Localization in sparse wireless sensor networks

Guo, Hao

2011

Guo, H. (2011). Localization in sparse wireless sensor networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/43986>

<https://doi.org/10.32657/10356/43986>

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

**LOCALIZATION IN SPARSE WIRELESS SENSOR
NETWORKS**

GUO HAO

**SCHOOL OF ELECTRICAL AND ELECTRONICS
ENGINEERING**

2011

LOCALIZATION IN SPARSE WIRELESS SENSOR NETWORKS

GUO HAO

School of Electrical and Electronics Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2011

Acknowledgements

I wish to take this opportunity to express my utmost gratitude and appreciation to the people who have assisted me in the preparation of this thesis.

First of all, I sincerely thank my supervisor Associate Professor Low Kay Soon for his patient guidance, support and encouragement throughout my graduate studies. It is because of his deep involvement in my thesis work, I could make this thesis possible. My special gratitude is extended to my co-supervisor Professor Er Meng Joo, who provided valuable guidance at different stages of my research.

It is my great pleasure to work with Miss Nguyen Hong Ahn and Ms. Win Nu Nu Win in Intelligent Systems Centre. They generously shared their knowledge, supported me to conduct experiments and provided valuable advice and suggestion to improve the quality of the experiment.

I appreciate Nanyang Technological University for the financial support by offering a research student scholarship. My thanks also go to all the staffs, technicians and graduate students in Control Engineering Laboratory and Intelligent Systems Centre for their support and help.

Contents

Acknowledgement	i
Contents	ii
Summary	vi
List of Figures	viii
List of Tables	xi

Chapter 1 Introduction.....	1
1.1 Wireless Sensor Networks and Localization.....	1
1.2 Motivations.....	3
1.3 Objectives.....	4
1.4 Main Contributions.....	5
1.5 Outline of the Thesis.....	7

Chapter 2 Literature Review on Localization Algorithms of Wireless Sensor Networks.....	9
2.1 Chapter Introduction.....	9
2.2 Typical WSN Localization Algorithms.....	11

2.3 Localization in Sparse Wireless Sensor Networks.....	20
2.4 Chapter Conclusion.....	24
Chapter 3 Distributed Likelihood Localization Algorithm.....	25
3.1 Chapter Introduction.....	25
3.2 Problem Formulation.....	26
3.3 Evaluation of the localization Algorithm	37
3.4 Chapter Conclusion.....	47
Chapter 4 Sensor Node Design and Implementation of Distributed Likelihood Localization Algorithm.....	49
4.1 Chapter Introduction.....	49
4.2 Sensor Node Architecture and Implementation.....	50
4.3 System Evaluation in Laboratory Environment	58
4.4 Outdoor experiment	69
4.5 Chapter Conclusion.....	73
Chapter 5 Enhanced Likelihood Localization Algorithm	74
5.1 Chapter Introduction.....	74
5.2 MLE based scheme	75
5.3 EKF based scheme	78
5.4 Comparison of MLE and EKF schemes	81
5.5 Enhanced Likelihood Localization Algorithm	83
5.6 Chapter Conclusion.....	89

Chapter 6 Localization of a Wireless Sensor Network using Maximum Likelihood Estimation with Negative Constraints..... 90

6.1 Chapter Introduction and Related Works	90
6.2 MLE Localization without Negative Constraints	92
6.3 Negative Constraints and Modeling	94
6.4 MLE Localization with Negative Constraints.....	96
6.5 Cramer-Rao Bound (CRB) Derivation	97
6.6 Position Estimation.....	101
6.7 Chapter Conclusion.....	104

Chapter 7 Simulation and Experimental Results of MLE Localization Algorithm with Negative Constraints..... 106

7.1 Chapter Introduction	106
7.2 The Negative Constraints Selection Factor k	107
7.3 Performance Comparison of MLE with and without Negative Constraints over Isotropic Networks	109
7.4 Performance under NLOS Environment	112
7.5 Localization in Uniform Random Distributed Networks	114
7.6 Computational cost and performance trend	117
7.7 Experimental Results.....	119
7.8 Chapter Conclusion.....	122

Chapter 8 Conclusion and Future Works 123

8.1 Summary.....	123
8.2 Future works	125

Author's Publications	127
Bibliography	129
Appendices.....	143
Appendix A Relation of Range Error Factor and Standard Deviation of RSSI Measurement.....	143
Appendix B The Covariance Matrix of Process Noise.....	146
Appendix C Information Matrix with Negative Constraint Information.....	148

Summary

In this thesis, we focus on the localization of sparse wireless sensor networks (WSN) through exploiting two types of new information, namely the deployment and negative constraint information.

For this research, we have proposed and developed a new distributed localization algorithm termed as the likelihood localization algorithm (LLA). Different from other algorithms, the LLA takes advantage of the deployment information through a deployment agent (DA). After a sensor node has been deployed, the system uses both the deployment and the inter-node radio information to improve the estimates of its' position through a maximum likelihood estimation (MLE) scheme. LLA has been implemented and evaluated using a low-cost microcontroller. Simulation and experimental results show that it outperforms the conventional approaches in terms of localization accuracy in both sparse and dense WSNs. Furthermore, an enhanced likelihood localization algorithm (ELLA) that hybridizes the MLE and an extended Kalman filter is also proposed to achieve better localization accuracy.

The negative constraint information and the Cramer-Rao bound of estimated position with negative constraint information have also been derived in this thesis. A new centralized MLE algorithm that uses both the negative constraints and inter-node distances information has been proposed and evaluated. The new algorithm outperforms

conventional centralized approaches in terms of localization accuracy especially for WSN with low network connectivity or high range measurement error. Furthermore, the new algorithm requires only comparable computational cost as other existing centralized methods.

List of Figures

Fig. 1.1 A sparse sensor network	4
Fig. 2.1 WSN localization algorithm classification	11
Fig. 2.2 Convex position estimation	16
Fig. 3.1 A sparse WSN	27
Fig. 3.2 Histogram distribution of pedometer measurements	28
Fig. 3.3 Real distance versus measured distance of the pedometer	28
Fig. 3.4 RSSI measurements versus distances	31
Fig. 3.5 Mapped distances in (3.5) versus real distances	32
Fig. 3.6 Histogram distribution of the measured distance in (3.6).....	33
Fig. 3.7 Localization in sparse network	40
Fig. 3.8 Localization error for each sensor node in sparse network	40
Fig. 3.9 Comparison of single- and bi-direction methods for sparse network	42
Fig. 3.10 Localization in dense network	44
Fig. 3.11 Localization error for each sensor node in dense network	45
Fig. 3.12 Comparison of single- and bi-direction methods for dense network	46
Fig. 3.13 Localization performance of LLA and DV-distance for a sparse network	47
Fig. 3.14 Localization performance of LLA and DV-distance for a dense network	48
Fig. 4.1 Sensor node block diagram	50
Fig. 4.2 System evaluation of a sparse network with single-direction approach	60
Fig. 4.3 System evaluation of a sparse network with bi-direction approach	61
Fig. 4.4 Average localization error under different measurement accuracy	62
Fig. 4.5 Average localization error with $p = 0.35$	63

Fig. 4.6 Average objective function value versus number of iterations/generations.....	66
Fig. 4.7 Experimental results for single-direction method	70
Fig. 4.8 Experimental results for bi-direction method	71
Fig. 4.9 Experiment error at every node	72
Fig. 5.1 Simulated network	81
Fig. 5.2 Localization error of LLA, ELLA, MLE and EKF based schemes	83
Fig. 5.3 Localization error under different pedometer and range errors	86
Fig. 5.4 localization error vs electronic compass accuracy	87
Fig. 5.5 Experimental localization result	88
Fig. 6.1 The negative constraint illustration	94
Fig. 6.2 CRBs on rms localization error without and with NC	100
Fig. 6.3 $\left \frac{\partial l_{ij}(\mathbf{X})}{\partial d_{ij}(\mathbf{X})} \right $ vs $d_{ij}(\mathbf{X})$	102
Fig. 6.4 Optimization procedure	105
Fig. 7.1 Effect of k on the localization accuracy	108
Fig. 7.2 Effect of k on the number of violated NCs	108
Fig. 7.3 Comparison of localization accuracy of MLE and MLE-NC	109
Fig. 7.4 Localization examples	110
Fig. 7.5 Localization accuracy improvement by NC under different range measurement error and connectivity	112
Fig. 7.6 Network under NLOS environment	113
Fig. 7.7 Localization accuracy under NLOS environments	114
Fig. 7.8 An example of uniform random distributed networks	115
Fig. 7.9 Localization coverage for uniform random distributed networks	116
Fig. 7.10 Localization error for uniform randomly distributed networks	117

Fig. 7.11 Objection function values vs iterations	119
Fig. 7.12 Experimental network and localization result	120
Fig. 7.13 Experimental system structure and data flow	121

List of Tables

Table 2.1 WSN localization algorithms	12
Table 4.1 Pseudo-code of <i>RSSI to Distance Translator</i>	53
Table 4.2 Pseudo-code of GNA	55
Table 4.3 Pseudo-code of PSO	56
Table 4.4 Average execution time for different number of neighboring nodes	64
Table 4.5 Average execution time for different pedometer error factor	65
Table 4.6 Probability of GNA being unstable	68
Table 4.7 PSO versus GNA	72
Table 5.1 Experiment results	89
Table 7.1 Computational costs	118
Table 7.2 Experimental results of the localization with different number of unknown nodes	122

Chapter 1

Introduction

1.1 Wireless Sensor Networks and Localization

Due to the advancement in low power microelectronic circuits, wireless sensor networks (WSNs) have attracted a lot of research interests in recent years. WSNs are systems that comprise a large number of wirelessly connected heterogeneous sensor nodes. The sensors are used to monitor physical or environmental conditions such as temperature, sound, vibration, pressure, motion or pollutants at different locations [1]. The sensor nodes are physically small, inexpensive and could be deployed in large numbers. Moreover, they consume little power to allow prolong operation for years. WSN can be used in many applications such as environmental monitoring and natural habitat study, medical instrumentation, industry and military applications [2]-[4].

Sensor node localization is an essential requirement in many WSN applications. The sensed data may not be useful without knowing the physical locations of the sensor nodes. For environmental monitoring application, sensors send the environmental conditions as well as the corresponding positions to the user center. Knowing the sensor nodes' physical locations is also a prerequisite in target tracking application [5]-[8]. Furthermore,

the location information can be used to improve routing efficiency and assist the system in determining if the nodes are redundant, and can thus be idle to save energy [9]-[11].

Since the network may consist of a large number of sensor nodes, it would be error prone if the location is manually assigned to each sensor node. It is also not practical if the network is to be setup in an emergency. One straightforward solution is to equip a global positioning system (GPS) to each sensor node. However, GPS is limited for outdoor applications. It is also costly and power consuming. Hence, GPS-less self localization has become an active research in recent years [12]-[51].

For the GPS-less localization systems, only a small fraction of the sensor nodes are localized by GPS and the remaining are localized through the information exchange between neighbor nodes. The WSN localization algorithms usually consist of two major phases, namely the information collection and position determination. The information collection phase is responsible for obtaining the necessary information such as distance between sensor nodes. For the position determination phase, its objective is to estimate unknown nodes' positions based on the information obtained from the first phase.

Since the amount of available inter-node information is directly dependent on the network connectivity, most existing localization schemes require the network to be heavily connected. In fact, self localization algorithms require the network to be globally rigid in order to localize every unknown node successfully [52]. A graph is globally rigid if and only if any motion of its vertices maintaining the edge constraints comes from a family of motions of all Euclidean space which are distance-preserving [100]. In the WSN context, there is only one set of nodes' relative positions satisfies the inter-node

constraints of a globally rigid WSN. For a sensor network that has sparse deployment density due to cost limitation or application requirements, there is less reported research works.

1.2 Motivations

Fig. 1.1 shows the perimeter deployment application using a sparse network. The sensor nodes are deployed along the border of a site for monitoring purpose. This leads to a non-rigid network. For GPS based localized system, the GPS signal may be blocked by the hostile forces during war. For forest monitoring application, the GPS signal could be blocked by the trees and becomes unavailable. Moreover, such system is typically a sparse network due to large coverage area. The use of conventional self-localization algorithms over this type of sparse network would lead to a low coverage network (i.e. a high percentage of the sensor nodes are not able to estimate their positions) with high localization error. In this class of WSN application, additional information besides distances between neighboring nodes are required to localize the whole network.

Fig. 1.1 A sparse sensor network

Motivated by the localization issues of sparse sensor network, two types of additional information namely the deployment and negative constraint information, are exploited and investigated in this research.

1.3 Objectives

The objectives of this thesis are as follows:

- Develop a new distributed WSN localization algorithm that uses the deployment information. The new algorithm should outperform the conventional methods in terms of localization accuracy, especially for sparse WSN that is non-rigid. Moreover, the new algorithm should have modest computation and memory requirements for each sensor node. Therefore it can be easily implemented using a low-cost microcontroller for distributed application.

- Investigate a new centralized WSN localization algorithm that uses the negative constraint information. The new algorithm should outperform other centralized methods in terms of localization accuracy, especially for WSN with low network connectivity. Furthermore, the new algorithm should have comparable computational requirement as the conventional centralized algorithms.
- Propose suitable models for WSN localization. The model can be used for theoretical formulation and simulation study to evaluate the systems performance under different parameters such as measurement errors, network connectivity and network topologies.
- Design and implement a prototype WSN and deployment agent system that can be used to validate the performance of the proposed algorithms in terms of localization accuracy and computational requirement.

1.4 Main Contributions

The main contributions of this thesis are as follows:

- **A new likelihood localization algorithm for sparse WSN**

A distributed localization algorithm, i.e. the likelihood localization algorithm (LLA), that utilizes deployment information has been developed and validated. The new algorithm outperforms the conventional approaches in terms of localization accuracy in both sparse and dense WSNs. Furthermore, the new algorithm can be easily implemented using a low cost microcontroller and is useful for distributed network.

- **Enhanced likelihood localization algorithm to improve localization**

An enhanced likelihood localization algorithm (ELLA) is proposed by modifying the system model and localization formulation to achieve better estimation accuracy. The ELLA hybridizes the LLA and extended Kalman filter to benefit from their advantages. Simulation and experimental results show that the localization accuracy of ELLA is 20% better than LLA.

- **Maximum likelihood estimation localization algorithm with negative constraints**

A new centralized maximum likelihood estimation (MLE) algorithm that uses both negative constraints and inter-node distance information for a WSN has been investigated and developed. As compared to the conventional centralized MLE localization, the new algorithm improves the localization accuracy by 20% while it requires only comparable computational cost.

- **Design and implementation of wireless sensor node and deployment agent's sensors**

A WSN has been setup together with a deployment agent who is equipped with a pedometer and an electronic compass. The developed WSN and deployment agent have been used to evaluate the proposed LLA and other localization algorithms in real environments under different scenarios.

The proposed LLA has been implemented using a low-cost microcontroller. Different optimization approaches, such as the Gauss-Newton algorithm and particle swarm optimization method, have also been implemented for benchmarking. The realized LLA system can be used to localize a sparse WSN in real time.

- **Simulation models for deployment, radio ranging and negative constraints information**

Simulation models for information such as deployment, radio ranging and negative constraints have been developed to reflect their statistical nature. The developed models can be used to simulate WSN localization algorithms under different environments and network topologies. The optimized parameters obtained from the simulation are used in real system design with minimum modification.

1.5 Outline of the Thesis

The rest of the thesis is structured as follows:

Chapter 2 reviews the related research works in WSN localization. The challenges and the limitations of the existing works for sparse WSN localization are also presented.

Chapter 3 presents in details the proposed LLA. The LLA is the first algorithm that utilizes deployment and inter-node distance information for WSN localization. Simulation models of deployment and radio ranging information are also discussed in this chapter. Various simulation results are also presented and compared with other methods. The results show that the proposed approach overcomes the weaknesses of existing methods.

Chapter 4 presents the architecture design of sensor node hardware. Microcontroller based real-time implementation is also discussed in this chapter. System performance of the implementation is also evaluated under different system parameters in terms of localization accuracy, computational costs and robustness.

Chapter 5 presents the proposed ELLA to enhance the localization accuracy. It has been evaluated in both simulation and experiment under different system parameters to justify its performance.

Negative constraint information is studied and its simulation model is discussed in Chapter 6. A MLE based localization algorithm, which utilizes both the negative and inter-node distance information has been investigated.

Simulation and experimental results under different environments and network conditions are presented and compared with existing methods in Chapter 7. The results show that it outperforms the conventional MLE approach by 20% in localization accuracy.

Chapter 8 summarizes this work and describes the directions of future research in WSN localization.

Chapter 2

Literature Review on Localization Algorithms of Wireless Sensor Networks

2.1 Introduction

Determining the sensor nodes' physical locations is one essential requirement and a key challenge in WSN applications. This can be achieved by equipping global positioning systems (GPS) to all the sensor nodes. However, such approach is costly, consumes high power and is limited for outdoor applications. Consequently, a number of GPS-less localization systems have been investigated [12]-[51].

The WSN localization algorithms usually consist of two major phases; namely the information collection and position determination. This process is shown in Fig. 2.1. The information collecting phase is responsible for obtaining the necessary information such as distance between sensor nodes. Based on the type of information obtained in this phase, the algorithms can be classified as range-free, range-based and mobile assistant based algorithms as depicted in Fig. 2.1. For the position determination phase, its objective is to estimate the unknown nodes' positions based on the information obtained from the first phase. In the second phase, the localization algorithms can be classified as constraint, distributed multilateration and centralized collaborative algorithms. In the constraint

based algorithms, the possible location of an unknown node is constrained into a smaller region. The geometric centre of this region is then used as the unknown node's estimated position. In the distributed multilateration algorithms, each node is responsible for computing its own estimated position. Each unknown node obtains distances to at least three reference nodes and uses multilateration to determine its position. For the centralized collaborative algorithm, it is assumed that there is a central node to collect information across the network. Moreover, the estimated positions are computed by the central node. The nodes in the network are localized collaboratively through Maximum Likelihood Estimation (MLE) or Multidimensional Scaling (MDS) techniques. In general, the centralized algorithms have better accuracy than the distributed algorithms as they have the knowledge of the entire network. However, more communications for the central nodes are required to collect local information. The communication cost will increase with the network size. Therefore, distributed algorithm is preferred for large scale network.

The rest of this chapter presents some of the typical localization algorithms for a WSN. The common features and the merits of various schemes will also be discussed.

Fig. 2.1 WSN localization algorithm classification

2.2 Typical WSN Localization Algorithms

Sensor node localization has been a topic of active research in recent years. Earlier research on the localization systems are infrastructure based [12]. However, recent research focuses more on self-localization algorithms [13]-[51] due to their lower cost, scalability and flexibility.

Self-localization algorithms require little infrastructure. They estimate the sensor node positions based on the network connectivity or inter-node distances. They usually consist of two major phases, namely the information collection and position determination. Table 2.1 lists the classification of WSN localization algorithms and their advantages and disadvantages. In the following subsections, several algorithms for both phases are reviewed.

2.2.1 Information Collecting Phase

To determine an unknown node's position, the distances to at least three non-collinear reference positions are needed. The information collecting phase is responsible for obtaining data needed by position determination phase. Based on the type of information

available, the self-localization algorithms can be classified as range-free, range-based and mobile assistant based algorithms. The details are discussed in the following sections.

Table 2.1 WSN localization algorithms

	Constraint based	Distributed Multilateration	Centralized Collaborative	Advantages	Disadvantages
Range-free	[14]-[20]	[24]	[21]-[23]	Low cost, no additional hardware is required; localization accuracy is independent on measurement error	Lower localization accuracy compared to range-based method or require high reference node density
Range-based	[35]	[24]-[34]	[36]-[43]	Higher range measurement accuracy usually leads to better localization accuracy.	Need additional hardware for the range measurement
Mobile Assistant	[47]-[49]	[50]-[51]		Mobile nodes are required	Can be used to localize sparse networks
Advantages	Do not require the knowledge of distances to reference nodes, suitable for range-free methods	Communication costs per node are independent on network size; do not require powerful central node	Sensor nodes in the network are localized collaboratively; high localization accuracy		
Disadvantages	Require high reference nodes density	Localization accuracy generally low compared to centralized collaborative algorithms	Communication costs per node increase with network size; require powerful central node		

a) Range-free Algorithms

Range-free algorithms [14]-[24] assume that the distance or angle information is not available for the sensor node. They use the network connectivity to approximate the nodes' locations. It has the advantage of low-cost with no additional hardware. Furthermore, the localization accuracy is not affected by the measurement error as in the range-based

algorithms. Their main disadvantage is the poorer localization accuracy as compared to the range-based algorithms.

b) Range-based Algorithms

Range-based algorithms [24]-[46] require distance measurements from a set of anchor nodes and use multilateration or maximum likelihood estimation techniques to find the locations of unknown nodes. The distance information can be obtained from RSSI (received signal strength index) [53]-[55], TOA (time of arrival) / TDOA (time difference on arrival) [56]-[59] or AOA (angle of arrival) [60]-[62] measurements. RSSI is low cost and requires little additional hardware. However, the RSSI measurements have larger errors as it is subject to the deleterious effects of fading channels. TOA and TDOA yield better accuracy at the expense of additional hardware requirements to receive a signal that has a smaller propagation speed than radio, such as ultrasound. AOA requires more sophisticated hardware such as antenna array to measure the angle of arrival for a received signal.

