *et al.*, 2014]. It is the optimal solution to the tracking problem when the assumptions of the linear Gaussian environment holds. However, this posterior is not necessarily Gaussian, and hence the filter is not certain to be optimal. Figure 1.3 shows the basic concept of Kalman filtering, where prior knowledge is used to first predict the state. The observed measurements are then used over time to update the state. On the other hand, grid-based methods provide the optimal solution if the state space is discrete and consists of a finite number of states. The preceding assumes the transitional densities and the likelihood to be known, but does not constrain their particular form.

  To address the non-linear cases, several approaches have been proposed such as the extended Kalman filter (EKF) and the unscented Kalman filter

(UKF) [Zhang *et al.*, 2013]. The main feature of these methods is that they approximate the non-linear function in the state dynamic and the observation model. The estimation error of the EKF is usually large due to linearization. The UKF has a better tracking performance since it can better approximate the non-linearity as compared to the EKF. However, it is found that the estimation error of UKF increases when the target's range is far in practical applications. Another well-known method for target tracking is the particle filter (PF) or the Monte-Carlo filter [Hong *et al.*, 2014]. The key idea is to recursively represent the posterior density function by a set of random samples or particles with associated weights according to the measurements. As the number of particles increases, the PF becomes an equivalent representation of the usual functional description of the posterior density, and thus approaches the optimal Bayesian estimate. The advantage of such methods over the previously described ones is the ability to deal with non-linear and non-Gaussian filtering.

## 1.5 Content Organization

This thesis tackles the problem of indoor localization in WSNs and carries several contributions to the domain. First, we introduce a new zoning-based localization method, through an observation model within a belief functions framework, using the WiFi RSSIs and the fingerprinting measurement technique. Second, we propose a solution for localization in wide surface areas by extending the introduced observation model. We also develop an Access Point selection algorithm

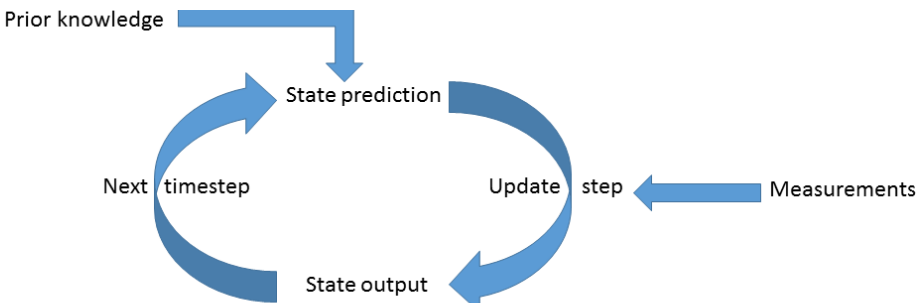



**Figure 1.3:** The basic concept of Kalman filtering.

that aims at selecting the best subset of networks in the localization process. We then, propose a novel tracking technique using the described observation model, the sensors' mobility, and the belief functions theory. Finally, we present an original decentralized approach for a more robust and less complex localization.

### 1.5.1 Organization of the manuscript

The rest of the manuscript is organized as follows:

In the second chapter, we give a brief overview of the belief functions theory. Multisensor data fusion explores efficient methods to transform information retrieved from different sources and at different points in time into a representation that provides effective support for decision making. We generally go over some of the data fusion techniques, motivating our usage of the belief functions theory. We then, introduce some of the well-known concepts of the belief functions theory, providing few examples, which are used throughout the manuscript.

In the third chapter, we propose a zoning-based localization method within the belief functions framework. The idea is to construct an observation model based on a fingerprinting database of WiFi RSSIs. For this purpose, the data acquisition and the localization phases are first described. Then, a parametric and a kernel-based model are proposed to represent the collected data. The belief functions theory is used to assign masses, combine evidence, and associate a confidence level to each zone. This amounts to a degree of confidence in the decision saying that the MN resides in each zone. Finally, we present the experimental setup, and analyze the performance of the method while comparing with other state-of-the-art techniques. The limitations of the proposed model are also discussed.

In the fourth chapter, we extend the observation model of the already described localization method. One major drawback of the latter is that a degradation of accuracy of the localization algorithm occurs when the targeted area is constituted of a large number of zones, due to failure in assigning discriminating evidence for the different zones. For this purpose, the model is extended via hierarchical clustering. The idea is to create a dendrogram of zones, through maximizing the inter- and intra-cluster similarity measurements. This helps reduce the number

of zones being classified at a time, and increase the discrepancy between the obtained clusters. On the other hand, we develop an Access Point selection algorithm to help select the best subset of networks in localization. The algorithm searches for the subset that maximizes the discriminative capacity and minimizes the redundancy at any level of the constructed hierarchy. The influence of each phase of the extended observation model is evaluated through experiments.

In the fifth chapter, we propose a novel tracking technique that uses the mobility of sensors with the already described observation model in the belief functions framework to track the sensors in real time. Considering that the target population is dependent elderly people, we can assume a maximum speed of movement of sensors in the indoor environment. This allows a prediction of the next possible destinations of the mobile sensor, and hence leading to a mobility model. The belief functions framework is used to propagate the previous step evidence till the current one. We present three mobility models. The first one is based on the original succession of zones. The second is based on the transition between created sub-zones and necessitates a specific data acquisition phase. The third is based on hidden Markov models, and aims at classifying trajectories between adjacent zones. The influence of the mobility models, the assumed maximum speed, and the number of points per trajectory are all studied experimentally.

In the sixth chapter, we present an original decentralized approach for zoning-based localization. The aim is to obtain a scalable, more robust, and less complex localization algorithm by partitioning the targeted area into several sectors and assigning a calculator to each one, which locally estimates the sensor's zone by running a local localization algorithm. The final decision is then made by fusing evidence from all calculators. We propose three decentralized architectures that differ in the geographic structure and the decision fusion. The first is distinct sectors with a local decision, the second is distinct sectors with local estimation, and the third is overlapping sectors with local decision. Moreover, we present a calculators placement method using circle packing and covering to reduce the number of required calculators and optimally locate their positions. Comparisons between the proposed decentralized approaches, our previously described centralized one, and other state-of-the-art techniques are provided.

Finally, in the seventh chapter, we provide concluding remarks, and an outlook on future research perspectives.

### 1.5.2 Publications

**Peer-reviewed international journal articles (2+1)**

- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Decentralized Kernel-based Localization in Wireless Sensor Networks Using Belief Functions." *IEEE Sensors Journal*, second revision, June 2018.
- **D.Alshamaa**, F. Mourad-Chehade, P. Honeine. "A Hierarchical Classification Method Using Belief Functions." *Signal Processing*, vol. 148, pp. 68-77, July 2018.
- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Tracking of Mobile Sensors Using Belief Functions in Indoor Wireless Networks." *IEEE Sensors Journal*, vol.14, no.1, pp. 310-319, January 2018.

**Peer-reviewed international conference articles (6)**

- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "The Belief Functions Theory for Sensors Localization in Indoor Wireless Networks." SMPS/BELIEF 2018, Compiègne, France, 17-21 September 2018.
- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Decentralized Sensor Localization by Decision Fusion of RSSI and Mobility in Indoor Environments." *26th European Signal Processing Conference (EUSIPCO)*. Rome, Italy, 3-7 September 2018.
- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "A Weighted Kernel-based Hierarchical Classification Method for Zoning of Sensors in Indoor Wireless Networks." *19th IEEE International Workshop on Signal Processing Advances for Wireless Communications (SPAWC)*. Kalamata, Greece, 24-28 June 2018.
- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Localization of Sensors

in Indoor Wireless Networks: An Observation Model Using WiFi RSS." *9th IFIP International Conference on New Technologies, Mobility and Security (NTMS).* Paris, France, 25-28 February 2018.

- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Mobility-based Tracking Using WiFi RSS in Indoor Wireless Sensor Networks." *9th IFIP International Conference on New Technologies, Mobility and Security (NTMS).* Paris, France, 25-28 February 2018.

- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Zoning-based Localization in Indoor Sensor Networks Using Belief Functions Theory." *17th IEEE International Workshop on Signal Processing Advances for Wireless Communications (SPAWC).* Edinburgh, Scotland, 3-6 July 2016.

**Peer-reviewed national conference articles (1)**

- **D. Alshamaa**, F. Mourad-Chehade, P. Honeine. "Classification Paramétrique Multi-classes à Croyance." *Colloque du GRETSI.* Juan-les-Pins, France, 3-5 septembre 2017.

# Chapter 2

# Multisensor Data Fusion

## Contents

*The decision-making process in localization and tracking systems rely on data coming from multiple sensors. Information retrieved from these sensors requires robust fusion approaches to be processed. One of these approaches is the belief functions theory (BFT), also called the Dempster-Shafer theory. This theory deals with uncertainty and imprecision with a theoretically attractive evidential reasoning framework. This chapter introduces first an overview on multisensor data fusion, describing the system architecture, the challenging problems, and some state-of-the-art algorithms. After that, the BFT is explained as a framework for data fusion, discussing its general concepts and the notations that will be used throughout the manuscript.*

## 2.1 Introduction

Sensors are used to provide a system with useful information concerning some features of interest in the system's environment. Combining the results of multiple sensors can provide more accurate and reliable information than using a single sensor. This allows either an improved accuracy using existing sensors or the same performance using smaller or cheaper ones. The potential advantages of multisensor data integration and fusion are described at the following levels. First, multisensor systems have an inherent redundancy. The fusion of redundant information can reduce the uncertainty, thus increasing the overall accuracy. This can also serve in increasing the reliability and robustness in case of sensor error or failure. In addition, complementary information from multiple sensors allows perceiving certain features in the environment, which cannot be perceived using just the information from each individual sensor operating separately. Also, more timely information may be provided by multiple sensors due to either the actual speed of operation of each sensor, or the parallel processing that might be achieved as part of the fusion process. Moreover, the use of multiple sensors allows an increase in coverage, both spatial and temporal. In fact, multiple sensors can observe a region larger than the one observable by a single sensor [Hall & Llinas, 1997].

## 2.2 System Architecture Issues

Two important issues to be studied in data fusion systems are sensors configuration and level of fusion [Varshney, 1997].

### 2.2.1 Sensors configuration

There are three basic configurations in wireless sensor networks, that differ by how the data are sensed. At first, we have the complementary configuration. A sensor configuration is called complementary if the sensors do not directly depend on each other, but can be combined in order to give a more complete image of the phenomenon under observation. Complementary sensors help resolve the problem of incompleteness. Another configuration is the competitive. A sensor configuration is competitive if each sensor delivers an independent measurement of the same property. The aim of competitive fusion is to reduce the effects of uncertain and erroneous measurements. The third configuration is the cooperative. A cooperative sensor configuration uses the information provided by two, or more, independent sensors to derive information that cannot be available from the single sensors. [Ruiz *et al.*, 2003].

### 2.2.2 Level of fusion

This means the level at which fusion takes place. Data aggregation can take place across sensors, across features, or across time.

- **Fusion across sensors:** In this situation, a number of sensors nominally measure the same property.
- **Fusion across features:** In this situation, a number of sensors measure different quantities associated with the same experimental situation.
- **Fusion across time:** In this situation, current measurements are fused with historical information. Often the current information is not sufficient to determine the system accurately and historical information has to be incorporated to determine the system accurately.

## 2.3 Challenging Problems

A number of problems confronts multisensor data fusion. These problems come from the data to be combined, imperfection and diversity of the sensor technologies, and the nature of the application environment. They are summarized in [Mahler, 2004; Rajive *et al.*, 1999]. We limit the scope here to challenges encountered in localization and tracking, providing examples from practical experiences.

- **Measurement:** The data measured by the sensors are imperfect and imprecise. For example, the antenna of the sensor to be localized and the software used to scan the network and measure the RSSIs of the WiFi signals provide uncertain measurements.
- **Environment:** The uncertainties in data do not come only from noisy measurements, but also from the surrounding environment. The temperature, humidity, furniture, doors, among others, result in a wide variation in the sensor's measurements.
- **Data dimensionality:** To tackle the previous two challenges, it is important to have redundant sources of information. They allow reducing the effect of imperfect or missing data. However, this leads to higher dimenionality, thus increasing the complexity of the data fusion algorithm. The same physical WiFi Access Point (AP) provides several networks that can be used as different sources of information in localization. These networks are redundant and the data fusion algorithm should deal with them.
- **Data modality:** The sensor network might collect homogeneous or heterogeneous data. An example of the former is receiving measurements from several APs, and of the latter is aggregating information from APs and motion sensors.
- **Operational timing:** The operation frequency of the sensors may be different leading to missing data. The APs send beacons at a certain frequency, and so does the sensor scanning the network. The difference between the two operations results in losing information at certain instants.

- **Static vs. dynamic phenomena:** The phenomenon under observation may be either time-invariant or varying with time. In the latter case, it is necessary to incorporate a recent history of measurements into the fusion process. An issue is raised here, often referred to by data freshness, considering how quickly data sources capture changes and update accordingly. This plays a vital role in the validity of the fusion results. An illustration of this problem is the indoor environment that changes with time. This leads to out-of-date measurements that do not correspond to the databases constructed at a previous time.

- **Processing framework:** The data fusion processing can be performed in a centralized or decentralized manner. Choosing the right framework is important and has a significant impact on the system's performance. Since data are processed locally, a decentralized approach is usually preferable in WSNs. This is much more efficient as compared to the centralized approach where high communication is required. In the latter, measurements are sent to a central processing unit for fusion.

## 2.4 Multisensor Data Fusion Algorithms

Multisensor data fusion algorithms have been proposed to tackle the challenges described above. We briefly describe here the probability theory [Durrant-Whyte & Henderson, 2008], the fuzzy logic theory [Zadeh, 1965], the rough set theory [Pawlak, 2012], and the belief functions theory [Shafer, 1976].

### 2.4.1 Probability theory

Probabilistic methods rely on the probability distribution to express data uncertainty. At the core of these methods lies the Bayes estimator, which enables fusion of pieces of data [Lillis *et al.*, 2006]. The Bayes estimator provides a method for computing the posterior probability distribution of the hypothetical state based on a given set of measurements and the prior distribution. However, both the prior distribution and the normalizing term contain integrals that cannot be evaluated analytically. An exceptional case is the Kalman filter that provides an exact

analytical solution due to enforcing simplifying constraints on the system dynamics to be linear-Gaussian [Mahfouz *et al.*, 2014]. However, it is inappropriate for applications whose error characteristics are not parameterized.

When dealing with non-linear system dynamics, the Monte Carlo simulation-based techniques such as Sequential Monte Carlo (SMC) and Markov Chain Monte Carlo (MCMC) are among the most powerful and popular methods of approximating probabilities. They are also very flexible as they do not make any assumption regarding the probability distributions to be approximated. Particle filters provide an alternative for Kalman filtering when dealing with non-Gaussian noise and non-linearity in the system [Fox *et al.*, 2003]. The idea is to deploy a weighted ensemble of randomly drawn samples as an approximation of the probability distribution of interest. They are used to approximate the posterior probability of the system state as a weighted sum of random samples. The random samples are usually predicted from the prior distribution with their weights updated according to the likelihood of the given measurement. However, they are computationally expensive as they may require a large number of particles to estimate the desired posterior probability density.

### 2.4.2 Fuzzy logic theory

The classical set theory allows elements to be either included in a set or not. This is in contrast with human reasoning, which includes a measure of imprecision or uncertainty. This approximate reasoning is modeled by fuzzy logic, which is a multivalued logic that allows intermediate values to be defined between conventional threshold values. Fuzzy systems allow the use of fuzzy sets to draw conclusions and to make decisions [Stover *et al.*, 1996]. Fuzzy sets differ from classical sets by allowing an object to be a partial member of a set. In such systems, the dynamic behavior is characterized by a set of linguistic fuzzy rules based on the knowledge of a human expert. The fuzzy rules are a set of *if-then* propositions containing linguistic variables. The antecedents of a fuzzy rule form a combination of sets through the use of logic operations. The fuzzy sets and rules are together the foundation of a rule-based inference system.

### 2.4.3 Rough sets theory

The rough set theory allows the approximation of possible states of the system based on the granularity of input data [Qian *et al.*, 2014]. Once approximated as rough sets, data pieces can be combined using fusion operators such as intersection or union. In order to perform fusion successfully, data granules must be neither too fine nor too rough. In the former case, they become singletons, and thus the rough set theory reduces to the classical probabilistic one. In the latter case, they become very large subsets, and thus the lower approximation of data is likely to be empty, resulting in total ignorance. An advantage of this theory is that it does not require any preliminary information such as membership function or data distribution. The fusion of imprecise data is done based on the internal structure or the granularity. However, the theory has not been often applied to data fusion problems as it is still not well understood by the fusion community.

### 2.4.4 Belief functions theory

The belief function theory (BFT) is an appealing framework for reasoning under uncertainty when imperfect data need to be aggregated through an information fusion process [Hégarat-Mascle *et al.*, 2003]. Indeed, imprecise and uncertain pieces of evidence can be efficiently represented and aggregated as part of the BFT. Combination rules are well-defined mathematical operators designed for this purpose. The theory does not assign a priori probabilities to unknown propositions. Instead, it assigns evidence only when the supporting information is available. In fact, it allows for explicit representation of total ignorance by assigning the entire mass to the frame of discernment at any time.

## 2.5 The Belief Functions Framework

The BFT was originally developed by Dempster [1967] and Shafer [1976]. It is also known as the Dempster-Shafer theory or evidence theory [Yager & Liu, 2008]. Similar to the Bayesian probability theory, the BFT is a theory of quantified beliefs. It is based on the notion of evidence and how different pieces are combined to

make inferences. The motivation behind using the BFT as a framework for multisensor data fusion in this thesis is its ability to handle the challenges described in Section 2.3. In particular, it allows us to manipulate missing RSSI packets, combine different types of sensors including redundant ones, model sensors reliability, and deal with uncertainty. The BFT can be interpreted as a generalization of the Bayesian probability theory that models correctly ambiguous data.

**Ex 2.1.** Suppose that someone tosses a coin and bet you on its outcome. If you do not know the person, you cannot trust that the coin is fair. However, in the Bayesian framework, this factor is not taken into account, and thus both outcomes of the coin are supposed to as equiprobable with $P(heads) = P(tails) = \frac{1}{2}$. The Bayesian approach corresponds to a situation where the coin has been tested before and found to be fair. In contrast, the belief functions framework defines this state of ignorance explicitly by assigning all belief masses to the disjunction of the possible outcomes. Hence, it only states that $P(\{heads; tails\}) = 1$ and does not add any information regarding the true probability distribution. If the coin is found to be fair later on, the belief is updated using the new evidence.

### 2.5.1 Representation of information

Given a question of interest, let $\Theta$ be a finite set of possible answers to the question, called *frame of discernment*. A hypothesis or proposition is a subset $A \subseteq \Theta$ of the frame of discernment saying that the truth lies in $A$. It is an element of the power set $P(\Theta)$,

$$P(\Theta) = 2^{\Theta} = \{A \mid A \subseteq \Theta\}. \tag{2.1}$$

A hypothesis consisting of only one element $A$, such that $|A| = 1$, is called a singleton, the operator $|\cdot|$ being the cardinal of the set $A$. The belief functions framework assigns a belief value to each hypothesis based on one or more pieces of evidence. In contrast to the Bayesian framework, the additivity of belief values is not required. This means that the belief in a hypothesis and in its complement can be less than 1. Instead of just having singletons with different probabilities as in the Bayesian framework, the cardinality of each hypothesis with an assigned belief

can vary. This provides the BFT with an additional dimension of uncertainty, allowing it to take ignorance into account.

There are several representations to quantify the belief within the belief functions framework, such as the mass function denoted by $m_S(\cdot)$, the belief function denoted by $bel_S(\cdot)$, the plausibility function denoted by $pl_S(\cdot)$, and the commonality function denoted by $q_S(\cdot)$, all determined according to an information source $S$. In the following, we introduce the various representations, and the rules for converting between them.

#### 2.5.1.1 Mass function

One fundamental function of the BFT is the mass function, also called the basic belief assignment (BBA). A mass function $m_S(\cdot)$ is a mapping from $P(\Theta)$ to the interval $[0, 1]$, defined according to a certain information source $S$. It satisfies:

$$\sum_{A \in P(\Theta)} m_S(A) = 1. \tag{2.2}$$

The mass $m_S(A)$ given to $A \in P(\Theta)$ stands for the proportion of evidence that is brought by the source $S$ and assigned to hypothesis $A$. Such an assignment to a set $A$ implies ignorance about the mass distribution over subsets of $A$. In Shafer's original work [Shafer, 1976], there is an additional constraint requiring that a mass function must not assign a positive value to the empty set,

$$m_S(\emptyset) = 0. \tag{2.3}$$

A mass function that satisfies equation (2.3) is said to be normalized. This constraint is not present in the works of Smets [1988, 1992], where the mass assigned to $\emptyset$ usually represents the possibility that the true value is not included in the frame of discernment. Smets [1992] argues that requiring $m_S(\emptyset) = 0$ corresponds to a closed-world assumption, while allowing $m_S(\emptyset) > 0$ corresponds to an open-world assumption.

Often, one is only interested in the sets $A \subseteq \Theta$ with positive mass values, $m(A) > 0$. Such sets are called focal sets. The set $F_{m_S}$ consisting of all fo-

cal sets corresponding to a mass function $m_S$ is defined as follows,

$$F_{m_S} = \{A \mid A \subseteq \Theta, m_S(A) \neq 0\}. \tag{2.4}$$

**Ex 2.2.** Consider an area constituted of three zones $Z_1$, $Z_2$, and $Z_3$, where the objective is to determine the zone of a sensor at a certain instant. Suppose that we use for this purpose an information source $S_1$ that relates the measured signal to the position of the sensor. After the source measures the signal from the sensor, it decides that the latter is in the area constituted of the zones $Z_1$ and $Z_3$, however it is sure about 60% of its decision. Thus the evidence that can be retrieved by $S_1$ is $m_{S_1}(\{Z_1, Z_3\}) = 0.6$, which corresponds to the percentage given above, and $m_{S_1}(\{Z_1, Z_2, Z_3\}) = 0.4$. Indeed, the remaining mass is transferred to the whole set, meaning ignorance. Suppose that another source $S_2$ decides that the sensor is four times more likely to be in zone $Z_2$ than in the area constituted of the zones $Z_1$ and $Z_3$, according to some other mobility measurements. Hence, the second piece of evidence suggests that $m_{S_2}(\{Z_2\}) = 0.8$ and $m_{S_2}(\{Z_1, Z_3\}) = 0.2$.

This example shows how to transform human reasoning into mass functions in the belief functions framework. We will use the same example throughout this chapter to clarify other concepts.

#### 2.5.1.2 Belief function

The term belief function is somewhat ambiguous since it is used both as a general term, thus enclosing all the different representations, and as a specific term referring to this representation. In this thesis, the term is used in the general sense unless explicitly stated otherwise. Here, we introduce the specific representation. The total amount of belief committed to a hypothesis $A \subseteq \Theta$ by the information source $S$, including all subsets $A_i \subseteq A$, is denoted by $bel_S(A)$. The function $bel_S(\cdot) : P(\Theta) \to [0, 1]$ is called a belief function, and is computed from the mass function as follows,

$$bel_S(A) = \sum_{A_i \subseteq A, A_i \neq \emptyset} m_S(A_i), \qquad \forall A \subseteq \Theta, A \neq \emptyset. \tag{2.5}$$

The belief function $bel_S(A)$ is interpreted as the lower bound for the probability of having the state falling in $A$.. It is also possible to compute the mass function from the belief function as follows,

$$m_S(A) = \sum_{A_i \subseteq A} (-1)^{|A \backslash A_i|} bel_S(A_i), \quad \forall A \subseteq \Theta, \tag{2.6}$$

where $|A \backslash A_i|$ designs the cardinal of the difference between sets $A$ and $A_i$.

#### 2.5.1.3 Plausibility function

In this framework, the belief not committed to $\bar{A}$, the negation of $A$, is not automatically committed to $A$, but it does make $A$ more credible or plausible. Thus, it is intuitive to define the plausibility function $pl_S(A)$ as the sum of beliefs not committed to $\bar{A}$,

$$pl_S(A) = 1 - bel_S(\bar{A}), \quad \forall A \subseteq \Theta. \tag{2.7}$$

Whereas the belief functions can be viewed as a lower bound for an unknown probability function under a lower- and upper probability interpretation, the plausibility can be viewed as its upper bound. It is also possible to compute the plausibility function from the mass function,

$$pl_S(A) = \sum_{A_i \cap A \neq \emptyset} m_S(A_i), \quad \forall A \subseteq \Theta. \tag{2.8}$$

#### 2.5.1.4 Commonality function

The commonality $q_S(A)$ represents the total mass committed to $A$ and to all of the supersets $A_i$, with $A \subseteq A_i$, by the source $S$. The commonality $q_S(A)$ therefore expresses how much mass potentially supports the set $A$. A commonality function $q_S : P(\Theta) \rightarrow [0, 1]$ is defined as,

$$q_S(A) = \sum_{A \subseteq A_i} m_S(A_i), \quad \forall A \subseteq \Theta. \tag{2.9}$$

In order to compute a mass function $m_S(\cdot)$ from a given commonality function $q_S(\cdot)$, the following equation can be used,

$$m_S(A) = \sum_{A \subseteq A_i} (-1)^{|A_i \backslash A|} q_S(A_i), \quad \forall A \subseteq \Theta. \tag{2.10}$$

**Ex 2.3.** Considering Ex. 2.2, we can compute the belief, plausibility, and commonality of all the zones and their subsets with respect to the two sources as shown in Table 2.1.

### 2.5.2 Combination rules

The purpose of aggregation of information is to meaningfully summarize and simplify a corpus of data whether the data is coming from a single source or multiple sources. Familiar examples of aggregation techniques include arithmetic averages, geometric averages, harmonic averages, maximum values, and minimum values. Combination rules are the special types of aggregation methods for data obtained from multiple sources. These multiple sources provide different assessments for the same frame of discernment. In order to solve inference problems, the mass functions representing different pieces of evidence need to be combined in a meaningful way. This is why combination rules are a major building block of the BFT. Typically, each piece of evidence is represented by a separate mass function. Combination rules are then used to successively merge all these mass functions in order to obtain a mass function representing all available evidence.

These rules can vary between conjunction, viewed as AND-based intersection, and disjunction, viewed as OR-based union. In a situation where all sources are considered reliable, a conjunctive operation is appropriate. In a case where there

**Table 2.1:** An example of different representation functions.

| source | $S_1$ | | | $S_2$ | | |
|---|---|---|---|---|---|---|
| subset | $bel_{S_1}(\cdot)$ | $pl_{S_1}(\cdot)$ | $q_{S_1}(\cdot)$ | $bel_{S_2}(\cdot)$ | $pl_{S_2}(\cdot)$ | $q_{S_2}(\cdot)$ |
| $\{Z_1\}$ | 0 | 1 | 1 | 0 | 0.2 | 0.2 |
| $\{Z_2\}$ | 0 | 0.4 | 0.4 | 0.8 | 0.8 | 0.8 |
| $\{Z_3\}$ | 0 | 1 | 1 | 0 | 0.2 | 0.2 |
| $\{Z_1, Z_2\}$ | 0 | 1 | 0.4 | 0.8 | 1 | 0 |
| $\{Z_2, Z_3\}$ | 0 | 1 | 0.4 | 0.8 | 1 | 0 |
| $\{Z_1, Z_3\}$ | 0.6 | 1 | 1 | 0.2 | 0.2 | 0.2 |
| $\{Z_1, Z_2, Z_3\}$ | 1 | 1 | 0.4 | 1 | 1 | 0 |

is one reliable source among many, the use of a disjunctive combination operation is justified. However, many combination rules lie between these two extremes. Dubois & Prade [1992] describe these types of combinations as conjunctive pooling, disjunctive pooling, and tradeoff. Surveys of different combination rules are given in [Sentz & Ferson, 2002; Smets, 2007]. The most important one is arguably Dempster's rule. It strongly emphasizes the agreement between multiple sources and ignores all the conflicting evidence through a normalization factor. Most other combination rules are variations of Dempster's rule and only differ in how they handle conflicting evidence.

#### 2.5.2.1 Dempster's rule

The Dempster's rule of combination was first introduced by Dempster [1967] and then reinterpreted by Shafer [1976] as a basis for the BFT. It allows combining normalized belief functions that are defined over the same frame of discernment and are based on independent arguments or bodies of evidence. The Dempster's rule of combination is purely a conjunctive operation (AND). Let $m_1(\cdot)$ and $m_2(\cdot)$ be normalized mass functions induced by distinct pieces of evidence that are defined over the same frame of discernment $\Theta$. The mass function $m_{1\bigoplus 2}(\cdot) = m_1(\cdot) \bigoplus m_2(\cdot)$ combined according to Dempster's rule $\bigoplus$ is the conjunctive combination followed by a normalized phase, computed as,

$$m_{1\bigoplus 2}(A) = \frac{\sum_{A_i \cap A_j = A} m_1(A_i) m_2(A_j)}{1 - \sum_{A_i \cap A_j = \emptyset} m_1(A_i) m_2(A_j)}, \qquad \forall A \in P(\Theta), A \neq \emptyset, \tag{2.11}$$

with $m_{1\bigoplus 2}(\emptyset) = 0$. Here, the denominator is a normalization assuring that the resulting mass function is normalized. Since empty intersection indicates a conflict, $\sum_{A_i \cap A_j = \emptyset} m_1(A_i) m_2(A_j)$ measures the total amount of conflict. It accounts for the products of mass values corresponding to all empty intersections of sets. This combination rule is easily extended to a higher number of sources by successively combining the pieces of evidence.

#### 2.5.2.2 Conjunctive rule

The conjunctive rule of combination is an adaptation of the Dempster's rule where unnormalized belief functions are allowed. The only difference to the Dempster's

rule is that the normalization step performed by the latter is omitted. Let $m_1(\cdot)$ and $m_2(\cdot)$ be two mass functions induced by distinct pieces of evidence and which are defined over the same frame of discernment $\Theta$. The mass function $m_{1\textcircled{\cap}2}(\cdot) = m_1(\cdot)\textcircled{\cap}m_2(\cdot)$ resulting from the combination using the conjunctive rule $\textcircled{\cap}$ is defined as,

$$m_{1\textcircled{\cap}2}(A) = \sum_{A_i \cap A_j = A} m_1(A_i)m_2(A_j), \qquad \forall A \in P(\Theta). \tag{2.12}$$

#### 2.5.2.3 Disjunctive rule

The disjunctive rule of combination is applied when only one of several pieces of evidence holds. Whereas Dempster's rule and the conjunctive rule correspond to an AND-like operation, the disjunctive combination rule represents an OR-like operation. Let $m_1(\cdot)$ and $m_2(\cdot)$ be two mass functions induced by distinct pieces of evidence that are defined over the same frame of discernment $\Theta$. The mass function $m_{1\textcircled{\cup}2}(\cdot) = m_1(\cdot)\textcircled{\cup}m_2(\cdot)$ resulting from the combination using the disjunctive rule $\textcircled{\cup}$ is defined as,

$$m_{1\textcircled{\cup}2}(A) = \sum_{A_i \cup A_j = A} m_1(A_i)m_2(A_j), \qquad \forall A \in P(\Theta). \tag{2.13}$$

There is no need for normalization since no conflict results from the disjunctive rule of combination. This is because the union $A_i \cup A_j$ is only empty if both sets are empty.

**Ex 2.4.** Considering Ex. 2.2, we try here to combine the evidence by the three combination rules. Table 2.2 shows the assigned evidence to sets obtained by conjunctive, disjunctive rules, and Dempster's combination rules.

#### 2.5.2.4 Dempster's rule limitation

One of the reasons why new combination rules kept getting proposed over time was Zadeh's criticism of Dempster's rule when significant conflict in the information is encountered [Zadeh, 1984]. An example of this criticism in the domain of zoning localization is given here.

**Ex 2.5.** Suppose we have three different zones $\{Z_1, Z_2, Z_3\}$ and two sources of information $\{S_1, S_2\}$, in the targeted area. Let the evidence retrieved from the two sources be as follows,

$$m_{S_1}(Z_1) = 0.99, m_{S_1}(Z_2) = 0.01,$$
$$m_{S_2}(Z_3) = 0.99, m_{S_2}(Z_2) = 0.01.$$

The result of combining these evidence using Dempster's rule is $m_{\oplus}(Z_2) = 1$, which means that there is absolute certainty that the mobile node is in zone $Z_2$. This has been interpreted as being counter intuitive because both sources provide confidence that zone $Z_2$ is highly unlikely. It should be noted that this criticism is not only limited to the BFT, but can be used against the probability theory as well. The conjunctive rule of combination leads to $m_{\textcircled{\cap}}(Z_2) = 0.0001$ and assigning the remaining mass to the empty set. Whereas the disjunctive rule of combination leads to the following, $m_{\textcircled{\cup}}(\{Z_1, Z_2\}) = 0.0099, m_{\textcircled{\cup}}(\{Z_1, Z_3\}) = 0.9801, m_{\textcircled{\cup}}(\{Z_2\}) = 0.0001$, and $m_{\textcircled{\cup}}(\{Z_2, Z_3\}) = 0.0099$.

As it can be noticed from the previous examples, the conjunctive rule is more specialized and informative than the disjunctive rule. So is the Dempster's rule that also neglects the conflict between the sources through normalization. This is noticed in Ex. 2.4 where the disjunctive rule assigns evidence to more general subsets, while both the conjunctive and Dempster's carry more important and

**Table 2.2:** An example of different combination rules.

| $m(\cdot)$ | combination rule | | |
|---|---|---|---|
| subset | conjunctive | disjunctive | Dempster's |
| $\{Z_1\}$ | 0 | 0 | 0 |
| $\{Z_2\}$ | 0.32 | 0 | 0.615 |
| $\{Z_3\}$ | 0 | 0 | 0 |
| $\{Z_1, Z_2\}$ | 0 | 0 | 0 |
| $\{Z_2, Z_3\}$ | 0 | 0 | 0 |
| $\{Z_1, Z_3\}$ | 0 | 0.12 | 0 |
| $\{Z_1, Z_2, Z_3\}$ | 0.2 | 0.88 | 0.385 |

specific evidence. However, the disjunctive rule of combination generates the most logical result when the sources are highly conflicting. This is clear in Ex. 2.5 where the disjunctive rule assigns the highest evidence to $\{Z_1, Z_3\}$, while Dempster's rule assigns a counter intuitive evidence to $\{Z_2\}$. Consequently, researchers have modified the Dempster's rule, attempting to represent the degree of conflict and the allocation of the BBA associated with it in the final result.

### 2.5.3 Discounting

Up till now, the uncertainties in the evidence are either considered in the assumptions of the frame $\Theta$, or taken into account by a function supporting this evidence. However, one might have some doubts regarding the reliability of the source that provides this information or piece of evidence. Such meta-knowledge can be considered using the discounting operation [Mercier *et al.*, 2008]. The sources of information might have the same or different degree of reliability, allowing to model the error rate of each source.

**Ex 2.6.** Suppose one wants to measure the temperature $T$, and uses a thermometer for this purpose, which indicates 37°C. The thermometer might be totally reliable, and thus the real temperature will be 37°C. If the thermometer is totally unreliable, no information can be retrieved from it, thus the real temperature can be any possible one. Practically, the thermometer is partially reliable with some incertitude on the measured values. If the incertitude of this thermometer is equal to one degree for example, the real temperature will be in the interval $[36°C, 38°C]$.

The BFT allows to take into account these uncertainties by providing the necessary adjustments. The obvious way to use discounting with combination rules is to discount the functions before combining them. Shafer [1976] proves that even a uniform discounting has advantages in eliminating the influence of a single function conflicting with all others, provided that all others do not conflict too much with each other, and the discount rate is neither too small nor too large. As described in the previous section, conflict between different sources of evidence is internal evidence that something is wrong in our assessment of one or more of these sources. Yet Dempster's rule for instance will sometimes ignore such

evidence and allow one or a few of the strongest sources to dominate the others. In this way, the final function after combination may approximate the strong, and possibly erroneous, one based on these sources alone. If all the functions to be combined are strongly conflicting, an average seems reasonable, whereas if one of them strongly conflicts with others while others do not, then it is better to eliminate the odd one than to allow it to dominate.

As previously indicated, it is appropriate to discount a function only if it fails to take into account some particular uncertainty that affects the evidence as a whole. In the case where the function is based on a statistical observation $x$, it will be based on the empirical fact of occurrence of $x$, which is known with less than certainty. We may think that we have observed $x$, but not for sure. This is due to a very fallible process of adjustment and transcription. For that reason, discounting the source of information is the most natural way to account for such uncertainties. We distinguish between two types of discounting, classical and contextual, described in the following.

#### 2.5.3.1 Classical discounting of information

The reliability of a source of information is classically taken into account by the discounting operation, which transforms the supporting function into a weaker, less informative one. When a piece of information, represented by a mass function $m_S(\cdot)$, is received from the source $S$, the classical discounting operation by a discount rate $\alpha_S \in [0, 1]$ can be applied to account the degree of reliability of $S$. It is defined as follows,

$$ {}^{\alpha}m_S(A) = \begin{cases} (1-\alpha_S)m_S(A), & \text{if } A \in P(\Theta),\ A \neq \Theta; \\ \alpha_S + (1-\alpha_S)m_S(A), & \text{if } A = \Theta; \\ 0, & \text{otherwise,} \end{cases} \tag{2.14} $$

A discount rate $\alpha_S$ equal to 1, means that the source is not reliable at all and thus the piece of information it provides cannot be taken into account, so the knowledge remains vacuous, ${}^{\alpha}m_S(\Theta) = 1$ and ${}^{\alpha}m_S(A) = 0$ for $A \neq \Theta$. On the contrary, a null discount rate indicates that the source is fully reliable and the piece of information is entirely accepted, ${}^{\alpha}m_S(\cdot) = m_S(\cdot)$. In practice, however,

the source has a certain degree of reliability $\beta_S = 1 - \alpha_S \in [0, 1]$ that is either known before or evaluated through theoretical models or experiments. By doing this, the amounts of evidence given to the subsets of $\Theta$ are reduced, and the remaining evidence is given to the whole set $\Theta$.

#### 2.5.3.2 Contextual discounting of information

The main idea of contextual discounting is based on the fact that the reliability of a source of information can be expected to vary according to the context itself, rather than fixed for the source of information.

**Ex 2.7.** Suppose a thermometer measures temperatures between $-300°C$ and $+300°C$. The thermometer might have a reliability of $\beta = 0.9$ in cold environments $[-300°C, 0]$, and a reliability of $\beta = 0.8$ in warm environments $[0, +300°C]$. We can make use of this information to assign different discount rates per context.

Let $\mathcal{W} = \{w_1, \ldots, w_L\}$ be a coarsening of $\Theta$, which means that $w_1, \ldots, w_L$ form a partition of $\Theta$. In the contextual model, we hold belief on the degree of reliability of the source of information conditionally on each $w_l, l \in \{1, \ldots, L\}$. For all $l \in 1, \ldots, L$, $\beta_S^l = 1 - \alpha_S^l$ represents the degree of reliability of the source $S$ knowing that the true answer of the question of interest belongs to $w_l$. Mercier *et al.* [2012] prove that a simple method to compute the contextual discounting ${}^{\alpha}m_S(A)$ of a mass function $m_S(A)$ consists in using its expression through the disjunctive rule of combination as in equation (2.13),

$$ {}^{\alpha}m_S(A) = m_S \textcircled{\cup} m_S^0(A), \tag{2.15} $$

such that $m_S^0(A)$ is defined as follows,

$$ m_S^0(A) = m_S^1 \textcircled{\cup} m_S^2 \textcircled{\cup} \ldots \textcircled{\cup} m_S^L(A), \tag{2.16} $$

where each $m_S^l(A), l \in \{1, \ldots, L\}$, is defined by,

$$ m_S^l(A) = \begin{cases} (1 - \alpha_S^l), & \text{if } A = \emptyset; \\ \alpha_S^l, & \text{if } A = w_l; \\ 0, & \text{otherwise.} \end{cases} \tag{2.17} $$

Liu *et al.* [2011] argue that the weighting factors for sources of information should be relative, not only with the support degree gained from the other sources, but also with the weights of those other sources. In the discounting process, the reliability discounting rule must be selected by the system designer according to the application. Two adjustments for contextual discounting can be done [Elouedi *et al.*, 2010],

- *Use of percentage of correct determination of context:* This approach consists in exploiting the reliability of the source for each context $w_l, l \in \{1, \ldots, L\}$ such that the truth is $w_l$. To this end, we construct a table having as rows the decided context and as columns the true context. The reliability rate is then obtained by looking at the column $l$ of the table, that shows the percentage of correct context determination such that the truth is $w_l$.

- *Use of a distance:* This approach consists in finding the reliability rate of the source of information $S$ by taking into account, for each context $w_l, l \in \{1, \ldots, L\}$, all the decisions given by that source. The Euclidean distance is proposed to measure the difference between the true context and the decided one. The source $S$ is said to be more reliable on context $w_l$ if the Euclidean distance between the decisions of the source and $w_l$ is smaller.

### 2.5.4 Decision making

Decision making under uncertainty is an important problem in real world applications. The BFT aims to model a decision maker's subjective evaluation of evidence. It allows one to express partial beliefs when complete information is not available. Some methods for using the BFT in decision making have been studied by Jaffray [1989], Yager [1992], Smets & Kennes [1994], and Strat [1990]. In order to make decisions based on the BFT, Smets [2002, 2005] argues that beliefs first need to be transformed to probabilities. In fact, we try to select the most likely hypothesis, which may be difficult to realize directly with the basics

of the BFT where mass functions are given not only to singletons but also to subsets of hypothesis. Some solutions exist to ensure the decision making within the theory of belief functions. The best known is the pignistic probability proposed by the Transferable Belief Model (TBM). Other criteria exist like the maximum of commonality and the maximum of plausibility [Smets, 2002].

The TBM is a model developed to represent someone's degree of beliefs. This model is based on the use of belief functions and is closely related to the model described by Shafer [1976]. Kennes & Smets [1990] detailed the TBM and compared it with the classical Bayesian model. In the TBM, it is distinguished between two aspects of beliefs, belief as weighted opinions, and belief for decision making [Smets, 2002]. The two levels are the credal level, where beliefs are held, and the pignistic level, where beliefs are used to make decisions. Only at the pignistic level it is possible to compute an expected value of a utility function, which is the basis for rational decision making. Decision making requires that we derive a probability function that can be used to compute expected utilities of each potential decision. It means that uncertainty at the pignistic level must be quantified by a probability function. However, it does not mean that beliefs at the credal level must also be quantified by a probability function. Usually these two levels are not distinguished and probability functions are used to quantify beliefs at both levels. Once these two levels are distinguished, as done in the TBM, the classical arguments used to justify the use of probability functions do not apply anymore at the credal level, where beliefs will be represented by belief functions. What is required is a transformation between the representation at the credal level and the probability function that must exist at the pignistic level. The probability functions needed to compute expected utilities are called pignistic probabilities to emphasize they do not represent beliefs, but are just induced by them. The detailed justification of why this probability function is adequate for decision making and its use to provide an operational definition to the belief function is detailed by Smets & Kennes [1994].

Transforming a mass function into a pignistic probability function is done via the pignistic transformation. Let $m_S(\cdot)$ denote a mass function with frame of

discernment $\Theta$ and let $BetP_S(\cdot)$ denote the corresponding pignistic probability function. The pignistic transformation of $m(\cdot)$ into $BetP(\cdot)$ is defined as

$$BetP_S(A) = \sum_{A \subseteq A_i} \frac{m_S(A_i)}{|A_i|}. \tag{2.18}$$

This transformation causes all mass values assigned to sets $A$ to be evenly distributed among the elements $A_i \in A$. Decisions are then achieved by computing the expected utilities of the acts using $BetP(\cdot)$ as the probability function needed to compute the expectations.

## 2.6 Conclusion

In this chapter, we provided a brief overview on multisensor data fusion, where we described the important issues to be studied and the challenging problems that confront it. After that, we highlighted some of the theories and algorithms that are used for this aim. Then, we focused on the BFT, one of those fusion algorithms, as a framework for evidence combination and association under uncertainty. We first introduced the general concepts of the theory, and then explained how it handles information through various representations. Afterwards, we detailed the combination rules that allow the fusion and integration of the different representations. Moreover, we explained the discounting operation that is used to account the reliability of the sources of information. Finally, we defined the decision making process carried by the BFT through the TBM.

# Chapter 3

# Zoning-based Localization

## Contents

*Localization is an important issue in WSNs to assist elderly people in distress. In this chapter, we propose a new zoning-based localization approach using the belief functions theory. We first formulate the problem, describing the network configuration, the data collection phase, and the localization phase. We then explain the proposed localization technique in the belief functions framework. Finally, we examine the performance of the proposed approach and compare it with other techniques.*

## 3.1 Introduction

Massive advances in wireless communications and electronics have enables the development of heavily distributed Wireless Sensor Networks constituted of hundreds of sensor nodes. These networks help in monitoring elderly people in hospitals, nursing homes, and health-care facilities [Liu *et al.*, 2013b]. Each person is equipped with a sensor in the form of bracelet, watch, or medallion. The objective becomes to localize the sensor, thus determining where the person resides. We consider the zoning-based localization problem in indoor environments where the aim is to find the zone of the sensor instead of its exact position. In fact, the determination of the zone where the sensor resides is sufficient for localization of elderly people. Once the zone of the sensor is determined, it is then easy to find the elderly person carrying it and assist him. It is noteworthy that large zones can be divided as convenient.