The range-based algorithms can be further classified as single-hop and multi-hop range based algorithms. Besides single-hop range information, multi-hop range based algorithm also utilizes the distances information between nodes that are several hops apart. Consider DV-distance [24] as an example. As most unknown nodes are not directly connected to the anchor nodes, DV-distance uses a multi-hop distance determination technique to find the distances between unknown and anchor nodes. The multi-hop distance determination technique uses the shortest path length between two nodes times a correction factor to infer the distance. The correction factor can be calculated as the

average ratio of the distance and shortest path length of two anchor nodes. The similar multi-hop distance determination technique can be found in several other localization algorithms [26], [38]. This technique performs quite well in isotropic networks but has limited performance in anisotropic networks. This is because the Euclidean distance between two nodes are not proportional to their shortest path in anisotropic networks.

Single-hop range based algorithms only utilize the range information between nodes in direct hop. In [27] and [31], successfully localized unknown nodes are promoted to “anchor” nodes to help localization of other unknown nodes. This effectively increases the density of “anchor” nodes. Its advantage is that they do not require multi-hop distance determination technique as in DV-distance. The drawback is the error propagation, due to the localization error from the promoted “anchor” nodes. Some centralized collaborative localization algorithms [36]-[37] require only the knowledge of single-hop inter-node distances. They view the entire network as a single object and use the inter-node measurement to form the objective function. The optimal coordinates that best fit the objective function are considered as the unknown nodes estimated positions. This type of algorithms needs a powerful central node to collect the inter-node information and perform optimization for the objective function.

c) Mobile Assistant Based Algorithms

In mobile assistant based algorithms [47]-[51], one or more mobile anchor nodes equipped with the GPS receivers broadcast their positions periodically and travel through the interested area to localize the unknown nodes. The main advantage of mobile assistant based algorithms is that they are suitable for sparse networks that are non-rigid. More details on the mobile assistant based algorithms will be discussed in Section 2.3.

2.2.2 Position Determination Phase

The purpose of this phase is to estimate unknown nodes' positions based on information obtained from the first phase. Depending on the method to obtain the unknown nodes' positions, the localization algorithms can be classified as constraint, distributed multilateration and centralized collaborative algorithms. The details are presented in the following sections.

a) Constraint Based Algorithms

Many range-free algorithms are also constraint based algorithms [14]-[20]. Without knowing the distances to some reference nodes, an unknown node's position is not deterministic. Instead, the unknown node's possible location can be constrained into a smaller region based on the network connectivity. The geometric centre of this region is then used as the unknown node's position.

Consider the Convex Constraint Satisfaction [16] as an example. If an unknown node can hear from 3 anchor nodes as shown in Fig. 2.2, its position is constrained to be the overlapped region of the three circles that represent the communication regions of the 3 anchor nodes. The central point of the overlapped region is considered to be the unknown node's location.

Fig. 2.2 Convex position estimation

b) Distributed Multilateration Algorithms

In distributed multilateration algorithms, each unknown node is required to obtain distances to at least three non-collinear reference nodes in the first phase. These reference nodes can be absolute anchor nodes or “pseudo-anchors” updated from the localized nodes. In the second phase, each unknown node uses multilateration to determine its position. In the following, we present the commonly used multilateration technique that is used in deriving an unknown node position [24], [26]-[27], [30]-[31].

From the measured distances d_i and known positions (x_i, y_i) of anchor nodes, the following system of equations can be derived.

$$\begin{aligned}
 (x_1 - x)^2 + (y_1 - y)^2 &= d_1^2 \\
 \vdots \\
 (x_n - x)^2 + (y_n - y)^2 &= d_n^2
 \end{aligned} \tag{2.1}$$

where the unknown node position is denoted as (x, y) .

The system can be linearized by subtracting the last equation from the first $n-1$ equations and rearranging the terms to give a proper linear equation in the form of $\mathbf{A}\boldsymbol{\theta}=\mathbf{b}$, where

$$\mathbf{A} = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix}, \quad (2.2)$$

$$\mathbf{b} = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix}. \quad (2.3)$$

The system can now be solved using standard least-square approach:

$$\hat{\boldsymbol{\theta}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (2.4)$$

This multilateration method is a linearized procedure without considering the statistic error distribution of the range measurement. This method is fast and has low computational cost. Other sophisticated multilateration methods which reflect the statistical characteristics of range measurement can be found in [28], [34] and [63].

The major advantage of distributed algorithms is the low communication cost for large scale network. As each sensor only uses its local information to compute its position, the communication cost in each node will not increase with the network size. However, distributed algorithms generally have lower localization accuracy as they do not utilize all the available information. In particular, most ranges between the neighboring nodes are not utilized when the node-anchor distances are being determined [30]. Some distributed algorithms also include an optional refinement phase after the first two phases [30]-[31].

The objective is to improve estimated positions using measured distances between the sensor nodes. By taking into account the inter-node information, the refinement procedures would lead to a better localization result. The underlying principle of refinement is that after the completion of the first two phases, each unknown node inside the network computes an estimate of its location during iteration. Since most unknown nodes are not directly connected to the anchor nodes, they use the estimated positions of their neighbors as the reference points for estimating their locations. Once an unknown node computed its new estimate, it broadcasts this estimate to its neighbors for them to update their position in the next iteration. This computation is repeated until a pre-specified criterion is met. This refinement may become costly if the number of iteration is not limited.

c) Centralized Collaborative Algorithms

Centralized collaborative algorithms [21]-[23], [36]-[43] assume there is a central node that collects information across the network. The estimated positions are computed by the central node. The nodes in the network are localized collaboratively through maximum likelihood estimation (MLE) or Multidimensional Scaling (MDS).

MDS is a statistical technique used to analyze the structure of similarity of data in a multidimensional space. It has been used extensively in many areas, including statistics, psychology, sociology, political science, and marketing [64]. The goal of multidimensional scaling is to find a low dimensional representation of a group of objects (e.g., sensor positions) such that the distances between objects fit the dissimilarity as well as possible. In the context of localization, the dissimilarity $\tilde{d}_{ij}$ is usually the measurement

of the actual geometric distance between two nodes. The MDS solution of sensor node positions can be obtained by minimizing the following loss function [37]:

$$S(\mathbf{X}) = \sum_{i=1}^{N-1} \sum_{j=i+1}^N w_{ij} (d_{ij}(\mathbf{X}) - \tilde{d}_{ij})^2 \quad (2.5)$$

where $\mathbf{X}$ is called the configuration matrix. $\mathbf{X} = [\mathbf{\theta}_1 \ \mathbf{\theta}_2 \ \dots \ \mathbf{\theta}_N]$; $\mathbf{\theta}_i = [x_i \ y_i]^T$ is the location coordinates of node i ; $d_{ij}(\mathbf{X}) = \|\mathbf{\theta}_i, \mathbf{\theta}_j\|$; and w_{ij} is the weight which is defined as:

$$w_{ij} = \begin{cases} 1, & \text{if } \tilde{d}_{ij} \text{ is observed} \\ 0, & \text{otherwise} \end{cases} \quad (2.6)$$

MDS-MAP [22], [38] is a popular MDS based algorithm. It can work in both range-free and range-based conditions, depending on the availability of the range information. For this approach, it computes the distances of all pairs of sensor nodes in the network to form the dissimilarity matrix. It performs very well for isotropic network as the dissimilarity matrix imposes more constraint to the result. However, its computational cost is high to manipulate a large amount of information.

In MLE based algorithms [36]-[37], [41], the system first forms a likelihood function for the unknown nodes' positions based on the inter nodes observations. The inter nodes observations are the distances between the sensor nodes estimated from RSSI or TOA measurements. The unknown nodes' positions are determined collaboratively through maximizing the likelihood function. Therefore, the localization problem is equivalent to an optimization problem [36]-[37], [41].

In [36], a MLE based centralized collaborative localization algorithm that uses RSSI measurements is proposed. It shows that if RSSI measurement has white Gaussian errors, the localization problem is equivalent to the following optimization problem:

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \left(\sum_{i=1}^{N_l} \sum_{j=i+1}^N w_{ij} ((\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2) \right) \quad (2.7)$$

where $\tilde{d}_{ij}$ is the measured distance between nodes i and j .

As mentioned in Section 2.1, the major advantage of centralized algorithm is its potential of achieving high localization accuracy. The disadvantage is the communication and computational cost per node increases with network size. Thus, direct application of the centralized algorithm is not practical in a large scale network. To improve the efficiency [39] and [99] partition a large network to several small local networks to lower the communication cost. In [38], each unknown node forms a local map with the nodes within a fixed radius. These local maps are localized using MDS separately and then merged to form a global map through common nodes.

2.3 Localization in Sparse Wireless Sensor Networks

Section 2.2 presents a literature review on the WSN localization algorithms. In these algorithms, the sensor node density or the network connectivity directly affects the amount of available inter-node distance information and hence the localization performance. A low network connectivity leads to either low coverage or large localization errors. In fact, self-localization algorithms that utilize only inter-node

distance information require the network graph to be globally rigid to successfully localize every unknown node [52].

For sparse networks with low deployment density due to cost limitation or application requirements, there is less reported work [47]-[51]. For forest monitoring application, the network is typically sparse due to large coverage area. As a result, the network connectivity could be lower than 4 on the average. For military application, the deployment of a sensor network along the camp site border for monitoring purpose leads to non-rigidity of the network graph. For this type of sparse network, the conventional self-localization algorithms that use only the inter-node distance information are not adequate. Additional information besides distances between neighboring nodes must be utilized to localize the whole network. The rest of this section presents some typical algorithms that can be used for sparse sensor network location. Their limitations and possible improvements are also discussed.

2.3.1 Typical localization algorithms for sparse networks

Most of the localization algorithms developed for the sparse WSNs use a mobile assistant [47]-[51]. In these algorithms, one or more mobile anchor nodes which are equipped with GPS receivers periodically broadcast their positions and travel through the interested area to help localizing the unknown nodes. The broadcasted positions are used by the receiving unknown nodes as reference to estimate their positions.

Path planning is a fundamental but not extensively studied problem in mobile assistant based algorithms. The localization accuracy is dependent on the trajectory of the mobile

anchor. In [51], an unknown node needs to receive at least three beacon packets from different anchor locations for a unique estimated position. Therefore, the mobile anchor needs to travel near to as many nodes as possible for good localization coverage. In general, an unknown node that is closer to the trajectory has better localization accuracy. In [48], a sensor node that is closer to the trajectory has more and longer chords of its range boundary circle, and consequently has more information to constrain the position estimation. Most of the approaches, however, only consider random movement for the mobile anchor. Works that discuss the problem of optimal mobile anchor trajectory is very limited. From [47]-[51], [65]-[66] there are the following guidelines on selecting a good trajectory:

1. The trajectory should travel near to as many sensor nodes as possible. Generally, a sensor node that is closer to the trajectory has more accurate estimated position.
2. For triangulation or multilateration based approaches, co-linearity should be avoided during the localization.
3. To reduce the power consumption, the path length shall be kept short.

From the above criteria, the optimal mobile anchor trajectory is intrinsically dependent on the network topology. Works [65]-[66] on the specific moving pattern without reference to the actual distribution of the WSN nodes are not adequate.

2.3.2 Proposed Algorithm using Deployment Information

In this thesis, a sparse WSN localization algorithm that utilizes both deployment and inter-node distance information is proposed. In this algorithm, it is assumed that a deployment agent (DA) equipped with a position sensor (such as a pedometer and an electronic compass) travels through the sensor field to deploy the sensor nodes. The position sensor keeps track the DA's travel and uses inter-node distance information to estimate the unknown nodes' positions through maximum likelihood estimation (MLE). The proposed scheme is termed as likelihood localizations algorithm (LLA). It has the following differences from the existing mobile-assistant based scheme:

1. The deployment path of the DA always covers the whole network. As all the sensors are assumed to be deployed along the path, this guarantees good coverage as each deployed node is able to obtain information from the DA.
2. The DA is equipped with a position sensor which consists of a pedometer and an electronic compass instead of a GPS receiver. In many situations, the GPS information may not be available. For forest monitoring and military applications, the GPS signal could be blocked by the trees or the hostile forces during war. The proposed approach does not suffer from this problem.
3. Most existing mobile-assistant based algorithms ignore the inter-node distance information. Though the connectivity is low in sparse network, the sensor nodes are not completely isolated from each other. Thus, inter-node distance information is fully utilized in the proposed approach to improve the localization accuracy.

4. Different from most existing mobile-assistant based algorithms, linear deployment path still can successfully localize each unknown nodes.

In the following chapters, we will investigate the LLA scheme in details.

2.4 Conclusion

This chapter presents a literature review on the state-of-the-art WSN localization algorithms. The localization algorithms usually consist of two major phases, namely the information collection and position determination. Based on the type of information obtained in the information collection phase, the algorithms can be classified as range-free, range-based and mobile assistant based algorithms. Based on the second phase, it can be classified as constraint, distributed multilateration and centralized collaborative algorithms.

Most localization schemes require good network connectivity. To localize this class of sparse networks, additional information besides distances between neighboring nodes must be used. In the next chapter, a new distributed GPS-less localization algorithm for sparse WSN is proposed. Different from the existing algorithms, the proposed approach takes advantage of the deployment information as well as the inter-node distance information to improve the localization accuracy.

CHAPTER 3

Distributed Likelihood Localization Algorithm

3.1 Introduction

As mentioned in Chapter 2, most of the existing WSN localization schemes work well only for dense networks that are rigid globally. For sparse network with low deployment density due to cost limitation or application requirements, there is less reported work. Localization algorithms for sparse WSNs are usually mobile-assistent based. In these algorithms, one or more mobile beacon nodes that are equipped with GPS receivers broadcast their positions periodically. They travel through the area of interest to localize the unknown nodes. In these algorithms, the path planning for the mobile beacon node is a difficult fundamental problem. For good coverage and accurate localization, the path of the mobile node is usually long. Furthermore, the algorithms rely heavily on the GPS information that may not always be available in applications such as forest monitoring or military applications.

In this chapter, we propose a new distributed GPS-less localization algorithm for sparse WSNs, namely the likelihood localization algorithm (LLA). Different from other algorithms, the proposed approach takes advantage of the deployment information from a deployment agent (DA). The DA could be a mobile robot, an unmanned aerial vehicle (UAV) or a person. The deployment information is the movement or location of the DA

when a node is deployed. After a sensor node has been deployed, the system uses both the deployment and the inter-node radio information to estimate the unknown nodes' positions through a maximum likelihood estimation (MLE) algorithm. Simulation results show that the proposed localization scheme works well for both sparse and dense networks. Despite the lack of absolute anchor nodes, the results show that this approach could outperform other localization schemes such as DV-distance that uses absolute anchor nodes.

3.2 Problem Formulation

Consider the perimeter deployment as shown in Fig. 3.1 for a sparse network. In this case, the DA is a walking person equipped with distance and heading sensors. The DA is walking from position A around an area of interests and ends at position B. It is assumed the real position of A is known a priori. The DA deploys the sensor nodes and his travel is measured during the deployment. Upon the deployment of each sensor node, the deployment position is first estimated and transmitted to the sensor node. Subsequently, the sensor node requests the other localized nodes to send their beacon packages to infer the ranges based on the received signal strength Indicator (RSSI) of the radio frequency (RF) signals. By utilizing both the deployment and radio ranging information with the proposed approach, a maximum likelihood function is constructed and a MLE method is used to compute the best location estimate. The MLE method is one of the most popular approaches in obtaining practical estimator. Different from other popular distributed position estimation method such as triangulation and multilateration, there is no requirement for the number of neighboring nodes. Furthermore, as information from two

independent sources is utilized, it can achieve a better result through data fusion of the information. Details of the proposed algorithm are explained in the following sections.

Fig. 3.1 A sparse WSN

3.2.1 Deployment Likelihood Function

To facilitate experimental measurements, the DA is a person equipped with an ultrasonic based pedometer [67] and an electronic compass. Using the pedometer, a fixed real distance of 1m has been measured 500 times. The resulting histogram distribution of the measured distances is shown in Fig. 3.2. The figure shows that the pedometer has an approximated Gaussian distributed error. Similar measurements have also been conducted for distances at 2m, 4m, 6m, 8m and 10m. The measured distances are plotted versus their corresponding real distances in Fig. 3.3. From the figure, it shows that the standard deviation of the measured distance is proportional to the real measured distance. For ease of analysis, the pedometer error factor p is introduced in this thesis as a ratio between the standard deviation and the measured distance V .

Fig. 3.2 Histogram distribution of pedometer measurements

Fig. 3.3 Real distance versus measured distance of the pedometer

For each sensor node i , its deployment position $\Phi_{di} = (x_{di}, y_{di})$ provided by the DA is considered to be near the exact position $\Phi_i = (x_i, y_i)$. The likelihood function of Φ_i is the following conditional probability density function:

$$L_d(\Phi_i) = P(\Phi_{di} | \Phi_i) \quad (3.1)$$

where the subscript ‘ d ’ denotes deployment measurement.

Φ_{di} is the summation of the previous deployed node ($i-1$) position and the measured moving vector between nodes i and $i-1$. Let $\tilde{\mathbf{v}}_i = [\tilde{v}_{xi} \ \tilde{v}_{yi}]$ be the measured moving vector between nodes i and $i-1$. If DA only uses the pedometer without an electronic compass, $\tilde{\mathbf{v}}_i$ is the measured movements in the x and y directions separately, i.e. $\tilde{v}_{xi}$ and $\tilde{v}_{yi}$ are uncorrelated. The deployment likelihood function is obtained as the product of their pdfs:

$$\begin{aligned} L_d(\Phi_i) &= P(\tilde{v}_{xi}) \times P(\tilde{v}_{yi}) \\ &= \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{(x_i - x_{di})^2}{\sigma_x^2} + \frac{(y_i - y_{di})^2}{\sigma_y^2}\right)\right) \end{aligned} \quad (3.2)$$

where the approximations $\sigma_x = \tilde{v}_{xi} \cdot p$ and $\sigma_y = \tilde{v}_{yi} \cdot p$ are used;

If a heading sensor is used by the DA, $\tilde{v}_{xi}$ and $\tilde{v}_{yi}$ are the measured moving vector projections along the x and y direction respectively. In this case, $\tilde{v}_{xi}$ and $\tilde{v}_{yi}$ are correlated. The exact deployment likelihood function becomes more complicated if it is formulated in the Cartesian coordinate system. We will address this issue in Chapter 5.

3.2.2 Radio Ranging Likelihood Function

If the sensor node i can receive the beacon packet of a localized node j , then the distance between the two nodes can be estimated through the RSSI measurement. Let this measured distance be $\tilde{d}_{ij}$ and the actual distance be $d_{ij} = \delta(\Phi_i, \Phi_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, then the likelihood function of the real position Φ_i will be:

$$L_{rj}(\Phi_i) = P(\tilde{d}_{ij} | d_{ij}) = P(\tilde{d}_{ij} | \delta(\Phi_i, \Phi_j)) \quad (3.3)$$

where the subscripts ' r ', ' j ' denote 'radio ranging' and localized node's index respectively. Because the real position Φ_j is unknown, its estimated position $\hat{\Phi}_j$ is used instead and (3.3) becomes

$$L_{rj}(\Phi_i) = P(\tilde{d}_{ij} | \delta(\Phi_i, \hat{\Phi}_j)) \quad (3.4)$$

To study the distribution of the RSSI measured distance, a series of measurements between a pair of RF modules (XBee ZNet 2.5) at different distances have been conducted in an outdoor car park. Fig. 3.4 presents the relation between the distance and the measured RSSI. From Fig. 3.4, the standard deviation of the measured RSSI is independent of the distance and the mean of the measured RSSI follows the log-shadowing model [68]-[69]. Given a measured RSSI between two sensor nodes, their distance can be estimated from the best-fit log-shadowing equation:

$$\tilde{d}_{ij} = 10^{\frac{P_0 - P_{ij}}{10\alpha}} \quad (3.5)$$

where $\tilde{d}_{ij}$ is the mapped distance, P_{ij} is the received RSSI value, P_o is the received RSSI value for 1m distance and α is the path-loss exponent. P_o and α are calibrated values in the best fit log-shadowing equation.

Fig. 3.4 RSSI measurements versus distances

Fig. 3.5 plots the relation between the mapped distance in (3.5) and the real distance. From Fig. 3.5, it is observed that the standard deviation of the mapped distance is proportional to the real distance. It is also noticed that the mapped distance in (3.5) is a biased estimate. The mean is generally larger than the real distance. A correction factor c can be used to reduce the bias as follows:

$$\tilde{d}_{ij} = c \cdot 10^{\frac{P_0 - P_{ij}}{10\alpha}} \quad (3.6)$$

It can be shown that $c = \exp(-\frac{1}{2}(\frac{\ln(10)}{10} \frac{\sigma_R}{\alpha})^2)$, where σ_R is the standard deviation of the RSSI measurement. In a typical RF channel, it is found that $c \approx 0.95$.

Fig. 3.5 Mapped distances in (3.5) versus real distances

Fig 3.6 shows the histogram distribution of the RSSI measured distance based on (3.6). The figure shows that the measurement error is approximately Gaussian distributed, i.e.:

$$\tilde{d}_{ij} \sim N(d_{ij}, \sigma_r^2) \quad (3.7)$$

where $\sigma_r = d_{ij} \cdot \gamma$ and γ is termed the range error factor. With (3.7), the likelihood function of (3.4) becomes

$$L_{rj}(\Phi_i) = \frac{1}{\sqrt{2\pi}\sigma_r} \exp\left(-\frac{(d_{ij} - \tilde{d}_{ij})^2}{2\sigma_r^2}\right) \quad (3.8)$$

In practice, the real distance d_{ij} is unknown and the approximation $\sigma_r = \tilde{d}_{ij} \cdot \gamma$ is used.

Then (3.8) can be approximated as

$$L_{rj}(\Phi_i) = \frac{1}{\sqrt{2\pi}\tilde{d}_{ij}\gamma} \exp\left(-\frac{(\delta(\Phi_i, \hat{\Phi}_j) - \tilde{d}_{ij})^2}{2(\tilde{d}_{ij}\gamma)^2}\right) \quad (3.9)$$

It should be noted that a sensor node can have communication with several nodes. Consequently, it may have several radio ranging likelihood functions. The number is dependent on the number of localized nodes that the sensor node can establish communication.

Fig. 3.6 Histogram distribution of the measured distance in (3.6)

For computation simplicity, the mapped distance is assumed to be normally distributed based on the results in Fig. 3.6 in this chapter to develop the LLA. Several works [36]-

[37],[56] have shown that the log-normal distribution could be a more accurate representation of the RSSI mapped distance if the RSSI measurement is normally distributed. In Chapter 5, the radio ranging likelihood function will be reformulated for the enhanced likelihood localization algorithm (ELLA).

3.2.3 Overall Likelihood Function and Position Estimation

A) Single-Direction Method

Let $\mathbf{J}_i$ represents all the localized nodes that can communicate to the unknown node i . As the deployment and RSSI ranging measurements are independent, the overall likelihood function $\bar{L}$ can be obtained by multiplying all the likelihood functions (3.2) and (3.9):

$$\begin{aligned}\bar{L}(\Phi_i) &= L_d(\Phi_i) \times \prod_{j \in \mathbf{J}_i} L_{ij}(\Phi_i) \\ &= \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{(x_i - x_{di})^2}{\sigma_x^2} + \frac{(y_i - y_{di})^2}{\sigma_y^2}\right)\right) \times \prod_{j \in \mathbf{J}_i} \frac{1}{\sqrt{2\pi}\tilde{d}_{ij}\gamma} \exp\left(-\frac{(\delta(\Phi_i, \hat{\Phi}_j) - \tilde{d}_{ij})^2}{2(\tilde{d}_{ij}\gamma)^2}\right)\end{aligned}\tag{3.10}$$

The position $\hat{\Phi}_i$ that minimizes (3.10) yields the maximum likelihood estimate. Taking natural logarithm of (3.10) yields:

$$\ln(\bar{L}(\Phi_i)) = C + S(\Phi_i)\tag{3.11}$$

where $C = -\ln(2\pi\sigma_x\sigma_y) - \sum_{j \in \mathbf{J}_i} \ln(\sqrt{2\pi}\tilde{d}_{ij}\gamma)$ is a constant independent of Φ_i , and the function S

is defined as:

$$S(\Phi_i) = \frac{(x_i - x_{di})^2}{\sigma_x^2} + \frac{(y_i - y_{di})^2}{\sigma_y^2} + \sum_{j \in \mathbf{J}_i} \frac{(\delta(\Phi_i, \hat{\Phi}_j) - \tilde{d}_{ij})^2}{(\tilde{d}_{ij}\gamma)^2} \quad (3.12)$$

To find the optimum position $\hat{\Phi}_i$ that maximizes $\bar{L}_i(\Phi_i)$, the function S must be minimized. Therefore, the localization problem becomes a least square optimization problem. This problem can be solved by using various optimization methods such as Gauss-Newton, genetic algorithm (GA), particle swarm optimization (PSO) etc. The objective function S is generally multi-modal, when the measurement error is large. The likelihood functions based on individual measurements have peak values at different locations due to the large measurement error. This leads to an overall multi-modal likelihood function. Typical local optimization algorithms as Gauss-Newton or the algorithm of Hooke and Jeeves usually converge to a local minimum that is nearest to the initial guess. More detailed discussion on the optimization will be discussed in Chapter 4.

In the following simulation study, the Gauss-Newton scheme is used due to its simple implementation.

B) Bi-Direction Method

In the single-direction method, an unknown node can only use the RSSI measurements from the sensor nodes that were deployed earlier. To fully utilize the RSSI information, the bi-direction method is proposed and investigated in this section. In some applications,

several of the deployed nodes may have knowledge of their real positions during the deployment. For example, the sensor node could be deployed at a land mark position that has known position. It is also possible that the nodes are deployed along a close path with the last node returns back to the first position. We term this sensor node as a landmark node. When a landmark node is encountered, the location of the previously deployed nodes could be updated by using the landmark node position and compute the likelihood function from the landmark node backward to the first node. The computation is similar to the localization procedure in the single-direction method. As the forward and backward likelihood functions $\bar{L}_f(\Phi_i)$ and $\bar{L}_b(\Phi_i)$ are independent, the overall likelihood function $\bar{L}(\Phi_i)$ can be obtained as their products:

$$\bar{L}(\Phi_i) = \bar{L}_f(\Phi_i) \times \bar{L}_b(\Phi_i) \quad (3.13)$$

Maximizing (3.13) is equivalent to minimizing the following sum of squares function:

$$S(\Phi_i) = \frac{(x_i - x_{di})^2}{\sigma_x^2} + \frac{(y_i - y_{di})^2}{\sigma_y^2} + \frac{(x_i - \bar{x}_{di})^2}{\bar{\sigma}_x^2} + \frac{(y_i - \bar{y}_{di})^2}{\bar{\sigma}_y^2} + \sum_{j \in \mathbf{J}_i} \frac{(\delta(\Phi_i, \hat{\Phi}_j) - \tilde{d}_{ij})^2}{(\tilde{d}_{ij}\gamma)^2} \quad (3.14)$$

where $(\bar{x}_{di}, \bar{y}_{di})$ is the backward deployment position, $\bar{\sigma}_x^2$ and $\bar{\sigma}_y^2$ are the x and y direction variance respectively; $\hat{\Phi}_j$ is the sensor node j 's forward estimated position if j is deployed before i . Otherwise, $\hat{\Phi}_j$ is j 's backward estimated position. It is noted that $\mathbf{J}_i$ in (3.14) generally contains more elements than in (3.12) as it includes the nodes deployed after i .

In the next section, it will be shown that the bi-direction method is more accurate than the single direction method as it uses more measured information to constrain the location estimation.

3.3 Evaluation of the localization Algorithm

In this section, the proposed algorithm is evaluated based on two network topologies. The first topology is a sparse network with uniformly distributed nodes on a closed circular path. The second topology is a dense network whose nodes are placed randomly with a uniform distribution within a square region. The performance metric is the average localization error, i.e. the average difference between the estimated and real positions of each node in the network. Moreover, it is assumed that the maximum communication range is 25 units.

3.3.1 *Sparse Network*

In this section, the simulation study for a sparse network is conducted. It is assumed that there are 30 nodes placed uniformly along a closed circular path that has a diameter of 100 units. Fig. 3.7 shows examples of the localized network using the single- and bi-direction methods. In these figures, the dash line is the deployment path and the circular dots represent the real positions of the sensor nodes. The square markers denote the estimated position using the LLA, and the crosses represent the estimated positions from the pedometer. From these results, it is observed that the proposed method improves the localization performance as compared to those using only the pedometer. Moreover, the bi-direction method is superior to the single-direction method. Fig. 3.8 summarizes the results from Fig. 3.7 by plotting the average localization error of each sensor node. The

data is obtained from the average of 100 sets of simulation trials. The figure shows that there are propagation errors if only the pedometer is used. Consequently, the localization accuracy decreases from the first node to the last node. Using the proposed approach, the propagation error is mitigated. This is the consequence of utilizing available RSSI information by the proposed method. For the bi-direction method, it achieves the localization accuracy more tightly than the single direction method.

Fig.3.8 also plots an upper bound of each sensor node's root mean square (rms) localization error of the single-direction method. The upper error bound is derived in the following way.

Let $l(\Phi_i) = \ln(\bar{L}(\Phi_i))$ be the log likelihood function of node i 's position. The covariance of the maximum likelihood estimator $\hat{\Phi}_i$ based on its overall likelihood function $\bar{L}(\Phi_i)$ is approximated as its asymptotic covariance,

$$\begin{aligned} \text{cov}_{\bar{L}}(\hat{\Phi}_i) &\approx -[E(\nabla_{\Phi_i} (\nabla_{\Phi_i} l(\Phi_i))^T)^{-1}]^{-1} \\ &= \begin{bmatrix} \frac{1}{\sigma_x^2} + \sum_{j \in \mathbf{J}_i} \frac{(x_i - x_j)^2}{\gamma^2 \cdot d_{ij}^4} & \sum_{j \in \mathbf{J}_i} \frac{(x_i - x_j)(y_i - y_j)}{\gamma^2 \cdot d_{ij}^4} \\ \sum_{j \in \mathbf{J}_i} \frac{(x_i - x_j)(y_i - y_j)}{\gamma^2 \cdot d_{ij}^4} & \frac{1}{\sigma_y^2} + \sum_{j \in \mathbf{J}_i} \frac{(y_i - y_j)^2}{\gamma^2 \cdot d_{ij}^4} \end{bmatrix}^{-1} \end{aligned} \quad (3.15)$$

Because $\bar{L}(\Phi_i)$ does not consider the localization error of its neighboring nodes, $\text{cov}_{\bar{L}}(\hat{\Phi}_i)$ is a lower bound on the covariance matrix of $\hat{\Phi}_i$. The upper bound of $\hat{\Phi}_i$'s covariance matrix can be obtained with assumption that the covariance matrix of all its neighboring nodes' estimated positions are the same as the last deployed node's upper bound, i.e.

$$\text{cov}_{upper}(\hat{\Phi}_i) = \text{cov}_{upper}(\hat{\Phi}_{i-1}) + \text{cov}_L(\hat{\Phi}_i) \quad (3.16)$$

Since the localization error of the neighboring nodes' estimated positions are generally smaller same as the last deployed node, $\text{cov}_{upper}(\hat{\Phi}_i)$ is an upper bound on its covariance matrix.

The upper bound on rms localization error in the figure is determined as:

$$\text{Error}_{upper}(i) = \sqrt{(\text{cov}_{upper}(\hat{\Phi}_{i-1}))_{1,1} + (\text{cov}_{upper}(\hat{\Phi}_{i-1}))_{2,2}} \quad (3.17)$$

(a)

(b)

Fig. 3.7 Localization in sparse network with
(a) single-direction method (b) bi-direction method

Fig. 3.8 Localization error for each sensor node

Fig. 3.9 shows the localization error under different pedometer and range error factors for the single and bi-direction methods. In real applications, the pedometer and range error factors can be obtained before the deployment by doing similar measurements as in Fig 3.3 and Fig. 3.5. The data points represent the average results of 100 simulation sets. The solid line represents the pedometer based localization result. From the figure, it is observed that the bi-direction method has less localization errors than the single-direction method. Furthermore, the position estimation is more accurate if the pedometer and range measurements have better accuracy. When the radio range error factor is small, i.e. the RSSI range measurement is relatively accurate; the LLA localization is much more accurate than the pedometer localization. For example, when the range error factor is 0.1, the localization error of the pedometer localization is 60% higher than the single-direction method and double the bi-direction method. However, when the range error factor is larger than 0.20, the single-direction method yields little advantage as the RSSI provides poor constraints on the localization. For the bi-direction method, it has better performance than the single-direction method. The bi-direction method with a range error factor of 0.2 can achieve the same localization accuracy as the single-direction method with a range error factor of 0.1.

Fig. 3.9 Comparison of single- and bi-direction methods for sparse network

3.3.2 Dense Network

In simulating the dense network localization, it is assumed that 100 nodes are being placed within a 100 units x 100 units sensor field. The sensor field is equally partitioned to 10 x 10 squares and the DA uses a zigzag path to deploy a sensor randomly in each of the square. Similar to the study as in section 3.3.1, the communication range is 25m. On average, each node has 6 neighboring nodes in single- direction method and 12 in the bi-direction method.

Fig. 3.10 shows one such sample result of the localized network for the single- and bi-direction methods. From the result, it is observed that the proposed methods track the sensor nodes' positions tightly; with the range error factor of 0.1 and pedometer error factor of 0.15, the distance between estimated and real positions is within 3 units.

Fig. 3.11 compares the average localization error between the pedometer and the LLA for each sensor node based on 100 trials. The results show that there is no obvious propagation error in both single- and bi-direction methods. Moreover, the bi-direction method has slightly less localization error.

(a)

(b)

Fig. 3.10 Localization in dense network

(a) single-direction method (b) bi-direction method

‘●’ unknown nodes’ real positions; ‘★’ starting point; ‘○’ estimated positions; ‘---’

deployment path

Fig. 3.12 shows the corresponding localization error under different pedometer and range error factors. Similar to the sparse network, it also follows the trend that the more accurate the pedometer and range measurements, the better the localization. When the range error factor is small, it is observed that the LLA performs well in dense networks.

This is due to the fact that the unknown node has more localized sensors within its range, which leads to more constraints yielding a better estimation. As in the sparse network, the bi-direction method yields better localization result than the single-direction method. However, the improvement is less significant as compared to the sparse network. Thus, the single-direction method may still be attractive for a dense network as it requires only halve the computation and communication requirements of the bi-direction method.

Fig. 3.11 Localization error for each sensor node in dense network

Fig. 3.12 Comparison of single and bi-direction methods for dense network

3.3.3 Comparison with other localization algorithms

Simulation studies have also been conducted for the DV-distance localization algorithm [24] using the same network topologies for comparison with the proposed algorithm. DV-distance is a very popular range-based distributed localization algorithm for WSNs with low anchor nodes density [30],[32]. In DV-distance, it usually requires 5% ~20% of the sensor nodes to be anchor nodes. Fig. 3.13 shows the results for the sparse network. In this case, it is assumed that 20% of the sensor nodes are anchor nodes for the DV-distance algorithm, and the bi-direction method is used for the LLA. The figure shows that the DV-distance approach yields an error of more than 18 units with $p=0.15$ while the LLA algorithm has an error of less than 4 units. Fig. 3.14 shows the

results for the dense network topology. In this case, it is assumed that 10% of the sensor nodes are anchor nodes for the DV-distance algorithm, and single-direction method was used for the LLA. From the figure, we observed that the LLA yields better result when the pedometer error factor is 0.15 or smaller. For a large pedometer error factor of 0.2, the LLA can still achieve comparable localization accuracy as DV-distance.

Fig. 3.13 Localization performance of LLA and DV-distance for a sparse network

3.4 Conclusion

In this chapter, a new distributed localization algorithm (i.e. LLA) is presented. It uses the deployment information and RSSI measurements to construct a likelihood function to localize the sensor nodes. Unlike most existing methods which require good network connectivity and are not suitable for sparse networks, the proposed approach can work

well in both dense and sparse sensor networks. Furthermore, it does not require the system to have anchor nodes to perform localization. The proposed system has been evaluated with simulation studies in this chapter. The results have demonstrated that it has good performance. When the proposed approach is benchmark with the DV-distance localization algorithm, it reduces the error from more than 18 units to less than 4 units.

In the next chapter, a microcontroller based realization of the proposed LLA schemes for a sparse WSN will be presented. The performance will be evaluated in experimental environment.

Fig. 3.14 Localization performance of LLA and DV-distance for a dense network

Chapter 4

Sensor Node Design and Implementation of Distributed Likelihood Localization Algorithm

4.1 Introduction

Many of the existing WSN localization works focus on the theoretical development [28]-[33]. They have not considered the computational cost and the implementation issue. In practice, there is high constraint on the computational power and memory size. Consequently, many sophisticated optimization approaches will not be feasible.

In chapter 3, the distributed likelihood localization algorithm (LLA) for a WSN has been presented. The simulation results show that the proposed LLA scheme works well for both sparse and dense networks. To determine the best estimate of the sensor node location, Chapter 3 shows that it is required to solve a nonlinear equation to find the best result. One possible approach to solve the problem is the Gauss-Newton Algorithm (GNA). However, the GNA is a local optimization method and it does not guarantee global convergence. An alternative is to use a global optimizer such as the Particle Swarm Optimization (PSO). In this chapter, both GNA and PSO have been studied and implemented using the same platform for the proposed LLA. Their effectiveness in

searching the optimal solutions under different operating conditions will also be investigated. This chapter also presents the hardware design and implementation. The performance will be evaluated in outdoor experimental environment. From the experimental results, the localization accuracy is shown to be similar to its simulation counterpart.

4.2 Sensor Node Architecture and Implementation

Fig. 4.1 shows the block diagram of the sensor node architecture developed in this study. It has five major components namely the RF system, RSSI to distance translator, GNA/PSO optimizer, transmission scheduler and data memory.

Fig. 4.1 Sensor node block diagram

As mentioned in Section 3.2, once the first sensor node is deployed with a known location, it starts to broadcast its location. For each subsequent deployment of unknown

node i , the sensor node receives its deployment information Φ_{di} through its RF system from the pedometer/compass system. Its transmission scheduler will then request its neighboring nodes to send their beacon messages that contain their estimated positions. Besides receiving the beacon message, the RSSI to distance translator also measures the RSSI values of the received beacons and translates them to a distance $\tilde{D}_i$. The GNA/PSO optimizer will then determine the sensor node's estimated position $\hat{\Phi}_i$, by combining the deployment and inter-node distances information using (3.10). The transmission scheduler responds to other nodes' requests and sends the sensor node's beacon package. The details of the various subsystems are elaborated in the following sections.

A. RF System

The RF system is used to receive beacon packages from the neighboring sensor nodes, and stride/heading information from the pedometer, as well as transmitting its beacon packages. The RF module (i.e. XBee ZNet 2.5 OEM) used in this study is able to measure the RSSI. The RF module uses the ZigBee communication protocols suite and operates at 2.4 GHZ frequency band. The transmitting antennas are quarter wavelength dipole type with omni-directional radiation patterns.

B. Transmission Scheduler

The transmission scheduler implements the MAC (Media Access Control) protocol to avoid data collisions. For ease of experiment, a simple polling process is used. After a sensor node has been deployed and received its pedometer information, it will begin to

poll its nearby sensor nodes to send its beacon packages. After the sensor has been successfully localized, it will switch to the listening mode, and wait for polling from other nodes.

C. RSSI to Distance Translator

The RSSI to distance translator implements a function **RtoD()** that contains the following equation:

$$\tilde{d}_j = c \cdot 10^{\frac{P_0 - P_j}{10\alpha}} \quad (4.1)$$

where $\tilde{d}_j$ is the measured distance; P_j is the received RSSI value; P_o is the received RSSI value for a 1m reference distance, α is the path-loss exponent and c is the bias reduction correction factor. P_o , α and c are measured and calibrated before the deployment. Their typical values are -30, 3.5 and 0.95 respectively.

In general, the RSSI measurements are subjected to burst interferences from various noises. To improve the estimation, the average RSSI measurements of 10 beacons are used to infer the distance. Table 4.1 shows the pseudo-code of *RSSI to Distance Translator*.

Table 4.1 Pseudo-code of *RSSI to Distance Translator*.

Beacon package contains:

j : sender's node ID

$\hat{\Phi}$: sender's estimated position

P : RSSI value of the package

Step 1: Poll a neighboring node, and extract its estimated position.

$$\hat{\Phi}_j = \hat{\Phi};$$

Step 2: Poll the node 10 times and compute the average RSSI value.

$$P_j = 0;$$

for $i = 1; i \leq 10; i++$

poll node j ;

$$P_j = P_j + P;$$

end

$$P_j = P_j / 10;$$

Step 3: Determine the distance

$$\tilde{d}_j = \mathbf{RtoD}(P_j); \quad \% \text{ Equation (4.1)}$$

Step 4: Go to **step 1** if there are other neighboring nodes.

D. GNA/PSO optimizer

D.1. GNA

The Gauss-Newton algorithm is well known for solving non-linear least square estimation problems [70]-[72]. It is an iterative algorithm and requires the user to provide an initial guess of its solution. Given m functions f_i ($i = 1, \dots, m$) of n

variables $\beta = (\beta_1, \beta_2, \dots, \beta_n)$, with $m \geq n$, the Gauss-Newton algorithm can be used to find the minimum of the sum of squares:

$$S(\beta) = \sum_{i=1}^m f_i^2(\beta) \quad (4.2)$$

Starting with an initial guess $\beta[0]$, the method proceeds by the iterations:

$$\beta[k+1] = \beta[k] + \Delta_k \quad (4.3)$$

with the increment Δ_k satisfying the normal equation:

$$(\mathbf{J}_f^T \mathbf{J}_f) \Delta_k = -\mathbf{J}_f^T \mathbf{f} \quad (4.4)$$

where $\mathbf{f}$ is the vector of functions f_i , and $\mathbf{J}_f$ is the Jacobin matrix of $\mathbf{f}$ with respect to $\beta[k]$.

For the localization problem, the estimated deployment position Φ_{di} is used as the initial guess. Table 4.2 shows the pseudo-code of the GNA, with a simple line-search procedure. From the pseudo-code, it is observed that the algorithm involves matrix inversion in step 3. The iteration will fail if the matrix $\mathbf{J}_f^T \mathbf{J}_f$ is singular. Theoretically, this matrix will never become singular in this localization problem. However, the matrix could be nearly singular due to the limited word length of the microcontroller, and occasionally may not converge to meet the stopping criterion. This issue will be discussed in the later section.

Table 4.2 Pseudo-code of GNA.