Zoning-based localization can be tackled by finding the exact locations of sensors and then determine their zones using their estimated locations. Many existing works have considered the exact localization problem. Indeed, the integration of a GPS-GSM system into sensors is widely used in outdoor tracking systems [Al-Khedher, 2011]. However, it has limitations in indoor environments [Oshin *et al.*, 2012]. For that reason, alternative solutions have been proposed. We consider the following approach. At first, two types of sensors are defined; anchor nodes (ANs), also called beacon nodes, of known positions, and non-anchor nodes of unknown positions, to be localized. Since we consider, in the general

setting, the case of moving sensors, we will refer to the non-anchor nodes by mobile nodes (MNs). The objective becomes to determine the position of any MN using collected measurements and information exchanged with the ANs. The remaining issue is to choose the appropriate type of signal and the measurement technique. This is extremely important as it plays a vital role in the accuracy of the localization algorithm.

Existing solutions consist in using signals that are available in closed environments, like ultra-wideband, WiFi, zigbee, Bluetooth, etc [Ahn & Yu, 2009; Bekkelien *et al.*, 2012; Disha, 2013; Liu *et al.*, 2007]. Typically, ANs broadcast signals in the network that are received by the MNs. The position is then estimated as a function of the received signals, using their strength, time of arrival, or any of the techniques described in Section 1.2.3. One of the advantages of WiFi signals over the others is its cost effectiveness due to the possibility to localize the position of almost every WiFi compatible device without installing any additional software. In addition, one can rely only on the Access Points (APs) present inside the building, with no additional hardware. Another advantage of using WLAN is that no line-of-sight (LoS) is required. The localization process consists then in finding the location of the MN according to the WiFi signals it collects from APs.

Several localization algorithms using Received Signal Strength Indicators (RSSIs) of WiFi signals have been developed. The RSSI-based techniques exploit the attenuation of the signal strength with the traveled distance to estimate the distances separating the APs from the MNs. They exhibit favorable properties with respect to power consumption, size, and cost, since no additional hardware is needed. Trilateration and connectivity-based methods have been applied for localization using distance estimation with the pathloss model. However, distance estimation using RSSI is really challenging, since the measurements of signals' powers can be significantly altered by the presence of additive noise, multipath fading, shadowing, and other interferences [Aruna *et al.*, 2015; Mourad *et al.*, 2009; Pak *et al.*, 2015; Sangthong *et al.*, 2012].

To overcome these issues, techniques that employ fingerprinting have been widely

implemented. They consist in collecting a database of exact reference locations, coupled to their corresponding WiFi signals strengths, received from the APs. Fingerprinting techniques are not sensitive to the propagation conditions, such as Non-line-of-sight (NLoS) and multipath. Indeed, they can be applied to any scenario and environment. They operate in two phases. In a preliminary phase, often called offline, the area of interest is discretized into cells and a database is built from the signal signatures, such as RSSIs. The database can be assembled using measured data or simulated data using a propagation model. In the online phase, the estimated signatures at each AP are compared to the database fingerprints. The advantage of such approach is that there is no need for a geometrical model that relates the signal strengths to the traveled distances. Instead, a radio-cartography is constructed by collecting measurements to cover the targeted area.

In a position-based localization approach, a huge database has to be constructed with collected measurements at exact reference positions covering the whole targeted area. This is time consuming especially if the targeted area is wide. Moreover, the WiFi signals are sensitive to the environmental conditions, which makes measurements at reference positions not highly reliable for exact localization. This also requires cumbersome learning models to achieve good accuracy. For that reason, fingerprints of WiFi RSSIs and zones numbers are collected and hence, the zoning problem can be addressed as a multi-class classification issue, to be resolved using one of the techniques described in Section 1.3.2. However, proposed approaches have not yet arrived to solve efficiently the localization problem [Sánchez-Rodríguez *et al.*, 2015]. For this reason, we aim to develop a localization algorithm that solves these issues and achieves a good accuracy.

In this chapter, we propose an original zoning-based localization technique that makes use of the belief functions theory (BFT) to combine evidence revealed at each AP. At the preliminary phase, the proposed method consists in constructing a fingerprinting database that associates to each zone a set of WiFi signals strengths collected from the APs. Each AP is then considered as a source of information and is used in the belief functions framework to set a mass function over the zones. In the online phase, once the MN is in an unknown zone, the

signals strengths are measured and the constructed mass functions are used to estimate the MN's zone. One advantage of this method is that it yields a set of possible solutions, sorted in a descent order of priority. The performance of the proposed approach is examined in a real experimental scenario, and is compared with other techniques.

## 3.2 Problem Formulation

The localization approach proposed in this chapter is centralized, thus the network has the topology illustrated in Figure 1.2(a). In such a topology, all collected data are transmitted to the central fusion station, where all processing and computations are conducted. Therefore, MNs only send and receive measurements, and do not perform any computation.

### 3.2.1 Network configuration

Consider an environment is divided into $N_Z$ zones denoted by $Z_j, j \in \{1, 2, \ldots, N_Z\}$. Having a MN moving within these zones, the objective of the method is to find instantly the zone where it resides. This is done by using the RSSIs of WiFi signals collected by the MN from the neighboring APs. Let $N_{AP}$ be the number of all available APs in the area of interest, denoted by $AP_k,\ k \in \{1, 2 \ldots, N_{AP}\}$. Let $\boldsymbol{\rho}_t$ be the vector of size $N_{AP}$ of RSSI measurements collected by the MN at the instant $t$ from all these APs,

$$\boldsymbol{\rho}_t = (\rho_{t,1}, \ldots, \rho_{t,N_{AP}}), \tag{3.1}$$

where $\rho_{t,k}$ is the RSSI of the signal with respect to $AP_k$ at instant $t$. Since not all APs are detected at each instant, we denote $I_{AP,t}$ the set of indices of the APs whose signals are detected by the MN at time $t$, and $\rho_{t,k}, k \in I_{AP,t}$, their measured RSSIs. The vector $\boldsymbol{\rho}_t$ is completed with zeros at positions where the APs are not detected. It is worth noting that only one MN is considered here, however the method can be applied in the same manner to as many MNs as needed. Table 3.1 lists the variables used in this chapter, along with their respective sizes.

### 3.2.2 Approach description

The aim of the proposed zoning-based localization algorithm is to find an observation model $\mathbb{O}$ that associates a confidence level to each zone for an input vector of RSSIs $\boldsymbol{\rho}_t$ at any instant $t$. This observation model is constructed in the offline phase, before the localization process, using collected data and a training process. The online phase follows later on once the model is identified and online measurements are collected.

#### 3.2.2.1 Offline phase

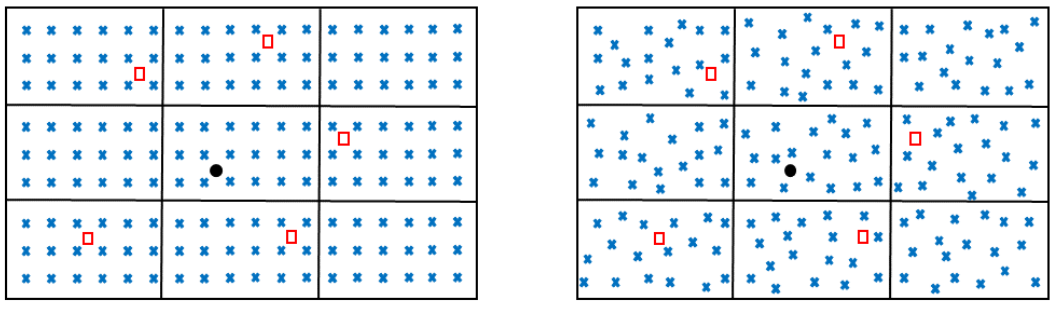
The offline phase starts with the fingerprinting process, that is a data collection in order to acquire a description of the environment and construct a database of measurements. The APs broadcast WiFi signals in the network. To construct the database, a MN moves freely in each zone of the targeted area and measures the RSSIs of WiFi signals from all APs. Suppose $\rho_{j,k,\ell}$ corresponds to the $\ell$-th RSSI measured inside zone $Z_j$ with respect to $AP_k$. Let $N_j$ be the number of RSSI measurements taken in zone $Z_j$. This implies that for a certain zone and a

**Table 3.1:** List of the variables used in Chapter 3, with their respective sizes.

| Notation | Variable | Size |
|---|---|---|
| $N_Z$ | number of zones | 1 |
| $N_{AP}$ | number of APs | 1 |
| $\rho$ | RSSI measurement | 1 |
| $N, N_D, o$ | number of RSSI measurements | 1 |
| $Z$ | MN's zone | 1 |
| $AP$ | Access Point | 1 |
| $Q(\cdot), \mathbb{I}(\cdot), \mathcal{K}(\cdot)$ | functions | 1 |
| $s$ | significance level | 1 |
| $\alpha, \epsilon$ | error rate | 1 |
| $m(\cdot), {}^{\alpha}m(\cdot), BetP(\cdot)$ | assigned evidence | 1 |
| $\mathcal{D}$ | database | $N_D \times N_{AP}$ |
| $\boldsymbol{\rho}$ | vector of RSSI measurement | $N_{AP} \times 1$ |
| $I_{AP,t}$ | indices of detected APs | $(\leq N_{AP}) \times 1$ |
| $\mathbb{O}(\cdot)$ | observation model | $N_Z \times 1$ |

given AP, a set of $N_j$ values is collected representing the variations of the RSSIs in this zone with respect to that AP. Let $N_D = \sum_{j=1}^{N_Z} N_j$ be the total number of measurements. A database $\mathcal{D}$ of $N_D \times N_{AP}$ RSSIs labeled to the zones is then obtained. This database describes the variability of the RSSIs within and between the zones. Figure 3.1 shows a grid of reference RSSI measurements collected in uniform and random distributions.

Afterwards, the observation model $\mathbb{O}$ is constructed using the collected database. This is a classification problem where the classifier takes the RSSIs vector as input and yields the zone as output. Moreover, the proposed model gives a degree of confidence of having the RSSI measured in each zone. By doing this, the model allows a ranking of the results and hence an order to follow in selecting zones if the first zone estimation is erroneous. Computations are done in the BFT framework as will be shown in details in Section 3.3. The proposed model permits combining several pieces of evidence, even if they are of different nature, as will be seen later on in Chapter 5. By combining different types of evidence, we aim to enhance the overall accuracy of the localization algorithm, detect an error generated by a sensor, and correct erroneous estimations.



(a) Uniform distribution. (b) Random distribution.

**Figure 3.1:** Illustration of fingerprinting configuration - × designates reference positions, □ designates WiFi Access Points, and • designates a MN.

#### 3.2.2.2 Online phase

In the online phase, also called localization phase, the observation model $\mathbb{O}$ is used to estimate the MN's zone. The MN to be localized measures a set of RSSIs from a certain number of APs, stores them in the vector $\boldsymbol{\rho}_t \in \mathbb{R}^{N_{AP}}$ and broadcasts $\boldsymbol{\rho}_t$ in the network. At the fusion center, the observation model $\mathbb{O}$ is applied to the vector $\boldsymbol{\rho}_t$ to instantly affiliate a confidence level to each zone of the targeted area,

$$\mathbb{O}(\boldsymbol{\rho}_t) = (m_{\mathbb{O},t}(Z_1), \ldots, m_{\mathbb{O},t}(Z_{N_Z})), \tag{3.2}$$

where $m_{\mathbb{O},t}(Z_j)$ is the level of confidence of having the MN of observation $\boldsymbol{\rho}_t$ residing in the zone $Z_j$ at the instant $t$. The zone having the highest confidence level is chosen, and considered to be the zone where the MN resides. Figure 3.2 illustrates the localization phase using the observation model $\mathbb{O}$.

## 3.3 Observation Model using the BFT

In this section, the objective is to determine the observation model $\mathbb{O} : \mathbb{R}^{N_{AP}} \rightarrow [0,1]^{N_Z}$. Let $\mathcal{Z} = \{Z_1, \ldots, Z_{N_Z}\}$ be the set of all possible zones and let $P(\mathcal{Z}) = 2^{\mathcal{Z}}$ be the set of all the subsets of $\mathcal{Z}$, i.e., $P(\mathcal{Z}) = \{\emptyset, \{Z_1\}, \ldots, \mathcal{Z}\}$. The empty set $\emptyset$ denotes impossible zone, which means that the MN resides outside $\mathcal{Z}$. The cardinal of $P(\mathcal{Z})$ is equal to $2^{|\mathcal{Z}|} = 2^{N_Z}$, where $|\mathcal{Z}|$ denotes the cardinal of $\mathcal{Z}$.

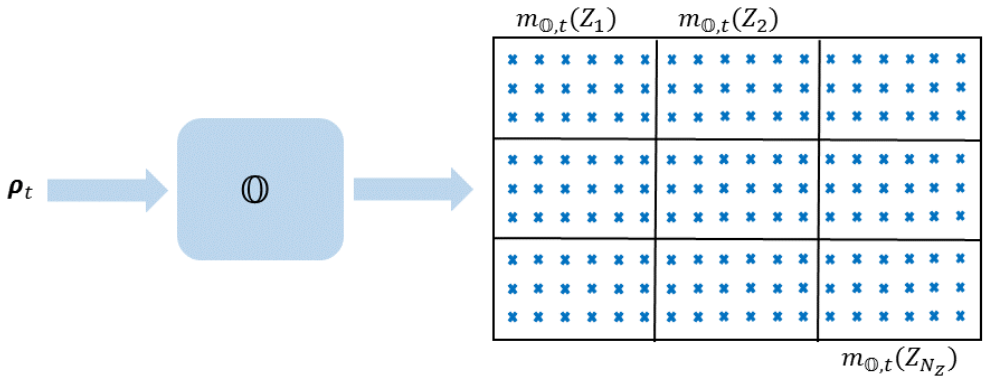



**Figure 3.2:** Illustration of the localization phase using the observation model $\mathbb{O}$.

The observation model is constructed as follows. At first, the RSSIs collected in the database $\mathcal{D}$ according to each AP are fitted into a distribution. We distinguish between two types of distributions, parametric and non-parametric. Supersets of single zones are also considered, and their RSSIs are also fitted. This allows us to take ambiguous information into consideration in a belief functions framework. This is important in our case especially that the used measurements are WiFi signals that are unstable and ambiguous, leading to uncertain estimations and decisions. The fitted distributions are then used to set mass functions over all the subsets of $P(\mathcal{Z})$. The APs, which are the sources of information, are discounted according to their error rate. Their evidence is then combined via the belief functions fusion rules, and a decision is taken by associating a confidence level to each zone. Taking uni-dimensional distributions according to single APs then combining their information allows the localization in case of missing data. This is common when using WiFi, as packets might be lost or not received by the MN. An additional advantage of the belief functions framework is that it allows combining evidence from different types of sensors. This is beneficial in correcting the estimations of the observation model. Moreover, it allows dynamic combination of evidence, and not restricted to only static cases. This is advantageous for the localization of MN as a function of time, where a previous evidence can be used as an information at a current instant.

### 3.3.1 Statistical representation of data

Having a set of $N_j \times N_{AP}$ observations $\rho_{j,k,\ell}, k \in \{1, \ldots, N_{AP}\}, \ell \in \{1, \ldots, N_j\}$, collected in zone $Z_j$, the aim of this section is to fit these observations to statistical distributions that represent the variation of the RSSIs inside the zone. Although a multi-dimensional distribution can be used for this purpose, we consider here the uni-dimensional case for the following reasons. [1] At first, the uni-dimensional distributions are easier for analysis and computations, especially when the number of APs is large. In addition, they allow considering the reliability or error rate of each AP to discount the assigned evidence. Moreover, uni-dimensional

[1] We will use the concept of multi-dimensional distribution in certain cases throughout the manuscript, which will be explicitly specified and explained.

distributions do not disable the process of localization when an AP, or more, are not detected for some reasons. The localization can still be performed by fusion of evidence of the detected APs only. The principle behind fitting data observations to distributions is to find the type of distribution and the values of its parameters that give the highest probability of producing the observed data. We distinguish between two types of distributions, parametric and non-parametric.

#### 3.3.1.1 Parametric modeling

Having the collected measurements of each zone in the offline phase, one approach is to parametrically model the data, by fitting them into one of the known parametric distributions [Delignette-Muller & Dutang, 2015]. Since we are dealing here with uni-dimensional distributions, we consider the set of $N_j$ observations $\rho_{j,k,\ell}, \ell \in \{1, \ldots, N_j\}$, collected in zone $Z_j$ with respect to the source $AP_k$. Then, we choose the types of distributions to be fitted. The set of candidate parametric distributions is listed in Table 3.2. Afterwards, we estimate the parameters of each considered distribution using the observations. Finally, we apply a statistical goodness of fit test to evaluate their fitting error. The problem is in the form of hypothesis testing where the null and alternative hypotheses are:

$H_0$: Sample data come from the stated distribution.

$H_a$: Sample data do not come from the stated distribution.

The Kolmogorov-Smirnov (K-S) test is used to test the hypotheses [Massey Jr, 1951]. For each considered distribution, the hypothesis $H_0$ is rejected at a significance level $s$ if the test statistic is greater than a critical value obtained from the K-S table [Facchinetti, 2009]. The significance level is chosen by convention, and can be set to 0.01, 0.02, or up to 0.05 if available distributions fail to fit with smaller levels. All the considered distributions are tested, and the accepted ones are ranked according to their statistics, the best fitting one being selected. This is done for each AP and for each zone, and also for each superset of single zones by using the union of their observations, thus obtaining the distributions $Q_{A,k}(\cdot), A \in P(\mathcal{Z}), k \in \{1, 2 \ldots, N_{AP}\}$. It is noteworthy that the observations of each subset of $P(\mathcal{Z})$ can be fitted to a different distribution type.

#### 3.3.1.2 Non-parametric modeling

The previously described parametric modeling approach is found to be effective when the RSSI measurements really follow a certain statistical law, yet it might fail to be representative elsewhere. When the assumptions of the parametric distribution fail, a more general non-parametric approach is required to estimate the probability density function of the measurements [Elgammal *et al.*, 2002]. One solution is to construct a histogram of the RSSIs [Moghtadaiee *et al.*, 2011]. However, this depends on the starting position of the bins and their number, and suffers from the curse of dimensionality as the number of bins grows exponentially with dimensions, thus making this solution unsuitable for most applications [Elgammal *et al.*, 2002]. For that reason, the kernel density estimation (KDE) is proposed to model the RSSI measurements. Figure 3.3 shows an example of real data RSSI measurements represented by their histogram, a parametric trial to fit them with a normal distribution, and a KDE with Gaussian kernel and

**Table 3.2:** List of parametric distributions.

| Distribution | pdf | Parameters |
|---|---|---|
| Beta | $\frac{1}{B(\alpha,\beta)}\rho^{\alpha-1}(1-\rho)^{\beta-1}$ | $\alpha, \beta$ |
| Exponential | $\lambda e^{-\lambda\rho}$ | $\lambda$ |
| Gamma | $\frac{1}{\Gamma(k)\theta^k}\rho^{k-1}e^{-\frac{\rho}{\theta}}$ | $k, \theta, \alpha, \beta$ |
| Generalized extreme value | $\frac{1}{\sigma}t(\rho)^{\xi+1}e^{-t(\rho)}$ | $\mu, \sigma, \xi$ |
| Generalized Pareto | $\frac{1}{\sigma}\left(1+\xi\frac{\rho-\mu}{\sigma}\right)^{-\left(\frac{1}{\xi}+1\right)}$ | $\mu, \sigma, \xi,$ |
| Inverse Gaussian | $\left(\frac{\lambda}{2\pi\rho^3}\right)^{\frac{1}{2}} e^{\frac{-\lambda(\rho-\mu)^2}{2\mu^2\rho}}$ | $\lambda, \mu$ |
| Logistic | $\frac{e^{-\frac{\rho-\mu}{s}}}{s\left(1+e^{-\frac{\rho-\mu}{s}}\right)^2}$ | $\mu, s$ |
| Log-logistic | $\frac{\frac{\beta}{\alpha}\left(\frac{\rho}{\alpha}\right)^{\beta-1}}{\left(1+\left(\frac{\rho}{\alpha}\right)^{\beta}\right)^2}$ | $\alpha, \beta$ |
| Lognormal | $\frac{1}{\rho}\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(\ln(\rho)-\mu)^2}{2\sigma^2}}$ | $\mu, \sigma$ |
| Nakagami | $\frac{2m^m}{\Gamma(m)\Omega^m}\rho^{2m-1}e^{-\frac{m}{\Omega}\rho^2}$ | $m, \Omega$ |
| Normal | $\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\rho-\mu)^2}{2\sigma^2}}$ | $\mu, \sigma$ |
| Rayleigh | $\frac{\rho}{\sigma^2}e^{-\frac{\rho^2}{2\sigma^2}}$ | $\sigma$ |
| Rician | $\frac{\rho}{\sigma^2}e^{-\frac{\rho^2+\nu^2}{2\sigma^2}} I_0\left(\frac{\rho\nu}{\sigma^2}\right)$ | $\nu, \sigma$ |
| t location-scale | $\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left[\frac{\nu+\left(\frac{\rho-\mu}{\sigma}\right)^2}{\nu}\right]^{-\left(\frac{\nu+1}{2}\right)}$ | $\mu, \sigma, \nu$ |
| Weibull | $\frac{k}{\lambda}\left(\frac{\rho}{\lambda}\right)^{k-1} e^{-\left(\frac{\rho}{\lambda}\right)^k}$ | $k, \lambda$ |

bandwidth $h = 1.6$. This is an example of failure of parametric modeling to represent the variations of RSSIs, and the ability of non-parametric modeling to better represent them.

Consider the computation of a uni-dimensional non-parametric distribution of observations of zone $Z_j$ according to a single source $AP_k$. For an observed RSSI $v$, let $\mathcal{S}_{j,k}(v)$ be a square with side $h_{j,k}$, centered on $v$, enclosing $o_{j,k}(v)$ RSSI measurements from the database. To find the number $o_{j,k}(\cdot)$ of measurements falling within $\mathcal{S}_{j,k}(\cdot)$ according to $AP_k$ for any uni- or multi-dimensional $v$, we consider the indicator function $\mathbb{I}(u)$ defined as,

$$\mathbb{I}(u) = \begin{cases} 1, & \text{if } |u| < \frac{1}{2}; \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

This function is known as a naive estimator. The quantity $\mathbb{I}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right)$ is then equal to unity if $\rho_{j,k,\ell}$ is inside $\mathcal{S}_{j,k}(\cdot)$ or 0 otherwise. The number of measurements within $\mathcal{S}_{j,k}(\cdot)$, centered on $(\cdot)$, is then computed as follows,

$$o_{j,k}(\cdot) = \sum_{\ell=1}^{N_j} \mathbb{I}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right). \tag{3.4}$$

The kernel density estimate $Q_{\{Z_j\},k}(\cdot)$ is obtained,

$$Q_{\{Z_j\},k}(\cdot) = \frac{o_{j,k}(\cdot)}{N_j \times h_{j,k}}. \tag{3.5}$$

Then, by substituting equation (3.4) in equation (3.5), we obtain,

$$Q_{\{Z_j\},k}(\cdot) = \frac{1}{N_j \times h_{j,k}} \sum_{\ell=1}^{N_j} \mathbb{I}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right). \tag{3.6}$$

This model solves the problem of bins locations of the histogram. However, the resulting density is bumpy, yielding discontinuous density estimates. Instead of assigning equal weights to all neighboring observations, the naive estimator is replaced by a smoother kernel $\mathcal{K}(u)$, such as Epanechnikov or Gaussian kernels. A list of kernel functions that can be used is provided in Table 3.3.

The kernel density estimate is then given by

$$Q_{\{Z_j\},k}(\cdot) = \frac{1}{N_j \times h_{j,k}} \sum_{\ell=1}^{N_j} \mathcal{K}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right). \tag{3.7}$$

These expressions can be easily applied for the estimation of the distribution $Q_{A,k}(\cdot)$ of observations of any set $A \in \mathcal{Z}$ having more than one single zone, by using the union of their observations as follows,

$$Q_{A,k}(\cdot) = \frac{1}{\sum_{j,Z_j \in A} N_j \times h_{A,k}} \sum_{\substack{j \\ Z_j \in A}} \sum_{\ell=1}^{N_j} \mathcal{K}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right). \tag{3.8}$$

Since the shape of the kernel has a small effect on the model [Eckert-Gallup & Martin, 2016], the Gaussian kernel is often considered due to the facility of its analytical derivations,

$$\mathcal{K}(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}. \tag{3.9}$$

**Table 3.3:** List of kernel functions for kernel density estimation.

| Kernel shape | Kernel function, $\mathcal{K}(u)$ |
|---|---|
| Uniform | $\frac{1}{2}$ |
| Triangular | $1 - \lvert u \rvert$ |
| Epanechnikov | $\frac{3}{4}\left(1 - u^2\right)$ |
| Quadratic | $\frac{15}{16}(1 - u^2)^2$ |
| Triweight | $\frac{35}{32}(1 - u^2)^3$ |
| Tricube | $\frac{70}{81}(1 - \lvert u \rvert^3)^3$ |
| Gaussian | $\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}$ |
| Cosine | $\frac{\pi}{4} \cos\left(\frac{\pi}{2}u\right)$ |
| Logistic | $\frac{1}{e^u + 2 + e^{-u}}$ |
| Sigmoid function | $\frac{2}{\pi} \frac{1}{e^u + e^{-u}}$ |
| Silverman kernel | $\frac{1}{2} e^{-\frac{\lvert u \rvert}{\sqrt{2}}} \sin\left(\frac{\lvert u \rvert}{\sqrt{2}} + \frac{\pi}{4}\right)$ |

In fact all these kernels have a comparable efficiency that is close to 100%, with the efficiency *eff* defined as,

$$eff = \sqrt{\int u^2 \mathcal{K}(u) du} \int \mathcal{K}(u)^2 du. \tag{3.10}$$

Figure 3.4(a) demonstrates the influence of the kernel shape on the fitting of the KDE. A Gaussian, a triangular, and an Epanechnikov kernels, all of bandwidth $h = 1.6$, are presented. As the figure clearly displays, there is no significant difference between the fitting distributions.

The problem is then to determine the bandwidth, or the smoothing parameter $h$. In Figure 3.4(b), three different values of the bandwidth of the Gaussian kernel are considered. As the figure shows, a small value of the bandwidth overfits the data and makes them hard to interpret, while a large value over-smooths the KDE and masks the structure of the data. A practical approach to estimate $h$, proposed by Habbema & Hermans [1977], is to maximize the pseudo-likelihood leave-one-out cross validation. If we consider the distribution of the observations of a zone $Z_j$ according to $AP_k$, the bandwidth will then be $h_{j,k} = \text{argmax}_h \, ML_{j,k}(h)$, where the quantity $ML_{j,k}(h)$ is computed as,

$$ML_{j,k}(h) = \frac{1}{N_j} \sum_{\ell=1}^{N_j} \log \left[ \sum_{\ell' \neq \ell} \mathcal{K} \left( \frac{\rho_{j,k,\ell'} - \rho_{j,k,\ell}}{h} \right) \right] - \log[(N_j - 1)h]. \tag{3.11}$$

It has been proven in literature that this criterion leads to a KDE that is the closest to the true model [Hall, 1982].

### 3.3.2 Mass assignment

The observation model consists in using the fitted RSSI distributions in the BFT as a framework for mass association and evidence fusion. As described in Chapter 2, a fundamental function of the BFT is the mass function, or BBA. A mass function $m_{AP_k,t}(\cdot)$ is a mapping from $P(\mathcal{Z})$ to the interval $[0, 1]$, defined according to a certain source $AP_k, k \in \{1, \ldots, N_{AP}\}$, and satisfying

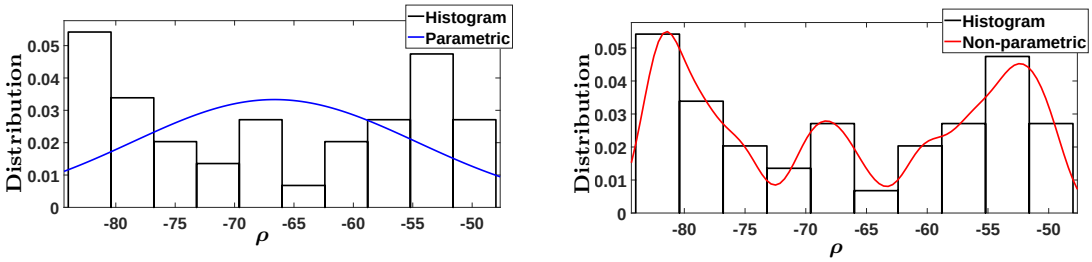
$$\sum_{A \in P(\mathcal{Z})} m_{AP_k,t}(A) = 1. \tag{3.12}$$

The mass $m_{AP_k,t}(A)$ given to $A \in P(\mathcal{Z})$ stands for the proportion of evidence, brought by the source $AP_k$ at instant $t$, saying that the observed variable belongs to $A$.

The objective is to define the APs BBAs, using the fitted distributions either parametrically, or non-parametrically. The distribution $Q_{A,k}(\cdot)$ represents, either parametrically or non-parametrically, the variations of the RSSIs in subset $A$ with respect to $AP_k$. Then, having an observation $\rho_{t,k}$ related to $AP_k, k \in \{1, \ldots, N_{AP}\}$, the mass $m_{AP_k,t}(A)$ is computed as follows,

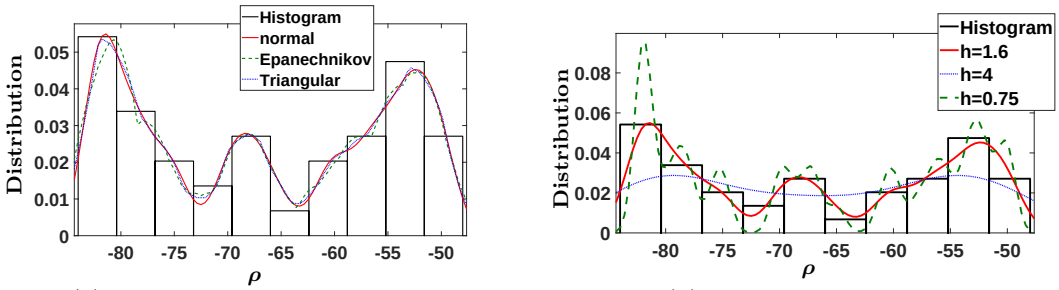
$$m_{AP_k,t}(A) = \frac{Q_{A,k}(\rho_{t,k})}{\sum_{A' \in P(\mathcal{Z}), A' \neq \emptyset} Q_{A',k}(\rho_{t,k})}, \quad A \in P(\mathcal{Z}), A \neq \emptyset. \tag{3.13}$$

In this work, we assume that $\mathcal{Z}$ covers all possible zones, that is, the node cannot be outside $\mathcal{Z}$. This means that $m_{AP_k,t}(\emptyset) = 0$, for all $AP_k$, at any time $t$. By taking all the subsets and not only the singletons, the proposed algorithm uses




(a) Parametric distribution. (b) Non-parametric distribution.

**Figure 3.3:** Fitting of parametric normal distribution in (a), and a KDE of Gaussian kernel in (b), of real data RSSIs of histogram in black.




(a) Different kernel functions. (b) Different bandwidths.

**Figure 3.4:** Influence of kernel shape and bandwidth parameter on KDE fitting.

all available pieces of evidence, even if they are uncertain about a single element. Note that $m_{AP_k,t}(A)$ is not the probability of having $\rho_{t,k}$ in $A$, but only an interpretation of the information brought by the source $AP_k$ by means of observation $\rho_{t,k}$, that is, $m_{AP_k,t}(A)$ can be higher than $m_{AP_k,t}(B)$ even if $A \subset B$.

**Ex 3.1.** Consider an area constituted of two zones, such that $\mathcal{Z} = \{Z_1, Z_2\}$. The set $\mathcal{Z}$ has three non-empty subsets $\{Z_1\}$, $\{Z_2\}$, and $\{Z_1, Z_2\}$, represented by their distributions in Figure 3.5. Consider two instant observations $\rho_t^{(1)}$ and $\rho_t^{(2)}$. The first is closer to the RSSI values of $Z_1$ and the second has its value in the middle of those of $Z_1$ and $Z_2$. By using equation (3.13), the observation $\rho_t^{(1)}$ is more likely to be of the entity $Z_1$ with the mass of $\{Z_1\}$ being higher than those of $\{Z_2\}$ and $\{Z_1, Z_2\}$. With respect to the observation $\rho_t^{(2)}$, the distributions of $Z_1$ and $Z_2$ are too close, leading to similar masses to both. Here, by considering the subset $\{Z_1, Z_2\}$ within the possibilities, a higher mass is associated to $\{Z_1, Z_2\}$ without dissociation, which represents better the ambiguity by describing rationally the observed evidence. The mass assigned to $\{Z_1, Z_2\}$ is higher than those of singletons only for observations where the distributions of singletons are two close, to avoid erroneous assignments and take advantage even of ambiguous data.

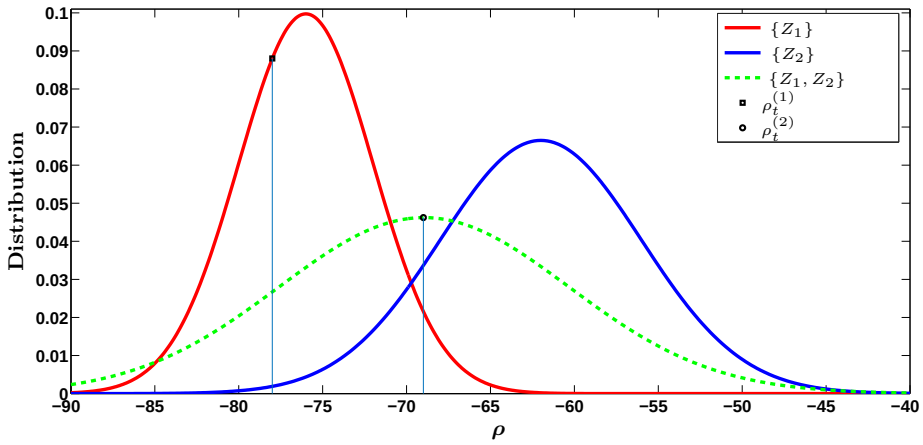



**Figure 3.5:** An example of mass assignments of some observations.

### 3.3.3 Discounting operation

The detected APs are not completely reliable. Indeed, each AP might yield an erroneous interpretation of evidence for some observations. This is due to the statistical modeling of the observations, based on their occurrence in the database. Another reason is the nature of the WiFi signals that are unstable and vary widely with various parameters. In order to correct this, one can discount the BBAs of equation (3.13) by taking into account the error rate of the AP. We discuss in the following two approaches to discount the evidence assigned by the APs, classical discounting and contextual discounting.

#### 3.3.3.1 Classical discounting

The reliability of a source is classically taken into account by the discounting operation, which transforms the supporting function into a weaker, less informative one. The discounted BBA ${}^{\alpha}m_{AP_k,t}(\cdot)$ related to $AP_k$ having an error rate $\alpha_k$ is deduced from the BBA $m_{AP_k,t}(\cdot)$ as follows [Mercier *et al.*, 2012],

$${}^{\alpha}m_{AP_k,t}(A) = \begin{cases} (1-\alpha_k)m_{AP_k,t}(A), & \text{if } A \in 2^{\mathcal{Z}},\ A \neq \mathcal{Z}; \\ \alpha_k + (1-\alpha_k)m_{AP_k,t}(A), & \text{if } A = \mathcal{Z}. \end{cases} \tag{3.14}$$

By doing this, the amounts of evidence given to the subsets of $\mathcal{Z}$ are reduced, and the remaining evidence is given to the whole set $\mathcal{Z}$.

To compute the error rate of a certain source $AP_k$, consider an observation $\rho_{\cdot,k}$ being truly in $A$. The source $AP_k$ is assumed not reliable if, according to $\rho_{\cdot,k}$, it associates more evidence to any subset other than $A$. Since the BBAs are defined using the statistical distributions related to each subset, then an AP is erroneous for all observations of $A$ when $Q_{A,k}(\rho_{\cdot,k})$ is less than any $Q_{A',k}(\rho_{\cdot,k})$, for any $A' \neq A$. Let $\epsilon_k(A)$ be the error rate related to $A$ with respect to $AP_k$. Then,

$$\epsilon_k(A) = \int_{\mathbb{D}_{A,k}} Q_{A,k}(\rho) d\rho, \tag{3.15}$$

such that $\mathbb{D}_{A,k}$ is the domain of error of subset $A$ according to $AP_k$, defined as,

$$\mathbb{D}_{A,k} = \{\rho \mid Q_{A,k}(\rho) \leq \max_{A' \in P(\mathcal{Z}), A' \neq A} (Q_{A',k}(\rho))\}. \tag{3.16}$$

The error rate $\alpha_k$ is then the average error of all subsets according to $AP_k$,

$$\alpha_k = \frac{\sum_{A \in P(\mathcal{Z})} \epsilon_k(A)}{|P(\mathcal{Z})|}. \tag{3.17}$$

The described discounting approach requires the calculation of integrals, which might be computationally expensive. Alternatively, the error can be empirically computed, by realizing experiments and recording the number of incorrect subset estimations of each AP. The error rate $\alpha_k$ is then the percentage of incorrect estimations.

#### 3.3.3.2 Contextual discounting

The described classical discounting approach assumes that each AP has an equal error rate with respect to all subsets. However, this is not always the case in practice, since an AP has a certain reliability regarding each subset. An AP will be more reliable to distinguish asymmetrical zones or areas, than symmetrical ones for instance. This is because in the latter case, the areas are more likely to have equal signals strengths, making them indistinguishable and thus increasing the error rate. For that reason, we consider here a contextual discounting approach to take into account the APs' reliability.

Let $\mathcal{A} = \{A_1, \ldots, A_L\}$ be a coarsening of $\mathcal{Z}$, which means that $A_1, \ldots, A_L$ form a partition of $\mathcal{Z}$. In this contextual model, we consider the degree of reliability of an AP conditionally on each subset $A_l, l \in \{1, \ldots, L\}$. For all $l \in 1, \ldots, L$, $\beta_k^l = 1 - \alpha_k^l$ represents the degree of reliability of $AP_k$ knowing that the observation belongs to $A_l$. Here, the considered partition is the set of single zones $\{\{Z_1\}, \ldots, \{Z_{N_Z}\}\}$, and thus the reliability of $AP_k$ with respect to the zone, or context, $Z_j$ will be $\beta_k^j$. As explained in Section 2.5.3.2, computing the contextual discounting consists in using its expression through the disjunctive rule of combination, leading to ${}^{\alpha}m_{AP_k}(A)$ given by

$${}^{\alpha}m_{AP_k}(A) = m_{AP_k,t} \textcircled{\cup} m^0_{AP_k,t}(A), \tag{3.18}$$

such that $m^0_{AP_k,t}(A)$ is defined as follows,

$$m^0_{AP_k,t}(A) = m^1_{AP_k,t} \textcircled{\cup} m^2_{AP_k,t} \textcircled{\cup} \ldots \textcircled{\cup} m^{N_Z}_{AP_k,t}(A), \tag{3.19}$$

where each $m^j_{AP_k,t}, j \in \{1,\ldots,N_Z\}$, is defined as,

$$m^j_{AP_k,t} = \begin{cases} (1-\alpha^j_k), & \text{if } A = \emptyset; \\ \alpha^j_k, & \text{if } A = A_l; \\ 0, & \text{otherwise.} \end{cases} \tag{3.20}$$

The error rate $\alpha^j_k(A)$ of subset $A$ such that the truth is $Z_j$ with respect to $AP_k$ is computed as,

$$\alpha^j_k(A) = \int_{\mathbb{D}_{A,k}} Q_{A,k}(\rho)d\rho, \tag{3.21}$$

such that $\mathbb{D}_{A,k}$ is the domain of error of subset $A$ according to $AP_k$, defined as,

$$\mathbb{D}_{A,k} = \{\rho \mid Q_{\{Z_j\},k}(\rho) \leq \max_{A' \in P(\mathbb{Z}), A' \neq A} (Q_{A',k}(\rho))\}. \tag{3.22}$$

However, computing these multi-dimensional integrals might be computationally expensive. For that reason, the reliability rate $\beta^j_k = 1 - \alpha^j_k$, of $AP_k$, is obtained by finding the percentage of correct subset determination such that the truth is $Z_j$. To this end, we construct a confusion matrix that describes the performance of the AP on a set of $N^l$ measurements to be tested in each zone. A confusion matrix $\mathcal{C} = c_{mn}, m \in \{1,\ldots,N_Z\}$ and $n \in \{1,\ldots,N_Z\}$, is a table where each line $m$ corresponds to a decision in favor to $Z_m$, and each column $n$ corresponds to the case where the truth is $Z_n$. The general term $c_{mn}$ is equal to the number of tested measurements of $Z_n$ that have been assigned to $Z_m$ by $AP_k$. The reliability rate is the percentage of correct estimations, computed as $\beta^m_k = \frac{c_{mn}}{N^n}$. The error rate is thus $\alpha^m_k = 1 - \frac{c_{mn}}{N^n}$.

### 3.3.4 Fusion of evidence

The mass functions ${}^{\alpha}m_{AP_k,t}(\cdot)$ are defined according to the RSSI vector $\rho_{t,k}, k \in I_{AP,t}$ retrieved from a certain number of APs. Combining the evidence consists in aggregating the information coming from all the APs [Kurdej & Cherfaoui, 2013]. The mass functions can then be combined using any of the methods described in Section 2.5.2. If there is a prior knowledge that the APs are reliable, and that the conflict between them is not large, the Dempster's rule of combination can

be considered, and thus the combined mass is computed as follows,

$$m_{\bigoplus,t}(A) = \frac{\sum_{\substack{A^{(k)}\in P(\mathcal{Z}) \\ \cap_k A^{(k)}=A}} \prod_{k\in I_{AP,t}} {}^{\alpha}m_{AP_k,t}(A^{(k)})}{1-\sum_{\substack{A^{(k)}\in P(\mathcal{Z}) \\ \cap_k A^{(k)}=\emptyset}} \prod_{k\in I_{AP,t}} {}^{\alpha}m_{AP_k,t}(A^{(k)})}, \tag{3.23}$$

for all the subsets $A \in P(\mathcal{Z})$, where $A^{(k)}$ is the subset $A$ with respect to the source $AP_k$. This fusion rule leads to a more informative and specialized mass function [Shafer, 1976]. The conjunctive rule can be also applied by considering only the numerator of equation (3.23), avoiding the normalization factor.

However, if the APs are conflicting, the previous rules generate counter-intuitive results. For that reason, we consider the disjunctive rule to combine the obtained evidence. By using the disjunctive rule, it is enough that at least one AP is reliable to acquire logical results. Therefore, the aggregated mass attributed to each subset $A$ is computed as,

$$m_{\bigcirc\!\!\!\!\cup,t}(A) = \sum_{\substack{A^{(k)}\in P(\mathcal{Z}) \\ \cup_k A^{(k)}=A}} \prod_{k\in I_{AP,t}} {}^{\alpha}m_{AP_k,t}(A^{(k)}), \tag{3.24}$$

Since the union $\cup_k A^{(k)}$ is never empty unless all the subsets $A^{(k)}$ are empty, there is no conflict resulting from the disjunctive rule of combination and therefore there is no need for normalization.

### 3.3.5 Confidence-based zone estimation

The computed mass function or BBA consists in an interpretation of the information brought by the observations at a given time $t$. It is a kind of belief or evidence, which is not a probability measure. An adequate notion of the BFT to attribute a confidence level to singleton sets is the pignistic level [Smets, 1993a]. Smets [1993b] argues that in order to make decisions, the belief represented by the BBA and held at the credal level, must induce a probability function at the pignistic level, as explained in Section 2.5.4. This is known as the pignistic transformation. It is defined as follows,

$$BetP_t(A) = \sum_{A\subseteq A'} \frac{m_{\bigoplus,t}(A')}{|A'|}, \tag{3.25}$$

where $A$ is a singleton of $P(\mathcal{Z})$. The mass obtained by Dempster's rule is shown in equation (3.25), but the conjunctive and disjunctive rules can be equivalently used. This probability function is used to make decisions using expected utilities theory. Its justification is based on rationality requirements and detailed by Smets & Kennes [1994]. The pignistic level is equivalent to the probability of having the observation belonging to the considered subset. One can also compute the pignistic level of higher-cardinal subsets. However, only the singletons are taken into consideration, as we are interested in determining a level of confidence for the original zones only. Hence, the level of confidence associated to each zone by the basic observation model at each instant $t$ can be computed as follows,

$$m_{\mathbb{O},t}(Z_j) = BetP_t(\{Z_j\}), j \in \{1, \ldots, N_Z\}. \tag{3.26}$$

The observation model $\mathbb{O}$ is therefore deduced,

$$\mathbb{O}(\boldsymbol{\rho}_t) = (m_{\mathbb{O},t}(Z_1), \ldots, m_{\mathbb{O},t}(Z_{N_Z})), \tag{3.27}$$

The zone having the highest confidence is thus selected. We also obtain a sorted list of zones to be used if needed. Figure 3.6 illustrates the different steps to construct the observation model $\mathbb{O}$.