Step 1: Initialization

$k=0$;

$\Phi_i[k] = \Phi_{di}$;

Step 2: Find necessary matrixes for GNA

$$\mathbf{f}(\Phi_i) = \begin{bmatrix} \frac{x_i - x_{di}}{\sigma_x} & \frac{y_i - y_{di}}{\sigma_y} & \frac{\delta(\Phi_i, \hat{\Phi}_{j1})}{\tilde{d}_{ij1} \cdot \gamma} - \frac{1}{\gamma} & \dots & \frac{\delta(\Phi_i, \hat{\Phi}_{jn})}{\tilde{d}_{ijn} \cdot \gamma} - \frac{1}{\gamma} \end{bmatrix};$$

$\mathbf{J}_f = \text{Jacobin}(\mathbf{f}, \Phi_i)$

$$= \begin{bmatrix} \frac{1}{\sigma_x} & 0 \\ 0 & \frac{1}{\sigma_y} \\ \frac{x_i - \hat{x}_{j1}}{\tilde{d}_{ij1} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_{j1})} & \frac{y_i - \hat{y}_{j1}}{\tilde{d}_{ij1} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_{j1})} \\ \vdots & \vdots \\ \frac{x_i - \hat{x}_{jn}}{\tilde{d}_{ijn} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_{jn})} & \frac{y_i - \hat{y}_{jn}}{\tilde{d}_{ijn} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_{jn})} \end{bmatrix};$$

Step 3: Determine the incremental direction

$$\psi = (\mathbf{J}_f^T \mathbf{J}_f)^{-1};$$

$$\Delta_k = -\psi \mathbf{J}_f^T \mathbf{f};$$

Step 4: Line search

$$\alpha = 1;$$

while $S(\Phi_i[k] + \alpha \Delta_k) \geq S(\Phi_i[k])$

$$\alpha = \alpha / 2;$$

end

$$\alpha_k = \alpha;$$

Step 5: Update

$$\Phi_i[k+1] = \Phi_i[k] + \alpha_k \Delta_k;$$

Step 6: Check for stopping criterion

$$\delta(\Phi_i[k+1], \Phi_i[k]) < 0.01$$

Update iteration index k and go to step 2 if the stopping criterion is not satisfied.

Table 4.3 Pseudo-code of PSO.

Step 1: Initialization

$k=0$;

Randomly generate:

A population of N particles, $\hat{\Phi}_i[k]$;

A initial set of velocities, $\mathbf{v}_i[k]$;

$pbest$ and $gbest$. $i=1, 2, \dots, N$

Step 2: Update $pbest$ and $gbest$.

for $i=1; i \leq N; i++$

 if ($S(\hat{\Phi}_i) < S(pbest_i)$) $pbest_i = \hat{\Phi}_i$;

 if ($S(pbest_i) < S(gbest)$) $gbest = pbest_i$

end

Step 3: Update velocities and particles

$\mathbf{v}_i[k+1] = \omega \mathbf{v}_i[k] + c_1 \lambda_1 (pbest_i - \hat{\Phi}_i[k]) + c_2 \lambda_2 (gbest - \hat{\Phi}_i[k])$

$\hat{\Phi}_i[k+1] = \hat{\Phi}_i[k] + \mathbf{v}_i[k+1]$

where ω is the inertia weight; λ_1 and λ_2 are random numbers between 0 and 1; The constants c_1 and c_2 are also known as the acceleration constants. They determine the speed a particle would accelerate towards the $pbest_i$ and the $gbest$.

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{k_{\max}} \times k$$

where $k_{\max}$ is the maximum iteration number.

Step 4: Check for stopping criterion

In this study, the stopping criterion is defined as $gbest$ has not been changed for the past 50 generations.

Update iteration index k and go to step 2 if the stopping criterion is not satisfied.

Step 5: The resulting $gbest$ gives the desired estimated position of the deployed sensor.

$N = 10$; $c_1 = 2$; $c_2 = 2$ are used in the system evaluation and experiment.

D.2. Particle Swarm Optimization

In this section, the Particle Swarm Optimization (PSO) [73] is used for the global optimizer for searching the sensor node positions. PSO has been used in various applications such as robotics, antenna design [74]-[79] etc. Similar to the other heuristic algorithms such as the genetic algorithm [80]-[81], the PSO is initialized with a population of random solutions, also called particles. Each particle keeps track of its best fitness solution (also known as *pbest*). Moreover, the global best fitness solution (also known as *gbest*) is also stored in the optimizer. In each time step, the PSO optimizer changes the velocity such that the particle will move towards its *pbest* and *gbest* with a random weight. In this investigation, each particle consists of 2 members, representing the x and y coordinates of the sensor node. Table 4.3 shows the pseudo code that has been used for the implementation.

In this study, population size of 10 is chosen for the PSO, because it requires the least computation time while it is able to achieve the same optimum points as larger population sizes, with the stopping criterion as the *gbest* has not been changed for the past 50 generations. Besides the population size, the acceleration constants $c1$ and $c2$ (Table 4.3) are the other parameters that affect the searching for optimum. They are the cognition and social components respectively that determine the speed a particle would accelerate towards the *pbest* and the *gbest*. From simulation study conducted with different pairs of acceleration constants, it is concluded that larger acceleration constants result in longer convergent time. However, the localization error is independent of the acceleration constants, when both parameters are larger than 1.6. For this study, we select $c1=2$, $c2=2$.

4.3 System Evaluation in Laboratory Environment

The proposed GNA and the PSO algorithms have been implemented on a microcontroller (Microchip PIC18LF4620). The microcontroller has 64k bytes Flash program memory and 3968 bytes SRAM data memory. It runs at a clock rate of 40 MHz. For ease of development, both algorithms have been coded in C language using floating point format. The C programs are then compiled using the Microchip MPLAB C18 compiler and download to the Flash memory through the MPLAB ICD2 debugger. The codes for transmission scheduler and RSSI to distance translator together occupy about 4k bytes program memory. For the GNA and PSO optimizers, the code requires about 8k and 12k bytes memory respectively.

A. System Performance in Laboratory Study

For ease of determining the effect of pedometer and ranging on measurement accuracy, we evaluate the performance of the two optimization approaches under laboratory environment in the following discussion. For this study, both the transmission scheduler and RSSI to distance translator programs are by-passed. The necessary information such as pedometer and RSSI distance measurements are randomly generated from their expected distributions.

The experimental network is a sparse network as shown in Fig. 4.2. It consists of 31 nodes being placed randomly along the path. In this example, a 0.2 pedometer error factor and 0.15 range error factor have been used. Their values have been chosen based on the typical range of experimental data. The communication range is assumed to be 60 units. Fig. 4.2 shows the localized network using single-direction method, and Fig. 4.3 shows the localized result using bi-direction method with both GNA and PSO optimization approaches. From these results, it is observed that localization by PSO is almost identical to the GNA.

Fig. 4.4 summarizes the localization error under different range and pedometer error factors for both single- and bi-direction methods. The localization results with the PSO are very similar to the GNA. The difference for each data point is less than 0.1 unit if $p < 0.3$. In summary, both GNA and PSO are able to search the same optimum with the same measurements. The slight differences are due to termination at different positions when stopping criterion is met, as the two algorithms may converge differently.

(a) GNA

(b) PSO

Fig. 4.2 System evaluation of a sparse network with single direction approach

(a) GNA

(b) PSO

Fig. 4.3 System evaluation of a sparse network with bi-direction approach

(a) GNA

(b) PSO

Fig. 4.4 Average localization error under different measurement accuracy

Fig. 4.5 Average localization error with $p = 0.35$

Fig. 4.5 shows the average localization error with $p=0.35$. It is observed that there are disparities between the localization results of these two approaches with large pedometer error ($p > 0.3$). From the figure, it is observed that the PSO generally has better localization result. This is because GNA sometimes converges to local minima due to its poor initial guesses.

B. Evaluation of Computational Costs

In this section, we study the computational costs of the two optimization approaches under different system parameters, such as pedometer error factor and number of neighboring nodes. Similar to the last section, the pedometer and RSSI distance measurements are randomly generated from their expected distribution. The number of neighboring nodes can be varied by adjusting the communication range. With a

communication range of 60 units, each sensor node has 2.5 neighboring nodes on average for single-direction method and 5 neighboring nodes for bi-direction method. From the evaluation, GNA generally requires less execution time than PSO. Moreover, GNA spends most of the execution time to compute the Hessian matrix of the objective function and about 15% of the execution time is used for objective function evaluations.

Table 4.4 compares the average execution time needed by the two algorithms to localize a single node for different network density. Here, the pedometer error factor is 0.2. From Table 4.4, it is observed that the PSO requires about twice the computational time of the GNA. Thus, local optimization method such as GNA is preferred for small pedometer error. From the Table, it is also observed that both the approaches need more computation time as the number of neighboring nodes increases.

Table 4.4 Average execution time for different number of neighboring nodes

(pedometer error factor = 0.2)

Number of neighboring nodes	GNA [s]	PSO [s]	Time ratio GNA/PSO
1	0.283	0.609	0.465
2	0.384	0.843	0.456
3	0.512	1.071	0.478
4	0.598	1.302	0.459
5	0.694	1.533	0.453
6	0.785	1.764	0.445
7	0.983	1.983	0.496

Table 4.5 shows how the pedometer accuracy affects the computational requirement for the two approaches. As the pedometer error factor becomes larger, the GNA requires more execution time to achieve the required accuracy. This is because the initial guess will be far from the optimal position. On the other hand, the PSO execution time is almost constant. This makes the PSO attractive for system that has large pedometer error factor.

Table 4.5 Average execution time for different pedometer error factor

Pedometer error factor	GNA [s]	PSO [s]	Time ratio GNA/PSO
0.05	0.297	0.944	0.315
0.10	0.330	0.902	0.366
0.15	0.378	0.986	0.383
0.20	0.449	1.017	0.441
0.25	0.528	0.896	0.589
0.30	0.556	0.989	0.562

Note that, after a minimum has been found the PSO also includes 50 additional generations to examine if the *gbest* would further change to ensure the global convergence. For the fairness of the comparison, the execution times in Table 4.4 and 4.5 do not include the 50 *gbest* checking generations for the PSO.

To show the performance trend, Fig. 4.6 plots the average objective function values versus the number of iterations or generations for these two algorithms. From the figure, it is observed that the GNA has faster convergent rate. This result is expected as the GNA starts from a position closer to the minimum, and it determines the descending direction directly by approximating the Hessian matrix ($\mathbf{J}_f^T \mathbf{J}_f$) of the objective function; while the PSO selects a better estimate through comparing the objective function values. Therefore, the GNA generally requires less iteration to converge to a minimum. However, the GNA may converge to a local minimum if the starting point is too far from the optimal point.

Fig. 4.6 Average objective function value versus number of iterations/generations

C. Stability Assessment of GNA

As mentioned in section 4.2, the GNA involves matrix inversion during its iteration. The iteration will fail if the matrix $\mathbf{J}_f^T \mathbf{J}_f$ is singular. The determinant of the matrix can be obtained as:

$$\begin{aligned}
|\mathbf{J}_f^T \mathbf{J}_f| = & \underbrace{\frac{1}{\sigma_x^2} \frac{1}{\sigma_y^2}}_{\text{first term}} \\
& + \underbrace{\frac{1}{\sigma_x^2} \left(\sum_{j \in \mathbf{J}_j} \frac{(y_i - \hat{y}_j)^2}{(\tilde{d}_{ij} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_j))^2} \right) + \frac{1}{\sigma_y^2} \left(\sum_{j \in \mathbf{J}_j} \frac{(x_i - \hat{x}_j)^2}{(\tilde{d}_{ij} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_j))^2} \right)}_{\text{second term}} \\
& + \underbrace{\left(\sum_{j \in \mathbf{J}_j} \frac{(x_i - \hat{x}_j)^2}{(\tilde{d}_{ij} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_j))^2} \right) \left(\sum_{j \in \mathbf{J}_j} \frac{(y_i - \hat{y}_j)^2}{(\tilde{d}_{ij} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_j))^2} \right) - \left(\sum_{j \in \mathbf{J}_j} \frac{(x_i - \hat{x}_j)(y_i - \hat{y}_j)}{(\tilde{d}_{ij} \cdot \gamma \cdot \delta(\Phi_i, \hat{\Phi}_j))^2} \right)^2}_{\text{third term}}
\end{aligned} \tag{4.5}$$

In (4.5), the first and second terms are always positive. The third term is zero when the current position estimation and the estimated positions of its neighboring nodes are collinear. Thus, $\mathbf{J}_f^T \mathbf{J}_f$ is non-singular in general but it could be nearly singular due to the limited precision of the microcontroller when the algorithm is implemented using a microcontroller. In such case, the inversion leads to an extremely large position increment Δ_k , and the estimated position of the next iteration may be out of the sensor field's boundary. Thus the algorithm becomes divergent. This happens when the pedometer error factor is large that makes the first 2 terms of (4.5) close to zero and the neighboring nodes are happen to be collinear.

Based on 1000 localization trials of the network depicted in Fig. 4.2, Table 4.6 shows the probability of instability. From Table 4.6, it shows that if the pedometer error factor is larger than 0.25 and is also larger than the range error factor, the algorithm may become unstable. Consequently, the estimated position does not converge to meet the stopping criterion.

Table 4.6 Probability of GNA being unstable, %

$\begin{matrix} r \\ p \end{matrix}$	0.05	0.1	0.15	0.2	0.25	0.3
0.05	0	0	0	0	0	0
0.1	0	0	0	0	0	0
0.15	0	0	0	0	0	0
0.2	0	0	0	0	0	0
0.25	0.2	0.1	0	0	0	0

In summary, it is noticed that both GNA and PSO have their advantages. When the accuracy of pedometer is high, local optimization is sufficient and these two methods have very similar localization result. In this case, GNA is preferred as it requires less computation and execution time. On the other hand, the PSO in this study is robust as it always returns an estimated position successfully. The GNA involves a matrix inversion during its iteration. Consequently, the optimum result will not be feasible when the matrix becomes singular or near singular due to the precision of the microcontroller. Though additional methods can be used to avoid the singular matrix for the GNA, this may significantly affect the convergent rate. For large pedometer error, the GNA may converge to a local minima and yield larger localization error than PSO. For systems that have large pedometer error factor, PSO shall be used for its robustness. Otherwise, GNA would be a better choice for its simplicity and low computational costs.

4.4 Outdoor experiment

The experimental measurement has been conducted surround a lake and a park located in the university campus. The network consists of 31 sensor nodes. Each sensor node is equipped with a RF module as mentioned in Section 4.2, which is able to measure the RSSI. Before the deployment, experiment has been conducted to calibrate the parameters used in the path-loss equation (4.1) by measuring the RSSI with respect to reference distance. From the calibration, it is determined that the range error factor $\gamma = 0.21$. For the pedometer, it consists of a 3-axes accelerometer based stride counter and an electronic compass (BASIC STAMP HM55B). Pre-deployment measurement shows that the pedometer error factor $p = 0.22$.

The experimental results are shown in Figs. 4.7 and 4.8. In the figures, the real position of the sensor nodes, the estimated positions from the pedometer as well as the single and bi-direction localization approaches have been marked on the satellite image map. From the figures, the results obtained from using the two optimization approaches are very close. Fig. 4.9 summarizes the errors of the localization process at every node along the deployment path. From the figure, it can be determined that the average errors for the single and bi-direction methods are 16.43 and 9.51 meters respectively. The average error from using the pedometer only (without using RSSI) is 19.5745 meters. The good performance of the bi-direction approach is expected as it makes use of more RSSI values available for processing than the single direction approach.

(a) GNA

(b) PSO

Fig. 4.7 Experimental results for single-direction method

($\times$ – pedometer, $\bullet$ – real position, $\blacksquare$ – LLA)

(a) GNA

(b) PSO

Fig. 4.8 Experimental results for bi-direction method

(x – pedometer, ● – real position, ■ – LLA)

Fig. 4.9: Experiment error at every node

(x – pedometer, o – single direction, ■ – bi-direction)

Table 4.7 PSO versus GNA

	PSO in outdoor experiment	PSO in laboratory Simulation	GNA in outdoor experiment	GNA in laboratory Simulation
Single-direction localization error	16.42m	14.52m	16.43m	14.51m
Bi-direction localization error	9.55m	9.81m	9.51m	9.81m
Execution time for single-direction	0.91s	0.92s	0.51s	0.48s
execution time for bi-direction	1.53s	1.51s	0.82s	0.72s

Table 4.7 shows the performance of the two methods. From the table, it is observed that the localization results of PSO and GNA are only slightly different. This is consistent with the earlier findings, i.e. both methods are able to find the same optimum value with the same given measurements. Moreover, it is observed that the GNA requires about half the execution time of the PSO.

4.5 Conclusion

In this chapter, a microcontroller has been used to implement the Gauss Newton algorithm and the particle swarm optimization approach for optimizing the sensor node localization with the LLA scheme proposed in Chapter 3. The performance of the proposed approaches have been evaluated and validated with experimental results. The results have shown that both of them have similar performance with improved accuracy. Moreover, the GNA requires less computation and execution time than the PSO especially when accuracy of the pedometer is high. For large pedometer error, the GNA may converge to a local minimum. In such cases, the PSO is preferred for its robustness as it always generates unique position estimation

In the next chapter, the system model and localization approach are reformulated with an improved representation to achieve better localization. Two new localization schemes namely maximum likelihood estimation (MLE) and EKF (extended Kalman filter) are proposed and investigated. To benefit from the advantages of both schemes, an enhanced likelihood localization algorithm (ELLA) that hybridizes both schemes is also proposed.

Chapter 5

Enhanced Likelihood Localization Algorithm

5.1 Introduction

As discussed in chapters 3 and 4, the LLA utilizes the inter-node distance and deployment information to estimate an unknown node's position through a maximum likelihood estimation algorithm. To minimize the system requirement, the deployment likelihood function of an unknown node's position is simplified as uncorrelated X and Y direction projections of the DA measurement in chapter 3. In addition, the measured distance based on RSSI is assumed to be approximately Gaussian distributed.

Several works [36]-[37],[56] have shown that the log-normal distribution could be a more accurate representation of the measured distance. To improve the localization accuracy, the system model and localization are reformulated in this chapter and a maximum likelihood estimation (MLE) based scheme is proposed to estimate the sensor node's position. In the new system model, the range likelihood function is reformulated based on log-normal distribution of the RSSI measurement error and the deployment likelihood function is reformulated in polar coordinate. With the assumption that the sensor node's position is the same as the DA's location when it is being deployed, the localization problem can be viewed as a tracking problem for the DA. Thus, an extended Kalman filter (EKF) based scheme is also proposed in this chapter to estimate the sensor

nodes' positions through tracking the DA. Though the EKF based scheme has better localization accuracy when the number of neighboring nodes is small, it has the disadvantage that the estimators may not be optimal if the measurement or the state transition model is non-linear. The simulation results show that the MLE scheme has superior localization accuracy than the EKF scheme when the measurements are highly non-linear. To benefit from the advantages of both schemes, an enhanced likelihood localization algorithm (ELLA) that hybridizes both schemes is proposed. Simulations and experimental results show that the localization accuracy of ELLA is 20% better than LLA.

5.2 MLE based scheme

In this study, Φ and ϕ denote the sensor node position in Cartesian and local polar coordinate system respectively. Furthermore, let $\tilde{v}_i$ and $\tilde{\varphi}_i$ be the range and compass measurements between node i and its previous deployed node $i-1$, then $\tilde{v}_i \cdot \exp(j\tilde{\varphi}_i)$ will be the deployment position vector measured by the DA. The error of the compass carried by the DA is assumed normal distributed with a variance σ_φ^2 .

To obtain the deployment likelihood function, we set the local polar coordinate system with the estimated position $\hat{\Phi}_{i-1}$ of node $i-1$ as origin. Let the position of node i in this local polar coordinate system be $\phi_i = r_i \cdot \exp(j\theta_i)$. Then the deployment likelihood function can be formulated as:

$$L_d(\phi_i) = \frac{1}{2\pi\sigma_v\sigma_\varphi} \exp\left(-\frac{(r_i - \tilde{v}_i)^2}{2\sigma_v^2} - \frac{(\theta_i - \tilde{\varphi}_i)^2}{2\sigma_\varphi^2}\right) \quad (5.1)$$

where $\sigma_v = \tilde{v}_i \cdot p$ and p is the pedometer error factor as defined in section 3.2.1.

If node i can measure the RSSI P_{ij} from node j , P_{ij} and the distance d_{ij} between these two nodes have the following relationship based on the log-shadowing model:

$$P_{ij} = P_0 - 10\alpha \log(d_{ij}) + X_\sigma \quad (5.2)$$

where X_σ is a random variable and it is usually assumed to be Gaussian distributed with 0 mean and variance σ_R^2 .

From (5.2) with a specific distance d_{ij} , P_{ij} is also Gaussian distributed, i.e.:

$$P_{ij} \sim N(P_0 - 10\alpha \log(d_{ij}), \sigma_R^2) \quad (5.3)$$

Therefore, the probability of a given measured P_{ij} can be obtained as:

$$P(P_{ij}) = \frac{1}{\sqrt{2\pi}\sigma_R} \exp\left(-\frac{(P_{ij} - P_0 - 10\alpha \log(d_{ij}))^2}{2\sigma_R^2}\right) \quad (5.4)$$

By the estimation theory, the likelihood function of ϕ_i based on the measured P_{ij} is equal to the probability of P_{ij} , i.e.:

$$L_{rj}(\phi_i) = \frac{1}{\sqrt{2\pi}\sigma_R} \exp\left(-\frac{(P_0 + 10\alpha \log(\psi(\phi_i, \hat{\phi}_j)) - P_{ij})^2}{2\sigma_R^2}\right) \quad (5.5)$$

where $\psi(\phi_A, \phi_B) = \sqrt{(r_A^2 + r_B^2 - 2 \cdot r_A \cdot r_B \cos(\theta_A - \theta_B))}$ is the distance between positions ϕ_A and ϕ_B .

Let $\mathbf{J}_i = \{j: \text{node } j \text{ establishes direct communication with node } i\}$. As the deployment and RSSI measurements are independent, the overall likelihood function $\bar{L}$ can be obtained by multiplying all the likelihood functions (5.1) and (5.5):

$$\begin{aligned}\bar{L}(\phi_i) &= L_d(\phi_i) \times \prod_{j \in \mathbf{J}_i} L_{rj}(\phi_i) \\ &= \frac{1}{2\pi\sigma_v\sigma_\phi} \exp\left(-\frac{1}{2}\left(\frac{(r_i - \tilde{v}_i)^2}{\sigma_v^2} + \frac{(\theta_i - \tilde{\phi}_i)^2}{\sigma_\phi^2}\right)\right) \\ &\quad \times \prod_{j \in \mathbf{J}_i} \left(\frac{1}{\sqrt{2\pi}\sigma_R} \exp\left(-\frac{1}{2} \frac{(P_0 - 10\alpha \log(\psi(\phi_i, \hat{\phi}_j)) - P_{ij})^2}{\sigma_R^2}\right)\right)\end{aligned}\quad (5.6)$$

The estimator $\hat{\phi}_i$ that maximizes (5.6) is the maximum likelihood estimate. Its equivalent least square optimization problem can be formulated as follows:

$$\hat{\phi}_i = \arg \min_{\phi_i} \left(\left(\frac{r_i - \tilde{v}_i}{\sigma_v}\right)^2 + \left(\frac{\theta_i - \tilde{\phi}_i}{\sigma_\phi}\right)^2 + \sum_{j \in \mathbf{J}_i} \left(\frac{P_0 - 10\alpha \log(\psi(\phi_i, \hat{\phi}_j)) - P_{ij}}{\sigma_R}\right)^2 \right) \quad (5.7)$$

This least square optimization problem can be solved by several methods such as Gauss-Newton or PSO approach. In the following discussion, the Gauss-Newton scheme is used.

Similar to the discussion in section 3.2.3, when a landmark node is encountered, bi-direction method can be formulated to improve the localization accuracy. The backward estimated state vector of the previously deployed nodes could be updated by using the landmark node position and compute the likelihood function from the landmark node to the first node. With both the forward and backward estimated positions of its neighboring nodes, the bi-direction estimated position $\hat{\Phi}_i = (x_i, y_i)$ of an unknown node i can be found as:

$$\begin{aligned} \hat{\Phi}_i = \arg \min_{\mathbf{x}_i} & \left(\left(\frac{\delta(\Phi_i, \hat{\Phi}_{i-1}) - \tilde{v}_i}{\tilde{v}_i p} \right)^2 + \left(\frac{\mathcal{G}(\Phi_i, \hat{\Phi}_{i-1}) - \tilde{\varphi}_i}{\sigma_\varphi} \right)^2 + \left(\frac{\delta(\Phi_i, \hat{\Phi}_{i+1}) - \tilde{v}_i}{\tilde{v}_i p} \right)^2 \right. \\ & \left. + \left(\frac{\mathcal{G}(\Phi_i, \hat{\Phi}_{i+1}) - \tilde{\varphi}_i}{\sigma_\varphi} \right)^2 + \sum_{j \in \mathbf{J}_i} \left(\frac{P_0 - 10\alpha \log(\delta(\Phi_i, \hat{\Phi}_j)) - P_{ij}}{\sigma_R} \right)^2 \right) \end{aligned} \quad (5.8)$$

where $\tilde{v}_i \cdot \exp(j\tilde{\varphi}_i)$ is the deployment position vector measured from node $i+1$ to i measured by the DA; $\delta(\Phi_A, \Phi_B) = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$ was defined in section 3.2.2 and

$$\mathcal{G}(\Phi_A, \Phi_B) = \begin{cases} \arcsin\left(\frac{y_A - y_B}{\delta(\Phi_A, \Phi_B)}\right) & , \text{ if } (x_A - x_B) \geq 0 \\ \pi - \arcsin\left(\frac{y_A - y_B}{\delta(\Phi_A, \Phi_B)}\right) & , \text{ if } (x_A - x_B) < 0 \end{cases} \quad (5.9)$$

5.3 Extended Kalman Filter

With the assumption that the sensor node i 's position is the same as the DA's location when it is being deployed, the localization problem can be considered as a tracking problem of DA. As the relationship of the RSSI information to the target position is non-linear, an Extended Kalman Filter (EKF) [83] approach is proposed to solve this tracking problem. The unscented Kalman filter and particle filters [84]-[85] are also popular data fusion techniques for target tracking problems with non-linear models. However these techniques usually have high computation requirement and consequently they may not be suitable for resource limited WSNs.

Let $\Phi(i) = [x_i \ y_i]^T$ be the state vector of the DA when deploying sensor node i , the DA has the following motion model

$$\Phi(i+1) = \Phi(i) + \mathbf{u}(i+1) + \mathbf{w}(i+1) \quad (5.10)$$

where $\mathbf{u}(i+1)$ is the input control vector, and it is obtained from the DA's measurement:

$$\mathbf{u}(i+1) = \exp\left(-\frac{\sigma_\varphi^2}{2}\right) \cdot [\tilde{v}_{i+1} \cdot \cos(\tilde{\varphi}_{i+1}) \quad \tilde{v}_{i+1} \cdot \sin(\tilde{\varphi}_{i+1})]^T \quad (5.11)$$

Here $\exp(-\frac{\sigma_\varphi^2}{2})$ is the correction factor to reduce the bias of DA's measurement in Cartesian coordinate system. $\mathbf{w}(i+1)$ is the process noise, whose covariance matrix is $\mathbf{Q}(i+1)$. In Appendix B, it is shown that:

$$\mathbf{Q}(i+1) = \exp(-\sigma_\varphi^2) \cdot \begin{bmatrix} \sigma_v^2 \cos^2(\tilde{\varphi}_{i+1}) + \sigma_\varphi^2 \tilde{v}_{i+1}^2 \sin^2(\tilde{\varphi}_{i+1}) & \sigma_v^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) - \sigma_\varphi^2 \tilde{v}_{i+1}^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) \\ \sigma_v^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) - \sigma_\varphi^2 \tilde{v}_{i+1}^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) & \sigma_v^2 \sin^2(\tilde{\varphi}_{i+1}) + \sigma_\varphi^2 \tilde{v}_{i+1}^2 \cos^2(\tilde{\varphi}_{i+1}) \end{bmatrix} \quad (5.12)$$

where σ_v and σ_φ are the same as defined in (5.1).

With the RSSI measurements $\mathbf{Z}(i) = [P_{ij1} \ P_{ij2} \ \dots \ P_{ijn}]^T$ from neighboring nodes $\mathbf{J}$ as observations, the observation model is given by

$$\mathbf{Z}(i) = \mathbf{h}_{\mathbf{J}}(\Phi(i)) + \mathbf{v}(i) \quad (5.13)$$

where $\mathbf{v}(i)$ is the measurement noise of the RSSI from nodes $\mathbf{J}$, and $\mathbf{v}(i)$ has the covariance matrix of $\mathbf{R}(i)$. $\mathbf{h}_{\mathbf{J}}$ is the measurement function and it is given by

$$h_j(\mathbf{X}(i)) = P_0 - 10\alpha \log(\sqrt{(x_i - \hat{x}_j)^2 + (y_i - \hat{y}_j)^2}) \quad (5.14)$$

The EKF operates in the following way:

Given the estimate $\hat{\Phi}(i|i)$ of $\Phi(i)$ with covariance matrix $\mathbf{P}(i|i)$, the predicted state $\hat{\Phi}(i+1|i)$ of sensor node $i+1$ can be calculated as

$$\hat{\Phi}(i+1|i) = \hat{\Phi}(i|i) + \mathbf{u}(i+1), \quad (5.15)$$

with the predicted state covariance

$$\mathbf{P}(i+1|i) = \mathbf{P}(i|i) + \mathbf{Q}(i+1). \quad (5.16)$$

The predicted measurement of sensor node $i+1$ is

$$\hat{\mathbf{Z}}(i+1|i) = \mathbf{h}_{\mathbf{J}}(\hat{\Phi}(i+1|i)). \quad (5.17)$$

Then the innovation, i.e. the difference between the measurement $\mathbf{Z}(i+1)$ and the predicted measurement $\hat{\mathbf{Z}}(i+1|i)$ is given by

$$\boldsymbol{\gamma}(i+1) = \mathbf{Z}(i+1) - \hat{\mathbf{Z}}(i+1|i), \quad (5.18)$$

with the covariance

$$\mathbf{S}(i+1) = \mathbf{H}_{\mathbf{J}}(i+1)\mathbf{P}(i+1|i)\mathbf{H}_{\mathbf{J}}^T(i+1) + \mathbf{R}(i+1),$$

where $\mathbf{H}_{\mathbf{J}}(i+1)$ is the Jacobian matrix of the measurement function $\mathbf{h}_{\mathbf{J}}$ with respect to the predicted state $\hat{\Phi}(i+1|i)$. The EKF gain is given by

$$\mathbf{K}(i+1) = \mathbf{P}(i+1|i)\mathbf{H}_{\mathbf{J}}^T(i+1)\mathbf{S}^{-1}(i+1), \quad (5.19)$$

and the state estimation will be updated as

$$\hat{\Phi}(i+1|i+1) = \hat{\Phi}(i+1|i) + \mathbf{K}(i+1)\boldsymbol{\gamma}(i+1) \quad (5.20)$$

with the covariance matrix

$$\mathbf{P}(i+1|i+1) = \mathbf{P}(i+1|i) - \mathbf{K}(i+1)\mathbf{S}(i+1)\mathbf{K}^T(i+1) . \quad (5.21)$$

5.4 Performance Comparison of MLE and EKF schemes

To evaluate the proposed schemes, simulation studies are conducted to compare the localization performance of MLE and EKF schemes as well as the LLA proposed in chapter 3. The simulations are conducted over a network that consists of 28 sensor nodes. The sensors are deployed along a closed path with an average inter-node distance of 15m as shown in Fig. 5.1. The perimeter of the path is 424m.

Fig. 5.1 Simulated network

Fig. 5.2 shows the simulated localization error under different number of neighboring nodes for the single direction method of the three schemes, with 0.05 standard deviation of the compass and a pedometer error factor of 0.2. To statistically study the localization

accuracy, the data points in Fig. 5.2 are the average results of 100 simulation sets. As the range error factor is defined as the ratio between the standard deviation of the measured distance to the real distance, the range error factor can be determined as follow for the LLA

$$\gamma = \sqrt{\exp\left(\left(\frac{\sigma_R \ln 10}{10\alpha}\right)^2\right) - 1} \quad (5.22)$$

From the results, it is observed that both the proposed MLE and EKF based schemes have better localization accuracy than the LLA localization. Furthermore, the MLE based scheme generally out performs EKF in terms of localization accuracy especially for large number of neighboring nodes. When the number of neighboring nodes is small ($n < 3$), EKF is only slightly better than MLE. The MLE based scheme ignores the estimation error of previous deployed node and neighboring nodes when derives the likelihood functions. This makes the estimated position bias towards the DA estimated position. While this bias is reduced when there are more neighboring nodes, as the RSSI measurements will play a more significant role on the position estimation. On the other hand, as the number of neighboring nodes becomes larger and there are more RSSI measurements available, EKF becomes less accurate compared to MLE. This is because the RSSI measurements to the sensor node locations are non-linear. The linearization of the observation function leads to non-optimal estimates of the states. Therefore, the MLE based scheme is more preferable for larger number of neighboring nodes. Furthermore, as discussed in Section 4.3.C, when GNA is used to estimate positions in the MLE based algorithm, the method may diverge when the positions of the neighboring nodes and the estimated positions are happened to be collinear and leads to a wrong estimated position.

More than three neighboring nodes greatly reduce the co-linearity chance. Therefore the MLE based algorithm with GNA is more preferable when the number of neighboring nodes is larger than 3.

Fig. 5.2 Localization error of LLA, ELLA, MLE and EKF based schemes

5.5 Enhanced Likelihood Localization Algorithm

Despite being widely used for state estimation within communication and navigation applications [86]-[90], EKF has the disadvantage that the estimators may not be optimal if the measurement or the state transition model is not linear [91]-[94]. In addition, if the initial estimate of the state is wrong or if the process is modeled incorrectly, the filter may quickly diverge owing to its linearization. The simulation results in section 5.4 show that the MLE based scheme has superior localization accuracy than the EKF when the

measurements are highly non-linear. To benefit from the advantages of both schemes, we propose to hybridize the MLE and EKF based schemes into a new localization algorithm, denoted here as enhanced likelihood localization algorithm (ELLA). Similar to LLA, ELLA also has two localization modes, single-direction and bi-direction methods. In the single-direction method, an unknown node selects the MLE or EKF based schemes to estimate its state vector based on its number of neighboring nodes. It uses EKF based schemes described in section 5.3 if its number of neighboring nodes is less than a pre-defined number n . Otherwise it uses the MLE based scheme to estimate its position. From the simulation results of Fig.5.2, the localization accuracy of MLE based scheme is higher than EKF when $n \geq 3$. Furthermore, when $n \geq 3$ the MLE based scheme with GNA optimization becomes stable as more than three neighboring nodes greatly reduces the collinearity chance of the estimated position and positions of neighboring nodes. Therefore, in the following study, $n = 3$ is used if not specified.

Note that it requires the knowledge of the covariance matrix $\mathbf{P}(i|i)$ of node i 's state, if node $i+1$ uses the EKF based scheme. This requires the MLE based scheme to provide the covariance of its maximum likelihood estimator.

Let $l(\phi_i) = \ln(\bar{L}(\phi_i))$ be the log likelihood function of node i 's state vector. The covariance of the maximum likelihood estimator $\hat{\phi}_i$ is approximated as its asymptotic covariance,

$$\begin{aligned} \text{cov}(\hat{\phi}_i) &\approx -[E(\nabla_{\phi_i} (\nabla_{\phi_i} l(\phi_i))^T |_{\phi_i=\hat{\phi}_i})]^{-1} \\ &= \begin{bmatrix} \frac{1}{\sigma_v^2} + (\frac{10\alpha}{\ln(10)})^2 \sum_{j \in \mathcal{J}_i} \frac{(\hat{r}_i - \hat{r}_j \cos(\hat{\theta}_i - \hat{\theta}_j))^2}{\psi(\hat{\phi}_i, \hat{\phi}_j)^4} & (\frac{10\alpha}{\ln(10)})^2 \sum_{j \in \mathcal{J}_i} \frac{(\hat{r}_i - \hat{r}_j \cos(\hat{\theta}_i - \hat{\theta}_j)) \hat{r}_i \hat{r}_j \sin(\hat{\theta}_i - \hat{\theta}_j)}{\psi(\hat{\phi}_i, \hat{\phi}_j)^4} \\ (\frac{10\alpha}{\ln(10)})^2 \sum_{j \in \mathcal{J}_i} \frac{(\hat{r}_i - \hat{r}_j \cos(\hat{\theta}_i - \hat{\theta}_j)) \hat{r}_i \hat{r}_j \sin(\hat{\theta}_i - \hat{\theta}_j)}{\psi(\hat{\phi}_i, \hat{\phi}_j)^4} & \frac{1}{\sigma_\phi^2} + (\frac{10\alpha}{\ln(10)})^2 \sum_{j \in \mathcal{J}_i} \frac{(\hat{r}_i \hat{r}_j \sin(\hat{\theta}_i - \hat{\theta}_j))^2}{\psi(\hat{\phi}_i, \hat{\phi}_j)^4} \end{bmatrix}^{-1} \end{aligned} \quad (5.22)$$

Let $\mathbf{f}(\hat{\phi}_i)$ be the transfer function of a state vector from the polar to Cartesian coordinate system:

$$\hat{\Phi}_i = \hat{\Phi}_{i-1} + \mathbf{f}(\hat{\phi}_i) \quad (5.23)$$

with the covariance

$$\mathbf{P}(i | i) = \text{cov}(\hat{\Phi}_i) = \text{cov}(\hat{\Phi}_{i-1}) + \mathbf{F}_i \text{cov}(\hat{\phi}_i) \mathbf{F}_i^T \quad (5.24)$$

where $\mathbf{F}_i$ is the Jacobian matrix of $\mathbf{f}(\hat{\phi}_i)$ with respect to $\hat{\phi}_i$.

In the bi-direction method, (5.8) is used to find the estimated state vector as the number of the neighboring nodes in the bi-direction method is large enough for the MLE based scheme.

Fig.5.2 also plots the localization results of the single-direction method of ELLA. From the figure, ELLA out performs both EKF and MLE based schemes in terms of localization accuracy. It is also noticed that, when $n > 4$ the localization results of ELLA and the MLE based scheme are very close. If before the deployment it is known that the number of neighboring nodes of each sensor node is high, it can simply use the MLE based scheme to save the communicational and computational costs.

Fig. 5.3 shows the simulated localization error under different pedometer and RSSI measurement error for the proposed ELLA, with 0.05 standard deviation of the compass. The simulated network is the same as depicted in Fig. 5.1. The communication range is assumed to be 40m. This leads to an average of 3 neighboring nodes of each sensor node for the single-direction method. From the results, it is observed that the position estimation is more accurate if the pedometer and the RSSI measurements have better accuracy. Moreover, when the RSSI measurement is relatively accurate, the proposed ELLA improves the localization accuracy by 20% than the LLA localization. In addition, the localization error can be further reduced by 25% in the bi-direction method.

Fig. 5.3 Localization error under different pedometer and range errors.

Fig. 5.4 localization error vs electronic compass accuracy

$p = 0.15$ and $\sigma_R = 1$ are used for this figure

Fig. 5.4 shows the effect of the compass accuracy on localization performance. From the figure, it is observed that if the standard deviation of the compass is less than 0.08 (4.6 degrees), the localization error is only slightly affected by the accuracy of the compass. An experiment over the same network as in simulation has been conducted. Pre-deployment measurement shows that the pedometer error factor $p = 0.22$, the standard deviation of the RSSI measurements $\sigma_R = 2.86$, and the standard deviation of the electronic compass is 0.04. Fig. 5.5 shows both the localized network with single and bi-direction method. Table 5.1 summarizes the localization result. It shows that the new approach outperforms the previous method in chapter 3. The localization accuracy is improved by about 17%.

(a) Single-direction

(b) Bi-direction

Fig. 5.5 Experimental localization result

● real positions; ▲ estimated position from the proposed scheme; X estimated positions from DA only

Table 5.1 Experiment results

Localization error	ELLA (simulation)	ELLA (experiment)	LLA (experiment)
DA only	11.05m	13.12m	13.12m
Single-direction	7.89m	8.27m	9.63m
Bi-direction	5.92m	5.47m	6.73m

5.6 Conclusion

In this chapter, the system model and localization are reformulated for more accurate representation to achieve better localization accuracy. Two new localization schemes namely the MLE and EKF based schemes that use the new system model have been proposed. The simulation results show that the EKF based scheme has better localization when the number of neighboring nodes is small. The MLE based scheme has superior localization accuracy than the EKF based scheme when the measurements are highly non-linear. To gain benefit from the advantages of both schemes, enhanced likelihood localization algorithm (ELLA) that hybridizes both schemes is proposed. Simulations and experimental results show that the localization accuracy of ELLA is 20 % better than the LLA.

Chapter 6

Localization of a Wireless Sensor Network using Maximum Likelihood Estimation with Negative Constraints

6.1 Introduction and related works

In chapters 3 to 5, we have investigated the use of deployment information for sparse sensor network localization. Simulation and experimental results show that the use of such information can effectively increase the localization accuracy, especially for WSN with non-rigid network graph. In this chapter, the use of negative constraint information for a centralized localization approach is explored.

Negative constraints have been used in some works [17],[45] together with the network connectivity information or inter-node distance measurements for the WSN localization. The negative constraints are based on the observation that if there is no communication link between two sensor nodes, then the distance between them should be longer than their communication range. In [17], an anchor node is used to give a repulsive virtual force to repulse the estimated position of an unknown node if it is out of the anchor node's communication range. As it is a distributed algorithm, it only uses the

negative constraints between nodes within two hops. In [45], only the negative constraints between unknown nodes and anchor nodes are utilized, while the negative constraints between two unknown nodes are ignored. Consider the centralized algorithm, the negative constraints can be used more advantageously since the central node has the knowledge of the whole network. However, this issue has not been well addressed and there is a lack of works on the effect of the negative constraints on the WSN localization performance.

In this chapter, a maximum likelihood estimation (MLE) WSN localization algorithm that uses the negative constraints and inter-node distances information from RSSI measurements is proposed. Different from other works that assume the communication coverage of the sensor node is a perfect circle, this chapter studies the communication range distribution of the sensor node based on the log-shadowing model. The likelihood function of the positions of the network is derived with the negative constraints and inter-node distances information.

To find the MLE estimate of the sensor nodes' positions, a least square optimization problem needs to be solved. The complexity of the optimization problem is largely dependent on the number of negative constraints being used. To reduce the computational costs, a novel iterative optimization procedure is proposed. In each procedure loop only the important negative constraints are being used for the localization.

The simulation and experimental works of the proposed approach will be presented in Chapter 7. The results show that the proposed algorithm improves the localization accuracy by 5-30% as compared to the conventional MLE localization without negative constraints.

6.2 MLE Localization without negative constraints

The MLE method is one of the most popular approach in obtaining practical estimator [82]. MLE localization algorithms for WSN without negative constraints can be readily found in some existing works [36]-[37].