**Ex 3.2.** We consider here a numerical example, assuming an area constituted of three zones $Z_1$, $Z_2$, and $Z_3$. Suppose we receive a new RSSI measurement belonging in reality to $Z_2$, and the following evidence, $m(Z_1) = 0.25$, $m(Z_2) = 0.2$, and $m(Z_3) = 0.02$ is assigned. In this case, we select $Z_1$ as the associated zone since $m(Z_1) > m(Z_2)$ even after normalizing the obtained masses. Whereas taking into account all available evidence, $m(\{Z_1, Z_2\}) = 0.2$, $m(\{Z_1, Z_3\}) =$

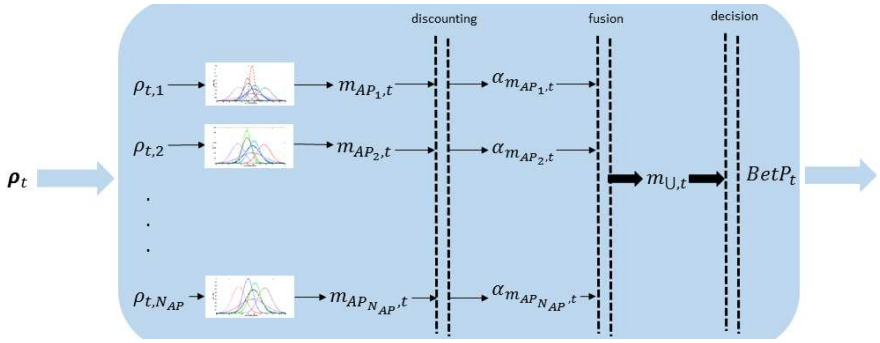



**Figure 3.6:** The block diagram of the observation model $\mathbb{O}$.

0.02, $m(\{Z_2, Z_3\}) = 0.3$, and $m(\{Z_1, Z_2, Z_3\}) = 0.01$, we get a confidence of $m_{\mathbb{O}}(Z_1) = 0.363$, $m_{\mathbb{O}}(Z_2) = 0.453$, and $m_{\mathbb{O}}(Z_3) = 0.183$. As we can see, the confidence level of $Z_2$ is now greater, and hence $Z_2$ is chosen. The computation of these confidence levels by the observation model is done by using the pignistic transformation. Hence, using the BFT to pass from the attributed masses to the confidence level through considering all evidence assigned to all subsets of $P(\mathcal{Z})$, enhances the decision making process.

## 3.4 Experimental Results

To evaluate the performance of the proposed method, real experiments were conducted in a sector of the University of Technology of Troyes, France. In the following, the experimental setup is first introduced. The illustration of the proposed method and its comparison to other techniques are shown afterwards.

### 3.4.1 Experimental setup

The real experiments are realized in a WLAN environment at the first floor of the statistical and operational research department at the University of Technology of Troyes, France. As shown in Figure 3.7, the considered sector of approximated area of 190 $m^2$ is partitioned into six offices, from both sides of a corridor that is divided into two zones, according to its architecture. This leads to eight zones in the considered area. A personal computer, with a "WiFi Scanner" software, distinguishes the APs of the network throughout their MAC addresses. It measures then the RSSIs of their transmitted signals. We used six of the APs detected at the considered area. A set of 30 measurements is taken in each zone, of which some are randomly used to construct the databases, and the others are kept for test and validation. The measurements are taken in random positions and orientations of the personal computer. Computations are conducted on version 7.11.2(R2010B) of Matlab on laptop with Microsoft Windows 7 and Intel Core i7 CPU. Table 3.4 summarizes the experimental setup parameters.

### 3.4.2 Illustration of the proposed method

In order to illustrate the proposed method, 70% of the collected RSSIs are first randomly selected at each zone to construct the database, which are 21 measurements, keeping 9 for testing per zone. Then, the RSSIs database is used to compute the mass functions by statistically fitting the parametric distributions listed in Table 3.2 and ranking the results. The Gaussian distribution was found to be the best fit for these zones. This leads to six graphs, each corresponding to an AP. Within each graph, eight functions represent the variations of RSSIs in the eight zones. Figure 3.8 represents the Gaussian distributions of the eight zones with respect to the first AP. As the figure shows, the overlapping between the functions representing the variations of the RSSIs in each zone is wide and hence, the zones' masses can be easily miscalculated considering only one AP. However, when considering all the APs and using the BFT framework to combine evidence,

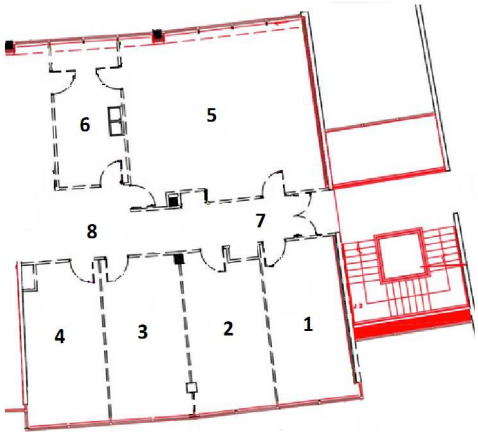



**Figure 3.7:** A sector of the first floor of the statistical and operational research department of the University of Technology of Troyes.

**Table 3.4:** Experimental setup parameters.

| Parameter | Notation | Value |
|---|---|---|
| Number of zones | $N_Z$ | 8 |
| Number of APs | $N_{AP}$ | 6 |
| Number of measurements per zone | $N_j$ | 30 |

we acquire interesting results.

Having 70% of the collected RSSIs in the database, we test the proposed method using Dempster's rule of combination and contextual discounting, localizing the 30% remaining measurements, that is 9 test points per zone. Table 3.5 shows the number of incorrect estimated zones if only the zone having the highest confidence level (1st choice) is considered and also in case the zone having the second highest confidence level (+2nd choice) is considered as well. The table shows also the total number of erroneous estimations over the 72 total test points in both cases. The results are encouraging, with an accuracy of 91.67% and thus 8.33% of erroneous estimations if the first choice is only considered. This is totally corrected if the second choice zone is also selected.

#### 3.4.2.1 Influence of discounting and combination

Two important concepts for zoning-based localization in a belief functions framework are the discounting and the combination of APs' evidence. In this paragraph, we measure the influence of the discounting technique, classical or contextual, and the combination rule, Dempster's, conjunctive, or disjunctive, on the performance of the proposed approach. Table 3.6 indicates the overall accuracy of the proposed method, obtained by the various combinations of discounting techniques and combination rules. As the table shows, the contextual discounting carries an enhancement of 3 to 4% as compared with the classical discounting.

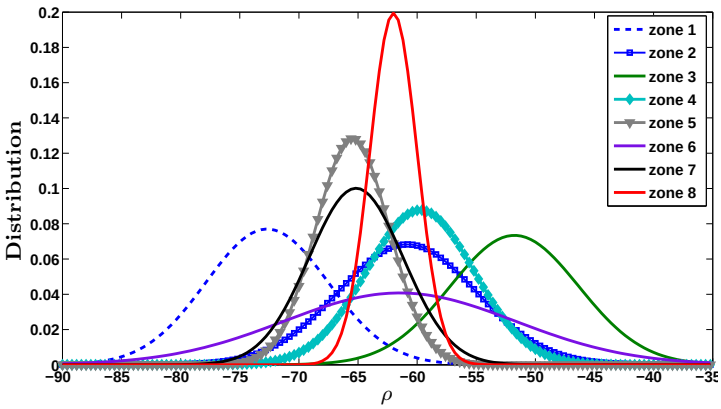



**Figure 3.8:** Gaussian functions of the eight zones with respect to the first AP.

This is due to considering the reliability of the APs per area or zone, which is important in the case of localization using RSSIs as aforementioned. On the other hand, a slight advantage to the Dempster's rule is recorded when evaluating the combination rules. This is because the used APs are found to be not highly conflicting. In fact, each pair of the used APs corresponds to the same physical AP, but on a different network. In addition, the three physical APs are installed at a centric position from the targeted area, providing similar information. By having non-conflicting APs, the Dempster's rule results in a more informative evidence assigned to the zones and their subsets.

**Table 3.5:** Number of incorrect estimated zones.

| Zone number | 1st choice | +2nd choice |
|---|---|---|
| 1 | 0 | – |
| 2 | 1 | 0 |
| 3 | 2 | 0 |
| 4 | 0 | – |
| 5 | 0 | – |
| 6 | 1 | 0 |
| 7 | 1 | 0 |
| 8 | 1 | – |
| Total | 6 | 0 |

**Table 3.6:** Influence of the discounting techniques and the combination rules on the overall accuracy (%) of the 1st zone choice.

| Combination rule | Discounting | |
|---|---|---|
| | Classical | Contextual |
| Dempster's | 87.50 | **91.67** |
| Conjunctive | 87.50 | 90.27 |
| Disjunctive | 84.72 | 88.89 |

#### 3.4.2.2 Influence of number of reference positions

In order to measure the performance of the method under low data availability, the ratio of the measurements used to construct the database is reduced to 60%, then to 50%, while testing over the other 40% and 50%, respectively. Table 3.7 shows the number of errors as a function of the ratio of the training-test database. As expected, the percentage of erroneous estimations increases with the 1st choice zone when the constructed database is reduced to 60%, then to 50%, leading to 11.46% of erroneous estimations in the former case and 14.17% in the latter. It is noticed that 2 out of the 120 measurements are not recovered by the second choice, rather by the third one. These erroneous points, located in zones 7 and 8, are estimated to be in zones 5 and 6 respectively as a 1st choice then in zones 2 and 3 as a 2nd choice. This is due to the insufficient built database that made it difficult to represent the variations of the small zones 7 and 8 as compared with the neighboring zones in the best possible way. The zones 7 and 8 constitute the corridor between the zones, and the estimated zones are adjacent and close to their actual positions.

#### 3.4.2.3 Influence of modeling and reference positions

As discussed in Section 3.3.1.2, the parametric modeling might not be always appropriate to represent the variations of the observations in a certain zone with respect to some AP. For that reason, we measure the influence of a non-parametric modeling or a KDE on the performance of the proposed approach in this paragraph. Table 3.8 provides a comparison of the performance of the proposed approach between a parametric and a KDE modeling. As the table shows, the

**Table 3.7:** Influence of number of reference positions on the number of errors.

| Ratio of training-test | | Number of errors | Total test points |
|---|---|---|---|
| 70% – 30% | 1st choice | **6** | 72 |
| | 2nd choice | 0 | 72 |
| 60% – 40% | 1st choice | 11 | 96 |
| | 2nd choice | 0 | 96 |
| 50% – 50% | 1st choice | 17 | 120 |
| | 2nd choice | 2 | 120 |

KDE does not have a huge influence on the overall accuracy. This is more clearly seen in a uniform distribution of reference position setting, where the variations of the RSSI measurements are found to follow a Gaussian distribution at a better significance level than in the case of a random distribution setting. It is worthy noting that this is not always true where, as will be seen later, the KDE strongly outperforms the parametric modeling in terms of accuracy.

As a result, it is important to try first a parametric modeling using the distributions of Table 3.2 and verify if the obtained results are satisfactory. If not, a KDE can be adopted using one of the kernel shapes indicated in Table 3.3, and computing the bandwidth through equation (3.11). In the cases where the observations closely follow a parametric distribution, the KDE is not worthy using as it adds unnecessary complexity, while it can be of significant importance if the conditions of the parametric distributions were not satisfied.

We now consider a random distribution of the reference positions instead of a uniform grid. An illustration of such distribution is shown in Figure 3.1(b). Table 3.8 shows the overall accuracy of the proposed approach in case of random distribution of reference positions, considering both parametric and non-parametric modeling. Compared with the results obtained when a uniform distribution of reference positions is considered, one can see that the overall accuracy decreases with the use of random distributions. This can be explained by the fact that a uniform grid allows a better coverage of the region of interest, while a random

**Table 3.8:** Influence of type of modeling and distribution of reference positions on the overall accuracy (%).

| Technique | | Type of modeling | |
|---|---|---|---|
| | | Parametric | KDE |
| Uniform | 1st choice | **91.67** | **91.67** |
| | 2nd choice | 100.00 | 100.00 |
| Random | 1st choice | 87.50 | **90.28** |
| | 2nd choice | 95.83 | 97.22 |

distribution does not always guarantee a good coverage of the region. Nevertheless, the results are still satisfactory, and random distributions can still be used for accurate localization when uniform grids are not applicable.

#### 3.4.2.4 Influence of number of zones

To study the influence of the number of zones $N_Z$ on the performance of the proposed approach, we consider the whole floor of the sector, as shown in Figure 3.9. The new considered floor has an area of 500 $m^2$ and is constituted of $N_Z = 21$ zones. In the new experimental setup, the overall accuracy of the method falls to 77.78%, considering a uniform distribution of reference positions, a $70\% - 30\%$ database split, and parametric modeling. This is due to the inability of the proposed observation model to assign discriminating evidence to the widely overlapping mass functions representing the different zones.

### 3.4.3 Comparison to other classification techniques

Since we are tackling the problem of zoning localization as a multi-class classification, we compare in this section the proposed method with two well-known classification techniques that are the naive Bayes (NB) and the Multinominal Logistic Regression (MLR). The NB classifiers are among the simplest probabilistic classifiers that assume independency between features to release probabilistic output [Liu *et al.*, 2013a]. The MLR is a natural extension of binary logistic regression to multi-class classification problems [Mauša *et al.*, 2012]. Both methods

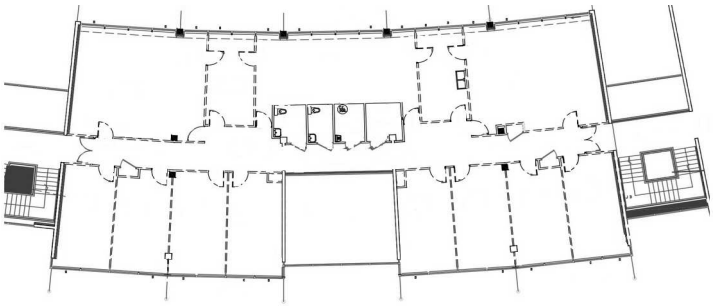


**Figure 3.9:** The first floor of the statistical and operational research department at the University of Technology of Troyes, France.

use the maximum likelihood estimation to evaluate the probability of the MN's zone. In addition, both methods yield probabilistic outputs, leading to first and second choice zones. Table 3.9 shows the overall accuracy over the test points obtained with our proposed method, compared with both NB and MLR as a function of the number of reference positions. The numbers in bold correspond to the best performing methods for a 1st choice zones. The results show that the proposed method outperforms the other classification techniques in terms of overall accuracy for different training-test ratios.

## 3.5 Conclusion

In this chapter, we proposed a new observation model for zoning-based localization in a belief functions framework using WiFi fingerprints. Different types of modeling were considered, namely the parametric and non-parametric distributions, to statistically describe the variation of the RSSI observations. The obtained distributions were used to define mass functions over the zones with respect to the available APs in the network. Once a new observation was carried for localization, the constructed mass functions were used to assign a mass for each zone with respect to each AP. The evidence attributed by each AP was

**Table 3.9:** Comparison of the proposed method to NB and MLR methods, in terms of overall accuracy (%), as a function of number of reference positions.

| Technique | | Ratio of training-test | | |
|---|---|---|---|---|
| | | $70\% - 30\%$ | $60\% - 40\%$ | $50\% - 50\%$ |
| NB | 1st choice | 86.11 | 83.33 | 79.17 |
| | 2nd choice | 95.83 | 93.75 | 90.83 |
| MLR | 1st choice | 88.89 | 84.38 | 82.50 |
| | 2nd choice | 97.22 | 94.79 | 92.50 |
| Proposed | 1st choice | **91.67** | **87.50** | **85.83** |
| | 2nd choice | 100.00 | 100.00 | 98.33 |

then discounted according to its error rate. Pieces of evidence were combined afterwards using fusion rule, and a confidence level was assigned to each zone through the pignistic transformation. The zone having the highest confidence was supposed to be the zone where the MN resides. The ranking of results allowed a second zone choice in case of erroneous first estimation. Experimental results showed that the proposed approach achieves a good overall accuracy, outperforming other classification techniques such as NB and MLR. However, the proposed observation model was found to be vulnerable to the number of zones in the targeted area. Therefore, in the next chapters, we introduce further improvements of this model, by extending it through hierarchical clustering and AP selection in Chapter 4, combining it to mobility models in Chapter 5, and providing a decentralized version of it in Chapter 6.

# Chapter 4

# Extended Observation Model

## Contents

*In this chapter, we aim at enhancing the zoning-based localization technique proposed in Chapter 3. The previously proposed technique is an observation model that is based on the belief functions theory and uses a fingerprinting database to assign confidence levels to the zones of the targeted area. A major drawback of the presented observation model is that it is vulnerable to the increase in the number of zones. To this end, we extend the model through hierarchical clustering to tackle large surface areas with a higher number of zones. We then develop an AP selection technique to choose the best subset of APs that helps increasing the overall accuracy and reducing the complexity of the localization approach. The performance of the new proposed model, which will be referred to as "extended observation model", is evaluated and compared with the previously described one, which will be referred to as "basic observation model". In addition, comparisons to other well-known techniques are provided.*

## 4.1 Introduction

In Chapter 3, we proposed an observation model for zoning-based localization. The proposed model is based on the BFT and uses a fingerprinting database to assign a confidence level to each zone of the targeted area. The APs are considered as sources of information, yielding an amount of evidence about having the MN in each zone. Mass functions are thus constructed at the offline phase using the collected RSSI measurements. Then in the online phase, once a new observation is carried for localization, constructed masses are used to assign evidence to each zone. We will refer to the proposed approach by a basic observation model (BOM). The advantage of this approach is that evidence can be associated to a set of zones rather than to singleton ones only. This allows to represent how strongly evidence supports a certain set, which is important especially in case of insufficient information regarding single zones. In addition, the BOM permits the use of any piece of evidence, even if it were unreliable. Indeed, a discounting technique is applied to correct such imperfect information, since received measurements are RSSIs that might be imprecise and incomplete due to packet loss and high variation of signals strengths. However, the performance of the BOM

degrades as the number of zones increases, as studied in Section 3.4.2.4. Moreover, when increasing the number of zones, a larger number of APs is required to cover the area and provide more evidence to achieve good performance. However, not all APs are useful, where some might be erroneous or redundant.

In this chapter, we propose an extended observation model (EOM) that solves all these problems of the BOM, through hierarchical clustering and AP selection, as follows. Given the database $\mathcal{D}$ of labeled RSSI observations, the zones are merged into clusters using an agglomerative hierarchical clustering method, leading to a dendrogram of clusters. An optimal level of clustering is selected from the obtained dendrogram, by optimizing the inter- and intra-clusters scatters. The hierarchy is reformed into two levels, the first consisting of the optimal selected clusters, and the second of the original zones in each cluster. Reducing the hierarchy to only two levels decreases considerably the complexity of the method, compared with classic hierarchical methods, with more robustness against error propagation. It also reduces the considered zones at a level, which makes it more efficient than flat techniques. Afterwards, the objective of the algorithm becomes to determine the correct cluster and the correct zone at the first and second levels respectively. Moreover, since APs might be beneficial in discriminating between some clusters, or zones, but harmful in others, an Access Point selection technique is carried at each level of the hierarchy, by maximizing the discriminative capacity, and minimizing the redundancy of the ensemble of APs. This creates a framework for the BFT that associates masses and combines evidence to determine a level of confidence of having the MN residing in each zone.

In this chapter, we have the same network configuration and objective as of Chapter 3. For that reason, we start by providing an overview on clustering analysis, discussing its general terms and some of the methods used in literature. We then describe the concept of feature selection, explaining its usage and some of its well-known approaches. Afterwards, we present the extended observation model by detailing the two-level hierarchy, the AP selection algorithm, and the confidence-based zone estimation. Finally, we evaluate the performance of the EOM as compared with the BOM and to other well-known techniques.

## 4.2 Clustering

The aim of clustering analysis is to organize a set of data items in clusters, in a way that data items that are in the same cluster are more *similar* or *dissimilar* to each other than they are to data items that are in other clusters [Shahbaba & Beheshti, 2014]. In the following, we attempt to briefly review a few concepts of cluster analysis, and some of the clustering methods used in literature.

### 4.2.1 Definition

The aim of cluster analysis is to partition a set of objects into clusters. Clustering is a mathematical tool for discovering structures in a dataset, where the data assigned to each cluster show some degree of similarity. This notion of similarity is defined in several ways, and varies as a function of the objective of the study. Clustering is often carried out without any knowledge regarding the belonging of data items to predefined classes, making it one of the most investigated approaches in unsupervised learning. Clustering analysis is used for vast number of applications, such as medical imaging, anomaly detection, robotics and tracking systems, human genetic clustering, etc. Surveys of these applications can be found in [Jain *et al.*, 1999; Kaufman & Rousseeuw, 2009]. Cluster analysis is an iterative process that requires a criterion in order to group data. Usually, a similarity metric is used, although on many occasions a dissimilarity measure is considered in accordance the application. A traditional way to measure distances is the Minkowski distance. The Manhattan, Euclidean and Chebyshev distances are special cases of the Minkowski distance. As an example of similarity/dissimilarity we have the cosine similarity, the Hellinger, Mahalanobis, Kullback-Leibler, and Hamming distances. A comprehensive survey on these measures is found in [Cha, 2007].

### 4.2.2 State-of-the-art methods

It is distinguished between two categories of clustering, hierarchical and flat.

- A hierarchical clustering algorithm yields a dendrogram representing the nested grouping of patterns and similarity levels at which groupings change.

The dendrogram can be broken at different levels to yield various clusterings of the data. Hierarchical clustering builds a hierarchy of clusters driving two strategies, agglomerative or divisive. Agglomerative hierarchical clustering algorithms, or the *bottom-up* approaches, merge a pair of clusters at each iteration [Murtagh & Contreras, 2012]. Most algorithms are variants of the single-link [Johnson, 1967], complete-link [Defays, 1977], and minimum-variance algorithms [Ward Jr, 1963]. In these methods, two clusters are merged based on minimum distance criteria. On the contrary, the divisive hierarchical clustering algorithm is a *top-down* clustering method [Sasirekha & Baby, 2013]. It works in a similar way to agglomerative clustering but in the opposite direction. This method starts with a single cluster containing all objects, and then successively splits resulting clusters until only clusters of individual objects remain. In the same manner, the single-link, complete-link, and minimum-variance algorithms are used. However, a cluster here is partitioned into smaller entities according to similarity metrics.

- A flat clustering algorithm obtains a single partition of the data instead of a clustering structure such as the dendrogram produced by a hierarchical technique. Flat methods have advantages in applications involving large data sets for which the construction of a dendrogram is computationally prohibitive. A problem accompanying the use of a flat algorithm is the choice of the number of desired output clusters. Dubes [1987] provides guidance on this key design decision. The flat clustering techniques generate clusters by optimizing a criterion function defined either locally or globally. Combinatorial search of the set of possible labelings for an optimum value of a criterion is clearly computationally prohibitive. In practice, the algorithm is typically run multiple times with different starting states, and the best configuration obtained from all the runs is used as the output clustering.

The two clustering approaches are illustrated in Figures 4.1 and 4.2, where $\{a, b, c, d, e, f, g, h\}$ is a set of clusters, $C_1, C_2$, and $C_3$ are parent nodes, and $R$ is a root node.

## 4.3 Feature Selection

The existing approaches that aim at improving an intelligent system's performance can be categorized into two groups. The first focuses on designing new machine learning algorithms for different applications at a higher level. The second focuses on crafting and selecting distinctive features, which lead to a significant improvement in the performance at a lower level of the system. Feature selection is considered an effective way to reduce the computational cost and improve the quality of features. In this section, we describe the concept of feature selection, explaining its usage and some of its well-known approaches.

### 4.3.1 Definition

Feature selection is an important step in data preprocessing for designing intelligent systems. This is especially important in high-dimensional datasets, where the goal is to seek the relevant features with the most predictive information from the original feature set. Often, many features in the datasets are irrelevant, thus introducing a prediction error. They might also be redundant, thus increasing the dimensionality of the algorithm. Feature selection reduces the dimension-

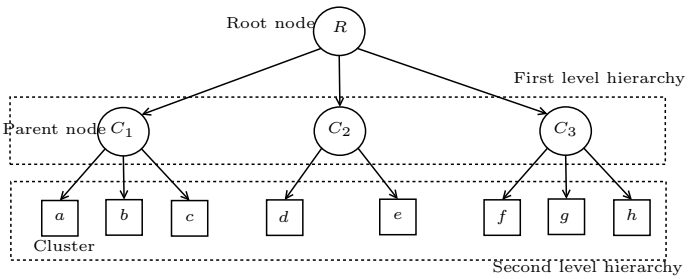



**Figure 4.1:** Hierarchical clustering.

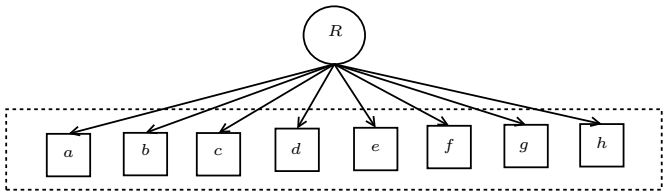



**Figure 4.2:** Flat clustering.

ality of datasets by eliminating many irrelevant and redundant features, which improves the performance of the learnt model and avoids overfitting. Moreover, this reduction helps to speed up the learning process and leads to a simple and understandable predictor model.

### 4.3.2 State-of-the-art methods

Feature selection algorithms are divided into two major categories, filter methods and wrapper methods.

- In filter methods [Gheyas & Smith, 2010; John *et al.*, 1994], a complete order of the features is provided using a relevance index. Methods for computing ranking indices include correlation coefficients, which assess the degree of dependence of individual variables with the outcome. More generally, methods that select features without optimizing the performance of a predictor are referred to as filter methods. These methods are called univariate filter models and can effectively identify and remove the irrelevant features independently of any learning algorithms. However, they are unable of removing redundant features. Since the possible dependency between features is disregarded, these methods lead to a weak learning model. On the other hand, multivariate filter models can handle both irrelevant and redundant features, thus improving the accuracy of the learning model as compared with the univariate filter methods. Nevertheless, the search strategy of the multivariate filter model involves only a single iteration and can easily be trapped into local optimum.

- In wrapper methods [Kohavi & John, 1997; Narendra & Fukunaga, 1977], the learning algorithm is used as a black box to score subsets of features according to their predictive power. For each subset of the original features, prediction scores can be regarded as evaluation outcomes. The best subset is consequently obtained by optimizing the objective function. Generally, wrapper methods are classified into greedy and random search approaches. The greedy search approach is based on the hill-climbing algorithm where a single feature is added or removed iteratively using greedy strategies, such

as sequential backward selection and sequential forward selection; whereas the random search approach applies randomness into its search strategy to explore a large portion of the solution space.

The interaction with the learning algorithm in the wrapper methods makes them outperform filter methods in terms of prediction accuracy. However, they continuously use the learning algorithm in the search process and thus they are computationally more expensive [Tabakhi & Moradi, 2015].

## 4.4 Extended Observation Model

The extended observation model (EOM) extends the basic observation model (BOM) presented in Chapter 3 by creating a two-level hierarchy and selecting the best APs at each level. This model is used when the number of zones in the targeted area is large. In this case, both the basic representation of the zones and the assignment of a discriminating evidence to each one become more difficult than in the case of small number of zones. A clustering technique is developed, leading to a two-level hierarchy composed of clusters, and the original zones in each cluster. Moreover, when increasing the number of zones, a larger number of APs is required to cover the area and provide more evidence to achieve good performance. However, not all APs are useful, where some might be erroneous, thus decreasing the overall accuracy, and others might be redundant, thus increasing the complexity of the localization algorithm. For that reason, an AP selection algorithm is developed and applied at each level of the created hierarchy. The same belief functions framework presented in Chapter 3 is created to associate masses, discount APs, and combine evidence. The model finally determines a level of confidence of having the MN residing in each zone. Table 4.1 lists the variables used in this chapter, along with their respective sizes. In the following, we present the clustering algorithm, the AP selection algorithm, and the confidence-based zoning approach.

### 4.4.1 Clustering algorithm

The idea behind developing a clustering algorithm is to reduce the number of zones being distinguished or classified at a certain time. Besides, the algorithm should group the zones in a way that facilitates discriminating between them in the online localization phase. Although no theoretical evidence or proof whether hierarchical or flat clustering models are better for classification [Babbar *et al.*, 2013], experiments throughout previous studies have shown that a better accuracy can be obtained by the former especially for a large number of classes [Dumais & Chen, 2000; Silla Jr & Freitas, 2011]. However, a large number of levels in the dendrogram causes slowness in the classification procedure, in addition to the risk of propagating any error in a top level all along the hierarchy [Babbar *et al.*, 2013; Dumais & Chen, 2000]. For that reason, we consider a hierarchical clustering approach to create a dendrogram of zones. Here, the agglomerative strategy is adopted since it is less complex than the divisive case with comparable performance [Manning *et al.*, 2008]. The latter is conceptually more complex

**Table 4.1:** List of the variables used in Chapter 4, with their respective sizes.

| Notation | Variable | Size |
|---|---|---|
| $N_Z$ | number of zones | 1 |
| $N_{AP}$ | number of APs | 1 |
| $N_C$ | number of clusters | 1 |
| $Z_j$ | zone | 1 |
| $AP_k$ | Access Point | 1 |
| $C_i$ | cluster | 1 |
| $h$ | bandwidth | 1 |
| $D_{KL}(\cdot)$ | Kullback-Leibler divergence | 1 |
| $\mathcal{E}(\cdot)$ | error function | 1 |
| $\mathcal{R}(\cdot)$ | redundancy function | 1 |
| $\eta$ | trade-off parameter | 1 |
| $\rho, \rho_{j,k,l}$ | RSSI measurement | 1 |
| $Q(\cdot)$ | distribution | 1 |
| $m_{\mathbb{O}}(\cdot), BetP^C(\cdot), BetP^i(\cdot)$ | assigned evidence | 1 |
| $\boldsymbol{\rho}, \boldsymbol{\rho}_{j,l}$ | vector of RSSI measurements | $N_{AP} \times 1$ |
| $F$ | set of APs | $N_{AP} \times 1$ |
| $F_s$ | subset of selected APs | $(\leq N_{AP}) \times 1$ |

since we need a second flat clustering algorithm as a subroutine to split each cluster. This step is recursively applied till we reach the individual units. The agglomerative strategy is shown in Figure 4.3. In order to avoid having observations of the same zone in different clusters, the proposed method considers the zones as units. Indeed, a distribution is assigned to each zone, by modeling its corresponding observations either parametrically or non-parametrically.

#### 4.4.1.1 Dissimilarity measure

As explained in Section 4.2, clustering requires a distance metric as a criterion to be performed. Similarity measures such as Minkowski distance are used for this purpose. The Manhattan, Euclidean and Chebyshev distances are special cases of the Minkowski distance. In contrary to the traditionally used similarity measures, we need a dissimilarity metric between clusters. In fact, we are interested in maximizing the distance between the zones rather than minimizing it. By maximizing the divergence between the zones, or clusters, it will be easier to distinguish between them later on in the localization phase. For that reason, statistical measures such as the Kullback-Leibler divergence, Hellinger distance, or total variation distance that measure the dissimilarity between statistical distribtions can be used [Basseville, 2013]. Among these, the Kullback-Leibler divergence is of significant interest due to its simplicity and mathematical properties. In addition the error between probability distributions can be defined in its terms. The Kullback-Leibler divergence or relative entropy of two distributions $Q_{Z_{j'}}$ and $Q_{Z_j}$ of a continuous random variable $\boldsymbol{\rho}$ is defined as,

$$D_{KL}(Q_{Z_{j'}}||Q_{Z_j}) = \int_{\boldsymbol{\rho}} \log\left(\frac{Q_{Z_{j'}}(\boldsymbol{\rho})}{Q_{Z_j}(\boldsymbol{\rho})}\right) Q_{Z_{j'}}(\boldsymbol{\rho}) d\boldsymbol{\rho}. \tag{4.1}$$

The relative entropy is asymmetric, i.e., $D_{KL}(u||v) \neq D_{KL}(v||u)$ in general, always positive and equal to zero when the two distributions are identical. The J-divergence symmetrizes the Kullback-Leibler divergence as follows [Nielsen & Nock, 2017],

$$D_J(Q_{Z_{j'}}||Q_{Z_j}) = D_{KL}(Q_{Z_{j'}}||Q_{Z_j}) + D_{KL}(Q_{Z_j}||Q_{Z_{j'}}). \tag{4.2}$$

In order to determine the dissimilarity between the zones, or clusters, all the measurements of a certain zone, or cluster, should be taken together. For that reason, the multi-dimensional distribution is considered to represent the variations of the observations in each zone, or cluster. Two zones that are similar with respect to one AP for instance, might be totally different with respect to the whole set of APs.

The proposed clustering method employs the J-divergence as the dissimilarity measure to construct the dendrogram. The methods starts by considering each zone as an independent cluster. At each iteration, it merges the two clusters whose distributions have the maximal divergence. Merging two clusters means here a merge of all the observations of the clusters and a computation of a new distribution according to the new set of observations. By maximizing the divergence, the clusters will be dissimilar, which helps in discriminating between zones of each cluster. The algorithm is iterated until all the zones are merged into one cluster. We discussed in Section 3.3.1 two types of modeling, parametric and non-parametric. In the following, we present the proposed clustering algorithm in each of the two cases.

In the case of parametric modeling, the observations of each zone are fitted to a multi-dimensional parametric distribution. This is done as explained in Section 3.3.1.1 and using the whole set of APs, instead of only one AP. The com-

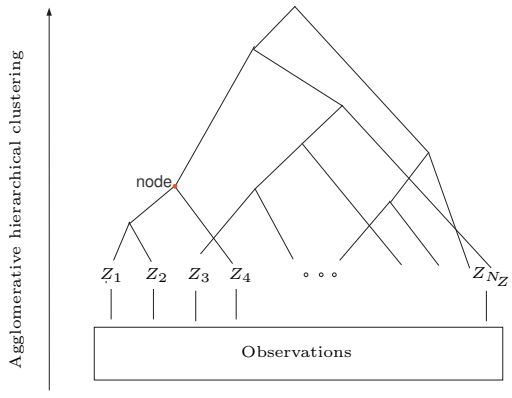



**Figure 4.3:** Agglomerative hierarchical clustering.

putation of the Kullback-Leibler divergence is then done using equation (4.7). This is a computation of a multi-dimensional integral that might be complex, especially for high number of APs. Fortunately, researchers have been working on deriving and approximating this divergence in the cases of each parametric distribution, and expressing it through only its parameters. Such studies can be found in [Barron & Sheu, 1991; Hershey & Olsen, 2007; Pérez-Cruz, 2008].

In the case of non-parametric or KDE modeling, it is not straightforward to compute the Kullback-Leibler divergence between the KDEs. We have shown using equation (3.7) that the univariate KDE of the RSSI observations of any zone $Z_j$ with respect to $AP_k$ is obtained as,

$$Q_{Z_j,k}(\cdot) = \frac{1}{N_j \times h_{j,k}} \sum_{\ell=1}^{N_j} \mathcal{K}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right). \tag{4.3}$$

The KDE is easily extended to the multivariate case for all $N_{AP}$ APs,

$$Q_{Z_j}(\cdot) = \frac{1}{N_j \times h_j^{N_{AP}}} \sum_{\ell=1}^{N_j} \mathcal{K}\left(\frac{\cdot - \boldsymbol{\rho}_{j,\ell}}{h_j}\right). \tag{4.4}$$

However, the same bandwidth is taken here on all axes, weighting all APs equally. When considering different bandwidths $h_{j,k}, j \in \{1,\ldots,N_Z\}, k \in \{1,\ldots,N_{AP}\}$, that vary according to each $AP_k$ in the zone $Z_j$, the KDE is obtained as follows,

$$Q_{Z_j}(\cdot) = \frac{1}{N_j} \sum_{\ell=1}^{N_j} \frac{1}{h_{j,1}\ldots h_{j,N_{AP}}} \mathcal{K}\left(\frac{\cdot - \rho_{j,1,\ell}}{h_{j,1}}, \ldots, \frac{\cdot - \rho_{j,N_{AP},\ell}}{h_{j,N_{AP}}}\right), \tag{4.5}$$

By using the product kernel $\mathcal{K}(u) = \mathcal{K}(u_1) \times \cdots \times \mathcal{K}(u_{N_{AP}})$, equation (4.5) can be expressed as,

$$Q_{Z_j}(\cdot) = \frac{1}{N_j} \sum_{\ell=1}^{N_j} \frac{1}{h_{j,1}\ldots h_{j,N_{AP}}} \prod_{k=1}^{N_{AP}} \mathcal{K}\left(\frac{\cdot - \rho_{j,k,\ell}}{h_{j,k}}\right), \tag{4.6}$$

such that the bandwidth $h_{j,k}$ is associated to zone $Z_j$ with respect to $AP_k$ using equation (3.11). It is noteworthy that by considering the product of kernels, we only assume kernels independence, which does not imply that we assume APs independence.

The hierarchy is built based on the Kullback-Leibler divergence between the distributions representing the zones or clusters. However, since a kernel density estimation is adopted here, the evaluation of the $D_{KL}(\cdot)$ is not trivial. The $D_{KL}(\cdot)$ between $Q_{Z_j}(\boldsymbol{\rho})$ and $Q_{Z_{j'}}(\boldsymbol{\rho})$ is defined as [Harmouche *et al.*, 2016],

$$D_{KL}(Q_{Z_j}||Q_{Z_{j'}}) = \int_{\boldsymbol{\rho}} \log\left(\frac{Q_{Z_j}(\boldsymbol{\rho})}{Q_{Z_{j'}}(\boldsymbol{\rho})}\right) Q_{Z_j}(\boldsymbol{\rho}) d\boldsymbol{\rho}. \tag{4.7}$$

The expected value of $\log\left(\frac{Q_{Z_j}(\boldsymbol{\rho})}{Q_{Z_{j'}}(\boldsymbol{\rho})}\right)$ with respect to $Q_{Z_j}(\boldsymbol{\rho})$ is given by,

$$E_{Q_{Z_j}(\boldsymbol{\rho})}\left[\log\left(\frac{Q_{Z_j}(\boldsymbol{\rho})}{Q_{Z_{j'}}(\boldsymbol{\rho})}\right)\right] = \int_{\boldsymbol{\rho}} \log\left(\frac{Q_{Z_j}(\boldsymbol{\rho})}{Q_{Z_{j'}}(\boldsymbol{\rho})}\right) Q_{Z_j}(\boldsymbol{\rho}) d\boldsymbol{\rho}. \tag{4.8}$$

Anderson *et al.* [2000] approximate the expected value as follows,

$$E_{Q_{Z_j}(\boldsymbol{\rho})}\left[\log\left(\frac{Q_{Z_j}(\boldsymbol{\rho})}{Q_{Z_{j'}}(\boldsymbol{\rho})}\right)\right] \approx \frac{1}{N_j}\sum_{\ell=1}^{N_j} \log\left(\frac{Q_{Z_j}(\boldsymbol{\rho}_{j,\ell})}{Q_{Z_{j'}}(\boldsymbol{\rho}_{j,\ell})}\right). \tag{4.9}$$

The $D_{KL}(\cdot)$ is then deduced using equations (4.8) and (4.9),

$$\begin{aligned} D_{KL}(Q_{Z_j}||Q_{Z_{j'}}) &= E_{Q_{Z_j}(\boldsymbol{\rho})}\left[\log\frac{Q_{Z_j}(\boldsymbol{\rho}_{j,\ell})}{Q_{Z_{j'}}(\boldsymbol{\rho}_{j,\ell})}\right]; \\ &\approx \frac{1}{N_j}\sum_{\ell=1}^{N_j} \log\frac{Q_{Z_j}(\boldsymbol{\rho}_{j,\ell})}{Q_{Z_{j'}}(\boldsymbol{\rho}_{j,\ell})}; \\ &\approx \frac{1}{N_j}\sum_{\ell=1}^{N_j} \log Q_{Z_j}(\boldsymbol{\rho}_{j,\ell}) - \log Q_{Z_{j'}}(\boldsymbol{\rho}_{j,\ell}). \end{aligned} \tag{4.10}$$

It is then easy to compute $D_{KL}(\cdot)$ by replacing $Q_{Z_j}(\cdot)$ and $Q_{Z_{j'}}(\cdot)$ with their kernel models obtained using equation (4.6). The divergence $D_J(\cdot)$, which is the symmetric divergence of $D_{KL}(\cdot)$ as expressed in equation (4.2), is computed by replacing $Z$ with the obtained cluster.

To construct the dendrogram of clusters, the algorithm starts by considering each zone as a separate cluster. The symmetric divergence, in either parametric or non-parametric case, is then computed between all pairs of clusters. The pair of clusters having the highest divergence are merged together, to form a parent

node as shown in Figure 4.3. This is recursively done all along the hierarchy until obtaining a final cluster that constitutes all the observations of the initial set of zones $Z_j, j \in \{1, \ldots, N_Z\}$. However, the usage of the whole dendrogram is computationally complex. In addition, an error in any node is propagated over the whole hierarchy. For that reason, the dendrogram is cut based on an optimal number of clusters criteria as explained in the next section.

#### 4.4.1.2 Two-level hierarchy

After the dendrogram is created, it should be cut based on the desired number of clusters. However, since there is no prior knowledge regarding this parameter, it is calculated by solving an optimization problem that takes into account both inter- and intra- clusters scatters. Several indices have been proposed to solve this problem [Islam *et al.*, 2015; Rokach & Maimon, 2005; Tibshirani *et al.*, 2001]. For the sake of simplicity and without loss of generality, we describe the embedding of the method proposed by Krzanowski & Lai [1988] to find the optimal number of clusters $N_C$ as follows,

$$N_C = \underset{r}{\operatorname{argmax}} \left| \frac{DIFF(r)}{DIFF(r+1)} \right|, \tag{4.11}$$

such that

$$DIFF(r) = (r-1)^{\frac{2}{N_{AP}}} W(r-1) - (r)^{\frac{2}{N_{AP}}} W(r), \tag{4.12}$$

where $W(r)$, the within cluster sums of squares, is computed as follows,

$$W(r) = \sum_{i=1}^{r} \sum_{\substack{j \\ Z_j \in C_i}} \sum_{\ell=1}^{N_j} ||\boldsymbol{\rho}_{j,\ell} - \mu_i||^2, \tag{4.13}$$

where $\boldsymbol{\rho}_{j,\ell}$ is an RSSI measurement taken in zone $Z_j$, $r$ takes its first value as $N_Z - 1$ and decreases by 1 at each iteration, $\mu_i$ is the mean of the observations of the cluster $C_i$, and $||\cdot||$ is the Euclidean norm operator. Equation (4.12) indicates the gain in within-cluster compactness resulting from the change from $r-1$ to $r$ clusters. Equation (4.11) serves to maximize the in between-cluster distance. In this way, both inter- and intra-cluster distances are optimized.

The dendrogram is cut at a certain level where $N_C$ clusters are obtained, denoted by $C_i$ with $i \in \{1, \ldots, N_C\}$. All clusters of each selected parent cluster are merged yielding a set of zones for each cluster, as it is shown in Figure 4.4. The set $I_i$ denotes the set of indices of the zones included in the cluster $C_i$, that is, $Z_j \in C_i, \quad \forall j \in I_i$. As a result, the output of the clustering algorithm is a two-level hierarchy. The first level is a set of $N_C$ clusters, $\{C_1, \ldots, C_{N_C}\}$, and the second level is a group of zones belonging to each cluster, $Z_j \in C_i, \forall j \in I_i$, $I_i$ being the set of indices of the zones included in the cluster $C_i$. The objective of the localization algorithm becomes to determine, for any new observation, the correct cluster at the first level, and the correct zone of the cluster at the second level.

### 4.4.2 Access Point selection algorithm

The APs are the sources of information, and hence the choice of reliable ones is indispensable for the localization process. Practically, a given installed physical WiFi AP transmits signals on different terminals, each to give access to a certain population (staff, residents, visitors, etc) and on different channel bands (2.4 GHz, 5 GHz, etc). The emitted power on each terminal is controlled and changed by the IT services as needed. Although information carries some redundancy, we aim here at using all available evidence to reach the best decision. Hence, what is meant here by AP selection is the choice of terminals and not only physical APs. The observations have $N_{AP}$ components, each one being related to a certain AP, of the set $F = \{AP_1, \ldots, AP_{N_{AP}}\}$. Having the two-level hierarchy, the AP selection is applied at the cluster level, and also at the zones level within each cluster, the

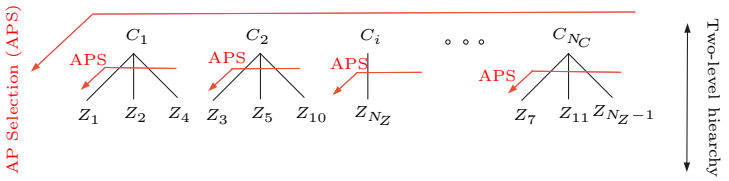



**Figure 4.4:** The two-level hierarchy and AP selection at each level.

aim being to select the most useful APs that are capable of discriminating zones of each cluster and between clusters.

The AP selection algorithm is applied equivalently at zones of each cluster and between clusters. For the sake of simplicity, unique notations for clusters and zones are considered in the following, that is, let $z$ denote either a cluster or a zone within a cluster, and let $N_z$ denote their numbers. A greedy AP selection method is adopted to maximize the discriminative capacity and minimize the redundancy of the selected APs. The APs cannot be treated independently, since one that might be useless by itself can provide a significant improvement in the performance when taken with others [Guyon & Elisseeff, 2003]. Let $F' \subseteq F$ denote one non-empty subset of $F$, the set of $N_{AP}$ APs. All the observations at the APs of $F'$ belonging to each entity $z_j$ are thus taken and fitted to a distribution denoted $Q_{z_j,F'}(\cdot)$, $j \in \{1,\ldots,N_z\}$. The distribution $Q_{z_j,F'}(\cdot)$ is either univariate or multivariate depending on the cardinal of $F'$. In what follows, we explain the aim of the AP selection algorithm at the discriminative and the redundancy levels, then we describe the bi-objective optimization to select the best subset of APs.

#### 4.4.2.1 Discriminative capacity

On one hand, having erroneous APs harms the performance of localization. But since the computation of the exact error rate is cumbersome especially for high dimensions, we use the discriminative capacity of APs, which is inversely proportional to the error rate [Jahromi, 2007]. Indeed, the farther the distributions $Q_{z_1,F'}(\cdot),\ldots,Q_{z_{N_z},F'}(\cdot)$ are one from another, the more discriminative the AP subset $F'$ is, and thus the less error rate is obtained. The Kullback-Leibler divergence is used to measure such quantity. The discriminative capacity of a subset of APs $F' \subseteq F$ is then defined as the cumulative distance between all zones or clusters according to $F'$,

$$DisC(F') = \sum_{a=1}^{N_z}\sum_{b=1}^{N_z} D_{KL}(Q_{z_a,F'}||Q_{z_b,F'}), \tag{4.14}$$

$D_{KL}(Q_{z_a,F'}||Q_{z_b,F'})$ being the Kullback-Leibler divergence measured between the distributions of the observations belonging to zones or clusters $z_a$ and $z_b$, while

considering only the APs of $F'$. Jahromi [2007] proved that the error rate to distinguish between two distributions is inversely proportional to the Kullback-Leibler divergence between them, and thus can be approximated for a subset $F'$ as,

$$\mathcal{E}(F') = 2^{-DisC(F')}. \tag{4.15}$$

Using equations (4.14) and (4.15), the larger the divergence $D_{KL}(Q_{z_a,F'}||Q_{z_b,F'})$, the less is the risk of error to distinguish between $z_a$ and $z_b$.

#### 4.4.2.2 Redundancy

On the other hand, redundancy reduction is an important factor in AP selection. Having redundant APs leads to a higher dimensionality and thus more complexity. Here, the coefficient of multiple correlation is used as it is a measure of how much an AP is dependent upon other ones. The square of the multiple correlation coefficient of $AP_k$ in a set of APs $F'$ with respect to all APs of $F'$ except itself, $F' \setminus \{AP_k\}$, is defined as follows,

$$R_k^2 = c_k^T R_{xx,k}^{-1} c_k, \tag{4.16}$$

where $c_k$ is the column vector with entries $d_{AP_{k'}AP_k}$ for $AP_{k'} \in F' \setminus \{AP_k\}$, $d_{AP_{k'}AP_k}$ being the correlation between $AP_{k'}$ and $AP_k$ computed using their observations, $c_k^T$ being the transpose of $c_k$, and $R_{xx,k}^{-1}$ the inverse of the matrix of entries $d_{AP_{k'}AP_{k''}}$ for all pairs $AP_{k'}$ and $AP_{k''}$ belonging to $F' \setminus \{AP_k\}$. The redundancy between all the APs of a set $F'$ is the average multiple correlation coefficient of all $AP_k \in F'$, namely

$$\mathcal{R}(F') = \sum_k \frac{R_k}{|F'|}, \text{ where } |F'| \text{ is the cardinal of } F'. \tag{4.17}$$

#### 4.4.2.3 Bi-objective optimization

The aim of the AP selection technique is to find the optimal subset $F_s \subseteq F$ such that both $\mathcal{E}(F_s)$ and $\mathcal{R}(F_s)$ are simultaneously minimized. This can be solved using any multi-objective optimization technique such as Pareto front [Ngatchou *et al.*, 2005]. Yet these solutions require the knowledge of $\mathcal{E}(\cdot)$ and $\mathcal{R}(\cdot)$ for all subsets, which is computationally unfeasible for large number of APs. For this

purpose, a greedy search algorithm with backward elimination strategy is applied to choose this subset. One starts with the whole set of APs and progressively eliminates an AP, whose elimination satisfies a function that considers the two objectives mentioned above. Let $F_y$ be the subset of APs chosen at iteration $y \geq 1$, with $F_0 = F$ and the cardinal of $F_y$, $|F_y| = N_{AP} - y$. The algorithm starts with the whole set of features $F = F_0$. At each iteration $y \geq 1$, all the subsets of $F_{y-1}$ having all but one elements, that is, $N_{AP} - y$ elements, are considered. Let $F_y^{(\lambda)}$, $\lambda = 1, \ldots, N_{AP} - y + 1$, denote these subsets. We define the function $g_y\big(F_y^{(\lambda)}\big)$ as follows,

$$g_y(F_y^{(\lambda)}) = \eta \frac{\mathcal{E}(F_{y-1}) - \mathcal{E}(F_y^{(\lambda)})}{\max(\mathcal{E}(F_{y-1}), \mathcal{E}(F_y^{(\lambda)}))} + (1-\eta) \frac{\mathcal{R}(F_{y-1}) - \mathcal{R}(F_y^{(\lambda)})}{\max(\mathcal{R}(F_{y-1}), \mathcal{R}(F_y^{(\lambda)}))}, \quad (4.18)$$

where $\eta \in [0, 1]$ is a tradeoff parameter chosen by the user to assign a weight for each objective. A positive value of $g_y\big(F_y^{(\lambda)}\big)$ means that the subset $F_y^{(\lambda)}$ is better than $F_{y-1}$ in optimizing the objectives. The greater $g_y(\cdot)$ is, the better the subset is. This leads to a selected subset at iteration $y$, $F_y = \arg\max_\lambda g_y\big(F_y^{(\lambda)}\big)$. A negative value of $g_y(F_y^{(\lambda)})$ means that there is no significant improvement in the objectives for the considered parameters and hence iterations stop when all $g_y\big(F_y^{(\lambda)}\big)$, $\lambda = 1, \ldots, N_{AP} - y + 1$, are negative and one chooses the set of APs $F_s = F_{y-1}$. This algorithm is applied at the clusters level to yield $F_s$ and at the zones level of each cluster $C_i$ to yield $F_{s,i}, i \in \{1, \ldots, N_C\}$, as shown in Figure 4.4.

### 4.4.3 Confidence-based zone estimation

The clustering and the AP selection algorithms lead to a two-level hierarchy, where the first level is a set of created clusters $C_i$, $i \in \{1, \ldots, N_C\}$, and the second level is the set of zones $Z_j \in C_i$, $\forall j \in I_i$, $I_i$ be the set of indices of the zones included in cluster $C_i$. Moreover, a selected subset of APs, $F_s$, at the clusters level, and selected subsets, $F_{s,i}$, at the zones level of cluster $C_i$. The masses assigned by the selected APs are discounted using either of the techniques presented in Section 3.3.3 to obtain ${}^{\alpha}m_{AP_k,t}(\cdot)$. For the fusion of evidence, the combination rules discussed in Section 3.3.4 are applied at the two levels of the

hierarchy, yielding unified mass functions. The unified mass functions are $m^{C}_{\oplus,t}(\cdot)$ that work on the subsets of $\{C_1, \ldots, C_{N_C}\}$, and also other functions $m^{i}_{\oplus,t}(\cdot)$ that work on the subsets of $\{Z_j, j \in I_i\}$, with $i \in \{1, \ldots, N_C\}$. This is the case if the Dempster's rule of combination is used. The unified mass functions become $m^{C}_{\textcircled{\cap},t}(\cdot)$ and $m^{i}_{\textcircled{\cap},t}(\cdot)$ if the conjunctive rule is used, and $m^{C}_{\textcircled{\cup},t}(\cdot)$ and $m^{i}_{\textcircled{\cup},t}(\cdot)$ if the disjunctive rule is used. After computing the masses of all clusters and zones within each cluster using either of the combination rules, equation (3.25) is applied at both levels, leading respectively to the functions $BetP^{C}_{t}(\cdot)$ and $BetP^{i}_{t}(\cdot)$ on the clusters and zones levels respectively. In order to compute pignistic levels for singletons, the equation (3.26) is used, yielding $BetP^{C}_{t}(\{C_i\})$, $i \in \{1, \ldots, N_C\}$, and $BetP^{i}_{t}(\{Z_j\})$, $j \in I_i$. Finally, to attribute a confidence level by the extended observation model to each zone at any instant $t$, the pignistic levels of zones and clusters are combined as follows,

$$m_{\mathbb{O},t}(Z_j) = BetP^{C}_{t}(\{C_i\}) \times BetP^{i}_{t}(\{Z_j\}), \tag{4.19}$$

with $j \in I_i, i \in \{1, \ldots, N_C\}$.

## 4.5 Experimental Results

The performance of the proposed method is evaluated through two experimental scenarios for localization. The experimental setups are first introduced, and the method is illustrated afterwards. The performance of the method is also compared with state-of-the-art classification techniques. Finally, to validate our proposed method, which serves also as a classification technique, we apply it in the domain of image processing for facial recognition using public databases, and compare its performance with well-known techniques.

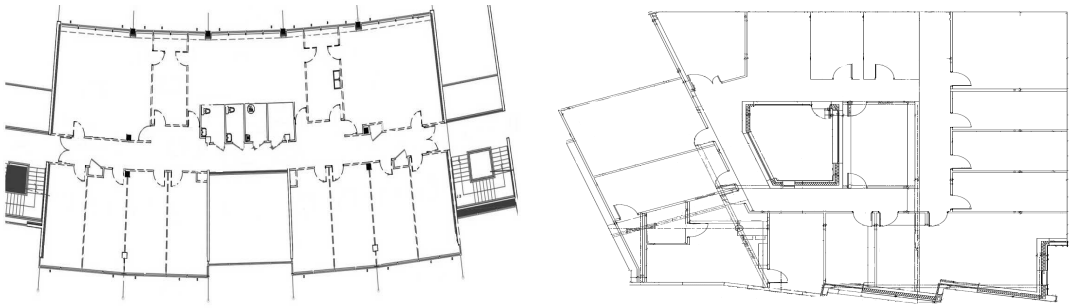
### 4.5.1 Experimental setups

Real experiments are realized in a WLAN environment at the first floor of the statistical and operational research department, and the first floor of the Living Lab at the University of Technology of Troyes, France. Figure 4.5 shows the layout plans of the two scenarios, which we will refer to as Experiment 1 and

Experiment 2 respectively. The layout plan of Figure 4.5(a) has an approximated area of 500 $m^2$, and is partitioned into 21 zones, with 23 detected APs. The layout plan of Figure 4.5(b) has an area of 550 $m^2$ and is partitioned to 19 zones, with 38 detected APs. Sets of 30 measurements are taken in each zone in random positions and orientations of the personal computer, and are used to construct each database. A new set of 20 measurements in each zone is collected after a month to test the proposed method, as measurements from the same day may be dependent. Table 4.2 summarizes the parameters of the two experimental setups.

### 4.5.2 Illustration of the proposed method

The collected RSSIs of the database are fitted parametrically or using the KDE. The basic observation model is then constructed by using the fitted distributions in the belief functions framework. This model is extended through the cluster-



(a) Layout plan 1. (b) Layout plan 2.

**Figure 4.5:** The first floor of the statistical and operational research department in (a) and the Living Lab in (b) at the University of Technology of Troyes, France.

**Table 4.2:** Experimental setup parameters.

| Parameter | Notation | Value | |
|---|---|---|---|
| | | Experiment 1 | Experiment 2 |
| Number of zones | $N_Z$ | 21 | 19 |
| Number of APs | $N_{AP}$ | 23 | 38 |
| Number of measurements per zone | $N_j$ | 30 | 30 |

ing algorithm presented in Section 4.4.1. By optimizing the intra-inter cluster distances, the dendrogram of clusters is cut at 7 and 6 clusters level, in Experiments 1 and 2 respectively. The clusters with the initial zones constituting them are considered, forming a two-level hierarchy. At each level, the AP selection technique presented in Section 4.4.2 is applied to determine the most useful APs. The proposed method EOM associates confidence levels to the zones of the targeted area. Table 4.3 compares the performance of the method as a function of the type of modeling. In both experiments, the performance achieved using KDE modeling is better, even though this comes at the cost of a higher computational complexity. The WiFi scanner software scans the network and measures the RSSIs at a 0.75 second interval. For that reason, we care more here about the overall accuracy, as the online execution time is less than the time needed by the software to scan the network.

We study the influence of the user-defined parameter $\eta$ employed in the AP selection technique on the performance of the method. Table 4.4 shows the overall accuracy and the online time of the localization algorithm for $\eta \in \{0.25, 0.5, 0.75\}$. The first row of the table corresponds to the performance of the method using the whole set of APs, without applying the AP selection algorithm. As the parameter $\eta$ increases, a larger weight is given to the discriminative capacity objective. This however adds redundant sources to the subset of APs, thus increasing the complexity and hence the online processing time. Decreasing the value of $\eta$ leads to removing redundant APs, thus reducing the complexity. However, the overall accuracy is severely affected especially for very low values of $\eta$.

Tables 4.5 and 4.6 show the results of applying the EOM to 420 and 380 test

**Table 4.3:** Influence of modeling on the overall accuracy and the processing time of the localization algorithm in Experiments 1 and 2.

| | Experiment 1 | | Experiment 2 | |
|---|---|---|---|---|
| Modeling | accuracy (%) | online time (s) | accuracy (%) | online time (s) |
| Parametric | 84.50 | 0.2183 | 85.42 | 0.2577 |
| KDE | **88.78** | 0.2418 | **90.21** | 0.2955 |

points in Experiments 1 and 2 respectively, and the influence of each phase on the overall accuracy and the processing time in both experiments. The accuracy over training data is considered alongside the accuracy over new data to show the highest accuracy the method can attain, and verify if there is an overfitting. In fact, if a significant difference between the accuracies on training data and on new data is noticed, this means there is overfitting in the used model. Besides, it is beneficial to keep an eye on the offline training time, since the computationally expensive training algorithms are not preferred. An estimation is said to be correct if the algorithm assigns the highest confidence level to the correct zone. As the table shows, modeling the data with associating and combining masses lead to an accuracy of less than 80% over new data. This low accuracy percentage is due to the wide overlapping of the various functions representing the distributions of the data in the different zones. In fact, this corresponds to the basic observation model previously discussed. However, when the two-level hierarchical clustering is carried out, a great enhancement in the overall accuracy is noted. This amelioration is at the expense of the processing time. It is clear that both the offline training time and the online localization time are almost doubled. In addition, the AP selection phase has a significant impact on the performance. An accuracy of around 90% is reached with a slight gain in the online test time, yet with an increase in the offline training time. Moreover, discounting the APs evidence raises the overall accuracy to 93% without a huge impact on the processing time.

**Table 4.4:** Influence of the parameter $\eta$ on the overall accuracy and the processing time of the localization algorithm in Experiments 1 and 2.

| | Experiment 1 | | Experiment 2 | |
|---|---|---|---|---|
| $\eta$ | accuracy (%) | online time (s) | accuracy (%) | online time (s) |
| - | 85.62 | 0.2674 | 86.44 | 0.3168 |
| 0.25 | 79.74 | 0.1872 | 83.89 | 0.2117 |
| 0.5 | 83.50 | 0.2259 | 86.11 | 0.2619 |
| 0.75 | **88.78** | 0.2418 | **90.21** | 0.2955 |

### 4.5.3 Comparison to state-of-the-art methods

In this section, the proposed method is compared to some of the well-known classification techniques. Flat classification methods such as k-nearest neighbors, naive Bayes, multinomial logistic regression (MLR), neural networks, and support vector machines (SVM) are considered. A 10-folded cross validation is used on the training database to train the classifiers and tune their parameters. The parameters that minimize the average error of all folds are considered. This aims to enhance the ability of the classifiers to generalize, with a better classification accuracy on the new data. For k-nearest neighbors, the optimal number of neighbors used to estimate the class membership is found to be 19 and 23 in Experiment 1 and Experiment 2 respectively. For naive Bayes and MLR, the maximum likelihood estimate is used to evaluate the probability of having the data instance belong to each class. As for neural networks, radial basis functions are used as activation functions for a one single hidden layer. The Gaussian kernel is used for SVM.

**Table 4.5:** Influence of each phase of the extended observation model on the accuracy and the complexity of the localization algorithm in Experiment 1.

| Experiment 1 | Accuracy (%) | | Time (s) | |
|---|---|---|---|---|
| EOM | training data | new data | offline | online |
| without clustering, AP selection, and discounting | 78.85 | 76.63 | 21 | 0.1007 |
| without AP selection and discounting | 87.21 | 84.26 | 56 | 0.2317 |
| without discounting | 88.52 | 86.73 | 76 | 0.2184 |
| as it is | **90.68** | **88.78** | 79 | 0.2418 |

**Table 4.6:** Influence of each phase of the extended observation model on the accuracy and the complexity of the localization algorithm in Experiment 2.

| Experiment 2 | Accuracy (%) | | Time (s) | |
|---|---|---|---|---|
| EOM | training data | new data | offline | online |
| without clustering, AP selection, and discounting | 83.90 | 81.61 | 39 | 0.1324 |
| without AP selection and discounting | 86.90 | 85.88 | 92 | 0.2736 |
| without discounting | 88.90 | 86.38 | 115 | 0.2577 |
| as it is | **92.68** | **90.21** | 119 | 0.2955 |

Moreover, the proposed method is compared with hierarchical techniques such as hierarchical support vector machines (HSVM) and random forests. Chen *et al.* [2004] develop an HSVM technique that solves a series of max-cut problems to recursively partition the classes into two-subsets, till pure leaf nodes that have only one class are obtained. Then, the classical SVM is applied to solve the binary two-subsets classification problem at each internal node. In addition, a random forest model is proposed in [Górak & Luckner, 2016] to localize sensors in indoor networks. Random forests is an ensemble of trees, obtained both by bootstrap sampling, and by randomly changing the feature set during learning [Breiman, 2001]. More precisely, at each node in the decision tree, a random subset of the input attributes is taken, and the best feature is selected from this subset instead of the set of all attributes. Górak & Luckner [2016] propose a straightforward random forest model and another modified one by defining a localization model for each AP that predicts the localization only when a signal is detected from that AP.

Tables 4.7 and 4.8 show the overall accuracy and the processing time of the proposed technique in the two experiments compared with these described methods. The proposed method outperforms all the other ones in terms of localization accuracy. On the other hand, its processing time is considered to be competitive to the others, yet with a clear advantage to naive Bayes for instance, and k-nearest neighbors that has no training phase. The indicated time for the latter is only to store the training data and calculate the optimal $k$ by a ten-folded cross validation.

### 4.5.4 Application in facial image recognition

The proposed method is in fact a classification technique. In the case of localization, the classes are the zones and the features are the APs. In the case of facial image recognition, the classes become the subjects whose faces are to be classified, and the features become a set of image pixels extracted from the images themselves.

Facial image recognition has gained a great attention in the recent years due to its

wide applications in video surveillance, database image matching, and security

**Table 4.7:** Comparison of performance between methods in terms of overall accuracy and processing time in Experiment 1.

| Experiment 1 | Accuracy (%) | | Time (s) | |
|---|---|---|---|---|
| Technique | training data | new data | offline | online |
| K-nearest neighbors | 83.42 | 82.83 | 11 | 0.1089 |
| Naive Bayes | 82.21 | 81.66 | 31 | 0.0883 |
| Multinomial logistic regression | 84.52 | 82.58 | 54 | 0.1198 |
| Neural networks | 85.68 | 84.72 | 68 | 0.1466 |
| Support vector machines | 88.68 | 85.55 | 79 | 0.1559 |
| Hierarchical support vector machines | 89.68 | 86.38 | 112 | 0.3771 |
| Random forests | 88.68 | 86.66 | 124 | 0.3966 |
| Basic observation model | 79.68 | 77.77 | 26 | 0.1018 |
| Extended observation model | **90.68** | **88.78** | 97 | 0.2418 |

**Table 4.8:** Comparison of performance between methods in terms of overall accuracy and processing time in Experiment 2.

| Experiment 2 | Accuracy (%) | | Time (s) | |
|---|---|---|---|---|
| Technique | training data | new data | offline | online |
| K-nearest neighbors | 80.42 | 78.82 | 14 | 0.1311 |
| Naive Bayes | 76.90 | 76.76 | 42 | 0.1042 |
| Multinomial logistic regression | 84.52 | 82.94 | 76 | 0.1559 |
| Neural networks | 86.81 | 85.82 | 88 | 0.1866 |
| Support vector machines | 88.77 | 86.47 | 96 | 0.1912 |
| Hierarchical support vector machines | 89.24 | 86.89 | 135 | 0.4077 |
| Random forests | 90.54 | 88.90 | 153 | 0.4667 |
| Basic observation model | 79.81 | 78.91 | 41 | 0.1374 |
| Extended observation model | **92.68** | **90.21** | 119 | 0.2955 |

measurements. The Extended Yale B [Lee *et al.*, 2005], the ORL [Samaria & Harter, 1994], and the AR [Martinez, 1998] face databases are used to evaluate the proposed classification method. Many state-of-the-art methods have been targeting these databases to solve the facial image recognition problem. In order to study the influence of the classification method only, the same data partition and feature extraction technique of each state-of-the-art method are taken.

The Extended Yale B database consists of 2414 frontal-face images for 38 individuals taken under various lighting conditions. Yang *et al.* [2013] investigate the use of Gabor features for sparse representation based classification with a learned Gabor occlusion dictionary. The authors randomly select half of the images for training (32 images per subject) and use the other half for testing. All images are cropped to $192 \times 168$. They demonstrate the results of the method versus the feature dimension while comparing to the classification techniques SVM, nearest neighbor, and linear regression classification. Khorsandi & Abdel-Mottaleb [2015] present a classification method based on a weighted sparse representation with dictionary learning. They also crop the images to $192 \times 168$, and use half of the images for training and the others for testing. The method uses the mutual information between the query sample and the training samples to give a weight for the latter to each class in the dictionary. We compare the results of these methods with those obtained by our proposed method using the same data portion and the same feature extraction technique. The results are presented in Tables 4.9 and 4.10.

The AR face dataset consists of more than 4000 images of 126 distinct subjects. Following the work of Huang *et al.* [2015], a subset of 1680 images for 120 subjects is constructed, where each image is $50 \times 40$ pixels. The authors propose a specific sparse representation-based classifier that incorporates the class information in the learning process. The method defines classes as groups that compete to represent the test sample. It considers $L_1$ and $L_2$ norm constraints to the classes and samples and is solved using convex optimization. Table 4.11 shows the results of different classification approaches.

The ORL database consists of 400 images of 40 subjects. Wang & Sun [2013]

propose a multiple kernel local Fisher discriminant analysis for face recognition. The authors present a method that searches for maximum discrimination between inter- and intra- classes scatters, producing nonlinear discriminant features with multiple base kernels. They select different numbers of images per individual to form the training set, and the rest for testing. The experiments are repeated 50 times and the average recognition accuracy is computed. All images are cropped and resized to $32 \times 32$ pixels, with 256 Grey levels per pixel. Each image is represented by a 1024-dimensional vector in the image space. The comparison results are shown in Table 4.12.

The results in the shown tables prove the competence of the proposed method as compared with well-known facial recognition techniques.

## 4.6 Conclusion

In this chapter, we extended the observation model presented in Chapter 3 to tackle the problem of zoning-based localization in wide surface areas with large number of zones. We proposed a clustering algorithm that aims at creating a two-level hierarchy of zones through agglomerative hierarchical clustering and

**Table 4.9:** Average face recognition accuracy (%) on the Extended Yale B database based on the Gabor feature robust representation. First and second best results are highlighted in bold.

| Method | Feature dimension | | | |
|---|---|---|---|---|
| | 56 | 120 | 300 | 504 |
| Nearest neighbors | 81.4 | 89.2 | 91.9 | 92.0 |
| Sparse representation classification | 92.6 | 95.6 | 97.4 | 97.9 |
| Support vector machines | 92.6 | 95.3 | 96.3 | 96.4 |
| Linear regression classification | **94.1** | 94.7 | 95.4 | 95.7 |
| Yang *et al.* [2013] | 92.7 | **95.6** | **97.9** | **99.0** |
| Proposed method | **93.9** | **96.2** | **98.3** | **98.7** |

optimization of the inter- and intra-cluster measures. We studied the parametric and KDE modeling and the execution of the clustering algorithm in each case.

**Table 4.10:** Average face recognition accuracy (%) on the Extended Yale B database based on weighted sparse representation. First and second best results are highlighted in bold.

| Method | Feature dimension | | | |
|---|---|---|---|---|
| | 30 | 56 | 120 | 504 |
| Nearest neighbors | 69.3 | 72.8 | 78.5 | 79.5 |
| Nearest subspace | **79.6** | 84.1 | 88.7 | 90.8 |
| Sparse representation classification | 75.7 | 84.8 | 93.9 | 96.8 |
| Khorsandi & Abdel-Mottaleb [2015] | 78.5 | **86.7** | **95.3** | **97.9** |
| Proposed method | **83.1** | **87.8** | **95.8** | **96.9** |

**Table 4.11:** Average face recognition accuracy (%) on the AR database.

| Method | Accuracy |
|---|---|
| Support vector machines | 68.10 |
| Linear regression classification | 68.75 |
| Sparse representation classification | 63.87 |
| Collaborative classification representation | 68.25 |
| Huang *et al.* [2015] | 77.14 |
| Proposed method | **79.64** |

**Table 4.12:** Average face recognition accuracy (%) on the ORL database.

| Method | Number of training images | | | |
|---|---|---|---|---|
| | 2 | 4 | 6 | 8 |
| Linear discriminant analysis | 78.2 | 84.9 | 88.6 | 95.5 |
| Locality preserving protection | 80.5 | 85.4 | 89.2 | 95.8 |
| Marginal Fisher analysis | 89.3 | 91.5 | 92.7 | 96.5 |
| Local Fisher discriminant analysis | 92.4 | 93.8 | 94.3 | 97.1 |
| Wang & Sun [2013] | **95.6** | **96.7** | **97.3** | **98.4** |
| Proposed method | **93.2** | **96.9** | **96.8** | **98.6** |

We then developed an AP selection algorithm, to be applied at each level of the created hierarchy, to choose the best subset of APs in terms of overall accuracy and redundancy. Afterwards, we described the confidence-based zone estimation by combining the obtained evidence at all levels in the belief functions framework. Finally, we illustrated the performance of the proposed method through experiments in two real scenarios, providing comparisons with other techniques. The proposed method, which is a classification technique at its core, was also validated in the domain of facial image recognition using well-known public databases.

Two main limitations of the proposed model are noticed at the accuracy and complexity levels. A major problem leading to most of the erroneous estimations are zones and areas that are symmetric to each other with respect to the APs. In such cases, the APs assign similar confidence levels to these zones, yielding erroneous estimations. To solve this issue, we present in Chapter 5 mobility models to be combined with the proposed observation model. In this way, erroneous estimations by the observed model can be corrected by taking into consideration the movement of the sensor, its next possible destinations, its trajectory, etc. In addition, the algorithm is more complex than other classical techniques in the domain, requiring at average double the time needed by them, which might be a serious issue especially in case of adjacent buildings and multi-floors, and dense APs networks. For such cases, we aim in Chapter 6 to enhance the basic observation model through a decentralized configuration instead of the hierarchical approach proposed here.

# Chapter 5

# Mobility-based Tracking

## Contents

*In this chapter, we propose a novel tracking technique that uses the mobility of sensors (MNs) with the previously described observation models in the belief functions framework to track the MNs in real time. Assuming a maximum speed of movement of MNs in the indoor environment, the next possible destinations of the MN are predicted, leading to a mobility model. The belief functions framework is used to propagate the previous step evidence till the current one. The mobility of the MN, along with information from the network, are used to obtain an accurate estimation of its position. The performance of the method is studied for different experimental scenarios.*

## 5.1 Introduction

The primary objective of MN tracking is to estimate the trajectory of a moving MN. As explained in Chapter 1, the tracking can be viewed as a sequential localization problem. Thus, it requires a real-time recursive location estimation algorithm. In Chapters 3 and 4, we performed localization using only RSSI information collected from the network. However, it is interesting to take advantage of additional information, to correct the estimated location. Such additional information can be inertial information, such as knowledge of past location and instantaneous speed, acceleration, or trajectory of the MN being tracked.

Tracking methods assume that the target motion can be represented by some known mathematical models that are sufficiently accurate. The most commonly used models are known as state-space models. We will call them here mobility models, considering that they depend on the previous state of the MN and the mobility to estimate the next state. It is noted that standalone mobility models are not recommended in practical applications, since uncertainty increases with time, and it becomes nearly impossible to have a confident estimation after a certain number of tracking trials. For that reason, the estimation should be updated at each time using information obtained by acquiring measurements from the network, referred to as an observation model. The problem of tracking becomes a matter of an estimation using a mobility model, and an estimation update using an observation model.

The simplest model for target tracking is the white-noise acceleration model. It assumes that the target acceleration is an independent process. The main attractive feature of this model is its simplicity. However, in practical applications, the acceleration is rarely independent with respect to time, and hence it is only applied when the movement is quiet small or random. Another simple model is the Wiener-process acceleration model that assumes that the acceleration is a process with independent increments, which means that it is supposed to be nearly constant. However, the assumption that the acceleration increment is independent with respect to time is hardly justifiable, except for its simplicity and mathematical tractability. A more general model is the polynomial one. The continuous trajectory can be approximated by an $n$-th degree polynomial to an arbitrary accuracy. Such a model amounts to assuming that the $n$-th derivative of the position is nearly constant. The two previously described models thus become special cases of the $n$-th degree polynomial model for $n = 1$ and 2 respectively. This model in its general setting does not appear very attractive for tracking, as it is difficult to develop an efficient method to determine systematically the coefficients of the polynomial. These three models have been thoroughly studied by Bar-Shalom *et al.* [2004]. Whenever white-noise models are not good enough, it is natural to consider a Markov process model. The Singer model assumes that the target acceleration is a zero-mean first-order stationary Markov process [Singer, 1970]. This model corresponds to a motion in between the nearly constant velocity and the nearly constant acceleration models [Kendrick *et al.*, 1981; Kumar & Zhou, 1984].

The tracking models are highly dependent on the choice of the state components, and thus the respective kinematic model. This is not a trivial problem, as the target dynamics, accuracy of approximations, and coordinate system, must all be taken into account. The Kalman filter assumes that the posterior density of the states at every time step is Gaussian, and thus it recursively computes the mean and the covariance of the Gaussian posterior [Mahfouz *et al.*, 2014]. It is the optimal solution to the tracking problem when the assumptions of the linear Gaussian environment holds. However, this posterior is not necessarily Gaussian, and hence the filter is not certain to be optimal. To address the non-linear cases,

several approaches have been proposed such as the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) [Zhang *et al.*, 2013]. The main feature of these methods is that they approximate the non-linear function in the state dynamic and the observation model. The estimation error of the EKF is usually large due to linearization. The UKF has a better tracking performance since it can better approximate the non-linearity as compared with the EKF. However, it is found that the estimation error of UKF increases when the target's range is far in practical applications. Another well-known method for target tracking relies on the particle filter (PF) or the Monte-Carlo filter [Hong *et al.*, 2014]. The key idea is to recursively represent the posterior density function by a set of random samples or particles with associated weights according to the measurements. As the number of particles increases, the PF becomes an equivalent representation of the usual functional description of the posterior density, and thus approaches the optimal Bayesian estimate.

In this chapter, we adapt the localization frameworks introduced in Chapters 3 and 4 in order to perform tracking in WSNs. Compared with the previous chapters, we now take advantage of the MN's mobility to enhance the obtained location estimate. The proposed tracking approach uses the belief functions theory to estimate the MNs zones by combining the evidence related to the MNs mobility and observations, as described in the following. The method makes use of the MNs mobility by assuming a maximum speed of movement of MNs in indoor environments. This allows a prediction of the next possible destinations of the MN, and hence leads to a mobility model. The belief functions framework is used to propagate the previous step evidence till the current one. Moreover, the MN measures its RSSI vector, and using the models described in Chapters 3 and 4, evidence is assigned to each zone of the targeted area. Mobility and observation evidence are then combined in the belief functions framework to determine a level of confidence of having the MN residing in each zone. To this end, three mobility-based tracking models are proposed. An enhancement in the overall accuracy of the zone estimates is recorded.

## 5.2 Problem Statement

The tracking problem consists in estimating the MN's zone in real time using its mobility and the signals strengths it collects from the surrounding APs. We adopt the same network configuration and notations as in the previous two chapters, where $N_Z$ is the number of zones of the targeted area, denoted by $Z_j$, $j = 1, 2, \ldots, N_Z$, $N_{AP}$ is the number of APs, denoted by $AP_k$, $k = 1, 2 \ldots, N_{AP}$, and $\boldsymbol{\rho}_t$ is the vector of RSSI measurements collected by the MN at the instant $t$ from surrounding APs. We suppose here $v_{max}$ to be the maximum speed of the MN in the indoor environment. In our application, $v_{max}$ is the maximum expected speed of movement of dependent elderly people in indoor networks. The aim of the proposed algorithm is to find a function $\mathbb{T} : \mathbb{R}^{N_{AP}} \rightarrow [0, 1]^{N_Z}$ such that $\mathbb{T}(\boldsymbol{\rho}_t) = (\mathcal{W}_t(Z_1), \ldots, \mathcal{W}_t(Z_{N_Z}))$, where $\mathcal{W}_t(Z_j)$ is the level of confidence of having the MN of observation $\boldsymbol{\rho}_t$ residing in the zone $Z_j$ at the instant $t$. It is worth noting that only one MN is considered here; however, the method can be applied in the same manner to as many MNs as needed.

On one hand, the method makes use of the mobility of the MN to track it in real time in indoor environments. Here, the MN is assumed to displace at a speed less than or equal a maximum speed $v_{max}$. By using this notion, one can have a clear idea about the next possible destinations of the MN depending on the architecture of the targeted area and the localization execution time. The first proposed mobility model is based on the original succession of zones and does not require any data acquisition phase. The second mobility model necessitates dividing the zones into sub-zones and collecting RSSI measurements in each one. The third mobility model is based on a hidden Markov model and uses the trajectory of the MN to update the zone estimate. All three mobility models thus affiliate some evidence $m_{\mathbb{M},t}(\cdot)$ to each zone.

On the other hand, the observation models described in Chapter 3 and 4 assign another evidence $m_{\mathbb{O},t}(\cdot)$ using the belief functions theory and the RSSI of the received signals. In a belief functions framework, all evidence is combined to determine at each instant a level of confidence $\mathcal{W}_t(\cdot)$ in each zone. Figure 5.1 illustrates the general framework of the proposed tracking technique, and Table 5.1

lists the variables used in this chapter, along with their respective sizes. In the following, we present the three mobility models, each with its architecture, mass association, and confidence-based zone estimation through combination with the observation models.

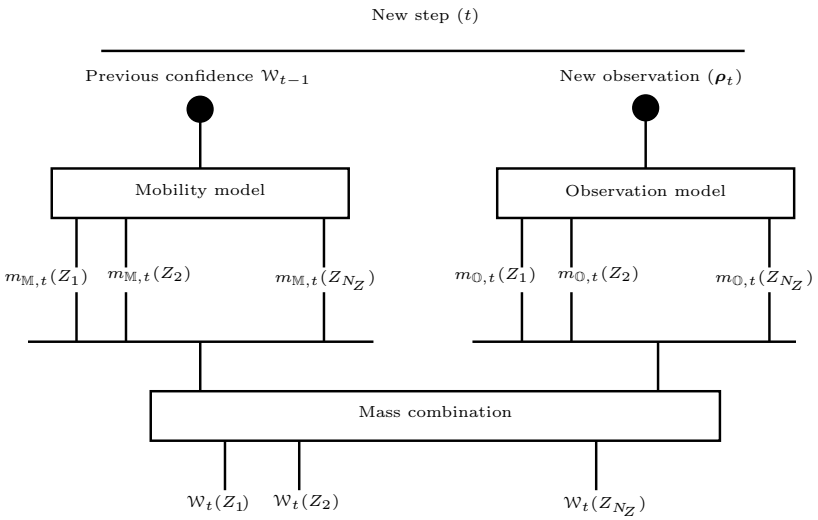



**Figure 5.1:** Belief functions framework to combine evidence from both observations and mobility.

**Table 5.1:** List of the variables used in Chapter 5, with their respective sizes.

| Notation | Variable | Size |
|---|---|---|
| $N_Z$ | number of zones | 1 |
| $N_S$ | number of states of HMM | 1 |
| $R, S$ | sequences | $\alpha$ |
| $\alpha$ | length of sequence | 1 |
| $d$ | distance | 1 |
| $v_{max}$ | maximum speed | 1 |
| $Q(\cdot)$ | distribution | 1 |
| $A$ | transition matrix | $N_S \times N_S$ |
| $B$ | emission model | $N_S \times 1$ |
| $\pi$ | initial state probabilities | $N_S \times 1$ |
| $g, h$ | forward and backward probabilities | $N_S \times 1$ |
| $m_{\mathbb{O}}(\cdot), m_{\mathbb{M}}(\cdot), \mathcal{W}(\cdot)$ | assigned evidence | $N_Z \times 1$ |
| $\mathbb{T}(\cdot)$ | tracking model | $N_Z \times 1$ |

## 5.3 First Mobility-based Tracking Model

The first mobility-based tracking model uses the maximum speed of the MN and the original succession of the zones in the targeted area, without any additional data acquisition phase, to determine the zone of the MN. We first describe the architecture of the mobility model, and then we explain how it assigns evidence to each zone. Finally, we show how the evidence assigned by the mobility model is combined with that of the observation model to determine a level of confidence of having the MN residing in each zone.

### 5.3.1 Architecture

Let $\Delta t_{loc}$ denote the time interval in which the localization algorithm is executed, and $d_{min,ij}$ denote the minimal geographical distance between two zones $Z_i$ and $Z_j$. The distance $d_{min,ij}$ is the shortest distance the MN is forced to travel to reach a point in one of the two zones, from where it then must travel to the other zone. In other words, it is the smallest of all the distances from a point in zone $Z_i$ to the closest point in the other zone $Z_j$. The maximum distance that the MN can travel is then deduced as $d_{max} = v_{max} \times \Delta t_{loc}$. Figure 5.2 illustrates the architecture of the first mobility-based tracking model, where the MN is in zone $Z_i$ at instant $t-1$, and we are interested in determining its zone $Z_j$ at instant $t$.

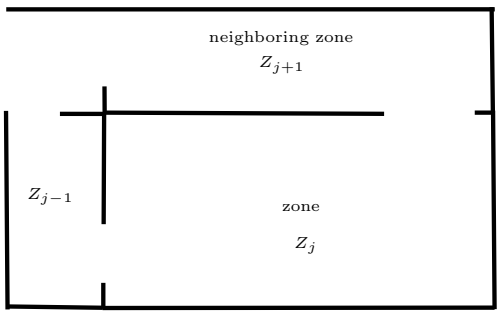



**Figure 5.2:** Illustration of the architecture of the first mobility model.

### 5.3.2 Mass association

Let $p_{ij}, i, j \in \{1, \ldots, N_Z\}$, denote the coefficient of transition from zone $Z_i$ to zone $Z_j$ within the localization period $\Delta t_{loc}$. Then,

$$p_{ij} = \begin{cases} 0, & \text{if } d_{max} < d_{min,ij}; \\ 1, & \text{if } d_{max} \geq d_{min,ij}. \end{cases} \tag{5.1}$$

Being in a zone $Z_i$ at instant $t-1$, the MN can be at instant $t$ in any zone $Z_j, j \in \{1, ..., N_Z\}$, of the ones having $p_{ij} = 1$. These zones are called the *following zones* of $Z_i$. The confidence $\mathcal{W}_{t-1}(Z_i)$ at time $t-1$ is then propagated to time $t$ by distributing it equally to its following zones, each one having $\frac{\mathcal{W}_{t-1}(Z_i)}{\sum_j p_{ij}}$. The mobility evidence given to a zone $Z_j$ at time $t$ is the aggregation of all evidence deduced from its preceding ones having $p_{ij} = 1$, $\forall i \in \{1, ..., N_Z\}$. This leads to a mobility mass at time $t$ computed in the following manner,

$$m_{\mathbb{M},t}(Z_j) = \sum_{i=1}^{N_Z} p_{ij} \times \frac{\mathcal{W}_{t-1}(Z_i)}{\sum_{f=1}^{N_Z} p_{if}}. \tag{5.2}$$

### 5.3.3 Confidence-based zone estimation

As described in Chapters 3 and 4, the observation model $\mathbb{O}$ assigns an evidence $m_{\mathbb{O},t}(\cdot)$ for each zone. The assigned evidence by the mobility model and the observation model are combined by aggregating the evidence of the two models to yield a confidence level $\mathcal{W}_t(\cdot)$ of having the MN residing in each zone at each instant $t$ as follows,

$$\mathcal{W}_t(Z_j) = m_{\mathbb{M} \bigoplus \mathbb{O},t}(Z_j) = \frac{m_{\mathbb{O},t}(Z_j) \times m_{\mathbb{M},t}(Z_j)}{\sum_{\chi=1}^{N_Z} m_{\mathbb{O},t}(Z_\chi) \times m_{\mathbb{M},t}(Z_\chi)}. \tag{5.3}$$

The advantage of this tracking model is that it is simple and does not require any additional data acquisition phase. In addition, one does not need inertial measurement units to keep track of the mobility of the MN. Prior knowledge about the maximum speed corresponding to the practical application can be considered. Inertial measurement units might be only used to collect this information in an offline phase in case it is not known.

## 5.4 Second Mobility-based Tracking Model

The accuracy of the first mobility model can be enhanced by taking advantage of our application, which is tracking of dependent elderly people. Although this model can be used for any other application, the maximum speed in our particular application is relatively low, allowing us to create transition zones between each pair of original consecutive zones. The model is designed in a way that the MN cannot move from a zone to another without passing through a transition zone within a localization period.

### 5.4.1 Architecture

Each zone $Z_i$ is divided into $N_{X_i}$ sub-zones $X_{i.\ell}$ according to its layout, $N_{X_i} - 1$ connection sub-zones and one main sub-zone $X_{i.N_{X_i}}$. Each connection sub-zone is a section area in front of a doorway connecting $Z_i$ with neighbor zones, as shown in Figure 5.3. Its dimensions are defined in a way to cover all possible positions in the zone $Z_i$ at which the MN can cross the doorway to go to a neighboring zone within the localization period. The main sub-zone $X_{i.N_{X_i}}$ is the remaining section area in $Z_i$. A MN being in the main sub-zone of zone $Z_i$ at time $t-1$, will remain in the same zone at time $t$. In the data acquisition phase, RSSI values are collected in each sub-zone and fitted to a multi-dimensional statistical distribution. Let $Q_{i.\ell}(\cdot)$ be the distribution representing the data of the connection sub-zone $X_{i.\ell}, \ell \in \{1, \ldots, N_{X_i} - 1\}$, and $Q_{i.N_{X_i}}(\cdot)$ be the distribution representing the main sub-zone $X_{i.N_{X_i}}$.

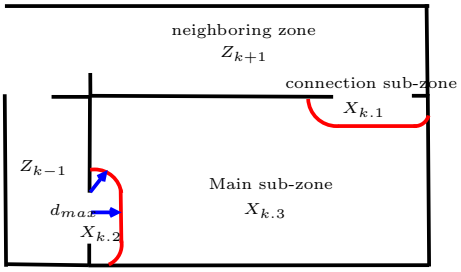



**Figure 5.3:** Illustration of the architecture of the second mobility model.

### 5.4.2 Mass association

Having the previous observation $\boldsymbol{\rho}_{t-1}$, membership weights $q_{t-1}(X_{i.\ell})$, $i \in \{1,\ldots,N_Z\}$ and $\ell \in \{1,\ldots,N_{X_i}\}$, are computed to quantify the membership of the MN to any sub-zone of each zone $Z_i$ at $t-1$. This is performed by evaluating the probability resulting from each fitted distribution with respect to the previous observation followed by a normalization phase,

$$q_{t-1}(X_{i.\ell}) = \frac{Q_{i.\ell}(\boldsymbol{\rho}_{t-1})}{\sum_{\chi=1}^{N_{X_i}} Q_{i.\chi}(\boldsymbol{\rho}_{t-1})}. \tag{5.4}$$

The confidence of each zone $Z_i$ at $t-1$ is then converted to its sub-zones,

$$\mathcal{W}^*_{t-1}(X_{i.\ell}) = \mathcal{W}_{t-1}(Z_i) \times q_{t-1}(X_{i.\ell}). \tag{5.5}$$

Let $r_{i.\ell,j}$ be the coefficient of transition from the connection sub-zone $X_{i.\ell}, \ell \in \{1,\ldots,N_{X_i}\}$ of zone $Z_i, i \in \{1,\ldots,N_Z\}$, to original zone $Z_j, j \in \{1,\ldots,N_Z\}$,

$$r_{i.\ell,j} = \begin{cases} 0, & \text{if } \ d_{max} < d_{min,i.\ell,j}; \\ 1, & \text{if } \ d_{max} \geq d_{min,i.\ell,j}, \end{cases} \tag{5.6}$$

where $d_{min,i.\ell,j}$ is the minimal distance between connection sub-zone $X_{i.\ell}$ and zone $Z_j$. The mass associated to each zone by the mobility model can be thus deduced,

$$m_{\mathbb{M},t}(Z_j) = \mathcal{W}^*_{t-1}(X_{j.N_{X_j}}) + \sum_{i=1}^{N_Z} \sum_{\ell=1}^{N_{X_i}-1} r_{i.\ell,j} \times \frac{\mathcal{W}^*_{t-1}(X_{i.\ell})}{\sum_{f=1}^{N_Z} r_{i.\ell,f}}. \tag{5.7}$$

The mass assigned to $Z_j$ at time $t$ is the aggregation from $t-1$ of the confidence of its main zone and a part of the confidence of the connection sub-zones of other zones that are capable of leading to it.