In almost all existing wireless systems, RSSI can be readily measured once two sensor nodes have established communication connection. Due to its low cost, RSSI has become a widely used ranging technology in WSN localization. The log-shadowing model [68]-[69] for the path loss in radio channel shows that the RSSI measurement P_{ij} for the signal transmitted between node i and j is related to their distance d_{ij} :

$$P_{ij} = P_0 - 10\alpha \log_{10} d_{ij} + X_{\sigma ij} \quad (6.1)$$

where P_0 is the RSSI measurement at 1m distance; α is the path-loss exponent; $X_{\sigma ij}$ is a normally distributed random number with 0 mean and variance of σ_R^2 , i.e. $X_{\sigma ij} \sim N(0, \sigma_R^2)$.

Given P_{ij} , the distance between the two nodes can be estimated as:

$$\tilde{d}_{ij} = 10^{(P_0 - P_{ij})/(10\alpha)} \quad (6.2)$$

From (6.1) and (6.2), it can be derived that the estimated distance $\tilde{d}_{ij}$ is log-normally distributed:

$$\ln \tilde{d}_{ij} \sim N(\ln d_{ij}, \sigma^2) \quad (6.3)$$

where $\sigma = (\sigma_R \ln 10)/(10\alpha)$.

Assume that a N -node sensor network comprises of N_I unknown nodes and N_2 anchor nodes, i.e. $N = N_I + N_2$. Let $\mathbf{X}$ be the vector of the unknown nodes' positions:

$$\mathbf{X} = [x_1 \quad \dots \quad x_{N_I} \quad y_1 \quad \dots \quad y_{N_I}]^T \quad (6.4)$$

Let $d_{ij}(\mathbf{X}) = \|\theta_i - \theta_j\|$, where $i = 1, \dots, N_I$, $j = 1, \dots, N$, and $\theta_i = [x_i \quad y_i]^T$ is the location coordinates of node i . The likelihood function of node positions $\mathbf{X}$ based on RSSI measured inter-node distance $\tilde{d}_{ij}$ between nodes i and j is the probability density function of $\tilde{d}_{ij}$, and from (6.3),

$$\begin{aligned} L_{ij}(\mathbf{X}) &= P(\tilde{d}_{ij}) \\ &= b_{ij} \exp\left(-\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2}\right) \end{aligned} \quad (6.5)$$

where $b_{ij} = \frac{1}{\tilde{d}_{ij} \sqrt{2\pi\sigma^2}}$.

With the independent assumption for the random-fading variables $X_{\sigma_{ij}}$ in (6.1), the likelihood function of $\mathbf{X}$ based on all the observed RSSI estimated distances can be formulated as the product of individual $L_{ij}(\mathbf{X})$:

$$L(\mathbf{X}) = \prod_{i=1}^{N_I} \prod_{j=i+1}^N L_{ij}(\mathbf{X}) \quad (6.6)$$

where

$$L_{ij}(\mathbf{X}) = \begin{cases} b_{ij} \exp\left(-\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2}\right) & , \text{ if } \tilde{d}_{ij} \text{ is observed} \\ 1 & , \text{ otherwise} \end{cases}$$

The position vector $\hat{\mathbf{X}}$ that maximizes (6.6) is the maximum likelihood estimate. It is also equivalent to the following optimization problem:

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \left(\sum_{i=1}^{N_1} \sum_{j=i+1}^N w_{ij} ((\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2) \right) \quad (6.7)$$

where

$$w_{ij} = \begin{cases} 1, & \text{if } \tilde{d}_{ij} \text{ is observed} \\ 0, & \text{otherwise} \end{cases} \quad (6.8)$$

6.3 Negative Constraints and Modeling

Consider Fig. 6.1 where there are two anchor nodes A_1 and A_2 . In addition, there is an unknown node i that is within the communication range of the anchor nodes. There are two possible estimated positions for sensor node i , namely i_1 and i_2 . With additional information that node i is out of sensor node j 's communication range, i 's real position is more likely nearer to i_1 than i_2 . This information is termed negative constraint (NC) [17],[45]. On the other hand, a RSSI measured inter-node distance is called positive constraint.

Fig. 6.1 The negative constraint illustration

Empirical studies [15] on real test beds have shown that the assumption of a perfect circular radio model is invalid for WSNs. The communication range of a sensor node usually varies in different directions. Two sensor nodes that cannot establish communication is generally because the signal strength between them is below a certain threshold. Let η be the signal strength threshold for communication between nodes i and j , and R be j 's communication range in certain direction, then from (6.1):

$$\eta = P_0 - 10\alpha \log_{10} R + X_{\sigma ij} \quad (6.9)$$

Hence, R is also log-normally distributed:

$$\ln R \sim N(\ln \bar{R}, \sigma^2) \quad (6.10)$$

where $\bar{R} = 10^{(P_0 - \eta)/(10\alpha)}$ is the nominal communication range.

Let $f_j(R)$ be the probability density function of j 's communication range distribution in any direction. From (6.10):

$$f_j(R) = \frac{1}{\sqrt{2\pi R \ln \bar{R}}} \exp\left(-\frac{(\ln R - \ln \bar{R})^2}{2\sigma^2}\right) \quad (6.11)$$

The likelihood function of node i 's position based on this negative constraint that i is out of j 's communication range can be formulated as:

$$\begin{aligned} L_{ij}(\mathbf{X}) &= \int_0^{d_{ij}(\mathbf{X})} f_j(R) dR \\ &= c_j(d_{ij}(\mathbf{X})) \end{aligned} \quad (6.12)$$

where $c_j(R)$ is the cumulative distribution function of j 's communication range:

$$c_j(R) = \frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{\ln R - \ln \bar{R}}{\sqrt{2}\sigma}\right) \quad (6.13)$$

$\operatorname{erf}(x)$ is the error function.

6.4 MLE Localization with negative constraints

As the negative and positive constraint information are independent, the overall likelihood function of $\mathbf{X}$ can be formulated as the product of the likelihood functions based on the inter-node observations of each pair of sensor nodes:

$$\bar{L}(\mathbf{X}) = \prod_{i=1}^{N_1} \prod_{j=i+1}^N L_{ij}(\mathbf{X}) \quad (6.14)$$

where,

$$L_{ij}(\mathbf{X}) = \begin{cases} b_{ij} \exp\left(-\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2}\right) & , \text{ if } \tilde{d}_{ij} \text{ is observed} \\ \frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{\ln d_{ij}(\mathbf{X}) - \ln \bar{R}}{\sqrt{2}\sigma}\right) & , \text{ otherwise} \end{cases} \quad (6.15)$$

Let $l(\mathbf{X}) = \ln(\bar{L}(\mathbf{X}))$, from (6.14) and (6.15)

$$\begin{aligned} l(\mathbf{X}) = & - \sum_{i=1}^{N_1} \sum_{j=i+1}^N (w_{ij} \left(\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2} - \ln(b_{ij}) \right) \\ & + (w_{ij} - 1) \ln\left(\frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{\ln d_{ij}(\mathbf{X}) - \ln \bar{R}}{\sqrt{2}\sigma}\right)\right) \end{aligned} \quad (6.16)$$

The maximum likelihood estimate $\hat{\mathbf{X}}$ that maximizes (6.15) is equivalent to the estimate that maximizes (6.16). Hence, we have the following:

$$\begin{aligned} \hat{\mathbf{X}} = \arg \min_{\mathbf{X}} & \left(\sum_{i=1}^{N_1} \sum_{j=i+1}^N (w_{ij} \left(\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2} \right) \right. \\ & \left. + (w_{ij} - 1) \ln \left(\frac{1}{2} + \frac{1}{2} \operatorname{erf} \left(\frac{\ln d_{ij}(\mathbf{X}) - \ln \bar{R}}{\sqrt{2}\sigma} \right) \right) \right) \end{aligned} \quad (6.17)$$

where

$$w_{ij} = \begin{cases} 1, & \text{if } \tilde{d}_{ij} \text{ is observed} \\ 0, & \text{otherwise} \end{cases}$$

6.5 Cramer-Rao bound (CRB) Derivation

In this section, the Cramer-Rao bound (CRB) of the network estimator $\hat{\mathbf{X}}$ that considered both the positive and negative constraints information is studied. The CRB [95]-[98] is a lower bound on the covariance of any unbiased estimator. The CRB on the covariance matrix of any unbiased estimator $\hat{\mathbf{X}}$ is $\operatorname{cov}(\hat{\mathbf{X}}) \geq \mathbf{F}_{\mathbf{X}}^{-1}$, where the Fisher information matrix (FIM) $\mathbf{F}_{\mathbf{X}}$ is defined as:

$$\mathbf{F}_{\mathbf{X}} = -E(\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}} l(\mathbf{X}))^T) = \begin{bmatrix} f_{1,1} & \cdots & f_{1,2N_1} \\ \vdots & \ddots & \vdots \\ f_{2N_1,1} & \cdots & f_{2N_1,2N_1} \end{bmatrix} \quad (6.18)$$

To derive the FIM with both positive and negative constraint information, $\mathbf{F}_{\mathbf{X}}$ is partitioned into 4 $N_1 \times N_1$ blocks:

$$\mathbf{F}_{\mathbf{X}} = \begin{bmatrix} \mathbf{F}_{xx} & \mathbf{F}_{xy} \\ \mathbf{F}_{xy}^T & \mathbf{F}_{yy} \end{bmatrix} \quad (6.19)$$

The FIM without negative constraint information can be found in [36] and is summarized in (6.20). To distinguish these two FIMs, $\mathbf{F}_X^p$ is used to denote the FIM without NC information, whereas $\mathbf{F}_X$ denote the FIM with NC information.

$$\mathbf{F}_X^p = \begin{bmatrix} \mathbf{F}_{xx}^p & \mathbf{F}_{xy}^p \\ \mathbf{F}_{xy}^{p^T} & \mathbf{F}_{yy}^p \end{bmatrix} \quad (6.20)$$

where

$$[\mathbf{F}_{xx}^p]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (w_{kj} p_{kj}^{[xx]}), & k = l \\ -w_{kl} p_{kl}^{[xx]}, & k \neq l \end{cases} \quad [\mathbf{F}_{xy}^p]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (w_{kj} p_{kj}^{[xy]}), & k = l \\ -w_{kl} p_{kl}^{[xy]}, & k \neq l \end{cases}$$

$$[\mathbf{F}_{yy}^p]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (w_{kj} p_{kj}^{[yy]}), & k = l \\ -w_{kl} p_{kl}^{[yy]}, & k \neq l \end{cases}$$

and

$$p_{kl}^{[xx]} = \frac{1}{\sigma^2 d_{kl}^4} (x_k - x_l)^2 \quad p_{kl}^{[xy]} = \frac{1}{\sigma^2 d_{kl}^4} (x_k - x_l)(y_k - y_l)$$

$$p_{kl}^{[yy]} = \frac{1}{\sigma^2 d_{kl}^4} (y_k - y_l)^2$$

The derivation of the FIM with NC information is shown in Appendix C, and it can be summarized in (6.21):

$$[\mathbf{F}_{xx}]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (w_{kj} p_{kj}^{[xx]} + (1-w_{kj}) n_{kj}^{[xx]}), & k = l \\ -w_{kl} p_{kl}^{[xx]} - (1-w_{kl}) n_{kl}^{[xx]}, & k \neq l \end{cases}$$

$$[\mathbf{F}_{xy}]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (w_{kj} p_{kj}^{[xy]} + (1-w_{kj}) n_{kj}^{[xy]}), & k = l \\ -w_{kl} p_{kl}^{[xy]} - (1-w_{kl}) n_{kl}^{[xy]}, & k \neq l \end{cases} \quad (6.21)$$

$$[\mathbf{F}_{yy}]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (w_{kj} p_{kj}^{[yy]} + (1-w_{kj}) n_{kj}^{[yy]}), & k = l \\ -w_{kl} p_{kl}^{[yy]} - (1-w_{kl}) n_{kl}^{[yy]}, & k \neq l \end{cases}$$

where

$$t_{kl} = \frac{\ln(d_{kl}) - \ln(\bar{R})}{\sqrt{2}\sigma} \quad m_{kl} = \frac{\ln(d_{kl}) - \ln(\bar{R})}{\sigma^2} + 2 + \frac{2e^{-t_{kl}^2}}{\sqrt{2\pi}\sigma(1 + \text{erf}(t_{kl}))}$$

$$c_{kl} = \frac{e^{-t_{kl}^2}}{\pi\sigma d_{kl}^2(1 + \text{erf}(t_{kl}))} \quad n_{kl}^{[xx]} = c_{kl} \left(1 - \frac{(x_k - x_l)^2}{d_{kl}^2} m_{kl}\right)$$

$$n_{kl}^{[xy]} = -c_{kl} \frac{(x_k - x_l)(y_k - y_l)}{d_{kl}^2} m_{kl} \quad n_{kl}^{[yy]} = c_{kl} \left(1 - \frac{(y_k - y_l)^2}{d_{kl}^2} m_{kl}\right)$$

The CRB of the root mean square (rms) localization error ε of the whole network is defined as (6.22) without NC, or (6.23) with NC as follows:

$$\varepsilon^p = \sqrt{\frac{1}{N_1} \sum_{i=1}^{N_1} \varepsilon_i^{p^2}} \geq \sqrt{\frac{1}{N_1} \text{tr}(\mathbf{F}_{\mathbf{x}}^{p^{-1}})} \quad (6.22)$$

$$\varepsilon = \sqrt{\frac{1}{N_1} \sum_{i=1}^{N_1} \varepsilon_i^2} \geq \sqrt{\frac{1}{N_1} \text{tr}(\mathbf{F}_{\mathbf{x}}^{-1})} \quad (6.23)$$

Consider the case that a network consists of 90 unknown and 10 anchor nodes that are placed within a 100 x 100 units sensor field. For an isotropic network, the sensor field is equally partitioned to 10 x 10 squares and there is a sensor randomly placed in each of

the square. The CRBs on rms localization error of the network can be calculated based on (6.22)-(6.23), and they are shown in Fig. 6.2. Since each realization of the network in simulation may not give the same CRBs, the CRBs shown in the figure are the average of 50 simulation trials. From the figure, we may conclude that the inclusion of NC information results in consistently lowered CRB. Furthermore, lower network connectivity or larger range measurement error lead to more significant improvement in localization accuracy. In this thesis, the network connectivity is defined as the average number of neighboring nodes that an unknown node has. A neighboring node is a sensor node that the unknown node can communicate with, and therefore use RSSI measurement to refer their inter-node distance.

Fig. 6.2 CRBs on rms localization error without and with NC

6.6 Position Estimation

The optimization problems defined in both (6.7) and (6.17) can be viewed as nonlinear least square (LS) problems. There are several iterative numerical optimization algorithms that can be used for such non-linear LS problems, for example, the Gauss-Newton, line-search and trust-region methods. An initial value is required for such iterative algorithms. In general, a good initial value would result in faster convergent to the global optimum. This can be achieved by estimating the initial positions using low cost localization algorithms such as DV-hop [24], DV-distance [24] or MDS [22],[37].

When compared to the objective function without NCs, the objective function (6.17) is much more complicated especially in the sparse networks. Let c be the network connectivity. There will be $\frac{1}{2}cN_1$ summation terms in the optimization problem defined in (6.7). However, if all the NCs are being used, the number of summation terms will be $\frac{1}{2}(N_1 + 2N_2)N_1$ as defined in (6.17). For a network with 100 sensor nodes and a connectivity of less than 10, direct minimization of (6.17) would require more than 10 times the computation. As many of the NCs provide little information for the position estimation, e.g. if two estimated positions are already very far from each other, then there is negligible difference between the results from MLE with and without the corresponding NC.

The NC between nodes i and j performs like a repulsive force that tends to make the distance between the estimated positions of these two nodes, i.e. $d_{ij}(\hat{\mathbf{X}})$, larger. Let $l_{ij}(\mathbf{X}) = \ln(L_{ij}(\mathbf{X}))$ be the log-likelihood of the NC between nodes i and j . The absolute

value of the partial derivative $\frac{\partial l_{ij}(\mathbf{X})}{\partial d_{ij}(\mathbf{X})}$ represents the magnitude of the repulsive force

between nodes i and j . The larger the value of $\left| \frac{\partial l_{ij}(\mathbf{X})}{\partial d_{ij}(\mathbf{X})} \right|$, the larger will be the repulsive

force, and hence the NC has more impact on the network localization. Fig 6.3 of the

amended thesis plots $\left| \frac{\partial l_{ij}(\mathbf{X})}{\partial d_{ij}(\mathbf{X})} \right|$ vs $d_{ij}(\mathbf{X})$. From the figure, if $d_{ij}(\hat{\mathbf{X}})$ is larger than 1.1 ~

1.2 times of $\bar{R}$, the NC has little impact on the localization, and therefore shall not be included for position estimation.

Fig. 6.3 $\left| \frac{\partial l_{ij}(\mathbf{X})}{\partial d_{ij}(\mathbf{X})} \right|$ vs $d_{ij}(\mathbf{X})$

An iterative procedure is developed in this study to select the NCs and find the position estimate. The procedure is shown as below and is summarized in the flowchart in Fig. 6.4.

Step 1: Obtain Initial Value

Obtain the initial value by localizing the network from other low cost algorithms (DV-distance, MDS, etc). In the following discussions, DV-distance is used.

Step 2: MLE Estimation

Estimate the sensor nodes positions without considering any NCs, through minimizing the objective function (6.5) and using the initial value obtained from Step 1.

Step 3: Negative Constraint Selection

Check if any NC is violated by the estimated positions. Go to step 6 if there is no new violation. Otherwise go to Step 4.

A negative constraint is said to be violated if:

- 1) There is no communication link between two associated sensor nodes; and
- 2) The distance between the two estimated positions is smaller than a predefined distance, $k \cdot \bar{R}$. I.e. if $d_{ij}(\hat{\mathbf{X}}) < k \cdot \bar{R}$, the NC between nodes i and j is violated. Here, k is termed as the negative constraints selection factor. The factor k should be chosen before localization. With a small k , very few NCs will be violated and the localization result will be similar to the result without any NCs. On the other hand, a large k leads to inclusion of many NCs and the computational cost will be increased significantly. As show in Fig. 6.3, a good k would be around 1.1.

Step 4: Objective Function Update

Update the objective function by including the new violated NCs. The new optimization problem becomes:

$$\begin{aligned} \hat{\mathbf{X}} = \arg \min_{\mathbf{X}} & \left(\sum_{i=1}^{N_1} \sum_{j=i+1}^N (w_{ij} \left(\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2} \right) \right. \\ & \left. + (w_{ij} - 1)v_{ij} \ln \left(\frac{1}{2} + \frac{1}{2} \operatorname{erf} \left(\frac{\ln d_{ij}(\mathbf{X}) - \ln \bar{R}}{\sqrt{2}\sigma} \right) \right) \right) \end{aligned} \quad (6.24)$$

where

$$v_{ij} = \begin{cases} 1, & \text{NC between } i \text{ and } j \text{ is violated} \\ 0, & \text{otherwise} \end{cases}$$

Step 5: Negative Constraint Optimization

Use the current estimated position as the initial value to minimize the new objective function (6.24) and find the new estimated positions. Go to step 6 if it exceeds the maximum number of loops, otherwise go to step 3.

Step 6: End of the optimization procedure.

Steps 3 to 5 form a procedure loop.

6.7 Conclusion

In this chapter, the likelihood function of sensor nodes' positions based on negative constraints information is derived and a maximum likelihood estimation based localization algorithm for WSNs is proposed. Furthermore, a novel iterative optimization

procedure is also proposed to reduce the computational cost. In the next chapter, we will evaluate the proposed localization scheme through various simulation and experimental studies.

Fig. 6.4 Optimization procedure

Chapter 7

Simulation and Experimental Results of MLE Localization Algorithm with Negative Constraints

7.1 Introduction

Chapter 6 proposed a maximum likelihood estimation (MLE) WSN localization algorithm that uses both the negative constraints (NC) and inter-node distance information based on RSSI measurements. In this chapter, we evaluate the proposed localization scheme through various simulation and experimental studies. The results show that the proposed algorithm improves the localization accuracy by 20% as compared to the MLE localization without negative constraints. Furthermore, the computational cost of the proposed algorithm is comparable to the conventional MLE localization.

To study the performance of the proposed approach under different network connectivity, the first simulated network from sections 7.2 to 7.4 uses an isotropic network that is placed within a 100 x 100 unit's sensor field. The sensor field is equally partitioned to 10 x 10 squares and there is a sensor placed randomly in each of the square. Therefore, there are a total of 100 sensor nodes. Assuming that 10% of the nodes are anchor nodes which are randomly selected. The range measurement is assumed log-

normally distributed, and σ ranges from 0.05 to 0.5 in different simulations. In section 7.5, simulation study over a non-isotopic network is also presented. Similar to earlier simulation study, 100 sensor nodes are randomly placed in the 100 x 100 unit's sensor field. In the simulations, the MLE and NC optimization were implemented using Matlab optimization toolbox with medium scale unconstrained nonlinear minimization. The algorithm used by the toolbox is Quasi-Newton method with a mixed quadratic and cubic line search. Simulation results show that the NC improves the localization accuracy by 5% to 30%. Moreover, the improvement is more significant for larger range measurement error and lower network connectivity.

7.2 The Negative Constraints Selection Factor k

As mentioned in Section 6.6, the negative constraints selection factor k is an important parameter. If k is too small, very few NCs will be violated and the optimization problem (6.18) is close to the optimization problem defined in (6.7). Consequently, the localization result will be similar to the result without NCs. On the other hand, large k leads to too many NCs inclusion. (6.18) will be close to (6.17) and the computational cost is increased significantly. The effect of k on localization accuracy and the number of violated NCs are plotted in Figs. 7.1 and 7.2 respectively. Each data points on the graph are the average of 50 simulation trials.