### 5.4.3 Confidence-based zone estimation

In the same manner as in the first model, the confidence level $\mathcal{W}_t(\cdot)$ of having the MN resides in each zone is obtained as follows,

$$\mathcal{W}_t(Z_j) = m_{\mathbb{M} \bigoplus \mathbb{O},t}(Z_j) = \frac{m_{\mathbb{O},t}(Z_j) \times m_{\mathbb{M},t}(Z_j)}{\sum_{\chi=1}^{N_Z} m_{\mathbb{O},t}(Z_\chi) \times m_{\mathbb{M},t}(Z_\chi)}. \tag{5.8}$$

## 5.5 Third Mobility-based Tracking Model

In this tracking approach, we construct our mobility model using a hidden Markov model (HMM). The idea is to create, for each transition between two consecutive zones, a trajectory that the MN follows to move from one zone to another. A HMM is constructed for each trajectory, yielding a probability that the MN has followed it. This probability is then combined with the evidence assigned by the observation model to determine a level of confidence that the MN resides in each zone. At first, we provide a general overview of the HMMs. We then present the architecture of the proposed mobility model, the way we use it to assign evidence, and the confidence-based zone estimation through combination with the observation model.

### 5.5.1 Hidden Markov Models

We provide here a general overview of the HMM, its definition, its parameters, and its statistical theory.

#### 5.5.1.1 Definition

A stochastic system is said to be a Markov process if the next state depends only on the present state. A HMM is a probabilistic model that can be used for representing a sequence of observations, and these observations can be either discrete or continuous, and can be either time dependent or independent [Eddy, 1996]. We start by presenting the following example.

**Ex 5.1.** Let us assume that someone is inside a closed room with no information about the weather outside. He receives postal mails through the mail slot every day of the week. The only way by which he can guess the weather is by looking at the state of the mails. For example, he expects the mails to be dry on a sunny day, and to be wet on a rainy day. Suppose that the aim is to model the weather as being sunny, soggy, or rainy based on the condition of the received mail. A three-state HMM model can be used. Here, the weather forms the states "sunny", "soggy", or "rainy". These can be represented by a set of hidden discrete states $S = \{\text{sunny, soggy, rainy}\}$. These states are shown as circles in

Figure 5.4. The reason why the states are described as hidden is because the observer is unaware of the nature of the states. In this example, assuming that the system is a Markov process, means that the weather of the next day depends only on today's weather.

With the states being hidden and the next state depending only on the previous state, the model in the above example is thus a HMM of the first order. A second order HMM assumes that the present state depends on the two previous states, and the third order HMM considers three previous states and so on.

#### 5.5.1.2 Parameters

Suppose we have an $N_S$-state HMM model $\Lambda$, where $N_S$ is the total number of states denoted $S = \{s_1, s_2, \ldots, s_{N_S}\}$. Whenever a sequence of length $\alpha$, $R = \{R_1, R_2, \ldots, R_\alpha\}$, is observed, the objective of the HMM is to determine the corresponding state sequence $S = \{s_1, s_2, \ldots, s_\alpha\}$. One of the three primary parameters of any HMM is the transition probability, which designates the probability of arriving at the next states for each present state. In Ex 5.1, it denotes

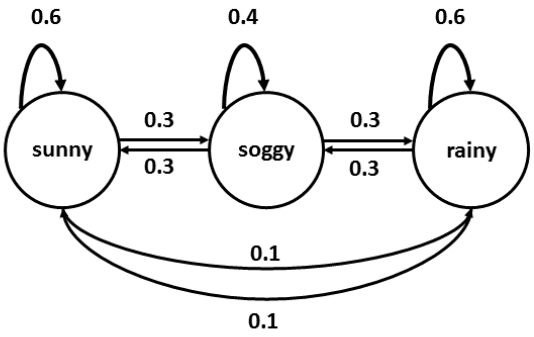



**Figure 5.4:** Illustration of a three state HMM - Transition probabilities.

**Table 5.2:** Illustration of a three state HMM - Emission probabilities.

| probability | weather | | |
|---|---|---|---|
| mail | sunny | soggy | rainy |
| dry | 0.9 | 0.3 | 0.1 |
| wet | 0.1 | 0.7 | 0.9 |

for instance, the probability of tomorrow being "rainy" if today is "sunny", which is 0.1 in Figure 5.4. These probabilities can be obtained from the data of weather patterns for the past years. Since it is a first order Markov process, we can represent the probability of all possible transitions from one state to another as a matrix $A$, which is termed as the *transition matrix*.

**Ex 5.2.** For the above example, the matrix $A$ will be,

$$A = \begin{pmatrix} 0.6 & 0.3 & 0.1 \\ 0.3 & 0.4 & 0.3 \\ 0.1 & 0.3 & 0.6 \end{pmatrix}$$

where each row corresponds to a present state, and each column corresponds to a next state of the HMM.

Since the model is a HMM, the actual states are hidden from the observer. In Ex 5.1, the observer is in a closed room and is unaware of the actual weather outside, and the only observable data is the dampness of the mails received each day. The observer classifies the mail as either being "Dry" or "Wet", and further proceeds on to model the possible state of weather for that day. One can generate a probability of each type of observation in each state. These probabilities are termed as *emission probabilities* and form matrix $B$.

**Ex 5.3.** In the emission probability matrix given below

$$B = \begin{pmatrix} 0.9 & 0.3 & 0.1 \\ 0.1 & 0.7 & 0.9 \end{pmatrix}$$

the rows correspond to the condition of the mail and the columns correspond to the three states of the weather outside.

Another parameter of the HMM is $\pi$, which is the set of probabilities of starting at different states. $\pi$ can be either uniform, random, or any vector generated from prior knowledge. Therefore, any HMM can be defined as $\Lambda = (A, B, \pi)$. We denote $A = \{a_{ij}\}_{i,j=1}^{N_S}$ the transition probability from state $i$ to $j$, and $B = \{b_i(R)\}_{i=1}^{N_S}$ the output probability distribution [Oudelha & Ainon, 2010].

#### 5.5.1.3 Statistical Theory of HMM

HMMs have a strong statistical basis and are used in different aspects. We cover here one aspect, which will be used in our application. Given a $N_S$-state HMM model $\Lambda$, and an observation sequence $R$, the aim is to evaluate the probability of observing the sequence $P(R|\Lambda)$. This is a problem of evaluating the observed sequence when we know the parameters of the HMM. In the simplest form, we can break down the evaluation of $P(R|\Lambda)$ as follows. Given a state sequence $S = \{s_1, \ldots, s_\alpha\}, 1 \leq \alpha \leq N_S$, we can compute the joint probability of the observed sequence and the state sequence,

$$P(R, S|\Lambda) = P(R|S, \Lambda) \times P(S|\Lambda). \tag{5.9}$$

This is the product of the probability of the observation sequence $R$ given $S$, and the probability of the state sequence $S$ given the model. The first term is obtained from the emission matrix $B$ as,

$$P(R|S, \Lambda) = \prod_{f=1}^{\alpha} b_{s_f}(R_f). \tag{5.10}$$

The second term is obtained from the transition matrix $A$ as,

$$P(S|\Lambda) = \prod_{f=1}^{\alpha} a_{s_{f-1}s_f}. \tag{5.11}$$

We can then derive $P(R|\Lambda)$ as the summation of $P(R, S|\Lambda)$ over all possible state sequences $S$ [Yen *et al.*, 1997],

$$P(R|\Lambda) = \sum_{\text{for all } S} P(R, S|\Lambda) = \sum_{\text{for all } S} \prod_{f=1}^{\alpha} a_{s_{f-1}} a_{s_f} b_{s_f}(R_f). \tag{5.12}$$

The total number of state paths increases quickly with the length of the sequence, and thus becomes computationally expensive and not feasible depending on $\alpha$. However, there exists a forward–backward algorithm that can be used to obtain $P(R|\Lambda)$, and reduces the computational burden. The expression in equation (5.12) can be transformed as follows,

$$P(R|\lambda) = \sum_{y=1}^{N_S} P(R_1, \ldots, R_f, s_f = y|\Lambda).P(R_{f+1}, \ldots, R_\alpha|s_f = y, \Lambda). \tag{5.13}$$

The probability of observing the sequence can thus be determined using forward and backward probabilities,

$$g_f(y) = P(R_1, \ldots, R_f, s_f = y|\Lambda). \tag{5.14}$$

$$h_f(y) = P(R_{f+1}, \ldots, R_\alpha|s_f = y, \Lambda). \tag{5.15}$$

The probabilities in equations (5.14) and (5.15) can be computed recursively,

$$g_{f+1}(y) = \left[\sum_{x=1}^{N_S} g_f(x)a_{xy}\right] b_y(R_{f+1}); \tag{5.16}$$

$$h_f(y) = \sum_{x=1}^{N_S} a_{yx}b_x(R_{f+1})h_{f+1}(x). \tag{5.17}$$

Therefore, the probability $P(R|\Lambda)$ is given by,

$$P(R|\Lambda) = \sum_{y=1}^{N_S} g_f(y)h_f(y). \tag{5.18}$$

This is the probability of observing a sequence $R$ of length $\alpha$ given an $N_S$-states HMM $\Lambda$.

### 5.5.2 Architecture

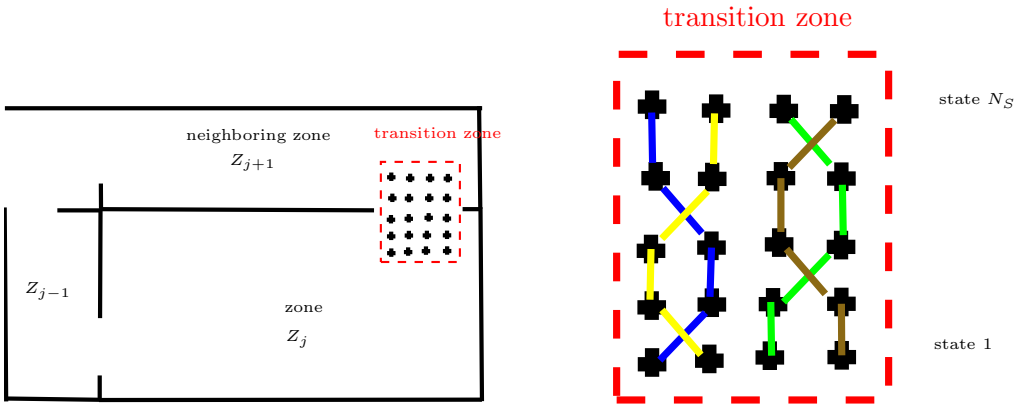
In this tracking approach, we make use of the trajectory of the MN in indoor environments. The objective is to detect a transition of the MN from any zone to another in a period of time. For that reason, we use the HMMs to determine a probability or likelihood that the MN has followed a certain trajectory. Each HMM $\Lambda$ is defined by three parameters, $\Lambda = (A, B, \pi)$, where $A$ is the transition matrix, $B$ is the emission model, and $\pi$ is the vector of initial state probabilities. As a state sequence $S = \{s_1, \ldots, s_\alpha\}$ is determined, we can observe a sequence $R = \{R_1, \ldots, R_\alpha\}$ since the states are hidden, corresponding to a vector of RSSI measurements at each state. We are interested in determining the probability $P(R|\Lambda)$, which is the probability of observing the sequence $R$, given the HMM model $\Lambda$. This probability is used as an evidence to be combined with that

obtained by the observation model $\mathbb{O}(\cdot)$ to determine a level of confidence of having the MN residing in each zone.

We construct a set of HMMs denoted as $\Lambda_{ij}, i, j \in \{1, \ldots, N_Z\}$, where $\Lambda_{ij}$ is an $N_S$-state HMM corresponding to a transition region or trajectory between zones $Z_i$ and $Z_j$. The parameter $N_S$ is the number of states chosen by the user in each transition region. In the offline phase, a transition region between each pair of neighboring zones is created as shown in Figure 5.5(a). This region is divided into $N_S$ states. At each state, a set of RSSI measurements is collected. Trajectories or sequences are constructed by randomly selecting a measurement from each state, as shown in Figure 5.5(b). All these constructed trajectories are considered as a database for each HMM.

The parameters of each HMM $\Lambda_{ij} = (A, B, \pi)$ are calculated as follows:

- Since at each state, except for the first and the last where there is only two options, the MN can equiprobably stay in its position, move to the state upfront, or move to the state behind, we define the $N_S \times N_S$ transition




(a) Construction of transition region between neighboring zones.

(b) Creation of $N_S$-state HMM from trajectories in each transition region.

**Figure 5.5:** Illustration of the architecture of the third mobility model.

matrix $A$ as,

$$A = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & \dots & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

- The emission model of each sequence is computed by modeling the offline collected RSSI measurements of each sequence with a multi-dimensional distribution as discussed in Section 3.3.1.
- Unless there is a prior knowledge regarding the starting state of the MN, the vector $\pi$ is defined as $\pi = [\frac{1}{N_S}, \dots, \frac{1}{N_S}]$.

### 5.5.3 Mass association

The objective of the proposed mobility model is to assign a mass or evidence that the MN has followed a trajectory, which is a transition between a zone and another. After constructing a HMM $\Lambda_{ij}$ for each transition between two zones as explained in the previous paragraph, it is time to use these HMMs to estimate the trajectory of the MNs. Once a sequence $R = \{R_1, \dots, R_\alpha\}$ is detected, each HMM $\Lambda_{ij}$ assigns a probability or likelihood that the MN has followed the state sequence corresponding to that HMM. The probabilities $P(R|\Lambda_{ij}), i, j \in \{1, \dots, N_Z\}$, are computed for each HMM using the equations (5.9) till (5.18). Thus, the probability of transitioning from any zone to another is computed. For pairs of zones where no transition is possible, the probability is zero. We define the transition coefficient $p_{ij}, i, j \in \{1, \dots, N_Z\}$, from $Z_i$ to $Z_j$ as follows,

$$p_{ij} = \begin{cases} P(R|\Lambda_{ij}), & \text{if } i \neq j; \\ 1 - \sum_{j=1}^{N_Z} P(R|\Lambda_{ij}), & \text{if } i = j. \end{cases} \tag{5.19}$$

The reason why we compute $p_{ij}$ for $i = j$ as such, is because we consider that the probability of the MN staying in the same zone, is the complement of all the probabilities that the MN move from that zone to all other zones. We cannot use the obtained derivations to determine a mass $m_{\mathbb{M},t}(\cdot)$ as we did in the previous two mobility models. This is because the obtained probabilities resemble a transition

from a zone to another only and not evidence that the MN is any zone at the current instant. In addition, the determination of $p_{ij}$ for $i = j$, as done here is not valid, except by the way we are transferring this probability to the next instant, and its combination with the observation model. In fact, if there is no transition, we will obtain large transition coefficients $p_{ii}$, for all $i \in \{1, \ldots, N_Z\}$, which results in false estimations.

### 5.5.4 Confidence-based zone estimation

The propagation of the probability assigned by the HMMs to the next instant is essential, especially that it allows the calculation of the transition coefficient as explained. The level of confidence assigned by the tracking model $\mathbb{T}$ to each zone $Z_j$ is computed as follows,

$$\mathcal{W}_t(Z_j) = \sum_{i=1}^{N_Z} m_{\mathbb{O},t-1}(Z_i) \times p_{ij}, \tag{5.20}$$

where $m_{\mathbb{O},t-1}(Z_i)$ is the mass associated by the observation model $\mathbb{O}(\cdot)$ at instant $t-1$. If there is no transition from a zone to another, the coefficient $p_{ij}$ for $i = j$ will be large as mentioned in the previous paragraph. This does not cause a problem when used in equation (5.20) because, if there is no transition, all coefficients $p_{ij}$ for $i = j$ will be large, and thus the mass $m_{\mathbb{O},t-1}(\cdot)$ is the deciding evidence. The zone already having a high confidence in the previous instant will still have high confidence, relative to the other zones.

## 5.6 Experimental Results

In this section, we evaluate the performance of the proposed tracking approaches through experiments in a real environment. We study the influence of each of the three mobility models when combined with each of the two observation models. We then study the influence of the parameters of the mobility models, which are $v_{max}$ and $\alpha$.

### 5.6.1 Experimental Setup

Experiments are realized in the same WLAN environment described in Chapter 4, which are the statistical and operational research department, and the Living Lab at the University of Technology of Troyes, France. The layout plans of the two environments are shown in Figure 4.5. The layout plan of Experiment 1 has an approximated area of 500 $m^2$, and is partitioned into 21 zones, with 23 detected APs. The layout plan of Experiment 2 has an area of 550 $m^2$ and is partitioned to 19 zones, with 38 detected APs. As described previously in Chapter 4, we refer to the observation model proposed in Chapter 3, as a basic observation model (BOM), and to the observation model proposed in Chapter 4, as an extended observation model (EOM).

### 5.6.2 Evaluation of performance

To evaluate the performance of the mobility-based tracking approaches, 10 trajectories of 50 observations each are considered. An important parameter in our models is the maximum speed of movement $v_{max}$. In our application, we aim to track dependent elderly people. For that reason, we use information regarding this group of people. Bohannon [1997] studies the comfortable and maximum walking speeds of 230 healthy adults aged 20-79 years. The corresponding technical term used to express the comfortable speed of the person is the gait speed. The study finds the following results. The mean comfortable gait speed for women in their 70s is $1.27m/s$, while it is $1.33m/s$ for men. The maximum gait speed for the same age group is $1.74m/s$ for women, while it is $2.07m/s$ for men. Montero-Odasso *et al.* [2004] study the gait speed of 100 elderly patients aged above 75 years. A maximum gait speed is found to be $0.8m/s$. Graham *et al.* [2010] study the walking speed of 174 in-hospital elderly people aged above 65 years. The study reports a maximum speed of $1.3m/s$ for relatively healthy people, while $0.7m/s$ for people suffering from poor health. We consider here $v_{max} = 1m/s$, as a maximum average speed of dependent elderly people in indoor networks; the influence of this parameter is studied in Section 5.6.2.2 with several values.

#### 5.6.2.1 Influence of mobility models

We study here the influence of the mobility models on the performance of the tracking approach when combined with the observation models BOM and EOM, for a maximum speed $v_{max} = 1m/s$. The time of execution of the localization algorithm is $\Delta t_{loc} = 0.75s$, which is the time needed by the software to scan the network and measure the RSSIs of the surrounding WiFi APs. In each transition region between neighboring zones, an 10-state HMM is created for the third mobility model. There is no general rule to determine the best number of states in our application. However, it is not important to determine it exactly. Any number of states that cover the transition region between the zones is acceptable. What really counts, is the chosen length of the observed sequence $\alpha$. We consider here a sequence length $\alpha = 8$. This means that after receiving a sequence of 8 observations, we use the HMMs to determine the probability that the MN has followed the 8-state sequence of the 10-state HMMs.

Table 5.3 shows the influence of the three mobility models on the performance of the tracking approach when combined with the BOM and EOM. As the table shows, an enhancement in the overall accuracy is noted when the observation models are combined with the proposed mobility models, yet at the expense of the processing time. The enhancement carried by the first mobility model is due to the presence of diametrically opposite erroneous zones with respect to APs, which are recovered by this model. A significant amelioration in the overall accuracy is noted upon using the second mobility model. The advantage of the latter is in the high accuracy achieved in assigning masses for sub-zones, due to the large number of APs selected inside each region. The third mobility model carries the most important enhancement on the overall accuracy. However, it adds a non-negligible computational complexity to the tracking algorithm.

#### 5.6.2.2 Influence of $v_{max}$

In this paragraph, we study the influence of the assumed maximum speed. In the previous section, it was supposed to be $v_{max} = 1m/s$. Here, we vary the maximum speed from $0.5m/s$ to $2.5m/s$, and study the performance of the first and second mobility models. Tables 5.4 and 5.5 show the influence of the maximum speed

$v_{max}$ on the overall accuracy, for Experiments 1 and 2 respectively. As the tables show, the overall accuracy of the tracking algorithm decreases as the maximum speed increases. This is due to the fact that the MN is supposed to cover larger distances in a shorter time, thus increasing the ambiguity of its next possible destinations. This makes the proposed method more convenient for tracking of dependent elderly people, where the maximum speed is supposed to be low.

#### 5.6.2.3 Influence of $\alpha$

In this paragraph, we study the influence of the length of the sequence $\alpha$. We vary $\alpha$ from 2 to 8, and study the performance of the third mobility model. Table 5.6 shows the influence of the sequence length $\alpha$ on the overall accuracy and the execution time. As the table shows, the overall accuracy of the tracking algorithm generally increases as the sequence length increases. This is due to augmenting the amount of information used by the HMMs to determine the likelihood that the MN has followed the different trajectories. However, enlarging the sequence length increases the execution time.

**Table 5.3:** Influence of the mobility models on the overall accuracy and the processing time of the tracking algorithm in Experiments 1 and 2.

| | Experiment 1 | | Experiment 2 | |
|---|---|---|---|---|
| Model | accuracy (%) | online time (s) | accuracy (%) | online time (s) |
| BOM | 77.77 | 0.1018 | 78.91 | 0.1374 |
| BOM + 1st mobility model | 78.54 | 0.1327 | 82.52 | 0.1616 |
| BOM + 2nd mobility model | 82.44 | 0.1561 | 84.71 | 0.1844 |
| BOM + 3rd mobility model | **84.91** | 0.1944 | **86.88** | 0.2078 |
| EOM | 88.78 | 0.2418 | 90.21 | 0.2955 |
| EOM + 1st mobility model | 90.38 | 0.2782 | 91.58 | 0.3162 |
| EOM + 2nd mobility model | 92.96 | 0.2956 | 93.40 | 0.3449 |
| EOM + 3rd mobility model | **94.48** | 0.3548 | **95.21** | 0.3911 |

**Table 5.4:** Influence of the maximum speed $v_{max}(m/s)$ on the accuracy (%) of the first and second mobility-based tracking approach in Experiment 1.

| Experiment 1 | $v_{max}(m/s)$ | | | | |
|---|---|---|---|---|---|
| Model | 0.5 | 1 | 1.5 | 2 | 2.5 |
| BOM + 1st mobility model | 79.89 | 78.54 | 78.32 | 77.24 | 76.84 |
| BOM + 2nd mobility model | **84.18** | 82.44 | 81.69 | 80.03 | 78.87 |
| EOM + 1st mobility model | 91.85 | 90.38 | 89.98 | 89.11 | 88.95 |
| EOM + 2nd mobility model | **94.07** | 92.96 | 91.12 | 90.87 | 89.19 |

**Table 5.5:** Influence of the maximum speed $v_{max}(m/s)$ on the accuracy (%) of the first and second mobility-based tracking approach in Experiment 2.

| Experiment 2 | $v_{max}(m/s)$ | | | | |
|---|---|---|---|---|---|
| Model | 0.5 | 1 | 1.5 | 2 | 2.5 |
| BOM + 1st mobility model | 83.47 | 82.52 | 82.48 | 81.91 | 81.73 |
| BOM + 2nd mobility model | **85.28** | 84.71 | 83.56 | 82.36 | 81.97 |
| EOM + 1st mobility model | 92.30 | 91.58 | 91.33 | 90.92 | 90.30 |
| EOM + 2nd mobility model | **94.79** | 93.40 | 92.88 | 91.19 | 90.57 |

**Table 5.6:** Influence of the sequence length $\alpha$ on the performance of the third mobility-based tracking approach in Experiments 1 and 2.

| | Experiment 1 | | Experiment 2 | |
|---|---|---|---|---|
| $\alpha$ | accuracy (%) | online time (s) | accuracy (%) | online time (s) |
| 2 | 88.95 | 0.2658 | 90.46 | 0.3055 |
| 4 | 90.35 | 0.2969 | 91.62 | 0.3287 |
| 6 | 92.77 | 0.3288 | 93.75 | 0.3569 |
| 8 | 94.48 | 0.3548 | 95.21 | 0.3911 |
| 10 | **94.62** | 0.3967 | **95.83** | 0.4344 |

## 5.7 Conclusion

In this chapter, we presented mobility-based algorithms for tracking for sensors in indoor wireless networks. At first, we proposed three tracking models that make use of the mobility of the sensor to propagate the evidence from a previous instant to the current one. The first mobility model is based on the original succession of the zones and uses the maximum speed to track the sensor. The second mobility model is similar to the first one but requires creating sub-zones and a new data acquisition phase to collect RSSI measurements in each created sub-zone. The third mobility model creates transition regions and construct hidden Markov models for each trajectory or sequence corresponding to a transition region. We also described how the associated evidence by the mobility models should be combined with observation models proposed in Chapters 3 and 4. Finally, we illustrated the performance of the tracking approaches through experiments in two real scenarios. The realized experiments demonstrated an enhancement in the overall accuracy carried by the mobility models to track the sensors in real time. We also showed that the accuracy of the first and second mobility-based tracking models decreases when the maximum speed increases. In addition, the accuracy of the third mobility-based tracking model increases as the sequence length increases. In addition to the enhancement in the overall accuracy, an important advantage of the proposed approaches, as compared with the state-of-the-art tracking methods, is that no inertial measurement units are needed to perform the tracking.

# Chapter 6

# Decentralized Localization

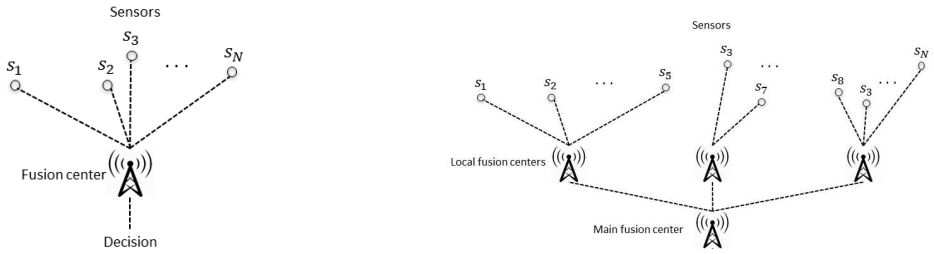
## Contents

*In this chapter, we propose a decentralized approach for sensors localization. The localization methods presented earlier are centralized, where all information are sent to one fusion center. Information processing is thus limited to only one entity. We introduce here an extension of the proposed centralized localization approach that allows us to localize mobile nodes in a decentralized framework, where several entities participate in the localization process. The targeted area is partitioned into several sectors, each of which having a local fusion center, namely a calculator, capable of receiving, processing, and emitting data. Each calculator runs then a local localization algorithm, using any of the observation and mobility models described in Chapters 3, 4, and 5, to estimate the zones of the mobile nodes. The fusion of all calculators estimates yields a final zone estimate. Various decentralized approaches are described, and their performance is evaluated through experiments, and compared against the state of the art.*

## 6.1 Introduction

The sensors deployed in the WSN, where localization and tracking are to be realized, are smart and have the capability of acquiring and processing data. These sensors are highly communication-intensive systems, as they keep communicating with each other and interacting with the environment [Zhou *et al.*, 2015]. However, they have limited resources regarding processing, memory, energy, and communication bandwidth [Nikolov & Haas, 2018]. The network topology is a key concept in designing such intelligent systems in WSNs, as it plays a vital role in minimizing various resource constraints [Chen *et al.*, 2011]. An efficient topology reduces the amount of communication required by the sensors to exchange information, and hence saves energy. A topology based on minimizing the distance between neighbor nodes, for instance, reduces the probability of losing a message during communication. Moreover, a well-designed topology can also reduce radio interference and facilitate data aggregation, thus reducing the amount of processing cycles and elongating the network lifetime [Velmani & Kaarthick, 2015]. Other important issues influenced by the topology are the robustness, the scalability, the performance, and the complexity of the designed system.

Three main topologies have been proposed in literature for localization in WSNs, outlined in the following; for more details, see [Cota-Ruiz *et al.*, 2016; Üney *et al.*, 2016; Yan *et al.*, 2017]. On one hand, there exists the centralized topology where sensors acquire data measurements and transmit them to the fusion center for processing [Talebi & Hemmatyar, 2014]. In such a topology, the sensors are not required to carry out complex computations. Although it can achieve high quality processing, the centralized topology results in unnecessary energy costs due to the transmission of measurements over long distances [Mamun, 2012]. On the other hand, the distributed topology treats equally all the sensors, each working as a local fusion center. In such topology, the sensors perform computations and exchange data with their neighboring sensors, located within their communication range. Since information processing is no longer limited to a single fusion center, the network is more robust to failures. However, developing relevant distributed algorithms remains a challenging issue. The third topology is the decentralized topology, also called clusterized topology, which takes the advantages of the preceding two topologies by partitioning the sensors into several clusters, each having its own fusion center or cluster head [Iliev & Paprotny, 2015]. Information is exchanged between the sensors of each cluster and transmitted to the local fusion center. The outputs of all fusion centers are combined to yield a final decision. Such a topology increases the scalability of the network, and reduces the energy consumption leading to a prolonged network lifetime [Üney *et al.*, 2016]. An illustration of the general structure of the centralized and decentralized topologies, considering a set of sensors $\{s_1, \ldots, s_N\}$, is shown in Figure 6.1.




(a) Centralized topology.

(b) Decentralized topology.

**Figure 6.1:** The general structure of the centralized and decentralized topologies for localization in WSNs.

# 6. DECENTRALIZED LOCALIZATION

In Chapter 3, we presented an observation model for localization of mobile sensors (MNs) in indoor environments, where the objective was to find the zone of the MN at each instant. The high precision of the model decreases as a function of the surface of the targeted area. In fact, as the number of zones increases, the proposed model fails to distinguish between the different zones, thus degrading the accuracy. For that reason, an extended observation model was proposed in Chapter 4. The advantage of the latter is that it manages to keep a high accuracy even in case of large number of zones due to its hierarchical approach. However, the centralized architecture adopted there is computationally complex and requires at average double the time needed by similar methods in the domain. The need for a fast, more scalable and robust, yet precise, method for real time localization applications, motivated our research for a decentralized approach.

In this chapter, we investigate decentralized architectures for zoning-based localization. Inspired by the decentralized topology, the proposed approach is decentralized for zoning-based localization. At first, we partition the targeted area into several sectors, depending on the environment characteristics, and then assign a calculator for each sector, which locally estimates the MN's zone. Each calculator performs a local localization algorithm, which can be only the zoning-based localization method proposed in Chapter 3, or the mobility-based tracking technique proposed in Chapter 5 by combining the basic observation model to any of the mobility models. It can also be the extended observation model proposed in Chapter 4 in case each considered sector consists itself of a large number of zones. The decision is then made by combining evidence from all calculators. In addition, we present a calculator placement technique to determine the optimal number of calculators and their positions.

In the following, we first formulate the problem. We then describe the decentralized approach by proposing three different architectures. Afterwards, we present an optimal calculators placement technique using two strategies. Then, we explain the local localization algorithm executed by each calculator. Finally, we evaluate the performance of the proposed approach against the previously described centralized approaches and the state of the art.

## 6.2 Problem Formulation

The objective of the proposed algorithm is to determine the zone of a mobile node in a decentralized architecture. This is done by assigning a confidence level $\mathcal{C}f_t(\cdot)$ to each zone at each instant $t$. Suppose the targeted area is partitioned into $N_C$ sectors, each having its own calculator, thus obtaining $N_C$ calculators in total, denoted $C_i$, $i \in \{1, ..., N_C\}$. Let $Z_j, j \in J_i$, be the set of zones of the $i$-th sector, $J_i$ being the set of indices of zones constituting sector $i$. As defined in the centralized approaches before, we denote $AP_k, k \in \{1, \ldots, N_{AP}\}$, the APs installed in the network. At the offline phase, a set of $N_j$ RSSI measurements $\rho_{j,k,\ell}, \ell \in \{1, \ldots, N_j\}$, are collected in the zone $Z_j$ with respect to AP $AP_k$. Then in the online phase, a MN to be localized uses the vector of RSSIs $\boldsymbol{\rho}_t \in \mathbb{R}^{N_{AP}}$ at time $t$. Since not all APs are detected at each instant, we denote $I_{AP,t}$ the set of indices of the APs whose signals are detected by the mobile node at time $t$ and $\rho_{t,k}, k \in I_{AP,t}$, their measured RSSIs. The vector $\boldsymbol{\rho}_t$ is completed with zeros at positions where APs are not detected. The calculators within its neighborhood receive the RSSIs vector and use it with the offline database to assign an evidence to each zone. Local estimations are then combined to reach a final decision of the MN's zone. In the following, $I_{C,t}$ denotes the set of indices of the calculators within the communication range of the MN at time $t$. Figure 6.2 shows two MNs $s_1$ and $s_2$ trying to communicate with the surrounding calculators. As the figure shows, the MN $s_1$ can communicate with $C_1$ and $C_2$ since they are in its sensing range, while $s_2$ can communicate with $C_3$ only. In these cases, $I_{C,t} = \{1, 2\}$ for $s_1$, while $I_{C,t} = \{3\}$ for $s_2$.

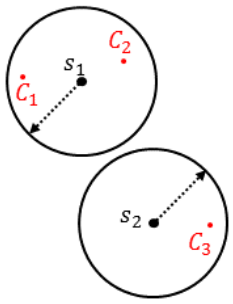



**Figure 6.2:** Illustration of communication between MNs and calculators.

## 6.3 Decentralized Approach

In this section, we present three decentralized approaches that differ by the geographic structure of the sectors and the fusion of evidence.

### 6.3.1 First decentralized approach

In this approach, we assign distinct sectors to each calculator, which means that the zones constituting a certain sector are totally different than those constituting another one; i.e., in any two sectors $i, i' \in \{1, \ldots, N_C\}$ with $i \neq i'$, we have $Z_j \neq Z_{j'},\ \forall j \in J_i, j' \in J_{i'}$. An example of such architecture is shown in Figure 6.3(a), where calculator $C_1$ is responsible for the blue sector constituted of four zones, $C_2$ for the red sector of two zones, and $C_3$ for the green sector of three zones. The Access Points are the sources of information for localization and are installed in the network. The mobile node moves inside the environment, collects observation measurements from APs, and sends the information to surrounding calculators. Let $\eta_{i,t}$ be the strength of the signal received by calculator $C_i$ from the mobile node. Each $C_i$ performs a local localization algorithm that will be explained in Section 6.5 and assigns certain evidence to each zone of its sector. Let $m_{i,t}(Z_j), j \in J_i$, be the assigned evidence by the local localization algorithm to the zones of sector $i$ at time $t$. Here, the calculator decides locally through normalizing $m_{i,t}(\cdot)$ on all zones of the sector, as follows,

$$\tilde{m}_{i,t}(Z_j) = \frac{m_{i,t}(Z_j)}{\sum_{q \in J_i} m_{i,t}(Z_q)}. \tag{6.1}$$

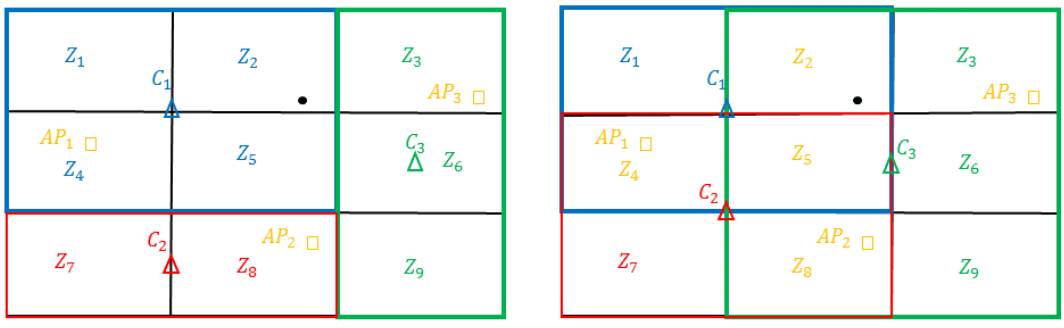
If only one calculator receives the node's message, the final decision will be identical to the local one, and hence the confidence assigned to each zone will be equal to the normalized mass, i.e., $\mathcal{C}f_t(Z_j) = \tilde{m}_{i,t}(Z_j)$. However, if more than one calculator detect the node's message, evidence is put together to reach a final decision. Since sectors are distinct, each zone has only one affiliated evidence. Thus, the fusion is all about weighting calculators decisions, followed by a final normalization phase on the whole set of zones. Since the strength of the signal decreases with the traveled distance, more confidence is given to calculators receiving stronger signals. The confidence attributed to each zone of the sector $i$ is

then computed,

$$\mathcal{C}f_t(Z_j) = \frac{\frac{1}{w_{i,t}} \times \tilde{m}_{i,t}(Z_j)}{\sum_{p \in I_{C,t}} \sum_{q \in J_p} \frac{1}{w_{i,t}} \times \tilde{m}_{p,t}(Z_q)}, \tag{6.2}$$

such that $w_{i,t} = \frac{\eta_{i,t}}{\sum_{x \in I_{C,t}} \eta_{x,t}}$ is the weight given to each calculator $C_i$.

In addition to its simplicity, an important advantage of such an approach is its scalability. Any expansion of the targeted area can be easily treated by adding the new calculators to the expanded area without any modification on the already existing architecture. Moreover, it reduces energy consumption, as mobile nodes are required to send information to calculators in their range. It also increases robustness, since the existence of several calculators renders the network more resistant to failures that are fatal in the case of a centralized approach. However, a major drawback of the described strategy is that it introduces a significant localization error when more than one calculator detect the MN's message. Specifically, the normalization phase of equation (6.1) made by the calculator corresponding to the sector where the MN does not really reside assigns comparatively high confidence values for zones that are, in fact, out-of-interest.




(a) Distinct sectors.

(b) Overlapping sectors.

**Figure 6.3:** Network architectures - △ designates calculators, □ designates Access Points, • designates a MN.

### 6.3.2 Second decentralized approach

To overcome the localization error problem introduced by making a local decision on each calculator level before fusion, we consider here another strategy. We adopt the same architecture as of the previous one, but instead of performing the normalization of masses at each calculator, the local estimations predicted by all calculators are combined first, then a final decision is made based on the obtained combined evidence. In the same manner, if only one calculator detects the node's message, the confidence associated to each zone is equal to the normalized mass computed by equation (6.1). If more than one calculator detect the sensor's message, the attributed confidence is obtained by normalizing over all the zones,

$$\mathcal{C}f_t(Z_j) = \frac{\frac{1}{w_{i,t}} \times m_{i,t}(Z_j)}{\sum_{p \in I_{C,t}} \sum_{q \in J_p} \frac{1}{w_{p,t}} \times m_{i,t}(Z_j)}. \tag{6.3}$$

The advantage of this strategy, over the previous one, is that it does not assign unwanted high evidence for zones of sectors where the mobile node does not really reside. Indeed, only an estimation is predicted locally at each calculator level without making a decision, which then is combined with all other calculators estimations to reach a final decision. A disadvantage of this proposed approach is noted if the localization algorithm degrades in accuracy when increasing the number of zones. More precisely, for this strategy to achieve a high accuracy, the localization algorithm should be able to assign discriminating evidence on wider areas, hence for larger number of zones.

### 6.3.3 Third decentralized approach

In the previous two approaches, distinct sectors have been studied. Here, we associate overlapping sectors to calculators, which means that certain zones correspond to more than one sector; i.e., In two sectors $i, i' \in \{1, \ldots, N_C\}, i \neq i', \exists j \in J_i, j' \in J_{i'}$, such that $j = j'$. Such architecture is shown in Figure 6.3(b), where sectors 1 and 3, for instance, overlap at zones $Z_2$ and $Z_5$, while the three sectors 1, 2 and 3 overlap at zone $Z_5$.

The design of such an architecture is done in the offline phase when collecting measurements and constructing databases. In each zone, the mobile node sends a

certain number of messages to surrounding calculators. The ones that receive the MN's messages are supposed to have their corresponding sector constituting the zone from which the message originated. In the same example of Figure 6.3(b), messages sent within $Z_1$ for instance are only detected by $C_1$, hence it is the first zone of sector 1. Therefore, the database constructed in $Z_1$ is only included in that sector. Messages originating from $Z_4$ are detected by $C_1$ and $C_2$, thus both sectors 1 and 2 cover that zone.

At the decision level, there is no difference with respect to the two preceding architectures if only one calculator detects the node's message in the online phase, where the confidence assigned to each zone is equal to $\tilde{m}_{i,t}(Z_j)$, computed using equation (6.1) for one specific $i$. However, if more than one calculator detect the MN's message, only zones at the intersection of all the sectors refereed in $I_{C,t}$ are considered. Local computations are carried out at the calculators. Then information is combined in the following manner,

$$m_{F,t}(Z_j) = \prod_{i \in I_{C,t}} \tilde{m}_{i,t}(Z_j), \qquad \forall j \in \bigcap_{i \in I_{C,t}} J_i. \tag{6.4}$$

The quantity $m_{F,t}(Z_j)$ is null for all the zones that are not at the intersection of the sectors corresponding to the detected calculators; i.e., $m_{F,t}(Z_j) = 0 \; \forall Z_j$ such that $j \notin \bigcap_{i \in I_{C,t}} J_i$. The confidence assigned to each zone is calculated by normalizing all combined evidence,

$$\mathcal{C}f_t(Z_j) = \frac{m_{F,t}(Z_j)}{\sum_{p \in I_{C,t}} \sum_{q \in J_p} m_{F,t}(Z_q)}. \tag{6.5}$$

This approach takes the advantages of both previous ones while solving their inconveniences. In fact, this strategy avoids assigning high confidence for out-of-interest zones as the first approach does, since a message received by the calculator indicates here for sure that the MN resides in one of the zones of its corresponding sector. Yet, it continues to profit of the local decision made at the calculators level. On the other hand, it does not introduce a localization error as in the case of the second approach, since there is no need to assign discriminating evidence to large number of zones. A local decision is being made instead, yet with benefiting of the fusion of evidence of several calculators as introduced in the second approach.

## 6.4 Calculators Placement

The calculators are the smart devices capable of exchanging information with the environment, through reception, processing, and transmission of signals. Since neither the number of required calculators nor their places are known a priori, it is important to develop a certain strategy to place them. One possible solution is to develop an accuracy-based algorithm-dependent strategy. The user measures the efficiency of the local localization algorithm by progressively increasing the number of zones and specifying an accuracy threshold. Once the overall accuracy falls below the threshold, no more zones are added, and thus sectors of approximately the same resulting number of zones or surface area are considered for each calculator. Calculators are then distributed in the same manner to cover the whole targeted area. However, such solutions incorporate randomness in the construction of the network and are case dependent.

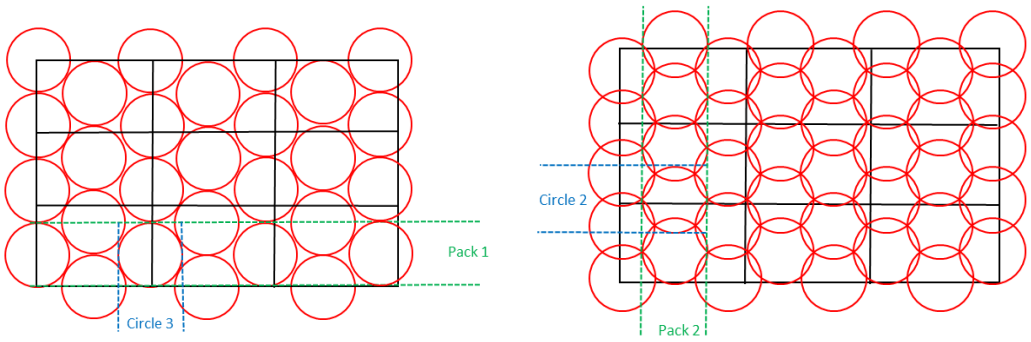
Here, we do not adopt such strategy, and we rather propose a more systematic solution based on the minimal number of required calculators criterion. The problem is formulated as follows. Suppose, without loss of generality, the targeted area is a rectangle of dimensions $L \times H$. Other forms are treated by considering the larger rectangle then removing parts that do not exist in the original area. Let $\delta$ be the sensing range of the mobile node. If the sensing range is not circular, we consider the largest circle inscribed in the scope. The objective is to minimize the number of calculators $N_C$ needed to cover the whole area, and determine their placements. For that reason, we pass from *circle around MN* to *circle around calculator*, since for a distance less than or equal to $\delta$, the MN belongs to the disk centered at the calculator of radius $\delta$.

This problem can be tackled as either packing of circles in a plane or covering a region with circles, which are treated in the domain of mathematics. The difference between the two is that the former prohibits overlapping between the circles [He & Dosh, 2017], while the latter completely covers the targeted area [Bánhelyi *et al.*, 2015]. Both approaches serve in generating an overlapping-sectors architecture, as discussed in Section 6.3.3. To generate a distinct-sectors architecture, it is then enough to assign manually the overlapping zones to a

corresponding sector. Examples of the two tiling arrangements are shown in Figures 6.4(a) and 6.4(b). In the following, we discuss the calculators' placement technique in each of the two cases.

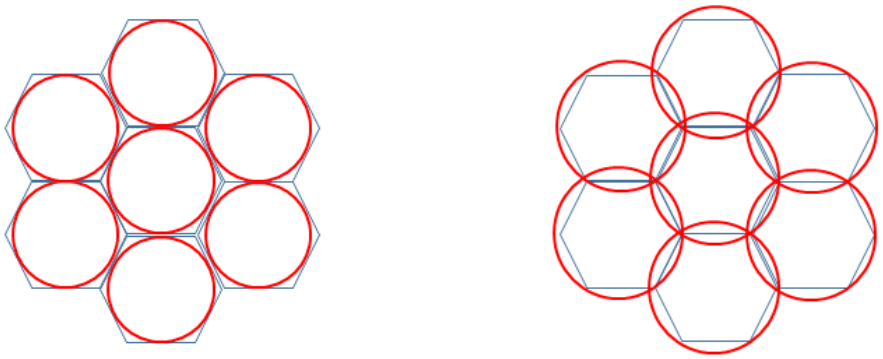
### 6.4.1 Circle packing

A circle packing is an arrangement of circles inside a given boundary where no couple of circles is allowed to overlap, with some or all of them being mutually tangent [He & Dosh, 2017]. It has been proved that the hexagonal tiling with circles inscribed in the hexagon, shown in Figure 6.5(a), is the optimal of all




(a) Circles packing. (b) Circles covering the region.

**Figure 6.4:** Tiling arrangements for optimal calculators placement.



(a) Hexagonal tiling for circle packing. (b) Hexagonal tiling for circle covering.

**Figure 6.5:** Hexagonal tilings.

possible plane packing strategies [Hales, 2000]. Let the radius of each circle be $\delta = 1$ unit, then its area is equal to $\pi$. As it is noted from the figure, the hexagon touches the circle at the midpoints of its sides, hence the distance between the midpoints of opposite sides is 2 units, thus the length of each hexagon side is $\frac{1}{\sqrt{3}}$. The area of the hexagon can be found by splitting it into six equilateral triangles and therefore the total area is equal to $6 \cdot \frac{1}{\sqrt{3}} \cdot 1 = 2\sqrt{3}$ square units. The number $N_C$ of required circles, hence calculators, is computed as follows,

$$\frac{\pi}{2\sqrt{3}} = \frac{N_C \times \pi\delta^2}{L \times H} \implies N_C = \text{ceil}\left(\frac{L \times H}{2\sqrt{3} \times \delta^2}\right), \tag{6.6}$$

where ceil$(\cdot)$ is a function that rounds the number into the upper integer. The packing density $\mathcal{D}$ is the ratio of the area covered by the circles to the total area,

$$\mathcal{D} = \frac{\text{number of circles} \times \text{area of a circle}}{\text{Total area}} = \frac{N_C \times \pi\delta^2}{L \times H} \approx 0.907. \tag{6.7}$$

This leads to a coverage of 90.7%. To deploy the circles, $B$ packs are placed horizontally, each containing $A$ circles. In a Cartesian coordinate system of origin the bottom left corner of the rectangle, the position $[x_a, y_b]$ of the center of the circle $a$ of pack $b$, $1 \leq a \leq A, 1 \leq b \leq B$, is determined as,

$$[x_a, y_b] = \begin{cases} \left[\sqrt{3}(a-1)\delta, (2b-1)\delta\right], \text{if } a \text{ is odd;} \\ \left[\sqrt{3}(a-1)\delta, 2(b-1)\delta\right], \text{if } a \text{ is even.} \end{cases} \tag{6.8}$$

### 6.4.2 Circle covering

A circle covering is an arrangement that aims at filling the whole plane with circles without leaving any gap neither between the circles nor between them and the boundary [Bánhelyi *et al.*, 2015]. It has been found that the optimal covering of a plane with circles is to circumscribe the circles about hexagons of a regular hexagon network [Kershner, 1939]. Such tiling is shown in Figure 6.5(b), where the circles are circumscribed around the hexagons, instead of being inscribed. Here, the minimum required number of circles is [Kershner, 1939],

$$N_C = \text{ceil}\left(\frac{2\sqrt{3}L \times H}{9 \times \delta^2}\right). \tag{6.9}$$

In this case, the distance between the centers of any two adjacent circles is equal to $\sqrt{3}\delta$. To cover the plane, $A$ packs are placed vertically, each containing $B$ circles. The position $[x_a, y_b]$ of the circle $b$ of pack $a$, $1 \leq a \leq A, 1 \leq b \leq B$, is,

$$[x_a, y_b] = \begin{cases} \left[\frac{3}{2}(a-1)\delta, \sqrt{3}(b-1)\delta\right], \text{if } a \text{ is odd;} \\ \left[\frac{3}{2}(a-1)\delta, (\sqrt{3}(b-1)+1)\delta\right], \text{if } a \text{ is even.} \end{cases} \tag{6.10}$$

The density of this tiling strategy is $\mathcal{D} = \frac{N_C \times \pi\delta^2}{L \times H} = \frac{2\pi\sqrt{3}}{9} \approx 1.209$.

### 6.4.3 Packing versus covering

The advantage of the circle packing approach is that it assigns fewer numbers of calculators than the circle covering approach. The former requires 11 calculators for instance to cover a $30m \times 30m = 900m^2$ region for MNs with $5m$ as sensing range, while the latter requires 14. However, the packing strategy keeps gaps making the MN undetected by any calculator in certain positions. Yet, since the sensing range is practically not circular, and here is supposed to be the greatest circle included in the antenna's lobe, then the 90.7% coverage will be practically acceptable especially for large surface areas. Though the circle covering approach leads to more calculators, it guarantees a 100% plane coverage, ensuring that all MNs are detected by at least one calculator at any position.

## 6.5 Local Localization Algorithm

Each calculator runs a local localization algorithm to localize MNs in its corresponding sector. The objective here is to determine a function $\mathfrak{I}_i(\cdot) : \mathbb{R}^{N_{AP}} \rightarrow [0,1]^{|J_i|}$ for each calculator $C_i$ such that $\mathfrak{I}_i(\boldsymbol{\rho}_t) = \left(m_{i,t}(Z_{j_1}), \ldots, m_{i,t}(Z_{j_{|J_i|}})\right)$, where $J_i = \{j_1, \ldots, j_{|J_i|}\}$, and $m_{i,t}(Z_j)$ is the evidence assigned to zone $Z_j$ at time $t$ due to new observation vector $\boldsymbol{\rho}_t \in \mathbb{R}^{N_{AP}}$.

A first solution to determine the function $\mathfrak{I}_i(\cdot) : \mathbb{R}^{N_{AP}} \rightarrow [0,1]^{|J_i|}$, is to use the basic observation model proposed in Chapter 3. In this case, the function $\mathfrak{I}_i(\cdot)$ denotes the basic observation model $\mathbb{O}(\cdot)$, the number of zones $|J_i|$ is $N_Z$, and the evidence $m_{i,t}(\cdot)$ assigned to the zones of sector $C_i$ is $m_{\mathbb{O},t}(\cdot)$. The calculator $C_i$ is thus responsible for assigning evidence to the zones of its sector. This is done by using

the offline database of RSSI measurements corresponding to only these zones. The measurements are modeled to represent the variability of the RSSIs in each zone with respect to all APs. Afterwards, the belief functions framework is created in the corresponding sector to assign the evidence $m_{\mathbb{O},t}(Z_j), j \in \{1,\ldots,N_Z\}$, which represents here $m_{i,t}(Z_j), j \in J_i$. The advantage of this model, in addition to its simplicity, is that it yields a good accuracy on areas with small number of zones, which can result in an efficient decentralized localization approach on both accuracy and complexity levels.

However, if the sectors formed by the calculators' placement technique are by themselves large, another more relevant solution will be to use the extended observation model proposed in Chapter 4 as a local localization algorithm to obtain a good accuracy. In this case, the clustering algorithm is used to group the zones of the sector $C_i$ to obtain a two-level hierarchy. The AP selection algorithm in each sector is also applied to choose locally the best subset of APs in terms of accuracy and redundancy. In the same manner as for the basic observation model, $\mathcal{I}_i(\cdot)$ denotes the extended observation model $\mathbb{O}(\cdot)$, and the number of zones $|J_i|$ denotes $N_Z$. The evidence $m_{i,t}(Z_j), j \in J_i$, will be equal to the confidence level $m_{\mathbb{O},t}(Z_j), j \in \{1,\ldots,N_Z\}$, assigned by $\mathbb{O}$ to the zones of its sector.

A third solution is to use both the observation and mobility models to localize the MNs locally in each sector, as described in the tracking technique of Chapter 5. In this case, the function $\mathcal{I}_i(\cdot)$ denotes the tracking model $\mathbb{T}(\cdot)$, the number of zones $|J_i|$ denotes $N_Z$, and the evidence $m_{i,t}(\cdot)$ assigned to the zones of sector $i$ is equal to $\mathcal{W}_t(\cdot)$.

## 6.6 Experiments

To evaluate the performance of the proposed method, real experiments are conducted in a WLAN environment. In the following, the experimental setup is first introduced. The performance of the proposed method is then evaluated, and is compared with the previously proposed centralized approaches, and against the state-of-the-art afterwards.

### 6.6.1 Experimental setup

Experiments are realized in the Living Lab of the University of Technology of Troyes, France. The targeted area of dimensions $22m \times 25m$, constituted of 21 zones, is shown in Figure 6.6. The MNs to be localized have a sensing range of $\delta = 6m$. The MNs scan the network and distinguish APs through their MAC addresses. They then measure the RSSIs and send the information to the corresponding fusion center. It is noted that 23 different AP networks are detected and hence are used as sources of information. To develop a decentralized localization algorithm, the three approaches described in Section 6.3 are used. To this end, the optimal placement technique presented in Section 6.4 is applied to determine the number and the position of the calculators, resulting in $N_C = 4.4 \approx 5$ calculators using the circle packing strategy, and $N_C = 5.8 \approx 6$ calculators using the circle covering strategy. Since there is no big difference in the number of calculators between the two strategies, the latter is adopted as it guarantees a 100% coverage. Figure 6.6 shows in red the regions of the calculators using the circle covering approach. The sectors corresponding to each calculator constitute all zones totally or partially included in the red region of the calculator. Each calculator then runs the local localization algorithm described in Section 6.5 to assign evidence to each zone of its sector. A set of 50 RSSI measurements is taken in each zone of which 30 are used to construct the kernel-based model, and 20 for testing in the online phase. An estimation is said to be correct if the algorithm assigns the highest evidence to the zone where the MN actually resides.

### 6.6.2 Evaluation of performance

The performance of the proposed decentralized approaches are evaluated by comparing them to the previously described centralized methods, which are the basic observation model (BOM) proposed in Chapter 3 and the extended observation model (EOM) proposed in Chapter 4. The BOM uses the BFT to merge all evidence using one calculator, while the EOM creates a hierarchy of clusters from the original zones using divergence similarity. Table 6.1 compares the overall accuracy of the various decentralized approaches with that of the BOM and the EOM. A kernel-based model and the AP selection technique are used for all the

methods. The third approach attains an accuracy of 91.43%, while the first and the second approaches suffer from high evidence assignment for out-of-interest zones, and low accuracy over large areas respectively.

To evaluate the robustness of the various techniques, the accuracy is recorded as a function of the number of failing fusion centers or calculators. For each new observation, a random set of calculators is supposed to have failed. The overall accuracy is then determined by the average of the obtained accuracy on all observations with respect to a certain number of failing calculators. Table 6.2 displays the overall accuracy of both decentralized and centralized approaches. The centralized approaches, BOM and EOM, will not be able to perform any localization if one or more calculators fail. In fact, there is only one calculator in such architectures and hence its failure means the failure of the whole network. On the other hand, the decentralized approaches can still manage to localize the MNs upon failure of a certain number of calculators, even though the overall accuracy degrades. This robustness is a clear advantage of the decentralized techniques as it allows localization upon calculators' failure, which cannot be achieved by the centralized methods.

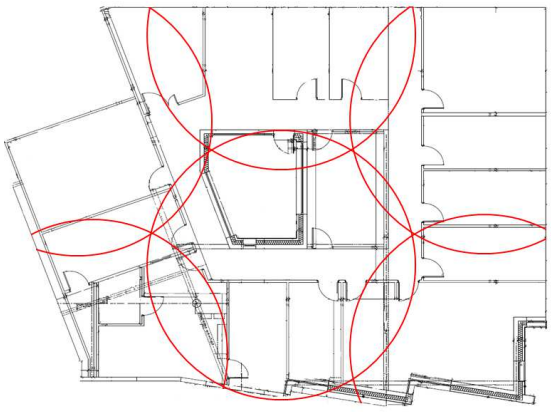


**Figure 6.6:** The Living Lab of the University of Technology of Troyes, France; A section area of $22m \times 25m$ covered by calculators with MNs of sensing range $\delta = 6m$.

### 6.6.3 Comparison to state-of-the-art methods

In this paragraph, the proposed methods are compared with well-known localization techniques. Koyuncu & Yang [2011] present a weighted $k$-nearest neighbors algorithm (WKNN) for indoor localization. To estimate the position of the MN, the new received measurement is compared with the elements in the fingerprint database using Euclidean distances. A set of $k$ smallest Euclidean distances is selected and the $k$-nearest neighbors algorithm is then applied. The algorithm averages the coordinates of the $k$-nearest neighbors of the MN, weighting each distance by a factor previously determined according to a mathematical model,

**Table 6.1:** Comparison between the decentralized and centralized localization approaches in terms of overall accuracy (%).

| Technique | $N_C$ | Overall accuracy (%) |
|---|---|---|
| Basic observation model | 1 | 78.91 |
| Extended observation model | 1 | 90.21 |
| First decentralized approach | 6 | 82.14 |
| Second decentralized approach | 6 | 86.90 |
| Third decentralized approach | 6 | **91.43** |

**Table 6.2:** Influence of fusion center failure on the overall accuracy of the decentralized and centralized approaches.

| Technique | Number of failed fusion centers | | | | | | |
|---|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| Basic observation model | 78.91 | - | - | - | - | - | - |
| Extended observation model | 90.21 | - | - | - | - | - | - |
| First decentralized approach | 82.14 | 76.19 | 64.28 | 52.30 | 28.57 | 23.54 | - |
| Second decentralized approach | 86.90 | 78.57 | 65.48 | 54.28 | 28.81 | **23.54** | - |
| Third decentralized approach | **91.43** | **81.67** | **70.24** | **58.33** | **29.76** | 22.62 | - |

to give its location estimate. Shang *et al.* [2004] propose a connectivity-based localization algorithm. The advantage of connectivity-based algorithms is that they do not rely on collected measurements. The MN's location is given as the intersection of the ranges of the APs detected by the MN. On the other hand, conventional classification techniques such as neural network (NN) [Dai *et al.*, 2016] and Support Vector Machines (SVM) [Zhang *et al.*, 2017] are applied. Table 6.3 compares the overall accuracy and the localization processing time of the various described techniques. As the table shows, the third approach outperforms other state-of-the-art techniques in terms of accuracy and processing time. The proposed method in all three approaches gains advantage of the simplicity of the local localization algorithm, which results in a relatively low complexity, and thus low processing time.

An important factor to take into account is the dependency of the localization method on the number of APs, which are the sources of information. A method that requires a high density of APs is not favored, as it is practically unfeasible in most of the cases due to the unavailability of sufficient APs in the network, or

**Table 6.3:** Comparison between various localization techniques in terms of overall accuracy (%) and processing time (s).

| Technique | accuracy (%) | online time (s) |
|---|---|---|
| Weighted K-nearest neighbors | 84.28 | 0.1265 |
| Connectivity | 86.67 | 0.1338 |
| Neural networks | 85.82 | 0.1866 |
| Support vector machines | 86.47 | 0.1912 |
| Basic observation model | 78.91 | 0.1374 |
| Extended observation model | 90.21 | 0.2955 |
| First decentralized approach | 82.14 | **0.0725** |
| Second decentralized approach | 86.90 | 0.0831 |
| Third decentralized approach | **91.43** | 0.0977 |

due to the installation cost. Table 6.4 shows the overall accuracy as a function of the number of available APs. It is noted that both WKNN and connectivity methods are highly sensitive to the density of APs, while the proposed method is less sensitive at this level with a maximum decrease of 8% in overall accuracy, upon a decrease in the number of detected APs from 23 to 5.

## 6.7 Conclusion

In this chapter, we presented a decentralized algorithm for localization of sensors in indoor wireless networks. At first, we proposed three decentralized approaches that differ in the geographic structure and the fusion of evidence. After that, we presented a calculator placement technique to optimally determine the number and the positions of the calculators responsible for the decentralized localization. Afterwards, we described the local localization algorithm that should be run by each calculator. This algorithm can be the basic observation model presented

**Table 6.4:** Influence of the number of detected APs on the overall accuracy (%) of the localization techniques.

| Technique | Number of detected APs | | | |
|---|---|---|---|---|
| | 5 | 10 | 15 | 23 |
| Weighted K-nearest neighbors | 67.14 | 72.62 | 74.05 | 84.28 |
| Connectivity | 65.48 | 69.29 | 76.67 | 86.67 |
| Neural networks | 79.05 | 81.19 | 84.76 | 85.82 |
| Support vector machines | 80.71 | 82.62 | 85.43 | 86.47 |
| Basic observation model | 74.52 | 76.90 | 77.62 | 78.91 |
| Extended observation model | 81.42 | 83.33 | 86.67 | 90.21 |
| First decentralized approach | 76.19 | 80.95 | 81.66 | 82.14 |
| Second decentralized approach | 78.10 | 81.43 | 83.81 | 86.90 |
| Third decentralized approach | **82.38** | **84.76** | **87.14** | **91.43** |

in Chapter 3, the extended observation model presented in Chapter 4, or the tracking technique presented in Chapter 5. Finally, we illustrated the performance of the proposed approaches through experiments conducted in Living Lab facilites. The realized experiments demonstrated the robustness of the proposed approach, its lower complexity as compared with the centralized techniques, and its competitiveness in terms of overall accuracy as compared with state-of-the-art localization methods. We also demonstrated the reliability of the proposed approaches against the density of APs in the network by studying their influence on the overall performance.

# Chapter 7

# Conclusion

## Contents



*This thesis addressed the problem of localization of sensors in indoor environments. First, a zoning-based localization method was proposed within the framework of the belief functions theory. This method was then extended using hierarchical clustering, allowing us to consider large-scale environments. In addition, an Access Point selection algorithm was presented to avoid erroneous and redundant sources. Next, information of the mobility of the sensors was considered to yield more accurate zone estimates. Finally, a decentralized version of the proposed localization method was described, to increase the robustness and scalability, and to reduce the complexity of the method. In this chapter, a summary of contributions is provided, along with some future research perspectives.*

## 7.1 Summary of Contributions

The presented work in this thesis tackled the problem of indoor localization in wireless sensor networks, and carried several contributions to the domain, summarized in the following.

In Chapter 3, we introduced a zoning-based method for localization in WSNs within the belief functions framework. An observation model was constructed to allow to estimate the sensor's zone using the RSSIs of the signals received from surrounding WiFi APs. Several configurations were investigated for the definition of the model. Different types of modeling of observations, discounting of APs decision, and combination of evidence were studied. The experimental results showed that the proposed method outperforms other classification techniques in terms of overall accuracy.

To improve the performance of the method when the number of zones in the targeted area is large, an extended observation model was proposed in Chapter 4. A two-level hierarchy of zones was created by optimizing the inter- and intra-cluster measures. An AP selection algorithm was also developed to choose the best subset of APs in terms of overall accuracy and redundancy. The results showed that this model yields a better accuracy, yet at the expense of a higher computational complexity. A main limitation of the extended model, is the erroneous estimations in the zones that are symmetric with respect to the APs, where similar RSSIs are received.

In order to overcome this problem, we proposed in Chapter 5 a tracking technique to take advantage of the mobility of sensors in indoor environments. Different mobility models were combined with the observation models of Chapters 3 and 4 in the belief functions framework. In this way, the erroneous estimations were corrected by taking into account the movement of the sensors, its next possible destinations, and its trajectory. The results showed that this combination gives more accurate zone estimates.

Finally, in Chapter 6, we presented a decentralized algorithm for localization. Several decentralized approaches that vary in the geographic structure and the

fusion of evidence were introduced. A calculator placement technique was described to optimally determine the number and the positions of the calculators responsible for the decentralized localization. The results demonstrated the robustness of the algorithm, its scalability, its lower complexity as compared to the centralized approaches, and its effectiveness in terms of overall accuracy.

## 7.2 Perspectives

In this thesis, several solutions were provided for localization of sensors in indoor wireless sensor networks. As part of future research, we would like to investigate the following aspects concerning improvements of our proposed methods.

- *Improving the observation models*
  In the definition of the extended observation model, a two-level hierarchy was proposed. In cases where the targeted area is too large, such as multi-floors and adjacent buildings, more than two levels in the constructed hierarchy will be required to obtain a good accuracy. Future work will search for solutions to systematically determine the number of levels needed to reform the hierarchy.

- *Improving the WiFi RSSI fingerprinting approach*
  The WiFI RSSI fingerprinting approaches, including our proposed methods, depend on the power of the received signal to determine the position of the sensor to be localized. An important factor in such algorithms is the position of the Access Points. Until now, Access Points are placed in buildings based only on the coverage and quality of service criteria. However, this installation method is not optimal for localization, where the power of signals in the zones is similar. It is important in the design of the WLAN to consider both criteria, coverage and localization. An example of such solution can be a bi-objective optimization function to maximize the coverage and minimize the error resulting from the position of the Access Points. Another solution can be to add Access Points to already installed WLANs, in a manner that is useful in localization.

- *Improving the Access Point selection algorithm*
  In the proposed Access Point selection algorithm, a trade-off parameter was defined. The role of the parameter was to assign a weight for each of the two objectives, maximizing the discriminative capacity and reducing the redundancy. However, the parameter was user-defined and chosen manually by the user according to the application. It is wiser to search for an algorithm that automatically determines the trade-off parameter in an optimal way. Moreover, in cases of dense Access Points networks, the proposed iterative greedy algorithm will be computationally complex and might fall in local minima. For that reason, solutions such as Gibbs sampling can be proposed to solve the problem.

- *Updating the databases*
  The RSSIs collected in each zone with respect to each WiFi AP vary with time. This is due to changes in the initial emitted power of the signals, change of atmospheric conditions, displacement of objects, etc. This leads to a database that does not correspond to the real situation. One solution is to manually collect new observations every period of time to keep the database updated. Another more appropriate solution is to develop an algorithm that automatically updates the database. The algorithm selects measurements that have high confidence of their zones, adds them to the database, and removes older measurements.

- *Improving the mobility models*
  The proposed mobility models assume a maximum speed of the sensors in the indoor environment. As the maximum speed increases, less accurate estimations are expected to be obtained if we need fast localization algorithms, which is naturally the case. In fact, when the maximum speed increases, the sensor can travel larger distances in the localization execution time, thus increasing the ambiguity, and therefore decreasing the efficiency of the mobility models. It is then important to equip the sensors with inertial units to collect real-time information about the mobility of the sensor. A model should then be created to relate the inertial information with the amount of belief assigned to each zone of the targeted area.

- *Considering other technologies*
  The problem of localization can be solved by camera-based solutions. In this case, image processing algorithms are to be proposed. The usage of other types of sensors such as thermal and infrared detectors is also of significant interest. Another possible solution is to use different radio signals such as Ultra-wideband or Bluetooth and combine them with the proposed WiFi models to enhance the overall accuracy.

- *Adapting the proposed solutions to suit more dependent elderly people*
  This project is realized in collaboration with the retirement home "Louis Pasteur" in Romilly-sur-Seine, France. The home welcomes elderly people and takes care of them. Implementing the proposed solutions requires equipping the elderly people with sensors in the form of bracelets, medallions, watches, etc. An important issue to study is the acceptability of the equipment by the elderly people. In addition, experiments should be realized using real data in such environments to validate the proposed methods. In addition, specific models that correspond to a population of people such as Alzheimer's patients can be considered.

# Appendix A

# Résumé de la thèse

## Contents

*Les réseaux de capteurs sans fil jouent un rôle de plus en plus important dans un grand nombre d'applications. L'une des problématiques principales des réseaux de capteurs sans fil est la géolocalisation des capteurs qui s'avère primordiale pour la plupart des applications. Dans ce manuscrit, nous abordons le problème de la localisation des capteurs et le problème du suivi de trajectoires dans les réseaux de capteurs sans fil. Nous appliquons et adaptons les solutions proposées pour localiser des personnes âgées dépendantes, accueillies dans des EHPADs. Les solutions proposées, basées sur des données de fingerprinting et de mobilité, se situent dans le cadre de la théorie des fonctions de croyance. Cette partie résume les travaux effectués autour de ces deux problèmes.*

# A.1 Introduction

Les réseaux de capteurs sans fil (RCSFs) sont des réseaux composés d'un grand nombre de capteurs intelligents [Akyildiz *et al.*, 2002]. Les capteurs, collaborant entre eux, révolutionnent le recueil et le traitement de l'information dans plusieurs domaines d'applications tels que le domaine militaire, environnemental, industriel, médical, etc. Dans ce qui suit, nous donnons un aperçu général des RCSFs. Nous exposons ensuite le problème de géolocalisation des capteurs par localisation exacte, zonage et suivi de trajectoires.

## A.1.1 Aperçu général des RCSFs

Les capteurs constituant un RCSF sont des capteurs intelligents. Ils sont munis d'un capteur physique, d'une unité de traitement et de stockage de données, d'une interface de communication sans fil et d'une source d'énergie [Zhou *et al.*, 2015]. Différents types de RCSFs peuvent être mis en œuvre selon la nature de l'environnement, y compris les réseaux terrestres, souterrains, sous-marins, multimédias et mobiles. La figure 1.1 illustre les cinq types principaux des RCSFs.

Un point essentiel dans le fonctionnement des réseaux de capteurs est le choix de la topologie. Il existe principalement trois types de topologies dans les RCSFs

définissant la façon dont les capteurs communiquent ensemble : centralisée, distribuée, et décentralisée [Cota-Ruiz *et al.*, 2016; Üney *et al.*, 2016; Yan *et al.*, 2017]. Dans le cas d'une topologie centralisée, les capteurs acheminent toutes les données mesurées vers la station de fusion centrale où elles seront stockées et traitées. Ce genre de réseaux fournit une haute qualité de traitement mais tout en consommant énormément d'énergie [Mamun, 2012]. Dans une topologie distribuée, les capteurs ont la capacité de traiter localement les données mesurées; ils n'échangent ainsi avec leurs voisins que des données traitées, d'habitude de taille réduite. Ce type de réseaux est beaucoup plus robuste à la défaillance que le mode centralisé; en effet, la panne d'un capteur en mode distribué n'affectera pas tout le réseau contrairement à une panne fatale du centre de fusion en mode centralisé. Toutefois, la topologie distribuée nécessite des algorithmes distribués qui sont souvent plus difficile à développer [Wang, 2008]. La topologie décentralisée se trouve aux juste milieu des deux autres topologies. Elle consiste à partitionner le réseau en des groupes de capteurs, appelés *clusters*, chacun ayant une tête de cluster ou *cluster head* chargée du traitement des données. Cette topologie permet d'assurer un équilibre entre la consommation d'énergie et la capacité de traitement, ce qui convient donc à des réseaux à grandes échelles [Üney *et al.*, 2016]. La figure 1.2 illustre ces trois principales topologies.

### A.1.2 Localisation exacte

La localisation exacte des capteurs dans les RCSFs est essentielle pour le traitement des données provenant de ces capteurs. Il s'agit de la détermination de leurs positions exactes dans l'environnement de surveillance. Une solution intuitive à ce problème est d'équiper tous les capteurs de GPS. Cependant, cette solution n'est pas optimale dans les milieux couverts, le service GPS y étant de qualité médiocre. Cette technologie reste limitée à une utilisation dans des environnements extérieurs [Oshin *et al.*, 2012].

Une solution alternative consiste à considérer deux types de capteurs, les ancres et les nœuds mobiles (MNs). Les ancres ont des positions connues, alors que les MNs ont des positions inconnues. Le problème de localisation se résume ainsi à estimer les positions des MNs en se basant sur les données échangées

avec les ancres. Il existe de nombreuses méthodes de localisation à base d'ancres répondant aux exigences des réseaux de capteurs sans fil. Entre autres, nous citons les techniques géométriques telles que l'angle d'arrivée ou le temps d'arrivée des signaux. Ces techniques nécessitent une synchronisation entre les capteurs et du matériel plus cher [Gezici *et al.*, 2005]. D'autres approches sont basées sur la mesure de la puissance des signaux reçus ou *received signal strength indicator* (RSSI). Comparées aux méthodes géométriques, les méthodes basées sur les RSSIs sont largement utilisées en raison de leur faible consommation en termes d'énergie et de leur faible coût, puisqu'aucun dispositif supplémentaire n'est nécessaire pour les capteurs. Cependant, ces méthodes sont sensibles à la présence du bruit et des interférences, surtout si elles estiment les distances inter-capteurs à partir des RSSIS [Patwari *et al.*, 2005; Zanella & Bardella, 2014]. Basées sur les RSSIs, les méthodes de fingerprinting consistent à collecter des données de RSSIs dans l'environnement surveillé et d'estimer les positions des capteurs en exploitant cette base de données [Mahfouz *et al.*, 2015]. Il s'agit de réaliser une cartographie de l'environnement au préalable pour pouvoir y situer ultérieurement les MNs grâce à leurs mesures de RSSIs.

### A.1.3 Localisation par zonage

L'inconvénient majeur des approches de localisation exacte basées sur le fingerprinting est la nécessité d'une grande base de données avec des points de référence exacts, ce qui est coûteux en temps et en ressources informatiques. Une autre approche pour aborder le problème de localisation est le zonage. La surface cible est partitionnée aux plusieurs zones; ensuite, une base de données est construite à partir des mesures de RSSIs prises aléatoirement dans chaque zone. Le problème de localisation par zonage est formulé comme un problème de classification multi-classes, où le but est d'identifier la zone où se trouve le MN en fonction de l'observation mesurée. Parmi les méthodes de classification, nous citons les machines à vecteur de support [Honeine *et al.*, 2013], les réseaux de neurones [Rojas, 2013], le Bayesien naïf [Narayanan *et al.*, 2013], la régression logistique [Liu *et al.*, 2014], les k-plus proches voisins [Souza *et al.*, 2014], les

arbres de décision [Kotsiantis, 2013], et les forêts d'arbres décisionnels [Breiman, 2001].

### A.1.4 Suivi de trajectoires

Une des applications les plus prometteuses des RCSFs est le suivi de trajectoires ou *tracking*. Le suivi de trajectoires est vu comme un problème de localisation séquentielle. L'objectif est de déterminer de manière récursive la position du MN. Les caractéristiques de la mobilité telles que la position, la vitesse ou l'accélération. peuvent être mesurées en utilisant les capteurs correspondants. Une première solution est d'estimer le mouvement du MN en utilisant des modèles de mobilité [Mahfouz *et al.*, 2014]. Toutefois, ces modèles ne peuvent pas être utilisés seuls, à cause de l'erreur de tracking qui s'accumule avec le temps. Pour cette raison, l'estimation par le modèle de mobilité doit être complétée par un modèle d'observation basé, par exemple, sur les mesures de RSSIs. Le problème de suivi de trajectoires consiste ainsi en une prédiction par le modèle de mobilité, suivie d'une correction par le modèle d'observation.

### A.1.5 Contributions

Dans cette thèse, nous nous intéressons à la géolocalisation indoor des capteurs par zonage. Nous abordons la problématique du zonage en raison de l'application qui consiste à localiser des personnes âgées dépendantes accueillies dans des EHPADS et où une localisation par zones est suffisante. Les méthodes sont basées sur les puissances des signaux WiFi échangés entre les capteurs. D'abord, en utilisant la théorie des fonctions de croyance et la technique de fingerprinting, nous obtenons une estimation de la zone du MN. Nous développons ensuite les méthodes proposées, au travers du clustering, dans le but de couvrir des surfaces plus grandes avec un nombre de zones plus élevé. Nous élaborons ensuite une méthode de sélection de paramètres, qui sont les bornes WiFi à privilégier.

Nous revisitons ensuite le travail proposé pour l'adapter au suivi de trajectoires. Pour cela, nous proposons des modèles de mobilité des capteurs. Les informations

obtenues par ces modèles sont combinées avec celles des méthodes de fingerprinting dans le cadre de la théorie des fonctions de croyance. Cette approche permet d'améliorer l'estimation de la zone du MN, en fusionnant toutes les informations disponibles.

D'autre part, nous présentons une approche décentralisée de la méthode de localisation, pour accroître la robustesse et diminuer la complexité de la solution. Les performances des méthodes proposées sont validées par des expérimentations sur des données réelles et comparées avec d'autres méthodes très connues dans le domaine.

## A.2 Fusion de données multi-capteurs

L'observation de certains phénomènes nécessitent la combinaison d'informations variées provenant d'un réseau hétérogène de capteurs au lieu de les utiliser d'une manière séparée. La combinaison de ces données permet, en général, l'obtention d'une information beaucoup plus riche et plus précise du phénomène observé à un certain instant. Différentes approches sont envisageables pour le faire. En effet, il est possible de fusionner des informations différentes recueillies à un même instant, ou alors un même type d'information recueilli à des instants différents. Il est également possible de combiner des données provenant de différents capteurs à des connaissance a priori provenant de bases de données collectées antérieurement ou à des résultats antérieurs de fusion [Hall & Llinas, 1997].

### A.2.1 Aperçu général de la fusion des données

Il existe trois principaux types d'architecture de fusion, qui varient en fonction du niveau d'abstraction des informations. Le premier type est la fusion des mesures; il s'agit d'une fusion directe des données multi-capteurs. Le deuxième est la fusion de primitives; c'est une représentation des données mesurées par des vecteurs caractéristiques qui sont alors fusionnés. Le troisième est la fusion de décisions, où l'on arrive à des décisions de plus haut niveau, obtenues à partir d'un calcul sur les données et qui sont ensuite fusionnées [Varshney, 1997].

Plusieurs approches ont été proposées pour la fusion de données multi-capteurs. L'approche par la théorie des probabilité utilise le théorème de Bayes pour mettre à jour la probabilité d'occurrence d'une hypothèse en fonction de ses précédentes occurrences et de ses nouvelles observations [Durrant-Whyte & Henderson, 2008]. L'approche par la théorie des ensembles flous est employée afin de tenir compte explicitement des informations imprécises au sein d'un système de décision. L'incertitude d'une donnée est modélisée par un ensemble de fonctions d'appartenance à différentes classes, permettant de représenter les différentes hypothèses à définir. Les fonctions d'appartenance sont définies pour chaque élément provenant d'une source à fusionner et les combinaisons sont ensuite réalisées [Zadeh, 1965]. L'approche par la théorie des fonctions de croyance permet de représenter l'incertitude en considérant des événements qui ne sont pas obligatoirement exclusifs ni exhaustifs. Cette théorie est une généralisation de l'inférence Bayésienne et nécessite la définition d'un ensemble d'hypothèse, nommé cadre de discernement. L'objectif du problème consiste à identifier une situation parmi ce cadre de discernement [Shafer, 1976]. Cette dernière approche présente l'avantage de mieux représenter l'ambiguïté existante dans les mesures, que la théorie probabiliste. Nous la présentons plus en détails ci-dessous.

### A.2.2 Théorie des fonctions de croyance

La théorie des fonctions de croyance (TFC) est appelée aussi la théorie de Dempster-Shafer ou encore la théorie de l'évidence [Shafer, 1976]. Cette théorie permet de tenir compte de l'imprécision de l'information et de son incertitude. Les imprécisions correspondent à une difficulté dans l'énoncé de la connaissance, parce que des connaissances numériques sont mal connues. Ceci est dû principalement à l'incapacité de l'instrument de mesures à fournir une mesure exacte. L'incertitude fait référence à la qualité de l'information. La source délivrant l'information peut être peu sûre d'elle ou susceptible de commettre des erreurs ou encore de donner de manière intentionnelle des informations erronées. L'information peut alors être complète et précise mais fausse. La TFC est un cadre qui permet la modélisation à la fois de l'imprécision et de l'incertitude.

### A.2.2.1 Représentation de l'information

Soit une question à laquelle on cherche une réponse et soit $\Theta$ un ensemble fini contenant des solutions possibles à cette question. Cet ensemble est appelé cadre de discernement et les éléments le constituant sont exclusifs. Une hypothèse ou proposition est un sous-ensemble $A \subseteq \Theta$ de cadre de discernement disant que la vérité se trouve dans $A$. C'est un élément de l'ensemble $P(\Theta)$ de tous les sous-ensembles de $\Theta$,

$$P(\Theta) = 2^{\Theta} = \{A \mid A \subseteq \Theta\}. \tag{A.1}$$

Une fonction $m_S(\cdot)$, appelée fonction de masse ou *basic belief assignment*, représente une connaissance imparfaite sur $\Theta$. Cette fonction est définie de $P(\Theta)$ à valeurs dans $[0, 1]$ selon une source $S$, avec comme contrainte,

$$\sum_{A \in P(\Theta)} m_S(A) = 1. \tag{A.2}$$

La quantité $m_S(A)$ représente la croyance sur le fait que $A$ contienne la réponse à la question. Il s'agit d'une interprétation de l'information apportée par la source $S$ sur cette hypothèse. La croyance peut être allouée à d'autres fonctions qui sont liées à la fonction de masse $m_S(\cdot)$, comme la fonction de croyance $bel_S(\cdot)$, la fonction de plausibilité $pl_S(\cdot)$ ou la fonction de communalité $q_S(\cdot)$.

### A.2.2.2 Combinaison des fonctions de croyance

La fusion de données est une solution intéressante pour l'obtention d'informations plus pertinentes. La phase de combinaison, dans le cadre de la théorie des fonctions de croyance, consiste donc à synthétiser, sous forme d'une fonction de croyance unique, l'ensemble des connaissances fournies par plusieurs fonctions. Une première règle de combinaison est la règle de Dempster représentée par $\bigoplus$ [Dempster, 1967]. La fonction de masse $m_{1 \bigoplus 2}(\cdot) = m_1(\cdot) \bigoplus m_2(\cdot)$, combinant deux fonctions de masse $m_1(\cdot)$ et $m_2(\cdot)$, est obtenue comme suit [Dempster, 1967],

$$m_{1 \bigoplus 2}(A) = \frac{\sum_{A_i \cap A_j = A} m_1(A_i) m_2(A_j)}{1 - \sum_{A_i \cap A_j = \emptyset} m_1(A_i) m_2(A_j)}, \qquad \forall A \in P(\Theta), A \neq \emptyset. \tag{A.3}$$

Le dénominateur sert à normaliser la fonction de masse obtenue. Cette règle, équivalente à l'intersection, permet l'obtention d'une fonction de masse plus informative que les fonctions de masses d'origine. Alternativement, la règle de combinaison conjonctive est équivalente à la règle de Dempster mais sans la phase de normalisation. Ces deux règles nécessitent que toutes les sources d'informations soient fiables et donc non conflictuelles. Puisque cette contrainte n'est pas toujours vérifiable, d'autres règles ont été proposées afin de tenir compte de la nature des sources d'informations [Smets, 2007]. C'est dans ce contexte que s'inscrit la règle de combinaison disjonctive [Dubois & Prade, 1992].

#### A.2.2.3 Affaiblissement des fonctions de masse

L'affaiblissement est une étape nécessaire lorsqu'on possède une information sur la fiabilité des sources d'information existantes. Elle consiste à modifier les fonctions de masse en utilisant les coefficients de fiabilité des sources [Mercier *et al.*, 2008]. L'affaiblissement classique consiste à pondérer chaque source $S$ par un coefficient $\alpha$ comme suit :

$$ {}^{\alpha}m_S(A) = \begin{cases} (1-\alpha_S)m_S(A), & \text{si } \ A \in P(\Theta),\ A \neq \Theta; \\ \alpha_S + (1-\alpha_S)m_S(\Theta), & \text{si } \ A = \Theta; \\ 0, & \text{sinon.} \end{cases} \tag{A.4} $$

Un facteur $\alpha_S$ égal à 1 signifie que la source $S$ n'est pas du tout fiable, et donc l'information qu'elle produit ne peut pas être prise en compte. Contrairement, un facteur $\alpha_S = 0$ signifie que la source $S$ est complètement fiable, et donc l'information est entièrement acceptée. Pratiquement, la source a un certain degré de fiabilité $\beta_S = 1 - \alpha_S \in [0, 1]$, qui est soit connu a priori, soit évalué à partir de modèles théoriques ou des expérimentations.

Une autre approche d'affaiblissement utilisée pour corriger l'information fournie par une source à partir d'une méta-connaissance, est l'affaiblissement contextuel. L'idée principale de ce dernier consiste à varier la fiabilité d'une source selon le contexte envisagé et donc selon les réponses considérées [Mercier *et al.*, 2012].

#### A.2.2.4 Prise de décision

Tout comme toutes les théories de fusion de données, la TFC a ont pour finalité la prise de décision. En ce qui concerne la partie décisionnelle, elle correspond au niveau pignistique [Smets, 2005]. La phase de décision s'appuie sur la distribution pignistique, notée $BetP(\cdot)$ obtenue à partir de la fonction de masse finale $m(\cdot)$. Elle est aussi appelée probabilité pignistique pour le jeu de probabilité sur les singletons qu'elle génère. La probabilité pignistque, notée $BetP(\cdot)$, consiste à repartir de manière équiprobable la masse d'une proposition $A_i$ sur les hypothèses contenues dans $A_i$ [Kennes & Smets, 1990]. Formellement, la probabilité pignistique $BetP(\cdot)$ est définie par :

$$BetP(A) = \sum_{A \subseteq A_i} \frac{m(A_i)}{|A_i|}. \tag{A.5}$$

Il s'agit d'une estimation de la probabilité que la bonne réponse tombe dans $A$, étant donné les informations mesurées. Enfin, l'ensemble $A$ ayant la plus grande probabilité pignistique est choisi. D'autres méthodes de prise de décision consistent à maximiser les fonctions de masses, les fonctions de croyance, les fonctions de plausibilité, etc.

## A.3 Localisation par zonage

La localisation des capteurs dans un RCSF est primordiale pour un grand nombre d'applications. Nous nous intéressons dans cette thèse à la localisation des personnes âgées dépendantes. Dans cette application, la détermination de la zone où se trouve la personne âgée est suffisante, sans besoin de localisation exacte. Pour cette raison, nous abordons le problème de localisation par zonage. Les méthodes de localisation basées sur les RSSIs sont considérées les plus populaires puisqu'elles présentent de nombreux avantages du point de vue de leur simplicité ainsi que du coût de matériel. Certaines de ces techniques exploitent l'atténuation de la puissance du signal avec la distance parcourue pour estimer les distances séparant les capteurs. Cependant, ces techniques ne sont pas toujours efficaces, puisque les signaux sont sensibles à la présence du bruit et des interférences. D'autres méthodes s'appuient sur la technique du fingerprinting

qui consiste à collecter des informations du réseau pour construire une base de données à utiliser pour la localisation.

Dans cette section, nous proposons une approche de localisation par zonage basée sur la théorie des fonctions de croyance et la technique du fingerprinting. Nous utilisons les signaux reçus des points d'accès WiFi (APs) et mesurons leurs puissances (RSSIs). Par la suite, nous définissons dans le cadre de la théorie des fonctions de croyance un modèle d'observation qui prend en entrée la puissance du signal échangé entre les APs et le MN et donne en sortie une estimation de la zone du MN en question.

### A.3.1 Problématique

Nous considérons un environnement partitionné en $N_Z$ zones, notées $Z_j, j \in \{1, 2, \ldots, N_Z\}$, et $N_{AP}$ APs de positions fixes, notées $AP_k,\ k \in \{1, 2, \ldots, N_{AP}\}$. L'approche proposée est centralisée; ainsi, toutes les données recueillies sont acheminées vers la station centrale de fusion, où tous les traitements et calculs seront effectués. Basée sur la technique du fingerprinting, notre méthode consiste à construire une base de données en offline, à utiliser par la suite en temps réel, pour estimer la zone du MN. Notons que plusieurs nœuds peuvent être localisés à la fois, puisqu'ils sont localisés chacun indépendamment des autres.

Pour construire la base de données, nous considérons $N_j$ positions de référence dans chaque zone, notées $\rho_{j,k,\ell}, \ell \in \{1, \ldots, N_j\}$, et générées de façon uniforme ou aléatoire dans la région. Les APs diffusent régulièrement des signaux dans la région étudiée.