From Fig. 7.1, it is observed that the localization error decreases as k increases for all the network topologies and range measurements. This is a consequence of more NCs are being included to constrain the position estimation with a larger k . However, the reduction in localization error is less significant when k is larger than 1.1. On the other

hand, it is also observed from Fig. 7.2 that the number of violated NCs is increased significantly when k is larger than 1.1. Therefore, it may be concluded from these results that a value of 1.1 is a reasonable choice for the factor k . In the later discussion, k is fixed at 1.1.

Fig. 7.1 Effect of k on the localization accuracy

Fig. 7.2 Effect of k on the number of violated NCs

7.3 Performance Comparison of MLE with and without Negative Constraints over Isotropic Networks

To study the effect of the NCs on localization accuracy, Fig. 7.3 compares the localization error of MLE without and with NCs (MLE-NC) under different network connectivity and range measurement error. From Fig. 7.3, the localization accuracy is improved consistently by including the NCs, under different network connectivity and range measurement error σ .

Fig. 7.3 Comparison of localization accuracy of MLE and MLE-NC

Fig. 7.4 shows an example of the localized network using both the methods. In this example, the average network connectivity is 7 and $\sigma = 0.2$. The localization errors of MLE and MLE-NC are 3.206m and 2.431m respectively. NC improves the localization accuracy by 24% in this particular example.

(a) MLE

(b) MLE-NC

Fig.7.4 Localization examples

‘●’ unknown nodes’ real positions; ‘*’ anchor nodes; ‘○’ estimated positions; ‘—’ localization error

Fig. 7.5 plots the percentage of the reduced localization error by the NCs under different network connectivity and range error. From Fig. 7.5, the localization accuracy with NC is improved by more than 20% when the network connectivity is less than 8. It is noticed that this improvement decreases as the connectivity increases. When the connectivity is larger than 9, the localization accuracy improvement from the NCs is less than 15%. This is because higher network connectivity means more positive constraints and less NCs are available. The estimated positions from only the positive constraints are more accurate, and fewer NCs are violated for higher network connectivity. Therefore, the improvement by the NCs is less significant for heavily connected networks.

Fig. 7.5 also shows the accuracy improvement by the NCs under different distance measurement error, i.e. σ . As shown in the figure, the improvement by the NCs increases with σ , especially when the distance measurement is relatively accurate ($\sigma < 0.25$). This is because less accurate estimated positions from MLE lead to more violated NCs. It is also noticed from (6.10) that a larger σ leads to more irregularity of the communication ranges. This implies that both the NCs and positive constraint information becomes less accurate with a larger σ . Therefore the improvement from the NCs is relatively consistent when $\sigma > 0.3$.

Fig. 7.5 Localization accuracy improvement by NC under different range measurement error and connectivity

7.4 Performance under NLOS Environment

In this simulation, we study the performance of NCs under non-line-of-sight (NLOS) environments. Several 10 unit length obstacles are placed within the sensor field in random places and orientations. As shown in Fig. 7.6, communication link that has an obstacle along the way will be blocked. Therefore, some NCs information would be misleading since any nearby nodes would still be out of communication coverage due to the obstacles. Fig 7.7 compares the localization errors of MLE and MLE-NC with different number of obstacles. The average network connectivity is 7 when there is no obstacle and $\sigma = 0.25$ is used in this simulation. It is noticed that the presence of obstacles causes the network connectivity to be reduced. Consequently, it leads to larger

localization error for both methods. Moreover, more obstacles also lower the localization coverage. Some sensor nodes or groups of sensor nodes are isolated from others due to the blocking of their communication links, and therefore they are not able to find their estimated positions. The localization error in Fig. 7.7 is the average localization error for 50 simulation trials. From Fig. 7.7, it is noted that the NCs can still improve the localization accuracy but the percentage improvement decreases from 20% to 14% with the increase of the number of obstacles.

Fig. 7.6 Network under NLOS environment

‘●’ unknown nodes; ‘⊗’ anchor nodes; ‘---’ communication links; ‘—’ obstacles

Fig. 7.7 Localization accuracy under NLOS environments

7.5 Localization in Uniform Random Distributed Networks

Localization performance for uniform random distributed network is also studied. In this simulation, 100 sensor nodes are randomly placed in the 100 x 100 unit's sensor field, and 10% of the sensor nodes are anchor nodes. Different from the networks simulated in sections 7.2 and 7.3, the network shown in Fig. 7.8 is generally not isotropic. The sensor node density varies in different part of the network, and it is possible that some unknown nodes cannot be localized. An unknown node will not be localized if the sensor node is isolated from the rest of the network or belongs to an isolated group that has less than three non-collinear anchor nodes. Fig. 7.9 plots the localization coverage of uniform random distributed networks under different communication ranges. The localization coverage is defined as the percentage of the unknown nodes that can be successfully

localized. From the figure, nearly all the unknown nodes can be localized when the communication range is larger than 22m, while the localization coverage drops to 70% with 12m communication range. The figure also shows the relationship between the network connectivity and communication range. The results show that the network connectivity increases almost linearly with the communication range.

Fig. 7.8 An example of uniform random distributed networks

‘●’ unknown nodes’ real positions; ‘●’ anchor nodes

Fig. 7.9 Localization coverage for uniform random distributed networks

To study the localization performance over uniform random distributed networks, Fig. 7.10 compares the localization error of MLE and MLE-NC under different communication ranges and range measurement errors. From Fig. 7.10, MLE-NC consistently outperforms MLE by 5% to 30%. Similar to section 7.3, the improvement is more significant for larger range measurement error and lower network connectivity.

Fig. 7.10 Localization error for uniform randomly distributed networks

7.6 Computational Cost and Performance Trend

For the position estimation, most computational time is spent on the MLE and the NC optimizations (Step 2 and 5 of the optimization procedure in section 6.6). Both the optimizations are least square optimization problems. Hence, the Gauss-Newton algorithm with line-search method can be used to perform the optimizations. Table 7.1 lists the average CPU time, the number of iterations and the number of objective function evaluations before meeting the stopping criterion. For this evaluation, the stopping criterion is set as change of the objective function is within 0.1. It should be noted that the NC optimization may consist of more than one procedure loops and consequently step 5 may be executed more than once. The values for the NC optimizations in Table 7.1 are sums of all procedure loops.

From Table 7.1, the NC optimization converges much faster than the MLE optimization. Even though its objective function is more complicated, NC optimization requires less than half the CPU time of the MLE optimization.

Table 7.1 Computational costs

	MLE optimization	NC optimization
CPU time	58.3 s	27.4s
No. of Iterations	146	57
No. of objective function evaluations	24168	9026

To show the performance trend, Fig. 7.11 plots the average value of the objective function versus the number of iteration. From the figure, the NC optimization converges faster as its initial value is closer to its minimum. Moreover, it is observed that the value of the objective function without NC also becomes smaller after the NC optimization. This implies that the MLE optimization generally terminates in a local minimum without NCs. The subsequent NCs optimization provides not only more information for the position estimation, but also make the estimate relatively closer to its global minimum.

Fig. 7.11 Objection function values vs iterations

7.7 Experimental Results

Experimental measurement has been conducted in a $90\text{m} \times 90\text{m}$ park located in the university campus. The network consists of 25 sensor nodes. 4 of them are anchor nodes that are placed at the corners of the sensor field. The rest of the sensor nodes are unknown nodes and are placed in a pseudo-random manner as shown in Fig. 7.12. Each sensor node is equipped with a XBee ZNet 2.5 OEM RF module, which is able to measure the RSSI as discussed previously in Section 4.2. Before the deployment, calibration has been conducted to determine the parameters used in the path-loss equation by measuring the RSSI with respect to a reference distance. From the calibration, the standard deviation of the logarithm of the measured distances is determined as $\sigma = 0.22$. The nominal communication range is 40m, which leads to the average connectivity of 7.62 for the unknown nodes.

Fig. 7.13 depicts the system structure and data flow. For the experiment, the sensor nodes collect their inter-node RSSI measurements asynchronously. A sensor node polls its neighboring sensor nodes in sequence and requests them to send a data packet. The sensor node measures the RSSI of the received packets and transmits it to a central coordinator, which uploads the data to a laptop. The laptop translates the RSSI measurements to distances, and estimates the sensor nodes' positions through the optimization procedure proposed in section 6.6.

Fig. 7.12 Experimental network and localization result

‘●’ unknown nodes’ real positions; ‘●’ anchor nodes; ‘×’ estimated positions without NCs; ‘---’ difference between estimated positions without NCs and real positions; ‘○’ estimated positions with NCs; ‘---’ difference between estimated positions with NCs and real positions

Fig. 7.13 Experimental system structure and data flow

The experimental results are shown in Fig. 7.12. The figure shows both the MLE localization without and with the NC. The average localization errors without and with the NC are 5.27m and 4.14m respectively. Hence, the negative constraints reduce the localization error by 21%.

To study the performance of the proposed localization scheme under different network topology and connectivity, a few unknown nodes are randomly taken out from the network and the rest of the sensors are re-localized by using the remaining positive and negative constraints information. Table 7.2 shows the localization error of the new networks. From Table 7.2, the localization error is reduced by about 20% if the NC is

considered. Moreover, the improvement is more significant when the network connectivity is low. This is consistent with the finding in section 7.3.

Table 7.2 Experimental localization results with different number of unknown nodes

Number of unknown nodes	Connectivity	Localization error without NCs (m)	Localization error with NCs (m)	Improvement by the NCs (%)
21	7.62	5.27	4.17	20.9
18	6.56	6.28	5.21	17.1
15	5.62	7.53	5.84	22.4
12	4.5	9.63	7.21	25.2

7.8 Conclusion

In this chapter, the performance of the proposed localization algorithm has been evaluated and validated with experimental results. The results show that the proposed MLE localization algorithm with negative constraints improves localization accuracy by 5 ~ 30% from the MLE localization without negative constraints. The improvement is more significant for low network connectivity and larger range measurement error system. The proposed algorithm requires comparable computational cost as the MLE localization without negative constraints information.

Chapter 8

Conclusion and Future Works

8.1 Summary

In this thesis, we have investigated the deployment and negative constraint information for WSN localization. We have proposed and developed distributed LLA and ELLA that utilize deployment information for sparse WSN localization. We have also proposed and developed a centralized MLE localization algorithm that utilizes negative constraint information. The proposed algorithms have been implemented and validated. Simulation and experimental results have shown they outperform other existing methods.

For the localization in sparse networks whose network graph is not globally rigid, the conventional self-localization algorithms which only use inter-node distances information are not adequate. To localize sparse WSNs, we have proposed and developed a distributed localization algorithm, namely likelihood localization algorithm (LLA) that utilizes additional deployment information. The new algorithm outperforms the conventional approaches in term of localization accuracy in both sparse and dense WSNs. Furthermore, the new algorithm is suitable for distributed network and can be easily implemented using low cost microcontrollers.

To conduct the experiment, the LLA has been realized and evaluated using a low-cost microcontroller. To estimate sensor positions in LLA, a non-linear least square optimization problem is involved. For the system's efficiency and robustness, different optimization approaches, such as Gauss-Newton algorithm and particle swarm optimization method, are carefully studied and compared through analysis and real experiments. The realized LLA localization system can be used to localize a sparse WSN in real time with its deployment.

To achieve better localization, ELLA has been proposed through reformulating the system model and localization for more accurate representation. Furthermore, ELLA hybridizes the maximum likelihood estimation based and extended Kalman filter based position estimation schemes to benefit from their advantages. Simulations and experimental results have shown that the localization accuracy of ELLA is 20% better than LLA.

Finally, we have also studied the negative constraint information and the theoretical work shows that negative constraint can lower Cramer-Rao bound of estimated position. Therefore, negative constraint information can be useful to improve the localization accuracy especially for networks with low connectivity. For more accurate localization, a new centralized MLE algorithm that uses both negative constraints and inter-node distances information has been investigated and developed. The new algorithm outperforms the conventional centralized approaches in terms of localization accuracy especially for WSN with low network connectivity or high range measurement error.

8.2 Future Works

- **Memetic Algorithm for Position Estimation**

Considering the tradeoff between the Gauss-Newton algorithm (GNA) and particle swarm optimization (PSO) method, memetic algorithm can be an alternative for the position estimation in LLA. Memetic algorithm hybridizes PSO and GNA. The algorithm employs a PSO framework and a GNA as a local searcher. The Memetic algorithm could benefit from the advantages of both methods, i.e. GNA's fast execution and PSO's robustness. Furthermore that Levenberg-Marquardt algorithm, quasi-Newton methods or other more sophisticated optimization methods can also be considered to improve the speed and the robustness of the localization system.

- **Obtain RSSI Ranging Parameters in Real-time or Other Ranging Methods**

Besides the deployment and negative constraint information, the inter-node distances information from RSSI measurement is utilized in both the proposed localization algorithms. In this thesis, the RSSI ranging parameters are determined off-line before the deployment. This often yields limited estimation accuracy, because the RSSI ranging parameters are subjected to the environmental changes. The estimation accuracy could be further improved, if the ranging parameters are estimated together with the unknown node's location in real time. Other ranging methods that are less dependent on the environment, such as ultra-sound and ultra-wide band (UWB) radio, can also be considered.

- **Hybrid Localization Algorithm**

The proposed MLE localization algorithm with negative constraint information has improved the localization accuracy as compared to the conventional MLE localization algorithm. However, it does not improve the localization coverage. The proposed method still needs the network graph to be globally rigid to localize every sensor node. A possible approach to improve the localization coverage is to hybrid the proposed algorithms, i.e. the MLE localization algorithm that utilizes deployment, negative constraint, as well as the inter-node distances information.

- **Distributed Algorithm with Negative Constraint Information**

A centralized algorithm with negative constraint information has been proposed in this thesis. Centralized algorithms have high localization accuracy potential, but they are computationally and communicational costly for a large scaled network. It would be interesting to develop a distributed localization algorithm with negative constraint information.

Author's Publications

- [1] H. Guo, K S Low, H.A. Nguyen, “Optimizing the Localization of a Wireless Sensor Network in Real Time Based on a Low Cost Microcontroller”, *IEEE Trans. on Industrial Electronics*, vol.58. no. 3, pp. 741-749. Mar. 2011.

- [2] H.A. Nguyen, H. Guo and KS Low, “Real Time Estimation of Sensor Node’s Position Using Particle Swarm Optimization with Log Barrier Constraint”, *IEEE Trans. Instrumentation & Measurements*, in Press.

- [3] H.A. Nguyen, K. S. Low and H. Guo, “Real Time Determination of Sensor Node Location in a Wireless Sensor Network using Particle Swarm Optimization”, *International Conference on Evolutionary Computing (EC '09)*, 23 – 25 March 2009, Prague, Czech Republic, pp.140 – 145.

- [4] K. S. Low, H.A. Nguyen and H. Guo, “Optimization of Sensor Node Locations in a Wireless Sensor Network”, *The 4th International Conference on Natural Computation (ICNC '08)*, 18 – 20 Oct 2008, Jinan, China.

[5] K. S. Low, H.A. Nguyen and H. Guo, “A Particle Swarm Optimization Approach for the Localization of a Wireless Sensor Network”, *IEEE International Symposium on Industrial Electronics (ISIE '08)*, 30 June – 2 July 2008, Cambridge, UK.

[6] H. Guo, K. S. Low, H.A. Nguyen and M. J. Er, “Localization in a Sparse Wireless Sensor Network Using Pedometer and Communication Ranging Measurements”, *IEEE Industrial Electronics, Control and Instrumentation Conference (IECON '07)*, 7 November- 10 November 2007, Taipei, Taiwan, pp. 2627 - 2632.

[7] H. Guo, K. S. Low, and M. J. Er, “ Probability Field Localization for Wireless Sensor Network”, *IEEE Industrial Electronics, Control and Instrumentation Conference (IECON '06)*, 7 November- 10 November 2006, Paris, France, pp. 3155-3160.

Bibliography

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, pp. 393-422, 2002.
- [2] K.S. Low, W.N.N. Win and M.J. Er, “Wireless Sensor Networks for Industrial Environments”, *Proc. Int. Conf. on Computational Intelligence for Modeling, Control and Automation*, 2005, pp. 271-276.
- [3] D. Culler, D. Estrin, and M. Srivastava, “Overview of Sensor Networks”, *IEEE Computer*, vol. 37, pp. 41-49, Aug. 2004.
- [4] D. Puccinelli, M. Haenggi, “Wireless sensor networks: applications and challenges of ubiquitous sensing”, *IEEE Circuits and Systems Magazine*, vol. 5, no. 3, pp. 19-31, 2005.
- [5] E.L. Souza, E.F. Nakamura, H.A. De Oliveira, “On the performance of target tracking algorithms using actual localization systems for wireless sensor networks”, *12th ACM Int. Conf. on Modeling, Analysis, and Simulation of Wireless and Mobile Systems*, 2009.
- [6] J. Lin, W. Xiao, F. L. Lewis, L. Xie, “Energy-Efficient Distributed Adaptive Multisensor Scheduling for Target Tracking in Wireless Sensor Networks”, *IEEE Trans. Instrumentation and Measurement*, vol. 58, no. 6, pp. 1886-1896, 2009.
- [7] T. Onel, C. Ersoy, H. Delic, “Information Content-Based Sensor Selection and Transmission Power Adjustment for Collaborative Target Tracking”, *IEEE Trans. Mobile Computing*, vol. 8, no. 8, pp. 1103-1116, 2009.

- [8] R. Rahman, M. Alanyali, V. Saligrama, "Distributed Tracking in Multihop Sensor Networks With Communication Delays", *IEEE Trans. Signal Processing*, vol. 55, no. 9, pp. 4656-4668, 2007.
- [9] I. Stojmenovic, "Position-based routing in ad hoc networks", *IEEE Commun. Mag.*, vol.40, no.7, pp.128 – 134, Jul. 2002.
- [10] Y. Xu, J. Heidemann, D. Estrin, "Geography-Informed energy conservation for ad hoc routing", *Proc ACM/IEEE Conf. Mobile Computing and Networking*, Jul. 2001, pp.70 - 84.
- [11] J.N. Al-Karaki, A.E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey ", *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6-28, 2004.
- [12] J. Hightower and G. Boriello. "Location systems for ubiquitous computing," *IEEE Computer*, vol. 34, pp. 57-66, 2001.
- [13] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Personal Commun.*, vol. 7, pp. 28-34, Oct 2000.
- [14] V. Vivekanandan, V.W.S. Wong, "Concentric Anchor Beacon Localization Algorithm for Wireless Sensor Networks", *IEEE Trans. Veh. Technol.*, vol.56, no.5, pp.2733 – 2744, 2007.
- [15] T. He, C. Huang, B.M. Blum, J.A. Stankovic, T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," *Proc. 9th Int. Conf. on Mobile computing and networking*, 2003, pp. 81-95.
- [16] L. Doherty, K.S.J. Pister, L.E. Ghaoui, "Convex position estimation in wireless sensor networks", *Proc. IEEE Conf. Computer and Communications Societies*, Apr. 2001. vol.3, pp. 1655-1663.

- [17] J.P. Sheu, P.C. Chen, C.S. Hsu, "A Distributed Localization Scheme for Wireless Sensor Networks with Improved Grid-Scan and Vector-Based Refinement", *IEEE Trans. Mobile Computing.*, vol.7, no.9, pp. 1110-1123, 2008.
- [18] J.P. Sheu, J.M. Li, C.S. Hsu, "A Distributed Location Estimating Algorithm for Wireless Sensor Networks", *Proc. IEEE Int. Conf. Sensor Networks, Ubiquitous, and Trustworthy Computing*, Jun. 2006. vol. 1, pp. 218-225.
- [19] M. Rudafshani, S. Datta, "Localization in Wireless Sensor Networks", *Proc. Int'l Conf. Information Processing in Sensor Networks*, Apr. 2007, pp. 51-60.
- [20] K. Yedavalli, and B. Krishnamachari, "Sequenced-based localization in Wireless sensor network," *IEEE Trans. Mobile Computing*, vol. 7, no. 1, pp. 81-94, Jan. 2008.
- [21] R. Stoleru, P. Vicaire, T. He, J.A. Stankovic, "StarDust: A Flexible Architecture for Passive Localization in Wireless Sensor Networks", *Proc. ACM Int. Conf. Embedded Networked Sensor Systems*, Oct. 2006, pp. 57-70.
- [22] Y. Shang, W. Ruml, Y. Zhang, M.P.J. Fromherz, "Localization from mere connectivity," *Proc. 4th ACM Int. Symp. On Mobile Ad Hoc Networking & computing. Annapolis: ACM press*, 2003, pp. 201-212.
- [23] L. Li, T. Kunz, "Cooperative Node Localization Using Nonlinear Data Projection", *ACM Trans. Sensor Networks*, vol. 5, no.1, , Feb. 2009.
- [24] D. Niculescu and B. Nath. "DV based positioning in ad hoc networks," *Journal of Telecommunication Systems*, 22(1/4), pp. 267-280, 2003.