Soit $\boldsymbol{\rho}_t$ le vecteur des RSSIs de taille $N_{AP}$ reçu par le MN à l'instant $t$ de tous les APs,

$$\boldsymbol{\rho}_t = (\rho_{t,1}, \ldots, \rho_{t,N_{AP}}), \tag{A.6}$$

où $\rho_{t,k}$ est la puissance du signal reçu de $AP_k$ à l'instant $t$. Puisque tous les APs ne sont pas nécessairement détectés à tout moment, on note $I_{AP,t}$ l'ensemble des indices des APs pour lesquels les signaux sont détectés par le MN à l'instant $t$ et $\rho_{t,k}, k \in I_{AP,t}$, leurs RSSIs mesurées. Le vecteur $\boldsymbol{\rho}_t$ est complété par des zéros aux positions où les APs ne sont pas détectés. La méthode de localisation consiste

par la suite à trouver un modèle $\mathbb{O} : \mathbb{R}^{N_{AP}} \rightarrow [0,1]^{N_Z}$ qui prend en entrée un vecteur RSSI $\boldsymbol{\rho}_t$, et donne en sortie une confiance $m_{\mathbb{O},t}(Z_j)$ pour chaque zone $Z_j$ à l'instant $t$.

### A.3.2 Représentation statistique des données

Le modèle d'observation est construit comme suit. Soit $\mathcal{Z} = \{Z_1, \ldots, Z_{N_Z}\}$ l'ensemble de toutes les zones et $P(\mathcal{Z}) = 2^{\mathcal{Z}}$ l'ensemble de tous les sur-ensembles de $\mathcal{Z}$, c'est à dire, $P(\mathcal{Z}) = \{\emptyset, \{Z_1\}, \ldots, \mathcal{Z}\}$. D'abord, les RSSIs de tous les sous-ensembles de $P(\mathcal{Z})$ sont modélisées par une distribution statistique. On distingue deux types de distributions, paramétriques et non-paramétriques. Les deux types de modélisation sont détaillés dans la Section 3.3.1. L'utilisation de l'un ou l'autre des deux types mène à une distribution statistique $Q_{A,k}(\cdot), A \in P(\mathcal{Z}), k \in \{1, 2 \ldots, N_{AP}\}$, qui représente les variations des RSSIs de chaque sous-ensemble $A$ par rapport à chaque $AP_k$.

### A.3.3 Affectation des masses

Une fonction de masse $m_{AP_k,t}(\cdot)$ est une fonction de $P(\mathcal{Z})$ à l'intervalle $[0,1]$, définie selon une source $AP_k, k \in \{1, \ldots, N_{AP}\}$, et satisfaisant

$$\sum_{A \in P(\mathcal{Z})} m_{AP_k,t}(A) = 1. \tag{A.7}$$

La masse $m_{AP_k,t}(A)$ est évaluée comme suit,

$$m_{AP_k,t}(A) = \frac{Q_{A,k}(\rho_{t,k})}{\sum_{A' \in P(\mathcal{Z}), A' \neq \emptyset} Q_{A',k}(\rho_{t,k})}, \quad A \in P(\mathcal{Z}), A \neq \emptyset. \tag{A.8}$$

### A.3.4 Affaiblissement

Nous considérons une approche d'affaiblissement contextuel parce que nous supposons que la fiabilité des APs varie selon les zones. En effet, un AP est plus fiable quand il distingue des zones asymétriques par rapport à lui, que dans le cas de zones symétriques.

Soit $\{A_1, \ldots, A_L\}$ une partition de $\mathcal{Z}$. Dans ce modèle contextuel, nous considérons le degré de chaque AP conditionné à chaque un sous-ensemble $A_l, l \in$

$\{1,\ldots,L\}$. Pour tout $l \in \{1,\ldots,L\}$, $\beta_k^l = 1-\alpha_k^l$ represente le degré de fiabilité de $AP_k$ sachant que l'observation appartienne à $A_l$. La partition que nous considérons est l'ensemble de singletons $\{\{Z_1\},\ldots,\{Z_{N_Z}\}\}$, et ainsi la fiabilité $AP_k$ par rapport au contexte $Z_j$ est $\beta_k^j$. La masse ${}^{\alpha}m_{AP_k}(A)$ obtenue après l'affaiblissement contextuel de la source $AP_k$ est calculée comme suit,

$$ {}^{\alpha}m_{AP_k}(A) = m_{AP_k,t} \textcircled{\cup} m^0_{AP_k,t}(A), \tag{A.9} $$

où $m^0_{AP_k,t}(A)$ est définie comme suit,

$$ m^0_{AP_k,t}(A) = m^1_{AP_k,t} \textcircled{\cup} m^2_{AP_k,t} \textcircled{\cup} \ldots \textcircled{\cup} m^{N_Z}_{AP_k,t}(A), \tag{A.10} $$

où chaque $m^j_{AP_k,t}, j \in \{1,\ldots,N_Z\}$, est calculée par

$$ m^j_{AP_k,t} = \begin{cases} (1-\alpha_k^j), & \text{si } A = \emptyset; \\ \alpha_k^j, & \text{si } A = A_l; \\ 0, & \text{sinon.} \end{cases} \tag{A.11} $$

Le taux d'erreur $\alpha_k^j(A)$ d'un sous-ensemble $A$ telle que la vérité est $Z_j$ par rapport à $AP_k$ est calculé comme suit,

$$ \alpha_k^j(A) = \int_{\mathbb{D}_{A,k}} Q_{A,k}(\rho)d\rho, \tag{A.12} $$

où $\mathbb{D}_{A,k}$ est le domaine d'erreur du sous-ensemble $A$ par rapport à $AP_k$, défini par

$$ \mathbb{D}_{A,k} = \{\rho \mid Q_{\{Z_j\},k}(\rho) \leq \max_{A' \in P(\mathcal{Z}), A' \neq A} (Q_{A',k}(\rho))\}. \tag{A.13} $$

### A.3.5 Fusion de l'information

Les fonctions de masse ${}^{\alpha}m_{AP_k,t}(\cdot)$ sont définies selon le vecteur de RSSIs $\rho_{t,k}, k \in I_{AP,t}$ récupéré d'un certain nombre d'APs. La fusion de l'information consiste à combiner les masses de tous les APs [Kurdej & Cherfaoui, 2013]. Les fonctions de masses peuvent être combinées par une des méthodes décrites dans la Section 2.5.2. La masse obtenue par la règle de Dempster est calculée comme suit,

$$ m_{\bigoplus,t}(A) = \frac{\sum_{\substack{A^{(k)} \in P(\mathcal{Z}) \\ \cap_k A^{(k)} = A}} \prod_{k \in I_{AP,t}} {}^{\alpha}m_{AP_k,t}(A^{(k)})}{1 - \sum_{\substack{A^{(k)} \in P(\mathcal{Z}) \\ \cap_k A^{(k)} = \emptyset}} \prod_{k \in I_{AP,t}} {}^{\alpha}m_{AP_k,t}(A^{(k)})}, \tag{A.14} $$

pour tous les sous-ensembles $A \in P(\mathcal{Z})$, où $A^{(k)}$ est le sous-ensemble $A$ par rapport à la source $AP_k$. L'avantage de l'utilisation de la TFC réside dans la modélisation de l'ambiguïté en affectant des masses à la fois aux singletons et à tous leurs sur-ensembles.

### A.3.6 Estimation de la confiance

Pour attribuer un niveau de confiance aux singletons, la transformation pignistique est utilisée dans le cadre de la TFC [Smets, 1993a]. Elle est définie par

$$BetP_t(A) = \sum_{A \subseteq A'} \frac{m_{\oplus,t}(A')}{|A'|}, \tag{A.15}$$

où $A$ est un singleton de $P(\mathcal{Z})$. Le niveau de confiance associé à chaque zone par le modèle d'observation à l'instant $t$ peut être calculé comme suit,

$$m_{\mathbb{O},t}(Z_j) = BetP_t(\{Z_j\}), j \in \{1, \ldots, N_Z\}. \tag{A.16}$$

Le modèle d'observation $\mathbb{O}$ est ainsi déduit,

$$\mathbb{O}(\boldsymbol{\rho}_t) = (m_{\mathbb{O},t}(Z_1), \ldots, m_{\mathbb{O},t}(Z_{N_Z})). \tag{A.17}$$

La zone ayant la plus grande confiance est alors sélectionnée.

### A.3.7 Expérimentations

Dans cette sous-section, nous évaluons les performances de la méthode proposée par des expérimentations dans un secteur de l'Université de Technologie de Troyes, France. Le secteur d'une surface de 190 $m^2$ est divisé en huit zones, comme le montre la figure 3.7. Un PC avec le logiciel *WiFi scanner* peut distinguer les APs par leurs adresses MAC. Le logiciel mesure les RSSIs des signaux émis. Nous utilisons six APs parmi tous les APs détectés dans la région cible. Un ensemble de 30 mesures est pris dans chaque zone, une partie étant sélectionnée pour la construction de la base de données, et le reste pour le test et la validation. Le tableau 3.4 résume les paramètres de la configuration expérimentale.

#### A.3.7.1 Illustration de la méthode proposée

Pour illustrer la méthode proposée, 21 mesures de RSSIs recueillies dans chaque zone sont sélectionnées pour la construction de la base de données et 9 mesures pour le test. Afin d'estimer la zone du MN qui se déplace, nous utilisons le modèle d'observation proposé pour différentes techniques de modélisation, d'affaiblissement et de fusion de l'information. Le tableau 3.5 montre le nombre des estimations erronées dans le cas d'une modélisation paramétrique, affaiblissement contextuel et la règle de combinaison de Dempster. Le tableau 3.6 montre l'influence de la technique d'affaiblissement et la règle de combinaison sur la performance de la méthode proposée. Nous pouvons voir que la plus faible erreur est obtenue avec la technique de Dempster et l'affaiblissement contextuel. Le tableau 3.8 compare les performances de la méthode pour des modélisations paramétrique et non-paramétrique. Puisqu'il n'y a pas une amélioration significative, la modélisation paramétrique est considérée parce qu'elle est moins complexe. Nous analysons la performance de la méthode pour une distribution aléatoire des positions de référence, comme le montre la figure 3.1(b). Nous observons une légère augmentation de l'erreur dans le tableau 3.8. Ceci peut être expliqué par le fait qu'une distribution uniforme permet une meilleure couverture de la région d'intérêt, tandis qu'une distribution aléatoire ne garantit pas toujours une bonne couverture. Néanmoins, les résultats sont toujours satisfaisants, et les distributions aléatoires permettent d'avoir une localisation par zonage assez exacte quand les distributions uniformes ne sont pas applicables. Pour étudier l'influence du nombre de zones sur la performance de l'approche proposée, nous considérons une surface de 500 $m^2$, constituée de 21 zones. La précision de la méthode diminue à 77.78% pour une modélisation paramétrique et une distribution uniforme de positions de références. L'approche proposée se trouve vulnérable à l'augmentation du nombre de zones.

#### A.3.7.2 Comparaison à d'autres méthodes

L'objectif de ce paragraphe est de fournir une comparaison de l'approche proposée à l'égard de deux techniques de classification multi-classes : le classifieur Bayésien naïf (NB) [Liu *et al.*, 2013a] et la régression logistique (MLR) [Mauša *et al.*, 2012].

Le tableau 3.9 montre les erreurs d'estimation pour différents nombres de points de références pour NB, MLR et la méthode proposée. Les résultats montrent que la méthode proposée est plus précise que les deux autres.

## A.4 Modèle d'observation avancé

Dans la section précédente, nous avons proposé un modèle d'observation pour la localisation des capteurs par zonage, en utilisant la TFC et la technique de fingerprinting. Nous appelons ce modèle, un modèle d'observation basique (BOM). Comme nous avons montré dans la partie d'expérimentations, la performance du BOM se dégrade lors de l'augmentation du nombre de zones. Dans cette section, nous développons un modèle d'observation avancé (EOM), qui résout les problèmes du BOM au travers du clustering hiérarchique et sélection des APs. Le but du EOM est de couvrir des surfaces plus grandes avec un nombre de zones plus élevé. Les zones sont regroupés dans des clusters, en utilisant la méthode du clustering ascendant hiérarchique, formant un dendrograme. Le nombre optimal des clusters est déterminé en optimisant les distances inter- et intra-clusters. L'hiérarchie est ensuite réformée sur deux niveaux, le premier comprenant les clusters sélectionnés et le deuxième comprenant les zones de chaque cluster. L'objectif de la méthode est ainsi de déterminer le cluster correct et la zone correcte dans le premier et le deuxième niveau respectivement. Nous développons ensuite une méthode de sélection de paramètres, qui sont les bornes WiFi à privilégier. Cette méthode permet d'améliorer la précision et de réduire la complexité de la méthode de localisation, en choisissant le meilleur ensemble de paramètres en termes de la capacité de discrimination et de la redondance.

### A.4.1 Algorithme de clustering

L'objectif du développement d'un algorithme de clustering est de réduire le nombre de zones à classifier en un instant. Par ailleurs, l'algorithme doit grouper les zones de façon qui facilite la discrimination entre elles dans la phase de localisation. Plusieurs approches de clustering ont été proposées, la plus utile dans notre cas étant le clustering hiérarchique ascendant [Babbar *et al.*, 2013; Manning *et al.*,

2008]. Le clustering hiérarchique ascendant ou *hierarchical agglomerative clustering* consiste à regrouper successivement des clusters en fonction d'une mesure de distance qui caractérise leur similarité ou leur dissimilarité. C'est un procédé itératif, partant d'un certain nombre de clusters initial. A chaque itération, les deux clusters les plus éloignés sont regroupés en un, et la distance avec les autres clusters est mise à jour. L'algorithme s'arrête lorsqu'il ne reste plus qu'un cluster ou lorsqu'un critère d'arrêt est atteint (par exemple, lorsque la distance décidant du prochain regroupement atteint un seuil). L'algorithme est illustré par la figure 4.3.

Nous nous intéressons à maximiser la distance entre les zones, pour que ça soit plus facile de les distinguer. Nous utilisons la distance de Kullback-Leibler pour évaluer la distance entre les différentes entités (clusters ou zones). La divergence de Kullback-Leibler de deux distributions $Q_{Z_{j'}}$ and $Q_{Z_j}$ d'une variable continue $\boldsymbol{\rho}$ est définie comme suit,

$$D_{KL}(Q_{Z_{j'}}||Q_{Z_j}) = \int_{\boldsymbol{\rho}} \log\left(\frac{Q_{Z_{j'}}(\boldsymbol{\rho})}{Q_{Z_j}(\boldsymbol{\rho})}\right) Q_{Z_{j'}}(\boldsymbol{\rho}) d\boldsymbol{\rho}. \tag{A.18}$$

Cette distance est non-symétrique, c'est à dire, $D_{KL}(u||v) \neq D_{KL}(v||u)$. La J-divergence symétrise la distance de Kullback-Leibler comme suit [Nielsen & Nock, 2017],

$$D_J(Q_{Z_{j'}}||Q_{Z_j}) = D_{KL}(Q_{Z_{j'}}||Q_{Z_j}) + D_{KL}(Q_{Z_j}||Q_{Z_{j'}}). \tag{A.19}$$

Le calcul de cette divergence entre les distributions paramétriques et non-paramétriques se trouve dans la section 4.4.1.1.

Cependant, la considération de tout le dendrograme est coûteux en ressources informatiques. De plus, ça risque de propager l'erreur tout au long de l'hiérarchie. Pour cela, le dendrograme est découpé basé selon un nombre optimal de clusters. Étant donné qu'il n'existe aucune connaissance préalable de ce nombre, il est calculé en résolvant un problème d'optimisation qui prend en compte les distances intra- et inter-clusters. Krzanowski & Lai [1988] proposent une technique pour trouver le nombre optimal de clusters $N_C$ comme suit,

$$N_C = \underset{r}{\operatorname{argmax}} \left| \frac{DIFF(r)}{DIFF(r+1)} \right|, \tag{A.20}$$

telle que,

$$DIFF(r) = (r-1)^{\frac{2}{N_{AP}}} W(r-1) - (r)^{\frac{2}{N_{AP}}} W(r), \tag{A.21}$$

où $W(r)$ est la somme des carrés des distances intra-clusters, calculée par

$$W(r) = \sum_{i=1}^{r} \sum_{\substack{j \\ Z_j \in C_i}} \sum_{\ell=1}^{N_j} ||\boldsymbol{\rho}_{j,\ell} - \mu_i||^2, \tag{A.22}$$

où $\boldsymbol{\rho}_{j,\ell}$ est la mesure de RSSI dans la zone $Z_j$.

Le dendrograme est découpé au niveau des $N_C$ clusters, notés $C_i, i \in \{1, \ldots, N_C\}$. Soit $I_i$ l'ensemble des indices des zones inclues dans le cluster $C_i$, c'est à dire, $Z_j \in C_i, \ \forall j \in I_i$. L'hiérarchie obtenue est ainsi de deux niveaux, le premier étant un ensemble de $N_C$ clusters, $\{C_1, \ldots, C_{N_C}\}$ et le deuxième étant un ensemble de zones appartenant à chaque cluster, $Z_j \in C_i, \forall j \in I_i$. La méthode se résume à déterminer, pour toute nouvelle observation, le cluster correct au premier niveau et la zone correcte au sein du cluster au second niveau. L'hiérarchie de deux niveaux est illustrée dans la figure 4.4.

### A.4.2 Algorithme de sélection de paramètres

L'objectif de la sélection de paramètres est de déterminer le meilleur ensemble parmi les $2^{N_{AP}} - 1$ sous-ensembles candidats de l'ensemble des APs qui satisfait les deux objectifs : la minimisation de l'erreur de localisation et la réduction de la dépendance entre les paramètres Tabakhi & Moradi [2015]. Soient $F = \{AP_1, \ldots, AP_{N_{AP}}\}$ l'ensemble de tous les APs, $z$ une entité qui désigne une zone ou un cluster et $N_z$ le nombre de ces entités. Nous proposons un algorithme de sélection de paramètres qui maximise la capacité discriminatoire et réduit la redondance des APs sélectionnés. Soient $F' \subseteq F$ un ensemble non vide de $F$ et $Q_{z_j,F'}(\cdot)$ une distribution representant les variations des RSSIs de l'entité $z_j$ par rapport à l'ensemble de paramètres $F'$.

D'une part, l'erreur de classification est inversement liée à la capacité discriminatoire des paramètres définie par [Jahromi, 2007]

$$DisC(F') = \sum_{a=1}^{N_z} \sum_{b=1}^{N_z} D_{KL}(Q_{z_a,F'} || Q_{z_b,F'}), \tag{A.23}$$

$D_{KL}(Q_{z_a,F'}||Q_{z_b,F'})$ est la distance de Kullback-Leibler entre les distributions des entités $z_a$ et $z_b$. L'erreur de l'ensemble $F'$ est ainsi déduite comme suit,

$$\mathcal{E}(F') = 2^{-DisC(F')}. \tag{A.24}$$

D'autre part, la dépendance est un facteur essentiel de la sélection pour avoir un sous-ensemble réduit. Le coefficient de corrélation multiple permet de mesurer le degré de dépendance d'une caractéristique par rapport aux autres. Le coefficient de la corrélation multiple d'une caractéristique $AP_k$ de $F'$ par rapport aux autres éléments de $F' \setminus \{AP_k\}$ est défini par

$$R_k^2 = c_k^T R_{xx,k}^{-1} c_k, \tag{A.25}$$

où $c_k$ est un vecteur colonne d'éléments $d_{AP_{k'}AP_k}$ pour $AP_{k'} \in F' \setminus \{AP_k\}$, $d_{AP_{k'}AP_k}$ étant la corrélation entre $AP_{k'}$ et $AP_k$, $c_k^T$ étant le vecteur transposé de $c_k$ et $R_{xx,k}^{-1}$ est l'inverse de la matrice d'éléments $d_{AP_{k'}AP_{k''}}$ pour toutes les paires $AP_{k'}$ et $AP_{k''}$ appartenant à $F' \setminus \{AP_k\}$. La redondance de tous les APs de $F'$ est la moyenne des coefficients de corrélation multiple pour tout $AP_k \in F'$,

$$\mathcal{R}(F') = \sum_k \frac{R_k}{|F'|}, \text{ où } |F'| \text{ est le cardinal de } F'. \tag{A.26}$$

L'objectif de la sélection de paramètres est de trouver l'ensemble $F_s \subseteq F$ tel que $\mathcal{E}(F_s)$ et $\mathcal{R}(F_s)$ sont simultanément minimisées. Une recherche exhaustive étant très coûteuse, un algorithme glouton est utilisé avec une stratégie d'élimination régressive. Commençant par l'ensemble complet des paramètres, nous supprimons successivement le paramètre le moins utile. Soit $F_y$ le sous-ensemble choisi à l'itération $y \geq 1$, avec $F_0 = F$ et le cardinal de $F_y$ égal à $|F_y| = N_{AP} - y$. A chaque itération $y \geq 1$, tous les sous-ensembles de $F_{y-1}$ ayant $N_{AP} - y$ éléments sont considérés. Soit $F_y^{(\lambda)}, \lambda = 1, \ldots, N_{AP} - y + 1$, le terme désignant ces sous-ensembles. La fonction bi-objective $g_y\big(F_y^{(\lambda)}\big)$ est alors définie comme suit,

$$g_y(F_y^{(\lambda)}) = \eta \frac{\mathcal{E}(F_{y-1}) - \mathcal{E}(F_y^{(\lambda)})}{\max(\mathcal{E}(F_{y-1}), \mathcal{E}(F_y^{(\lambda)}))} + (1-\eta) \frac{\mathcal{R}(F_{y-1}) - \mathcal{R}(F_y^{(\lambda)})}{\max(\mathcal{R}(F_{y-1}), \mathcal{R}(F_y^{(\lambda)}))}, \tag{A.27}$$

où $\eta \in [0, 1]$ contrôle le compromis entre les deux fonctions objectives. L'ensemble $F_y^{(\lambda)}$ qui a la plus grande valeur positive de $g_y(F_y^{(\lambda)})$ est sélectionné à l'itération $y$.

Si toutes les valeurs sont négatives, alors il n'y a plus d'amélioration possible, et donc l'ensemble final est $F_s = F_{y-1}$. Le sous-ensemble optimal obtenu sera utilisé dans la suite pour la classification. Cet algorithme est appliqué au niveau des clusters pour générer $F_s$ et au niveau des zones de chaque cluster pour générer $F_{s,i}, i \in \{1, \ldots, N_C\}$, comme le montre la figure 4.4.

### A.4.3 Estimation de la confiance

La TFC est appliquée au premier niveau de l'hiérarchie pour générer les niveaux pignistiques $BetP_t^C(\{C_i\})$, $i \in \{1, \ldots, N_C\}$ correspondant aux clusters et au second niveau de l'hiérarchie pour générer les niveaux pignistiques $BetP_t^i(\{Z_j\})$, $j \in I_i$ correspondant aux zones de chaque cluster $C_i$. Le EOM attribue un niveau de confiance à chacune des zones en combinant les niveaux pignistiques des clusters et des zones,

$$m_{\mathbb{O},t}(Z_j) = BetP_t^C(\{C_i\}) \times BetP_t^i(\{Z_j\}). \tag{A.28}$$

### A.4.4 Expérimentations

Nous évaluons les performances de la méthode proposée par des expérimentations au premier étage de l'équipe M2S et au Living Lab de l'Université de Technologie de Troyes, France. Les deux plans sont illustrés dans la figure 4.5. Le tableau 4.2 résume les paramètres des deux installations expérimentales. Le Tableau 4.3 évalue les performances de la méthode en fonction du type de modélisation. Dans les deux expérimentions, une meilleure performance est atteinte par la modélisation non-paramétrique. Nous étudions également l'influence du paramètre $\eta$. Le tableau 4.4 montre la précision et le temps d'exécution de la méthode de localisation pour $\eta \in \{0.25, 0.5, 0.75\}$. Nous évaluons ensuite l'influence de chaque phase de l'EOM sur la performance de la méthode. Les tableaux 4.5 et 4.6 montrent l'amélioration apportée par chaque phase et son influence sur le temps d'exécution.

La méthode proposée est comparée à d'autres méthodes de classification multi-classes. Les méthodes de classification telles que les k-plus proches voisins, le Bayésien naïf, la régression logistique et les machines à vecteur support sont

considérées. De plus, la méthode proposée est comparée avec les approches hiérarchiques telles que les machines à vecteur support hiérarchique et les forêts d'arbres décisionnels. Les tableaux 4.7 et 4.8 montrent la précision de l'approche proposée dans les deux expérimentations en comparaison avec les méthodes de classification mentionnées. La méthode proposée surpasse les autres en termes de précision avec un temps d'exécution compétitif.

## A.5 Suivi de trajectoire

Le suivi de trajectoires est une application très populaire des RCSFs. Le suivi de trajectoire consiste à estimer de manière récursive la position du MN. En utilisant un modèle de mobilité qui décrit le mouvement du MN, nous pouvons d'abord prédire son état futur à partir de son état actuel. Ensuite, l'état prédit est mis à jour en utilisant les observations du réseau, à savoir, un modèle d'observation. Plusieurs méthodes de fusion de données peuvent être utilisées pour combiner les informations de mobilité et les observations. Le filtre de Kalman (KF) peut être utilisé dans le cas d'un modèle d'observation linéaire. Dans le cas de la non-linéarité, le filtre de Kalman étendu (EKF) et le filtre de Kalman sans parfum (UKF) peuvent être utilisés. Cependant, de telles approches effectuent des linéarisations et des approximations conduisant à une performance sous-optimale et parfois à la divergence. Le filtre particulaire (PF) est également utilisé pour le suivi de trajectoires. Un tel filtre a plus de potentiel que le filtre de Kalman dans le cas de bruits non gaussiens et de modèles non linéaires. Cependant, la génération d'échantillons et l'étape de ré-échantillonnage rendent les algorithmes employant ce filtre plus complexes en termes de calculs que le filtre de Kalman. Plusieurs techniques de suivi utilisant des informations de mobilité ont été proposées dans la littérature. En plus des mesures de RSSIs, ces techniques emploient un modèle de mobilité pour affiner l'estimation de la position grâce à sa position antécédente.

Dans cette section, nous effectuons le suivi de trajectoire en utilisant les techniques de localisation introduites aux sections précédentes. Ainsi, en utilisant la TFC et la technique de fingerprinting, nous définissons les modèles d'observation

pour donner une première estimation de la zone du MN. Ensuite, nous combinons cette première estimation à un modèle de mobilité pour obtenir une estimation plus affinée de la zone du MN. Dans ce but, trois modèles de mobilité sont proposés.

### A.5.1 Problématique

Nous considérons la même configuration et les mêmes notations des sections précédentes, où $N_Z$ est le nombre de zones de la région cible, notées $Z_j,\ j = 1, 2, \ldots, N_Z$, $N_{AP}$ est le nombre de APs, notés $AP_k,\ k = 1, 2 \ldots, N_{AP}$ et $\boldsymbol{\rho}_t$ est le vecteur de RSSI reçu par le MN à l'instant $t$ des APs détectés. Soit $v_{max}$ la vitesse maximale du MN dans le milieu couvert. Dans notre application, $v_{max}$ est la vitesse maximale prévue du mouvement des personnes âgées dépendantes à l'intérieur. L'objectif de l'algorithme proposé est de trouver une fonction $\mathbb{T} : \mathbb{R}^{N_{AP}} \rightarrow [0, 1]^{N_Z}$ telle que $\mathbb{T}(\boldsymbol{\rho}_t) = (\mathcal{W}_t(Z_1), \ldots, \mathcal{W}_t(Z_{N_Z}))$, où $\mathcal{W}_t(Z_j)$ est le niveau de confiance d'avoir le MN dans chacune des zones $Z_j, j = 1, 2, \ldots, N_Z$ à l'instant $t$.

D'une part, la méthode profite de la mobilité du MN pour donner une première estimation de sa zone. Le MN est supposé se déplacer avec une vitesse inférieure ou égale à la vitesse maximale $v_{max}$. Sur cette base, nous pouvons avoir une idée des destinations possibles du MN selon l'architecture de la région cible et le temps d'exécution de l'algorithme de localisation. Le premier modèle de mobilité est basé sur la succession originale des zones et ne nécessite aucune phase d'acquisition de données supplémentaire. Le deuxième modèle de mobilité divise les zones en des sous-zones et requiert une phase additionnelle d'acquisition de données dans chaque souos-zone. Le troisième modèle de mobilité est basé sur les modèles de Markov cachés et utilise la trajectoire du MN pour le suivre. Chacun de ces trois modèles attribue une masse $m_{\mathbb{M},t}(\cdot)$ à chaque zone.

D'autre part, les modèles d'observation décrits dans les sections précédentes attribuent une autre masse $m_{\mathbb{O},t}(\cdot)$. Dans le cadre de la TFC, la masse du modèle de mobilité est combinée avec celle du modèle d'observation pour attribuer un niveau de confiance $\mathcal{W}_t(\cdot)$ à chaque zone.

### A.5.2 Premier modèle de mobilité

Soit $\Delta t_{loc}$ l'intervalle de temps dans lequel l'algorithme de localisation est exécuté et $d_{min,ij}$ la distance géographique minimale entre $Z_i$ et $Z_j$. La distance maximale que le MN peut couvrir est ainsi déduite, $d_{max} = v_{max} \times \Delta t_{loc}$. La figure 5.2 illustre l'architecture du premier modèle de mobilité. Soit $p_{ij}, i, j \in \{1, \ldots, N_Z\}$, le coefficient de transition de la zone $Z_i$ à la zone $Z_j$ pendant la période de localisation $\Delta t_{loc}$,

$$p_{ij} = \begin{cases} 0, & \text{si } d_{max} < d_{min,ij}; \\ 1, & \text{si } d_{max} \geq d_{min,ij}. \end{cases} \tag{A.29}$$

La masse attribuée par le modèle de mobilité est calculée par

$$m_{\mathbb{M},t}(Z_j) = \sum_{i=1}^{N_Z} p_{ij} \times \frac{\mathcal{W}_{t-1}(Z_i)}{\sum_{f=1}^{N_Z} p_{if}}. \tag{A.30}$$

La confiance associée à chacune des zones est alors déterminée par

$$\mathcal{W}_t(Z_j) = m_{\mathbb{M} \bigoplus \mathbb{O},t}(Z_j) = \frac{m_{\mathbb{O},t}(Z_j) \times m_{\mathbb{M},t}(Z_j)}{\sum_{\chi=1}^{N_Z} m_{\mathbb{O},t}(Z_\chi) \times m_{\mathbb{M},t}(Z_\chi)}. \tag{A.31}$$

### A.5.3 Deuxième modèle de mobilité

Dans ce modèle, chaque zone est divisée à $N_{X_i}$ sous-zones $X_{i.\ell}$ selon son architecture, qui sont les $N_{X_i} - 1$ sous-zones de connexion et une sous-zone principale $X_{i.N_{X_i}}$, comme le montre la figure 5.3. Soit $Q_{i.\ell}(\cdot)$ la distribution représentant les variations des RSSIs dans chacune des sous-zones de connexion $X_{i.\ell}, \ell \in \{1, \ldots, N_{X_i} - 1\}$ et $Q_{i.N_{X_i}}(\cdot)$ la distribution représentant les variations des RSSIs dans la sous-zone principale $X_{i.N_{X_i}}$. Des poids d'appartenance du MN à chacune des sous-zones sont calculés :

$$q_{t-1}(X_{i.\ell}) = \frac{Q_{i.\ell}(\boldsymbol{\rho}_{t-1})}{\sum_{\chi=1}^{N_{X_i}} Q_{i.\chi}(\boldsymbol{\rho}_{t-1})}. \tag{A.32}$$

La confiance de chaque zone $Z_i$ à $t-1$ est transférée à ses sous-zones,

$$\mathcal{W}^*_{t-1}(X_{i.\ell}) = \mathcal{W}_{t-1}(Z_i) \times q_{t-1}(X_{i.\ell}). \tag{A.33}$$

Soit $r_{i.\ell,j}$ le coefficient de transition de la sous-zone de connexion $X_{i.\ell}, \ell \in \{1,\ldots,N_{X_i}\}$ de la zone $Z_i, i \in \{1,\ldots,N_Z\}$, à la zone originale $Z_j, j \in \{1,\ldots,N_Z\}$,

$$r_{i.\ell,j} = \begin{cases} 0, & \text{si } d_{max} < d_{min,i.\ell,j}; \\ 1, & \text{si } d_{max} \geq d_{min,i.\ell,j}, \end{cases} \tag{A.34}$$

où $d_{min,i.\ell,j}$ est la distance minimale entre la sous-zone de connexion $X_{i.\ell}$ et la zone $Z_j$. La masse associée à chaque zone par le modèle de mobilité est ainsi déduite,

$$m_{\mathbb{M},t}(Z_j) = \mathcal{W}^*_{t-1}(X_{j.N_{X_j}}) + \sum_{i=1}^{N_Z} \sum_{\ell=1}^{N_{X_i}-1} r_{i.\ell,j} \times \frac{\mathcal{W}^*_{t-1}(X_{i.\ell})}{\sum_{f=1}^{N_Z} r_{i.\ell,f}}. \tag{A.35}$$

La confiance associée à chacune des zones est alors déterminée par

$$\mathcal{W}_t(Z_j) = m_{\mathbb{M}\bigoplus\mathbb{O},t}(Z_j) = \frac{m_{\mathbb{O},t}(Z_j) \times m_{\mathbb{M},t}(Z_j)}{\sum_{\chi=1}^{N_Z} m_{\mathbb{O},t}(Z_\chi) \times m_{\mathbb{M},t}(Z_\chi)}. \tag{A.36}$$

### A.5.4 Troisième modèle de mobilité

Dans ce modèle, nous profitons de la trajectoire du MN pour le suivre. L'objectif est de détecter une transition d'une zone à une autre dans une période de temps. Pour ce faire, nous utilisons les modèles de Markov cachés (HMM) pour déterminer une probabilité que le MN a suivi une certaine trajectoire. Chaque HMM $\Lambda$ est définie par trois paramètres, $\Lambda = (A, B, \pi)$, où $A$ est la matrice de transition, $B$ est le modèle d'émission et $\pi$ est le vecteur d'états initiaux. Quand une séquence d'états $S = \{s_1,\ldots,s_\alpha\}$ est déterminée, nous pouvons observer une séquence $R = \{R_1,\ldots,R_\alpha\}$. Nous nous intéressons à la probabilité $P(R|\Lambda)$, qui désigne la probabilité d'observer la séquence $R$ pour un modèle $\Lambda$ donné. Le calcul détaillé de cette probabilité se trouve dans la section 5.5.1.3.

Nous construisons des HMMs, notés $\Lambda_{ij}, i,j \in \{1,\ldots,N_Z\}$, où $\Lambda_{ij}$ est un HMM de $N_S$-états correspondant à une transition de la zone $Z_i$ à la zone $Z_j$. Le paramètre $N_S$ est le nombre d'états et est choisi par l'utilisateur. Dans la phase offline, une région de transition est construite entre chaque paire de zones voisines, comme le montre la figure 5.5(a). Cette région est divisée en $N_S$ états. A chaque état, des mesures de RSSIs sont recueillies. Les trajectoires sont créées en sélectionnant

aléatoirement une mesure dans chaque état, comme le montre la figure 5.5(b). Les paramètres de chaque HMM $\Lambda_{ij} = (A, B, \pi)$ sont calculés comme suit :

- La matrice de transition $A$ de taille $N_S \times N_S$ est définie comme suit,

$$A = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & \dots & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

- Le modèle d'émission de chaque séquence est calculé en modélisant les mesures de RSSIs de chaque séquence par une distribution multi-dimensionnelle comme discuté dans la section 3.3.1
- Le vecteur $\pi$ est défini par $\pi = [\frac{1}{N_S}, \dots, \frac{1}{N_S}]$.

L'objectif de ce modèle est d'associer une masse ou preuve indiquant que le MN a suivi une certaine trajectoire, qui est une transition d'une zone à une autre. Une fois une séquence $R = \{R_1, \dots, R_\alpha\}$ est détectée, chaque HMM $\Lambda_{ij}$ attribue une probabilité que le MN a suivi la trajectoire correspondante à cet HMM. Les probabilités $P(R|\Lambda_{ij}), i, j \in \{1, \dots, N_Z\}$, sont calculées en utilisant les équations (5.9) à (5.18). Ainsi, la probabilité de transition entre toutes paires de zones est déterminée. Nous définissons le coefficient $p_{ij}, i, j \in \{1, \dots, N_Z\}$, de $Z_i$ à $Z_j$ comme suit,

$$p_{ij} = \begin{cases} P(R|\Lambda_{ij}), & \text{si } \ i \neq j; \\ 1 - \sum_{j=1}^{N_Z} P(R|\Lambda_{ij}), & \text{si } \ i = j. \end{cases} \tag{A.37}$$

Enfin, les probabilités associées par les HMMs sont propagées à l'instant suivant. La confiance attribuée par ce modèle de suivi $\mathbb{T}$ à chaque zone $Z_j$ est calculée comme suit,

$$\mathcal{W}_t(Z_j) = \sum_{i=1}^{N_Z} m_{\mathbb{O},t-1}(Z_i) \times p_{ij}, \tag{A.38}$$

où $m_{\mathbb{O},t-1}(Z_i)$ est la masse associée par le modèle d'observation $\mathbb{O}(\cdot)$ à l'instant $t-1$.

### A.5.5 Expérimentations

Cette sous-section évalue les performances de la méthode de suivi de trajectoires que nous venons de proposer. Nous considérons 10 trajectoires de 50 observations chacune. Dans notre application, nous nous intéressons à la localisation des personnes âgées dépendantes. Pour cela, nous considérons une vitesse maximale $v_{max} = 1m/s$, un nombre d'états $N_S = 10$ et une longueur de séquence $\alpha = 8$. Le choix de ces valeurs est motivé dans les sections 5.6.2 et 5.6.2.1. Le tableau 5.3 montre l'influence de chacun des modèles de mobilité lors de la combinaison avec les modèles d'observation, sur les performances de la méthode de suivi de trajectoires. Nous pouvons voir que la plus faible erreur est obtenue avec le troisième modèle de mobilité pour les deux expérimentations.

Nous étudions l'impact des différents paramètres des modèles de mobilités. En premier lieu, nous varions la vitesse maximale de $0.5m/s$ à $2.5m/s$ et nous analysons les performances des premiers deux modèles de mobilité. Les tableaux 5.4 et 5.5 montrent l'influence de $v_{max}$ sur la précision de l'approche de suivi dans les deux expérimentations. Comme le tableau le montre, la précision de l'approche proposée diminue en augmentant la vitesse maximale. Quant à l'impact de la longueur de la séquence $\alpha$, nous remarquons tout d'abord que la précision de l'approche proposée augmente lors de l'augmentation de la taille de la séquence observée $\alpha$. Cependant, l'augmentation de la taille de la séquence induit une augmentation de la complexité du calcul.

## A.6 Localisation décentralisée

Dans cette section, nous explorons les architectures décentralisées pour la localisation par zonage. Inspirée par la topologie décentralisée, nous proposons l'approche décentralisée suivante. Tout d'abord, nous partitionnons la région cible en différents secteurs, et nous assignons un calculateur à chaque secteur. Chacun des calculateurs estime la zone du MN en appliquant un algorithme de localisation local. Nous prenons la décision ensuite en combinant les décisions de tous les calculateurs. En plus, nous présentons une stratégie pour déterminer le nombre optimal de calculateurs et leurs positions.

### A.6.1 Problématique

L'objectif de la méthode proposée est de déterminer la zone du MN dans une architecture décentralisée, en associant une confiance $\mathcal{C}f_t(\cdot)$ à chacune des zones à l'instant $t$. Supposons que la région cible est partitionnée en $N_C$ secteurs, notés $C_i$, $i \in \{1, ..., N_C\}$. Soit $Z_j, j \in J_i$, l'ensemble de zones constituant le secteur $i$. Les calculateurs qui détectent le signal émis par le MN reçoivent le vecteurs des RSSIs de tous les APs détectés. Les estimations locales sont ainsi combinées afin d'estimer la zone du MN. Dans ce qui suit, $I_{C,t}$ représente l'ensemble des indices des calculateurs à portée de la communication du MN à l'instant $t$.

### A.6.2 Première approche décentralisée

Dans cette approche, nous assignons des secteurs distincts à chaque calculateur, c'est à dire, dans deux secteurs $i, i' \in \{1, \ldots, N_C\}$ avec $i \neq i'$, nous avons $Z_j \neq Z_{j'}$, $\forall j \in J_i, j' \in J_{i'}$. Un exemple de cette architecture est illustré par la figure 6.3(a). Soit $\eta_{i,t}$ la puissance du signal reçu par le calculateur $C_i$ du MN. Chaque $C_i$ exécute l'algorithme de localisation local et assigne une certaine preuve à chacune de zones de son secteur.

Soit $m_{i,t}(Z_j), j \in J_i$, la masse assignée par l'algorithme de localisation local aux zones du secteur $i$ à l'instant $t$. La masse $m_{i,t}(\cdot)$ est normalisée sur toutes les zones de ce secteur,

$$\tilde{m}_{i,t}(Z_j) = \frac{m_{i,t}(Z_j)}{\sum_{q \in J_i} m_{i,t}(Z_q)}. \tag{A.39}$$

La confiance associée à chacune des zones du secteur $i$ est ainsi calculée,

$$\mathcal{C}f_t(Z_j) = \frac{\frac{1}{w_{i,t}} \times \tilde{m}_{i,t}(Z_j)}{\sum_{p \in I_{C,t}} \sum_{q \in J_p} \frac{1}{w_{i,t}} \times \tilde{m}_{p,t}(Z_q)}, \tag{A.40}$$

où $w_{i,t} = \frac{\eta_{i,t}}{\sum_{x \in I_{C,t}} \eta_{x,t}}$ est le poids de chaque calculateur $C_i$.

### A.6.3 Deuxième approche décentralisée

Dans cette approche, nous considérons des secteurs distincts comme dans l'approche précédente. Par contre, au lieu de prendre une décision par chaque

calculateur, nous prenons uniquement les estimations des calculateurs. Les estimations des calculateurs sont ainsi combinées pour arriver à une décision finale de la zone du MN. La confiance associée à chaque zone est obtenue par

$$\mathcal{C}f_t(Z_j) = \frac{\frac{1}{w_{i,t}} \times m_{i,t}(Z_j)}{\sum_{p \in I_{C,t}} \sum_{q \in J_p} \frac{1}{w_{p,t}} \times m_{i,t}(Z_j)}. \tag{A.41}$$

### A.6.4 Troisième approche décentralisée

Dans cette approche, nous associons des secteurs de chevauchement aux calculateurs, c'est à dire, dans deux secteurs $i, i' \in \{1, \ldots, N_C\}, i \neq i', \exists j \in J_i, j' \in J_{i'}$, tel que $j = j'$. Un exemple de cette architecture est illustré dans la figure 6.3(b). La masse obtenue par combiner les décisions de tous les calculateurs est calculaée par

$$m_{F,t}(Z_j) = \prod_{i \in I_{C,t}} \tilde{m}_{i,t}(Z_j), \qquad \forall j \in \bigcap_{i \in I_{C,t}} J_i. \tag{A.42}$$

La quantité $m_{F,t}(Z_j)$ est nulle pour toutes les zones qui ne sont pas à l'intersection des secteurs correspondants aux calculateurs détectés; c'est à dire, $m_{F,t}(Z_j) = 0 \ \forall Z_j$ telle que $j \notin \bigcap_{i \in I_{C,t}} J_i$. La confiance associée à chacune des zones est calculée en normalisant toutes les preuves combinées,

$$\mathcal{C}f_t(Z_j) = \frac{m_{F,t}(Z_j)}{\sum_{p \in I_{C,t}} \sum_{q \in J_p} m_{F,t}(Z_q)}. \tag{A.43}$$

### A.6.5 Placement des calculateurs

Les calculateurs sont des dispositifs intelligents capables d'échanger des informations avec l'environnement. Puisqu'il n'existe aucune connaissance préalable du nombre ou des positions des calculateurs, il est important de développer une stratégie pour les placer. Nous proposons une solution basée sur le nombre optimal de calculateurs requis pour couvrir toute la région cible. Supposons que la région cible est un rectangle de dimensions $L \times H$. D'autres formes sont traitées en considérant le plus grand rectangle et en enlevant les parties qui ne sont pas dans la région originale. Soit $\delta$ la portée de détection du MN. Si sa portée n'est pas circulaire, nous considérons le plus grand cercle inscrit dans la portée. L'objectif est de minimiser le nombre de calculateurs $N_C$ requis pour

couvrir toute la région et de déterminer leurs positions. Pour ce faire, nous proposons la technique d'empilement de cercles. Nous citons l'empilement compact de cercles dans cette sous-section. L'autre stratégie des *circles covering the region* est similaire. La seule différence est que la dernière ne laisse aucun espace non couvert, nécessitant un nombre plus élevé de calculateurs. Les détails de ces deux stratégies se trouvent dans la section 6.4.

L'empilement compact de cercles est un arrangement de cercles dans une région où aucun chevauchement est permis [He & Dosh, 2017]. Hales [2000] démontrent que le pavage hexagonal est le plus optimal de tous les pavages, comme le montre la figure 6.5(a). Le nombre $N_C$ de cercles requis est calculé par

$$\frac{\pi}{2\sqrt{3}} = \frac{N_C \times \pi\delta^2}{L \times H} \implies N_C = \text{ceil}\left(\frac{L \times H}{2\sqrt{3} \times \delta^2}\right), \tag{A.44}$$

où ceil$(\cdot)$ est la fonction qui arrondi le chiffre à la partie entière par excès. Pour déployer les cercles, $B$ paquets sont placés horizontalement, chacun comprenant $A$ cercles. Dans un système de coordonnées cartésiennes d'origine au coin inférieur gauche du rectangle, la position $[x_a, y_b]$ du centre du cercle $a$ du paquet $b$ est déterminée par

$$[x_a, y_b] = \begin{cases} \left[\sqrt{3}(a-1)\delta, (2b-1)\delta\right], \text{si } a \text{ est impair;} \\ \left[\sqrt{3}(a-1)\delta, 2(b-1)\delta\right], \text{si } a \text{ est pair.} \end{cases} \tag{A.45}$$

### A.6.6 Algorithme de localisation local

Chaque calculateur exécute un algorithme de localisation local pour déterminer la zone du MN dans son secteur correspondant. L'objectif est de trouver une fonction $\mathcal{I}_i(\cdot) : \mathbb{R}^{N_{AP}} \to [0,1]^{|J_i|}$ pour tout calculateur $C_i$ tel que $\mathcal{I}_i(\boldsymbol{\rho}_t) = \left(m_{i,t}(Z_{j_1}), \ldots, m_{i,t}(Z_{j_{|J_i|}})\right)$, où $J_i = \{j_1, \ldots, j_{|J_i|}\}$ et $m_{i,t}(Z_j)$, est la masse attribuée à $Z_j$ à l'instant $t$ pour une nouvelle observation $\boldsymbol{\rho}_t \in \mathbb{R}^{N_{AP}}$.

Une première solution est d'utiliser les modèles d'observation. Dans ce cas, la fonction $\mathcal{I}_i(\cdot)$ désigne le modèle d'observation $\mathbb{O}(\cdot)$, le nombre de zones $|J_i|$ est $N_Z$ et les masses $m_{i,t}(Z_j), j \in J_i$ attribuées aux zones du secteur $C_i$ seront $m_{\mathbb{O},t}(Z_j), j \in \{1, \ldots, N_Z\}$, associées par $\mathbb{O}$ aux zones de son secteur.

Une deuxième solution est d'utiliser les modèles d'observation et de mobilité, comme expliqué dans la section précédente. Dans ce cas, la fonction $\mathfrak{I}_i(\cdot)$ désigne le modèle de suivi de trajectoire $\mathbb{T}(\cdot)$, le nombre de zones $|J_i|$ est $N_Z$ et les masses $m_{i,t}(Z_j)$ attribuées aux zones du secteur $i$ seront les confiances $\mathcal{W}_t(Z_j), j \in \{1, \ldots, N_Z\}$.

### A.6.7 Expérimentions

Dans cette sous-section, nous utilisons une portée de communication de $\delta = 6m$. Le nombre optimal de calculateurs obtenu par l'empilement compact de cercles par *circle covering the plane* est de $N_C = 4.4 \approx 5$ et $N_C = 5.8 \approx 6$ respectivement.. Puisqu'il n'y a pas une grande différence dans le nombre de calculateurs, nous prenons 6 calculateurs pour garantir une couverture complète de la région cible. Le Tableau 6.1 compare la précision des approches décentralisées à celle du BOM et du EOM décrits dans les sections précédentes. Nous pouvons voir que la troisième approche atteint une meilleure performance que toutes les autres. Nous analyserons ensuite la robustesse des approches proposées. Le tableau 6.2 montre l'impact du nombre de calculateurs en panne sur les performances des méthodes. Les approches centralisées, BOM et EOM, n'arrivent pas à effectuer la localisation si le calculateur tombe en panne. Par contre, les approches proposées peuvent toujours localiser les MNs, bien que la performance se dégrade.

Ce paragraphe compare l'approche proposée à deux techniques de localisation bien connues : la localisation basée sur les connectivités [Shang *et al.*, 2004] et la localisation utilisant l'algorithme des k-plus proches voisins pondérés [Koyuncu & Yang, 2011]. Nous comparons aussi l'approche proposée aux techniques conventionnelles de classification multi-classes, telles que les machines à vecteur support et les réseaux de neurones. Le tableau 6.3 compare les performances des différentes méthodes. Nous pouvons voir que la plus faible erreur est obtenue avec la troisième approche décentralisée. L'avantage des approches proposées est la simplicité de l'algorithme de localisation local, ce qui réduit la complexité de l'approche et ainsi le temps d'exécution.

## A.7 Conclusion et perspectives

Cette thèse a abordé le problème de localisation des capteurs dans les réseaux de capteurs sans fil. Tout d'abord, nous avons proposé une méthode de localisation par zonage dans le cadre de la théorie des fonctions de croyance. Cette méthode a été développée au travers du clustering hiérarchique, pour effectuer la localisation dans le but de couvrir des surfaces plus grandes avec un nombre de zones plus élevé. Ensuite, nous avons proposée une technique de sélection de paramètres, qui sont les bornes WiFi à privilégier. Nous revisitons ensuite ce travail pour l'adapter au suivi de trajectoire. Cette méthode permet d'améliorer l'estimation de la zone en tenant compte du mouvement du capteur. Finalement, une version décentralisée a été présentée, pour accroitre la robustesse et l'applicabilité, et réduire la complexité de la méthode proposée. Cette section résume les contributions majeures de cette thèse et propose quelques perspectives futures.

### A.7.1 Contributions principales

Dans la section A.3, nous avons introduit une méthode de localisation par zonage dans le cadre de la théorie des fonctions de croyance. Un modèle d'observation a été construit pour estimer la zone du capteur mobile en utilisant les mesures de RSSIs reçues des APs WiFi. Plusieurs configurations ont été explorées pour la définition du modèle. Différents types de modélisation des observations, affaiblissement des sources et des règles de combinaison de l'information ont été étudiées. Les résultats expérimentaux ont montré que la méthode proposée surpasse d'autres technique de classification en termes de précision.

Pour améliorer la performance de la méthode pour un nombre élevé de zones, un modèle d'observation élaboré a été proposé dans la section A.4. Une hiérarchie de deux niveaux a été construite par optimiser les distances inter- et intra-clusters. Une technique de sélection de APs a été également développée pour choisir le meilleur ensemble de APs en termes de précision et redondance. Les résultats obtenus ont montré que ce modèle atteint une précision plus grande, mais au prix d'une augmentation significative de la complexité du calcul.

Dans la section A.5, nous avons proposé une technique de suivi de trajectoires qui profite de la mobilité des capteurs dans le milieu couvert. Plusieurs modèles de mobilité ont été proposés dans le cadre de la théorie des fonctions de croyance et combinés avec les modèles d'observation déjà décrits. Les résultats obtenus ont montré l'efficacité de ces modèles pour améliorer les estimations des zones.

Finalement, dans la section A.6, nous avons présenté un algorithme décentralisé pour la localisation par zonage. Plusieurs approches décentralisées ont été étudiées. Une stratégie optimale pour placer les calculateurs a été décrite en utilisant l'empilement compact de cercles. Les résultats obtenus ont montré la robustesse de l'approche, sa faible complexité par rapport aux approches centralisées et son efficacité en termes de précision.

### A.7.2 Perspectives

Dans le cadre de travaux futures, nous tenons à étudier les aspects suivants concernant l'amélioration des méthodes proposées.

- *Amelioration des modèles d'observation*
  Dans la définition du modèle d'observation avancé, une hiérarchie de deux niveaux a été proposée. Dans les cas où la région cible est très large, comme les immeubles à plusieurs étages, plus que deux niveaux sont requis. Les travaux futurs comprendront une sélection systématique du nombre de niveaux optimal.

- *Amélioration de l'approche de fingerprinting*
  Les approches de fingerprinting par RSSIs se basent sur les puissances des signaux reçus pour déterminer la position du capteur mobile. Un facteur important dans ces algorithmes est les positions des APs. Jusqu'à maintenant, les APs sont installés selon le critère de couverture. Nous pouvons développer un algorithme d'optimisation pour maximiser la couverture et minimiser l'erreur résultant des positions des APs. Une autre solution consiste à ajouter des APs d'une manière efficace pour la localisation.

- *Amélioration de l'algorithme de sélection des APs*
  Dans l'algorithme de sélection des APs, un paramètre de compromis a été défini. Le rôle de ce paramètre est d'attribuer un poids à chacune des fonctions objectives : la maximisation de la capacité discriminatoire et la minimisation de la redondance. Toutefois, le paramètre a été défini par l'utilisateur manuellement. Nous pouvons chercher un algorithme qui détermine le paramètre de compromis de manière automatique.

- *Mise à jour de la base de données*
  Les RSSIs recueillis dans chaque zone varient avec le temps. C'est à cause des changements de la puissance initiale émise, les conditions atmosphériques, etc. Nous pouvons développer un algorithme qui sélectionne de nouvelles mesures, ayant une grande confiance, et qui les ajoute à la base de données.

- *Amélioration des modèles de mobilité*
  Les modèles de mobilité proposés supposent une vitesse maximale des capteurs. Lorsque cette vitesse augmente, des estimations moins précises sont obtenues si on a besoin d'un algorithme de localisation rapide, qui est naturellement le cas. Nous pouvons ainsi équiper les capteurs par des unités de mesure inertielles qui peuvent collecter des informations plus précises sur la mobilité des capteurs en temps réel afin d'améliorer la localisation.

# Bibliography


Ahn, H.S. & Yu, W. (2009). Indoor localization techniques based on wireless sensor networks. In *Mobile Robots-State of the Art in Land, Sea, Air, and Collaborative Missions*, InTech. 51

Akyildiz, I.F. & Stuntebeck, E.P. (2006). Wireless underground sensor networks: Research challenges. *Ad Hoc Networks*, **4**, 669–686. 3

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. & Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, **38**, 393–422. 2, 161

Akyildiz, I.F., Melodia, T. & Chowdury, K.R. (2007). Wireless multimedia sensor networks: A survey. *IEEE Wireless Communications*, **14**, 32–39. 4

Al-Khedher, M.A. (2011). Hybrid GPS-GSM localization of automobile tracking system. *International Journal of Computer Science & Information Technology*, **3**, 75–85. 50

Alemdar, H. & Ersoy, C. (2010). Wireless sensor networks for healthcare: A survey. *Computer Networks*, **54**, 2688–2710. 7

Amundson, I. & Koutsoukos, X.D. (2009). A survey on localization for mobile wireless sensor networks. In *Mobile Entity Localization and Tracking in GPS-less Environnments*, 235–254, Springer. 4

Anderson, E.C., Williamson, E.G. & Thompson, E.A. (2000). Monte carlo evaluation of the likelihood for Ne from temporally spaced samples. *Genetics*, **156**, 2109–2118. 91

## BIBLIOGRAPHY


Aruna, M., Ganesan, R. & Renold, A. (2015). Optimized path planning mechanism for localization in wireless sensor networks. In *International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, 171–177. 51

Babbar, R., Partalas, I., Gaussier, E. & Amini, M.R. (2013). On flat versus hierarchical classification in large-scale taxonomies. In *Advances in Neural Information Processing Systems*, 1824–1832. 87, 175

Bahr, A., Leonard, J.J. & Fallon, M.F. (2009). Cooperative localization for autonomous underwater vehicles. *The International Journal of Robotics Research*, **28**, 714–728. 14

Bánhelyi, B., Palatinus, E. & Lévai, B.L. (2015). Optimal circle covering problems and their applications. *Central European Journal of Operations Research*, **23**, 815–832. 142, 144

Bar-Shalom, Y., Li, X.R. & Kirubarajan, T. (2004). *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons. 20, 111

Barron, A.R. & Sheu, C.H. (1991). Approximation of density functions by sequences of exponential families. *The Annals of Statistics*, **19**, 1347–1369. 90

Basseville, M. (2013). Divergence measures for statistical data processing, an annotated bibliography. *Signal Processing*, **93**, 621–633. 88

Bekkelien, A., Deriaz, M. & Marchand-Maillet, S. (2012). Bluetooth indoor positioning. *Master's Thesis, University of Geneva*. 51

Bohannon, R.W. (1997). Comfortable and maximum walking speed of adults aged 20—79 years: reference values and determinants. *Age and Ageing*, **26**, 15–19. 127

Breiman, L. (2001). Random forests. *Machine Learning*, **45**, 5–32. 18, 102, 164

Cha, S.H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *City*, **1**, 1–9. 82

Chen, L., Carpenter, G., Greenberg, S., Frolik, J. & Wang, X.S. (2011). An implementation of decentralized consensus building in sensor networks. *IEEE Sensors Journal*, **11**, 667–675. 5, 134

Chen, Y., Crawford, M.M. & Ghosh, J. (2004). Integrating support vector machines in a hierarchical output space decomposition framework. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 949–952, IEEE. 18, 102

Chriki, A., Touati, H. & Snoussi, H. (2017). SVM-based indoor localization in wireless sensor networks. In *International Wireless Communications and Mobile Computing Conference (IWCMC)*, 1144–1149, IEEE. 16

Corke, P., Wark, T., Jurdak, R., Hu, W., Valencia, P. & Moore, D. (2010). Environmental wireless sensor networks. *Proceedings of the IEEE*, **98**, 1903–1917. 7

Cota-Ruiz, J., Rivas-Perea, P., Sifuentes, E. & Gonzalez-Landaeta, R. (2016). A recursive shortest path routing algorithm with application for wireless sensor network localization. *IEEE Sensors Journal*, **16**, 4631–4637. 5, 135, 162

Dai, H., Ying, W.h. & Xu, J. (2016). Multi-layer neural network for received signal strength-based indoor localisation. *IET Communications*, **10**, 717–723. 150

Defays, D. (1977). An efficient algorithm for a complete link method. *The Computer Journal*, **20**, 364–366. 83

Delignette-Muller, M.L. & Dutang, C. (2015). fitdistrplus: An R package for fitting distributions. *Journal of Statistical Software*, **64**, 1–34. 58

Dempster, A.P. (1967). Upper and lower probabilities induced by a multivalued mapping. *The Annals of Mathematical Statistics*, **38**, 325–339. 33, 39, 167

Di Francesco, M., Das, S.K. & Anastasi, G. (2011). Data collection in wireless sensor networks with mobile elements: A survey. *ACM Transactions on Sensor Networks (TOSN)*, **8**, 7–41. 4

Disha, A. (2013). A comparative analysis on indoor positioning techniques and systems. *International Journal of Engineering Research and Applications*, **3**, 1790–1796. 51

Dubes, R.C. (1987). How many clusters are best?-an experiment. *Pattern Recognition*, **20**, 645–663. 83

Dubois, D. & Prade, H. (1992). On the combination of evidence in various mathematical frameworks. In *Reliability Data Collection and Analysis*, 213–241, Springer. 39, 168

Dumais, S. & Chen, H. (2000). Hierarchical classification of Web content. In *ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 256–263, ACM. 87

Durrant-Whyte, H. & Henderson, T.C. (2008). Multisensor data fusion. In *Springer Handbook of Robotics*, 585–610, Springer. 31, 166

Eckert-Gallup, A. & Martin, N. (2016). Kernel density estimation (KDE) with adaptive bandwidth selection for environmental contours of extreme sea states. In *OCEANS MTS/IEEE Monterey*, 1–5, IEEE. 61

Eddy, S.R. (1996). Hidden markov models. *Current Opinion in Structural Biology*, **6**, 361–365. 119

Elgammal, A., Duraiswami, R., Harwood, D. & Davis, L.S. (2002). Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, **90**, 1151–1163. 59

Elouedi, Z., Lefèvre, E. & Mercier, D. (2010). Discountings of a belief function using a confusion matrix. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, 287–294, IEEE. 45

Facchinetti, S. (2009). A procedure to find exact critical values of Kolmogorov-Smirnov test. *Statistica Applicata*, 337–359. 58

Fox, V., Hightower, J., Liao, L., Schulz, D. & Borriello, G. (2003). Bayesian filtering for location estimation. *IEEE Pervasive Computing*, **2**, 24–33. 32

Gara, F., Saad, L.B., Ayed, R.B. & Tourancheau, B. (2015). RPL protocol adapted for healthcare and medical applications. In *International Wireless Communications and Mobile Computing Conference (IWCMC)*, 690–695, IEEE. 8

Gezici, S., Tian, Z., Giannakis, G.B., Kobayashi, H., Molisch, A.F., Poor, H.V. & Sahinoglu, Z. (2005). Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. *IEEE Signal Processing Magazine*, **22**, 70–84. 13, 163

Gheyas, I.A. & Smith, L.S. (2010). Feature subset selection in large dimensionality domains. *Pattern Recognition*, **43**, 5–13. 85

Górak, R. & Luckner, M. (2016). Modified random forest algorithm for WiFi indoor localization system. In *International Conference on Computational Collective Intelligence*, 147–157, Springer. 102

Graham, J.E., Fisher, S.R., Bergés, I.M., Kuo, Y.F. & Ostir, G.V. (2010). Walking speed threshold for classifying walking independence in hospitalized older adults. *Physical Therapy*, **90**, 1591–1597. 127

Guyon, I. & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, **3**, 1157–1182. 94

Habbema, J. & Hermans, J. (1977). Selection of variables in discriminant analysis by F-statistic and error rate. *Technometrics*, **19**, 487–493. 62

Hales, T.C. (2000). Cannonballs and honeycombs. *Notices-American Mathematical Society*, **47**, 440–449. 144, 188

Hall, D.L. & Llinas, J. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE*, **85**, 6–23. 28, 165

Hall, P. (1982). Cross-validation in density estimation. *Biometrika*, **69**, 383–390. 62

Hancke, G.P. (2012). Industrial wireless sensor networks: A selection of challenging applications. In *European Conference on Antennas and Propagation (EUCAP)*, 64–68, IEEE. 3

Harmouche, J., Delpha, C. & Diallo, D. (2016). Incipient fault amplitude estimation using KL divergence with a probabilistic approach. *Signal Processing*, **120**, 1–7. 91

Hauschildt, D. & Kirchhof, N. (2010). Advances in thermal infrared localization: Challenges and solutions. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 1–8, IEEE. 11

He, K. & Dosh, M. (2017). Packing unequal circles into a square container by partitioning narrow action spaces and circle items. *Equilibrium*, **8**, 18–41. 142, 143, 188

Hégarat-Mascle, S.L., Richard, D. & Ottlé, C. (2003). Multi-scale data fusion using Dempster-Shafer evidence theory. *Integrated Computer-Aided Engineering*, **10**, 9–22. 33

Hershey, J.R. & Olsen, P.A. (2007). Approximating the Kullback-Leibler divergence between Gaussian mixture models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IV–317, IEEE. 90

Honeine, P., Noumir, Z. & Richard, C. (2013). Multiclass classification machines with the complexity of a single binary classifier. *Signal Processing*, **93**, 1013–1026. 17, 163

Hong, F., Zhang, Y., Zhang, Z., Wei, M., Feng, Y. & Guo, Z. (2014). WaP: Indoor localization and tracking using WiFi-assisted particle filter. In *IEEE Conference on Local Computer Networks (LCN)*, 210–217, IEEE. 21, 112

Huang, S., Yang, Y., Yang, D., Huangfu, L. & Zhang, X. (2015). Class specific sparse representation for classification. *Signal Processing*, **116**, 38–42. 104, 106

Hussain, M.A. & kyung Sup, K. (2009). WSN research activities for military application. In *International Conference on Advanced Communication Technology (ICACT)*, 271–274, IEEE. 6

Hussain, S., Schaffner, S. & Moseychuck, D. (2009). Applications of wireless sensor networks and RFID in a smart home environment. In *Annual Communication Networks and Services Research Conference (CNSR)*, 153–157, IEEE. 7

Iliev, N. & Paprotny, I. (2015). Review and comparison of spatial localization methods for low-power wireless sensor networks. *IEEE Sensors Journal*, **15**, 5971–5987. 135

Islam, M.A., Alizadeh, B.Z., van den Heuvel, E.R., Bruggeman, R., Cahn, W., de Haan, L., Kahn, R.S., Meijer, C., Myin-Germeys, I. & van Os, J. (2015). A comparison of indices for identifying the number of clusters in hierarchical clustering: A study on cognition in schizophrenia patients. *Communications in Statistics: Case Studies, Data Analysis and Applications*, **1**, 98–113. 92

Jaffray, J.Y. (1989). Linear utility theory for belief functions. *Operations Research Letters*, **8**, 107–112. 45

Jahromi, O.S. (2007). *Multirate statistical signal processing*. Springer Science & Business Media. 94, 95, 177

Jain, A.K., Murty, M.N. & Flynn, P.J. (1999). Data clustering: a review. *ACM Computing Surveys*, **31**, 264–323. 82

John, G.H., Kohavi, R. & Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *Machine Learning Proceedings 1994*, 121–129, Elsevier. 85

Johnson, S.C. (1967). Hierarchical clustering schemes. *Psychometrika*, **32**, 241–254. 83

Kaufman, L. & Rousseeuw, P.J. (2009). *Finding groups in data: an introduction to cluster analysis*, vol. 344. John Wiley & Sons. 82

Kendrick, J.D., Maybeck, P. & Reid, J. (1981). Estimation of aircraft target motion using orientation measurements. *IEEE Transactions on Aerospace and Electronic Systems*, **2**, 254–260. 20, 111

Kennes, R. & Smets, P. (1990). Fast algorithms for Dempster-Shafer theory. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, 14–23, Springer. 46, 169

Kershner, R. (1939). The number of circles covering a set. *American Journal of Mathematics*, **61**, 665–671. 144

Khedo, K.K., Sathan, D., Elaheebocus, R., Subramanian, R.K. & Rughooputh, S. (2010). Overlapping zone partitioning localisation technique for RFID. *International Journal of Ubiquitous Computing*, **1**, 20–32. 16

Khorsandi, R. & Abdel-Mottaleb, M. (2015). Classification based on weighted sparse representation using smoothed $L^0$ norm with non-negative coefficients. In *IEEE International Conference on Image Processing (ICIP)*, 3131–3135, IEEE. 104, 106

Kim, B.H., Roh, D.K., Lee, J.M., Lee, M.H., Son, K., Lee, M., Choi, J. & Han, S. (2001). Localization of a mobile robot using images of a moving target. In *IEEE International Conference on Robotics and Automation (ICRA)*, 253–258, IEEE. 10

Ko, J., Lu, C., Srivastava, M.B., Stankovic, J.A., Terzis, A. & Welsh, M. (2010). Wireless sensor networks for healthcare. *Proceedings of the IEEE*, **98**, 1947–1960. 7

Kohavi, R. & John, G.H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, **97**, 273–324. 85

Kotsiantis, S.B. (2013). Decision trees: a recent overview. *Artificial Intelligence Review*, **39**, 261–283. 17, 164

Kotsiantis, S.B., Zaharakis, I. & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerging Artificial Intelligence Applications in Computer Engineering*, **160**, 3–24. 17

Koyuncu, H. & Yang, S.H. (2011). A 2D positioning system using WSNs in indoor environment. *International Journal of Electrical and Computer Sciences*, **11**, 70–77. 149, 189

Krzanowski, W.J. & Lai, Y. (1988). A criterion for determining the number of groups in a data set using sum-of-squares clustering. *Biometrics*, 23–34. 92, 176

Kumar, K. & Zhou, H. (1984). A current statistical model and adaptive algorithm for estimating maneuvering targets. *Journal of Guidance, Control, and Dynamics*, **7**, 596–602. 20, 111

Kurdej, M. & Cherfaoui, V. (2013). Conservative, proportional and optimistic contextual discounting in the belief functions theory. In *International Conference on Information Fusion (FUSION)*, 2012–2018, IEEE. 67, 172

Lee, D.L. & Chen, Q. (2007). A model-based WiFi localization method. In *International conference on Scalable information systems (ICSIS)*, 40–47, ICST. 16

Lee, K.C., Ho, J. & Kriegman, D.J. (2005). Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, 684–698. 104

Lee, S.H., Lee, S., Song, H. & Lee, H.S. (2009). Wireless sensor network design for tactical military applications: Remote large-scale environments. In *IEEE Military communications conference (MILCOM)*, 1–7, IEEE. 6

Lillis, D., Toolan, F., Collier, R. & Dunnion, J. (2006). Probfuse: a probabilistic approach to data fusion. In *ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 139–146, ACM. 31

Lin, M., Wu, Y. & Wassell, I. (2008). Wireless sensor network: Water distribution monitoring system. In *IEEE Radio and Wireless Symposium (RWS)*, 775–778, IEEE. 7

Liu, D., Li, T. & Liang, D. (2014). Incorporating logistic regression to decision-theoretic rough sets for classifications. *International Journal of Approximate Reasoning*, **55**, 197–210. 17, 163

Liu, H., Darabi, H., Banerjee, P. & Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, **37**, 1067–1080. 51

Liu, L., Shao, L. & Rockett, P. (2013a). Human action recognition based on boosted feature selection and naive Bayes nearest-neighbor classification. *Signal Processing*, **93**, 1521–1530. 76, 174

Liu, Y.C., Ou, Y.K., Lin, S.N. & Fang, C.W. (2013b). A study of the indoor walking navigation system for patients with early-stage Alzheimer's disease. In *International Conference on Computer, Networks and Communication Engineering (ICCNCE)*, Atlantis Press. 16, 50

Liu, Z.g., Dezert, J., Pan, Q. & Mercier, G. (2011). Combination of sources of evidence with different discounting factors based on a new dissimilarity measure. *Decision Support Systems*, **52**, 133–141. 45

Lv, X., Mourad-Chehade, F. & Snoussi, H. (2015). Decentralized localization using radio-fingerprints and accelerometer in WSNs. *IEEE Transactions on Aerospace and Electronic Systems*, **51**, 242–257. 15

Mahfouz, S., Mourad-Chehade, F., Honeine, P., Farah, J. & Snoussi, H. (2013). Decentralized localization using fingerprinting and kernel methods

inwireless sensor networks. In *European Signal Processing Conference (EUSIPCO)*, 1–5, IEEE. 5

Mahfouz, S., Mourad-Chehade, F., Honeine, P., Farah, J. & Snoussi, H. (2014). Target tracking using machine learning and Kalman filter in wireless sensor networks. *IEEE Sensors Journal*, **14**, 3715–3725. 20, 32, 111, 164

Mahfouz, S., Mourad-Chehade, F., Honeine, P., Farah, J. & Snoussi, H. (2015). Kernel-based machine learning using radio-fingerprints for localization in WSNs. *IEEE Transactions on Aerospace and Electronic Systems*, **51**, 1324–1336. 15, 163

Mahler, R.P. (2004). "Statistics 101" for multisensor, multitarget data fusion. *IEEE Aerospace and Electronic Systems Magazine*, **19**, 53–64. 30

Mamun, Q. (2012). A qualitative comparison of different logical topologies for wireless sensor networks. *Sensors*, **12**, 14887–14913. 5, 135, 162

Manning, C.D., Raghavan, P., Schütze, H. *et al.* (2008). *Introduction to information retrieval*, vol. 1. Cambridge university press Cambridge. 87, 175

Martinez, A.M. (1998). The AR face database. *CVC Technical Report*, **24**. 104

Massey Jr, F.J. (1951). The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American statistical Association*, **46**, 68–78. 58

Mauša, G., Grbac, T.G. & Bašić, B.D. (2012). Multivariate logistic regression prediction of fault-proneness in software modules. In *International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 698–703, IEEE. 76, 174

Mercier, D., Quost, B. & Denœux, T. (2008). Refined modeling of sensor reliability in the belief function framework using contextual discounting. *Information Fusion*, **9**, 246–258. 42, 168

Mercier, D., Lefèvre, É. & Delmotte, F. (2012). Belief functions contextual discounting and canonical decompositions. *International Journal of Approximate Reasoning*, **53**, 146–158. 44, 65, 168

Misra, S., Reisslein, M. & Xue, G. (2008). A survey of multimedia streaming in wireless sensor networks. *IEEE Communications Surveys & Tutorials*, **10**, 18–39. 4

Moghtadaiee, V., Dempster, A.G. & Lim, S. (2011). Indoor localization using FM radio signals: A fingerprinting approach. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 1–7, IEEE. 59

Montero-Odasso, M., Schapira, M., Varela, C., Pitteri, C., Soriano, E.R., Kaplan, R., Camera, L.A. & Mayorga, L. (2004). Gait velocity in senior people an easy test for detecting mobility impairment in community elderly. *Journal of Nutrition Health and Aging*, **8**, 340–343. 127

Moreno, L., Armingol, J.M., Garrido, S., De La Escalera, A. & Salichs, M.A. (2002). A genetic algorithm for mobile robot localization using ultrasonic sensors. *Journal of Intelligent and Robotic Systems*, **34**, 135–154. 11

Mourad, F., Snoussi, H., Abdallah, F. & Richard, C. (2009). Anchor-based localization via interval analysis for mobile ad-hoc sensor networks. *IEEE Transactions on Signal Processing*, **57**, 3226–3239. 51

Mukhopadhyay, S.C. (2015). Wearable sensors for human activity monitoring: A review. *IEEE Sensors Journal*, **15**, 1321–1330. 8

Murad, M., Sheikh, A.A., Manzoor, M.A., Felemban, E. & Qaisar, S. (2015). A survey on current underwater acoustic sensor network applications. *International Journal of Computer Theory and Engineering*, **7**, 51. 4

Murtagh, F. & Contreras, P. (2012). Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, **2**, 86–97. 83

Narayanan, V., Arora, I. & Bhatia, A. (2013). Fast and accurate sentiment classification using an enhanced naive Bayes model. In *International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*, 194–201, Springer. 17, 163

Narendra, P.M. & Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, **9**, 917–922. 85

Neuzil, J., Kreibich, O. & Smid, R. (2014). A distributed fault detection system based on IWSN for machine condition monitoring. *IEEE Transactions on Industrial Informatics*, **10**, 1118–1123. 7

Ngatchou, P., Zarei, A. & El-Sharkawi, A. (2005). Pareto multi objective optimization. In *International Conference on Intelligent systems Application to power systems (ISAP)*, 84–91, IEEE. 95

Nielsen, F. & Nock, R. (2017). Generalizing skew Jensen divergences and Bregman divergences with comparative convexity. *IEEE Signal Processing Letters*, **24**, 1123–1127. 88, 176

Nikolov, M. & Haas, Z.J. (2018). Encoded sensing for energy efficient wireless sensor networks. *IEEE Sensors Journal*, **18**, 875–889. 134

Noury, N., Hervé, T., Rialle, V., Virone, G., Mercier, E., Morey, G., Moro, A. & Porcheron, T. (2000). Monitoring behavior in home using a smart fall sensor and position sensors. In *International Conference On Microtechnologies in Medicine and Biology (MMB)*, 607–610, IEEE. 8

Okello, N., Fletcher, F., Musicki, D. & Ristic, B. (2011). Comparison of recursive algorithms for emitter localisation using TDOA measurements from a pair of UAVs. *IEEE Transactions on Aerospace and Electronic Systems*, **47**, 1723–1732. 13

Oliveira, L.M. & Rodrigues, J.J. (2011). Wireless sensor networks: A survey on environmental monitoring. *Journal of Communication*, **6**, 143–151. 7

Oshin, T., Poslad, S. & Ma, A. (2012). Improving the energy-efficiency of GPS based location sensing smartphone applications. In *IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 1698–1705. 9, 50, 162

Oudelha, M. & Ainon, R.N. (2010). HMM parameters estimation using hybrid baum-welch genetic algorithm. In *International Symposium in Information Technology (ITSIM)*, 542–545, IEEE. 121

Pak, J.M., Ahn, C.K., Shmaliy, Y. & Lim, M.T. (2015). Improving reliability of particle filter-based localization in wireless sensor networks via hybrid particle/FIR filtering. *IEEE Transactions on Industrial Informatics*, **11**, 1089–1098. 51

Patwari, N., Ash, J.N., Kyperountas, S., Hero, A.O., Moses, R.L. & Correal, N.S. (2005). Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, **22**, 54–69. 14, 163

Pawlak, Z. (2012). *Rough sets: Theoretical aspects of reasoning about data*, vol. 9. Springer Science & Business Media. 31

Pérez-Cruz, F. (2008). Kullback-Leibler divergence estimation of continuous distributions. In *IEEE International Symposium on Information Theory(ISIT)*, 1666–1670, IEEE. 90

Polkowski, L. (2013). *Rough sets in knowledge discovery 2: applications, case studies and software systems*, vol. 19. Physica. 17

Qian, Y., Zhang, H., Sang, Y. & Liang, J. (2014). Multigranulation decision-theoretic rough sets. *International Journal of Approximate Reasoning*, **55**, 225–237. 33

Rajive, J. *et al.* (1999). *Multisensor fusion: a minimal representation framework*, vol. 11. World Scientific. 30

Ravi, N., Shankar, P., Frankel, A., Elgammal, A. & Iftode, L. (2006). Indoor localization using camera phones. In *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, 49–49, IEEE. 10

Robles, R.J. & Kim, T.h. (2010). Applications, systems and methods in smart home technology: A review. *International Journal of Advanced Science And Technology*, **15**, 37–48. 7

Rojas, R. (2013). *Neural networks: a systematic introduction*. Springer Science & Business Media. 17, 163

Rokach, L. & Maimon, O. (2005). Clustering methods. In *Data Mining and Knowledge Discovery Handbook*, 321–352, Springer. 92

Rong, P. & Sichitiu, M.L. (2006). Angle of arrival localization for wireless sensor networks. In *IEEE Communications Society on Sensor and Ad Hoc Communications and Networks (SECON)*, 374–382, IEEE. 13

Ruiz, L.B., Nogueira, J.M. & Loureiro, A.A. (2003). Manna: A management architecture for wireless sensor networks. *IEEE communications Magazine*, **41**, 116–125. 29

Samaria, F.S. & Harter, A.C. (1994). Parameterisation of a stochastic model for human face identification. In *IEEE Workshop on Applications of Computer Vision (WACV)*, 138–142, IEEE. 104

Sammut, C. & Webb, G.I. (2011). *Encyclopedia of machine learning*. Springer Science & Business Media. 17

Sánchez-Rodríguez, D., Hernández-Morera, P., Quinteiro, J.M. & Alonso-González, I. (2015). A low complexity system based on multiple weighted decision trees for indoor localization. *Sensors*, **15**, 14809–14829. 52

Sangthong, J., Dokpikul, P. & Promwong, S. (2012). Indoor positioning based on IEEE 802.15. 4a standard using trilateration technique and UWB signal. In *Progress in Electromagnetics Research Symposium (PIERS)*. 51

Sasirekha, K. & Baby, P. (2013). Agglomerative hierarchical clustering algorithm-a. *International Journal of Scientific and Research Publications*, 83. 83

Sentz, K. & Ferson, S. (2002). *Combination of evidence in Dempster-Shafer theory*, vol. 4015. Citeseer. 39

Shafer, G. (1976). *A mathematical theory of evidence*, vol. 42. Princeton university press. 31, 33, 35, 39, 42, 46, 68, 166

Shahbaba, M. & Beheshti, S. (2014). Mace-means clustering. *Signal Processing*, **105**, 216–225. 82

Shang, Y., Rumi, W., Zhang, Y. & Fromherz, M. (2004). Localization from connectivity in sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, **15**, 961–974. 150, 189

Shi, G. & Ming, Y. (2016). Survey of indoor positioning systems based on ultra-wideband (uwb) technology. In *Wireless Communications, Networking and Applications*, 1269–1278, Springer. 12

Silla Jr, C.N. & Freitas, A.A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, **22**, 31–72. 87

Singer, R.A. (1970). Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, **4**, 473–483. 20, 111

Smets, P. (1988). Belief functions versus probability functions. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, 17–24, Springer. 35

Smets, P. (1992). The nature of the unnormalized beliefs encountered in the transferable belief model. In *Uncertainty in Artificial Intelligence*, 292–297. 35

Smets, P. (1993a). Belief functions: the disjunctive rule of combination and the generalized Bayesian theorem. *International Journal of approximate reasoning*, **9**, 1–35. 68, 173

Smets, P. (1993b). Quantifying beliefs by belief functions: an axiomatic justification. In *International Joint Conference on Artifical Intelligence-Volume 1*, 598–603, Morgan Kaufmann Publishers Inc. 68

Smets, P. (2002). Decision making in a context where uncertainty is represented by belief functions. *Belief Functions in Business Decisions*, **88**, 17–61. 45, 46

Smets, P. (2005). Decision making in the TBM: the necessity of the pignistic transformation. *International Journal of Approximate Reasoning*, **38**, 133–147. 45, 169

Smets, P. (2007). Analyzing the combination of conflicting belief functions. *Information Fusion*, **8**, 387–412. 39, 168

Smets, P. & Kennes, R. (1994). The transferable belief model. *Artificial Intelligence*, **66**, 191–234. 45, 46, 69

Smirnov, A., Shilov, N. & Kashevnik, A. (2012). Ontology-based mobile smart museums service. In *International Conference on Advances in Future Internet (AFIN)*, 48–54. 16

Souza, R., Rittner, L. & Lotufo, R. (2014). A comparison between k-optimum path forest and k-nearest neighbors supervised classifiers. *Pattern Recognition Letters*, **39**, 2–10. 17, 163

Steele, R., Lo, A., Secombe, C. & Wong, Y.K. (2009). Elderly persons' perception and acceptance of using wireless sensor networks to assist healthcare. *International Journal of Medical Informatics*, **78**, 788–801. 8

Stover, J.A., Hall, D.L. & Gibson, R.E. (1996). A fuzzy-logic architecture for autonomous multisensor data fusion. *IEEE Transactions on Industrial Electronics*, **43**, 403–410. 32

Strat, T.M. (1990). Decision analysis using belief functions. *International Journal of Approximate Reasoning*, **4**, 391–417. 45

Stuntebeck, E.P., Pompili, D. & Melodia, T. (2006). Wireless underground sensor networks using commodity terrestrial motes. In *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 112–114, IEEE. 3

Tabakhi, S. & Moradi, P. (2015). Relevance–redundancy feature selection based on ant colony optimization. *Pattern Recognition*, **48**, 2798–2811. 86, 177

Talebi, H. & Hemmatyar, A.A. (2014). Asynchronous track-to-track fusion by direct estimation of time of sample in sensor networks. *IEEE Sensors Journal*, **14**, 210–217. 5, 135

Tibshirani, R., Walther, G. & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **63**, 411–423. 92

Tsirmpas, C., Rompas, A., Fokou, O. & Koutsouris, D. (2015). An indoor navigation system for visually impaired and elderly people based on radio frequency identification (RFID). *Information Sciences*, **320**, 288–305. 8

Üney, M., Mulgrew, B. & Clark, D.E. (2016). A cooperative approach to sensor localisation in distributed fusion networks. *IEEE Transactions on Signal Processing*, **64**, 1187–1199. 5, 6, 135, 162

Varshney, P.K. (1997). Multisensor data fusion. *Electronics & Communication Engineering Journal*, **9**, 245–253. 29, 165

Velmani, R. & Kaarthick, B. (2015). An efficient cluster-tree based data collection scheme for large mobile wireless sensor networks. *IEEE Sensors Journal*, **15**, 2377–2390. 5, 134

Wang, S., Fidler, S. & Urtasun, R. (2015). Lost shopping! monocular localization in large indoor spaces. In *IEEE International Conference on Computer Vision (ICCV)*, 2695–2703, IEEE. 16

Wang, Y. (2008). Topology control for wireless sensor networks. In *Wireless Sensor Networks and Applications*, 113–147, Springer. 5, 162

Wang, Z. & Sun, X. (2013). Multiple kernel local Fisher discriminant analysis for face recognition. *Signal processing*, **93**, 1496–1509. 104, 106

Ward Jr, J.H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, **58**, 236–244. 83

Xiong, H., Tang, J., Xu, H., Zhang, W. & Du, Z. (2018). A robust single GPS navigation and positioning algorithm based on strong tracking filtering. *IEEE Sensors Journal*, **18**, 290–298. 9

Yager, R.R. (1992). Decision making under Dempster-Shafer uncertainties. *International Journal of General System*, **20**, 233–245. 45

Yager, R.R. & Liu, L. (2008). *Classic works of the Dempster-Shafer theory of belief functions*, vol. 219. Springer. 33

Yan, J., Liu, H., Pu, W. & Bao, Z. (2017). Decentralized 3D target tracking in asynchronous 2D radar network: Algorithm and performance evaluation. *IEEE Sensors Journal*, **17**, 823–833. 5, 135, 162

Yang, M., Zhang, L., Shiu, S.C. & Zhang, D. (2013). Gabor feature based robust representation and classification for face recognition with gabor occlusion dictionary. *Pattern Recognition*, **46**, 1865–1878. 104, 105

Yen, Y., Fanty, M. & Cole, R. (1997). Speech recognition using neural networks with forward-backward probability generated targets. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 4, 3241–3244, IEEE. 122

Yiu, S., Dashti, M., Claussen, H. & Perez-Cruz, F. (2017). Wireless RSSI fingerprinting localization. *Signal Processing*, **131**, 235–244. 15

Zadeh, L.A. (1965). Information and control. *Fuzzy Sets*, **8**, 338–353. 31, 166

Zadeh, L.A. (1984). Review of a mathematical theory of evidence. *AI Magazine*, **5**, 81–83. 40

Zanella, A. & Bardella, A. (2014). RSS-based ranging by multichannel RSS averaging. *IEEE Wireless Communications Letters*, **3**, 10–13. 14, 163

Zhang, L., Chew, Y.H. & Wong, W.C. (2013). A novel angle-of-arrival assisted extended kalman filter tracking algorithm with space-time correlation based motion parameters estimation. In *International Wireless Communications and Mobile Computing Conference (IWCMC)*, 1283–1289, IEEE. 21, 112

Zhang, L., Li, Y., Gu, Y. & Yang, W. (2017). An efficient machine learning approach for indoor localization. *China Communications*, **14**, 141–150. 150

Zhou, B., Yang, S., Sun, T. & Grattan, K.T. (2015). A novel wireless mobile platform to locate and gather data from optical fiber sensors integrated into a WSN. *IEEE Sensors Journal*, **15**, 3615–3621. 2, 134, 161

Zhuang, L., Goh, K.M. & Zhang, J.B. (2007). The wireless sensor networks for factory automation: Issues and challenges. In *IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, 141–148, IEEE. 7

Zou, H., Jin, M., Jiang, H., Xie, L. & Spanos, C.J. (2017). Winips: WiFi-based non-intrusive indoor positioning system with online radio map construction and adaptation. *IEEE Transactions on Wireless Communications*, **16**, 8118–8130. 12

Zuo, J., Liu, S., Xia, H. & Qiao, Y. (2018). Multi-phase fingerprint map based on interpolation for indoor localization using ibeacons. *IEEE Sensors Journal*, **18**, 3351–3359. 12

# Daniel ALSHAMAA

## Doctorat : Optimisation et Sûreté des Systèmes

Année 2018


## Localisation *indoor* des capteurs : application pour des personnes âgées dépendantes


Cette thèse porte sur le problème de localisation des personnes âgées dépendantes à l'aide de réseaux de capteurs sans fil. Chaque personne est équipée d'un bracelet, comprenant un capteur capable de mesurer les signaux WiFi. Nous abordons le problème de localisation par zonage, où le but est de déterminer la zone où se trouve la personne. Il s'agit d'un problème de classification multi-classes, que nous traitons en associant la flexibilité des méthodes d'apprentissage statistique à la théorie de Dempster-Shafer pour la fusion de l'information avec incertitude. Nous proposons en premier un modèle d'observation exploitant la puissance des signaux échangés entre les capteurs avec la technique de fingerprinting. Nous développons ensuite le modèle proposé, au travers du regroupement hiérarchique, dans le but de couvrir des surfaces plus grandes. Nous élaborons ensuite une méthode de sélection des bornes WiFi à privilégier. Cette méthode permet d'améliorer la précision et de réduire la complexité de la méthode de localisation. Nous proposons également des modèles de mobilité que nous combinons au modèle d'observation, afin de corriger les estimations en fusionnant toutes les preuves disponibles. D'autre part, nous présentons une approche décentralisée de la méthode de localisation, pour accroître la robustesse et diminuer la complexité. Les performances des méthodes proposées sont validées par des expérimentations sur des données réelles, et évaluées en comparaison avec d'autres méthodes très connues dans le domaine.



Mots clés : réseaux de capteurs (technologie) – théorie de Dempster-Shafer – traitement du signal – services basés sur la localisation – fusion multi-capteurs.


## Indoor Localization of Sensors: Application to Dependent Elderly People


This thesis deals with the problem of localization of dependent elderly people using wireless sensor networks. Each person is equipped with a bracelet that consists of a sensor capable of measuring the WiFi signals. We tackle the problem of localization by zoning, where the objective is to determine the zone where the person resides. It is formulated as a problem of multi-class classification, which we treat by associating the flexibility of statistical learning methods to the theory of Dempster-Shafer for fusion of information with uncertainty. First, we propose an observation model exploiting the power of the exchanged signals between the sensors with the fingerprinting technique. Afterwards, we extend the proposed model, through hierarchical clustering, in order to cover larger surface areas. We then develop a method for selecting the WiFi terminals in favor. This method helps enhancing the overall accuracy and reducing the complexity of the localization method. We also propose mobility models, which we combine with the observation model, in order to correct the estimations by aggregating all available evidence. In addition, we present a decentralized approach of the localization method, to increase the robustness and reduce the complexity. The performance of the proposed methods is validated through experiments on real data, and evaluated in comparison with other well-known methods in the domain.



Keywords: sensor networks – Dempster-Shafer theory – signal processing – location-based services – multisensor data fusion.


**Thèse réalisée en partenariat entre :**

**Ecole Doctorale** "Sciences pour l'Ingénieur"