- [25] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AoA," *IEEE Joint Conf. Computer and Communications Societies*, Mar. – Apr. 2003, vol. 3, pp. 1734-1743.
- [26] C. Savarese, J. Rabaey, and K. Langendoen, "Robust positioning algorithm for distributed ad-hoc wireless sensor networks," *USENIX technical Annual Conf.*, Monterey, CA, 2002, pp. 317-328.
- [27] S. Meguerdichian, S. Slijepcevic, V. Karayan, M. Potkonjak, "Localized algorithms in wireless ad-hoc networks: Location discovery and sensor exposure." *Proc. 2nd ACM Int. Symp. Mobile Ad Hoc Networking & Computing. Long Beach: ACM Press*, 2001, pp. 106-116.
- [28] Y.T. Chan, H. Yau Chin Hang, P.K. Ching. "Exact and approximate maximum likelihood localization algorithms", *IEEE Trans. Veh. Technol.*, vol.55, no.1, pp.10 – 16, Jan. 2006.
- [29] K. Yu, Y.J. Guo, M. Hedley, "TOA-based distributed localisation with unknown internal delays and clock frequency offsets in wireless sensor networks", *IET Signal Processing*, vol.3, no.2, pp. 106-118, Mar. 2009.
- [30] K. Langendoen, N. Reijers, "Distributed Localization in Wireless sensor networks: a quantitative comparison", *Computer Networks*, vol. 43, pp. 499-518, 2003.
- [31] J. Liu, Y. Zhang, F. Zhao, "Robust Distributed Node Localization with Error Management", *Proc. 7th ACM int. symposium on Mobile ad hoc networking and computing*, May 22-25, 2006.
- [32] H. Lim, J. C. Hou, "Distributed Localization for Anisotropic Sensor Networks", *ACM Trans. Sensor Networks*, Vol. 5, No.2, Article 11, Mar. 2009.

- [33] Q. Shi, C. He, H. Chen, L. Jiang, "Distributed Wireless Sensor Network Localization Via Sequential Greedy Optimization Algorithm ", *IEEE Trans. Signal Processing*, vol. 58, no. 6, pp. 3328-3340, 2010.
- [34] S. Zhu, Z. Ding, "A simple approach of range-based positioning with low computational complexity ", *IEEE Trans. Wireless Communications*, vol. 8, no. 12, pp. 5832-5936, 2009.
- [35] A. Savvides, H. Park, M. Srivastava, "The bits and flops of the N-hop multilateration primitive for node localization problems", *1st ACM Int. workshop on Wireless Sensor Networks and Application (WSNA), Atlanta, GA*, 2002, pp. 112-121.
- [36] N. Patwari, A.O. Hero III, M. Perkins, N.S. Correal, R.J. O'Dea, "Relative Location Estimation in Wireless Sensor Networks", *IEEE Trans. Signal Processing*. vol.51, no.8, pp. 2137 - 2148, Aug. 2003.
- [37] X.R. Li, "Collaborative Localization With Received-Signal Strength in Wireless Sensor Networks", *IEEE Trans. Veh. Technol.*, vol.56, no.4, pp.3807 – 3817, Nov. 2007.
- [38] Y. Shang, W. Ruml, Y. Zhang, M.P.J. Fromherz, "Localization from connectivity in sensor networks", *IEEE Trans. Parallel and Distributed Systems*, vol. 15, no. 11, pp.961 – 974, Nov. 2004.
- [39] F. Chan, H.C. So, W.-K. Ma, "A Novel Subspace Approach for Cooperative Localization in Wireless Sensor Networks Using Range Measurements", *IEEE Trans. Signal Processing*., vol.57, no.1, pp. 260-269, Jan. 2009.

- [40] K.W.K. Lui, W.-K. Ma, H.C. So, F.K.W. Chan, F. K. W. "Semi-Definite Programming Algorithms for Sensor Network Node Localization With Uncertainties in Anchor Positions and/or Propagation Speed", *IEEE Trans. Signal Processing.*, vol.57, no.2, pp. 752-763, Feb. 2009.
- [41] N. Patwari, J.N. Ash, S. Kyperountas, A.O. Hero III, R.L. Moses, N.S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks", *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54-69, 2005.
- [42] F. Chan, H.C. So, "Efficient Weighted Multidimensional Scaling for Wireless Sensor Network Localization", *IEEE Trans. Signal Processing*, vol. 57, no. 11, pp. 4548-4533, 2009.
- [43] K. Yu, Y.J. Guo, "Anchor-free localisation algorithm and performance analysis in wireless sensor networks", *IET Communications*, vol. 3, no. 4, pp. 549-560, 2009.
- [44] J. A. Costa, N. Patwari, A. O. Hero III, "Distributed Weighted-Multidimensional Scaling for Node Localization in Sensor Networks", *ACM Trans. Sensor Networks*, Vol. 2, No.1, pp. 39-64, Feb. 2006.
- [45] R. Peng, M.L. Sichitiu, "Robust, probabilistic, constraint-based localization for wireless sensor networks", *IEEE 2nd Annual Communications Society Conf. on Sensor and Ad Hoc Communications and Networks*, Sept. 2005, pp. 541-550.
- [46] A. Savvides, W.L. Garber, R.L. Moses, M.B. Srivastava, "An analysis of error inducing parameters in multihop sensor node localization", *IEEE Trans. Mobile Computing*, vol. 4, no. 6, pp. 567-577, 2005.

- [47] B. Xiao, H. Chen, S. Zhou, "Distributed Localization Using a Moving Beacon in Wireless Sensor Networks", *IEEE Trans. Parallel and Distributed Systems*, vol.19, no.5, pp. 587-600, May. 2008.
- [48] K.F. Ssu, C.H. Ou, H.C. Jiau, "Localization with mobile anchor points in wireless sensor networks", *IEEE Trans. Veh. Technol.*, vol.54, no.3, pp.1187 – 1197, May. 2005.
- [49] H. Chen, Q. Shi, R. Tan, H.V. Poor, K. Sezaki, "Mobile element assisted cooperative localization for wireless sensor networks with obstacles ", *IEEE Trans. Wireless Communications*, vol. 9, no. 3, pp. 956-963, 2010.
- [50] E. Kim, K.Kim, "Distance Estimation With Weighted Least Squares for Mobile Beacon-Based Localization in Wireless Sensor Networks", *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 559-562, 2010.
- [51] S. Mihail, R. Vaidyanathan, "Localization of wireless sensor networks with a mobile beacon", *IEEE Conf. on Mobile Ad-hoc and Sensor Systems*, Oct. 2004, pp.174-183.
- [52] J. Aspnes, T. Eren, D.K. Goldenberg, A.S. Morse, "A theory of network localization", *IEEE Trans. Mobile Computing*, vol.5, no.12, pp.1663 – 1678, Dec. 2006.
- [53] K. Benkic, M. Malajner, P. Planinsic, Z. Cucej, "Using RSSI value for distance estimation in wireless sensor networks based on ZigBee ", *IEEE 15th conf. Systems, Signals and Image Processing*, Jun. 2008, pp. 303-306.

- [54] F. Cabrera-Mora, J. Xiao, "Preprocessing technique to signal strength data of wireless sensor network for real-time distance estimation", *IEEE Int. conf. Robotics and Automation*, May. 2008, pp. 1537-1542.
- [55] S. Hara, D. Zhao, K. Yanagihara, J. Taketsugu, K. Fukui, S. Fukunaga, K. Kitayama, "Propagation characteristics of IEEE 802.15.4 radio signal and their application for location estimation", *IEEE Conf. Vehicular Technology Conference*, 30 May -1 Jun. 2005, vol.1, pp.97-101.
- [56] Q. Shi, N. Correal, S. Kyperountas, F. Niu, "Performance comparison between TOA ranging technologies and rssi ranging technologies for multi-hop wireless networks", *IEEE 62nd conf. Vehicular Technology*, Sept. 2005, pp. 434-438.
- [57] B. Alavi, K. Pahlavan, "Modeling of the TOA-based distance measurement error using UWB indoor radio measurements", *IEEE Communications Letters*, vol. 10, no. 4, pp. 275-277, 2006.
- [58] S. Gezici, Z. Tian; G.B. Giannakis, H. Kobayashi, A.F. Molisch, H.V. Poor, Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks ", *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70-84, 2005.
- [59] Y. Bai, X. Lu, "Research on UWB indoor positioning based on TDOA technique", *IEEE 9th int. conf. Electronic Measurement & Instruments*, Aug. 2009, pp. 1167-1170.
- [60] J.R. Gonzalez, C.J. Bleakley, "Ultrasonic Orientation-Location Algorithm Based on ToA and AoA Measurements", *IEEE 10th Int. Symp. Spread Spectrum Techniques and Applications*, Aug. 2008, pp. 800-805.

- [61] R. Klukas, M. Fattouche, "Line-of-sight angle of arrival estimation in the outdoor multipath environment", *IEEE Trans. Veh. Technol*, vol.47, no.1, pp.342 - 351, Feb. 1998.
- [62] S. Yuan, M.Z. Win, "On the accuracy of localization systems using wideband antenna arrays", *IEEE trans. Communications*, vol. 58, no. 1, pp.270-280, 2010.
- [63] J.K. Nelson, M.P. Gupta, J.E. Almodovar, W.H. Mortensen, "A Quasi EM Method for Estimating Multiple Transmitter Locations", *IEEE Signal Processing Letters*, vol. 16, no. 5, pp. 354-357, 2009.
- [64] I. Borg and P. Groenen, *Modern Multidimensional Scaling*, NewYork: Springer-Verlag, 1997.
- [65] D. Koutsonikolas, S.M. Das, Y.C. Hu, "Path Planning of Mobile Landmarks for Localization in Wireless Sensor Networks", *Proc. 26th IEEE Int. Conf. on Distributed Computing Systems Workshops*, 2006, pp. 86-86.
- [66] R. Huang, G.V. Zaruba, "Static Path Planning for Mobile Beacons to Localize Sensor Networks", *proc. 5th Annual IEEE Int. Conf. on Pervasive Computing and Communications Workshops*, 2007, pp. 323-330.
- [67] H. Guo, K.S. Low, and M.J. Er, "Localization in a Sparse Wireless Sensor Network Using Pedometer and Communication Ranging Measurements," *Proc. IEEE Conf. Industrial Electronics, Control and Instrumentation*, 2007, pp. 2627-2632.
- [68] T.S. Rappaport, *Wireless Communication: Principles and Practice*, Englewood cliffs, NJ: Prentice-Hall, 1996.

- [69] T. K. Sarkar, Z. Ji, K. Kim, A. Medouri, and M. Salazar-Palma, "A survey of various propagation models for mobile communication," *IEEE Antennas and Propagation Magazine*, vol. 45, no. 3, pp. 51-82, June 2003.
- [70] S.Y. Xue, S.X. Yang, "Power system frequency estimation using Supervised Gauss-Newton algorithm", *Proc. Systems, Man and Cybernetics, Conference (ISIC '07)*, 2007, pp. 3761-3766.
- [71] J.D. Zaeytjld, A. Franchois, C. Eyraud, J.-M. Geffrin, "Full-Wave Three-Dimensional Microwave Imaging With a Regularized Gauss-Newton Method - Theory and Experiment", *IEEE Trans. on Ant. and Propag.*, vol. 55, no. 11, pp. 3279-3292, Nov. 2007.
- [72] T. Zhao, A. Nehorai, "Information-Driven Distributed Maximum Likelihood Estimation Based on Gauss-Newton Method in Wireless Sensor Networks", *IEEE Trans. Signal Processing*, vol. 55, no. 9, pp. 4669-4682, 2007.
- [73] K.S. Low, H.A. Nguyen and H. Guo, "A Particle Swarm Optimization Approach for the Localization of a Wireless Sensor Network" *IEEE Int. Symp. on Industrial Electronics (ISIE '08)*, July 2008, pp. 1820-1825.
- [74] A. Chatterjee, K. Pulasinghe, K. Watanabe, K. Izumi, "A particle-swarm-optimized fuzzy-neural network for voice-controlled robot systems" *IEEE Trans. on Ind. Electron.*, vol. 52, no. 6, pp. 1478-1489, Dec. 2005.
- [75] L. Dos Santos Coelho, B.M. Herrera, "Fuzzy Identification Based on a Chaotic Particle Swarm Optimization Approach Applied to a Nonlinear Yo-yo Motion System", *IEEE Trans. on Ind. Electron.*, vol. 54, no. 6, pp. 3234-3245, Dec. 2007.

- [76] Y. Song, Z. Chen, Z. Yuan, "New Chaotic PSO-Based Neural Network Predictive Control for Nonlinear Process", *IEEE Trans. on Neural Network*, vol. 18, no. 2, pp. 595-601, Mar. 2007.
- [77] T. Huang, A.S. Mohan, "A Microparticle Swarm Optimizer for the Reconstruction of Microwave Images", *IEEE Trans. on Ant. and Propag.* vol. 55, no. 3, pp. 568-576, Mar. 2007.
- [78] N. Jin, Y. Rahmat-Samii, "Advances in Particle Swarm Optimization for Antenna Designs: Real-Number, Binary, Single-Objective and Multiobjective Implementations", *IEEE Trans. on Ant. and Propag.*, vol. 55, no. 3, pp. 556-567, Mar. 2007.
- [79] L. Lizzi, F. Viani, R. Azaro, A. Massa, "Optimization of a Spline-Shaped UWB Antenna by PSO", *IEEE Ant. And Wireless Propag. Letter*, vol. 6, pp. 182-185, Mar. 2007.
- [80] F.-J. Lin; P.-K. Huang and W.-D. Chou, "Recurrent-Fuzzy-Neural-Network-Controlled Linear Induction Motor Servo Drive Using Genetic Algorithms", *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp 1449-1461, Jun. 2007.
- [81] K.-S. Low and T.-S. Wong , "A Multiobjective Genetic Algorithm for Optimizing the Performance of Hard Disk Drive Motion Control System", *IEEE Trans. on Ind. Electron.*, vol. 54, no. 3, pp. 1716-1725, Jun. 2007.
- [82] S.M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Englewood cliffs, NJ: Prentice-Hall, 1993.
- [83] Ian R. Petersen, *Robust Kalman Filtering for Signals and Systems with Large Uncertainties*, Birkhauser Boston, 1999.

- [84] L. Angrisani, A. Baccigalupi, R.S.L. Moriello, "Ultrasonic time-of-flight estimation through unscented Kalman filter ", *IEEE Trans. Instrumentation and Measurement*, vol. 55, no. 4, pp. 1077-1084, 2006.
- [85] M.M. Olama, S.M. Djouadi, I.G. Papageorgiou, C.D. Charalambous, "Position and Velocity Tracking in Mobile Networks Using Particle and Kalman Filtering With Comparison", *IEEE Trans Vehicular Technology*, vol. 57, no. 2, pp. 1001-1010, 2008.
- [86] S. Bolognani, L. Tubiana, and M. Zigliotto, "EKF-based sensorless IPM synchronous motor drive for flux-weakening applications," *IEEE Trans. Ind. Appl.*, vol. 39, no. 3, pp. 768-775, May/Jun. 2003.
- [87] S. Wu, L. Hong, and J. R. Layne, "2D rigid-body target modeling for tracking and identification with GMTI/HRR measurements," *IEE Proc. Control Theory and Applications.*, Jul. 2004, vol. 151, no. 4, pp. 429-438.
- [88] H. Zhao and R. Shibasaki, "A novel system for tracking pedestrians using multiple single-row laser-range scanners," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 2, pp. 283-291, Mar. 2005.
- [89] D. Ma, M.J. Er, H.B. Lim, "A Comprehensive Study of Kalman Filter and Extended Kalman Filter for Target Tracking in Wireless Sensor Networks", *IEEE Int. Conf. Systems, Man and Cybernetics*, 2008, pp. 2792-2797.
- [90] J.R. Katkuri, V.P. Jilkov, X.R. Li, "A comparative study of nonlinear filters for target tracking in mixed coordinates", *Proc. 42nd IEEE Conf. Southeastern Symposium on System Theory*, 2010, pp. 202-207.

- [91] W. Xiao, J. Wu, L. Shue, Y. Li, L. Xie, "A Prototype Ultrasonic Sensor Network for Tracking of Moving Targets", *1st IEEE Conf. Industrial Electronics and Applications*, May. 2006, pp. 1511-1516.
- [92] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control* Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [93] X. Kai, C. Wei, L. Liu, "Robust Extended Kalman Filtering for Nonlinear Systems With Stochastic Uncertainties ", *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 40, no. 2, pp. 399-405, 2010.
- [94] M.S. Schlosser, K. Kroschel, "Limits in tracking with extended Kalman filters ", *IEEE Trans. Aerospace and Electronic Systems*, vol. 40, no. 4, pp. 1351-1359, 2004.
- [95] Y. Fu, Z. Tian, "Cramer-Rao Bounds for Hybrid TOA/DOA-Based Location Estimation in Sensor Networks", *IEEE Signal Processing Letters*, vol. 16, no. 8, pp. 655-658, 2009.
- [96] H. Shi, X. Li, Y. Shang, D. Ma, "Cramer-Rao Bound Analysis of Quantized RSSI Based Localization in Wireless Sensor Networks", *Proc. IEEE 11th Int. Conf. Parallel and Distributed Systems*, Jul. 2005, pp. 32-36.
- [97] E.G. Larsson, "Cramer-Rao bound analysis of distributed positioning in sensor networks", *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 334-337, 2004.
- [98] C. Chang; A. Sahai, "Estimation bounds for localization", *IEEE 1st Annual Communications Society Conf. Sensor and Ad Hoc Communications and Networks*, Jan. 2004, pp. 415-424.

- [99] O.H. Kwon, H.J. Song, "Localization through Map Stitching in Wireless Sensor Networks", *IEEE Tran. Parallel and Distributed Systems*, vol. 19, no.1, pp. 93-105, 2008.
- [100] H. Maehara, "Distance Graphs in Euclidean Space." *Ryukyu Math. J.* no. 5, pp. 33-51, 1992.

Appendices

Appendix A

Relation of Range Error Factor and Standard Deviation of RSSI Measurement

The measured distance from RSSI in LLA is given in (3.6):

$$\tilde{d}_{ij} = c \cdot 10^{\frac{P_0 - P_{ij}}{10\alpha}} \quad (\text{AA.1})$$

where, P_{ij} is the received RSSI value; P_0 is the received RSSI value for 1m distance and α is the path-loss exponent; c is the correct factor to reduce the bias of $\tilde{d}_{ij}$.

Assume P_{ij} has white Gaussian noise, i.e.

$$P_{ij} \sim N(\bar{P}_{ij}, \sigma_R^2) \quad (\text{AA.2})$$

Take natural logarithm of both sides of (AA.1)

$$\ln(\tilde{d}_{ij}) = \ln(c) + \frac{\ln(10)P_0}{10\alpha} - \frac{\ln(10)P_{ij}}{10\alpha} \quad (\text{AA.3})$$

From (AA.2) and (AA.3), $\tilde{d}_{ij}$ is log-normally distributed, i.e.:

$$\ln(\tilde{d}_{ij}) \sim N(\ln(c) + \frac{\ln(10)P_0}{10\alpha} - \frac{\ln(10)\bar{P}_{ij}}{10\alpha}, (\frac{\ln(10)}{10\alpha}\sigma_R)^2) \quad (\text{AA.4})$$

From (AA.4), we can obtain the mean and variance of $\tilde{d}_{ij}$ as follows,

$$\begin{aligned} E(\tilde{d}_{ij}) &= \exp(\ln(c) + \frac{\ln(10)P_0}{10\alpha} - \frac{\ln(10)\bar{P}_{ij}}{10\alpha} + \frac{1}{2}(\frac{\ln(10)}{10\alpha}\sigma_R)^2) \\ &= c \cdot \exp(\frac{1}{2}(\frac{\ln(10)}{10\alpha}\sigma_R)^2)d_{ij} \end{aligned} \quad (\text{AA.5})$$

$$\begin{aligned} \text{var}(\tilde{d}_{ij}) &= (\exp((\frac{\ln(10)}{10\alpha}\sigma_R)^2) - 1) \cdot \exp(2(\ln(c) + \frac{\ln(10)P_0}{10\alpha} - \frac{\ln(10)\bar{P}_{ij}}{10\alpha}) + (\frac{\ln(10)}{10\alpha}\sigma_R)^2) \\ &= c^2 \cdot \exp((\frac{\ln(10)}{10\alpha}\sigma_R)^2) \cdot (\exp((\frac{\ln(10)}{10\alpha}\sigma_R)^2) - 1) \cdot d_{ij}^2 \end{aligned} \quad (\text{AA.6})$$

After the bias reduction,

$$E(\tilde{d}_{ij}) = c \cdot \exp(\frac{1}{2}(\frac{\ln(10)}{10\alpha}\sigma_R)^2)d_{ij} = d_{ij} \quad (\text{AA.7})$$

(AA.7) results the equation of the correction factor c ,

$$c = \exp(-\frac{1}{2}(\frac{\ln(10)}{10}\frac{\sigma_R}{\alpha})^2) \quad (\text{AA.8})$$

Substitute (AA.8) into (AA.6),

$$\text{var}(\tilde{d}_{ij}) = (\exp((\frac{\ln(10)}{10\alpha}\sigma_R)^2) - 1) \cdot d_{ij}^2 \quad (\text{AA.9})$$

The range error factor is defined as the ratio of the measured distance to the real distance, i.e.

$$\gamma = \sqrt{\text{var}(\tilde{d}_{ij}) / d_{ij}} \quad (\text{AA.10})$$

The substitution of (AA.9) into (AA.10) results (5.22),

$$\gamma = \sqrt{\exp\left(\left(\frac{\sigma_R \ln 10}{10\alpha}\right)^2\right) - 1} \quad (\text{AA.11})$$

Appendix B

The Covariance Matrix of Process Noise

The motion model of The DA is given in (5.10):

$$\Phi(i+1) = \Phi(i) + \mathbf{u}(i+1) + \mathbf{w}(i+1) \quad (\text{AB.1})$$

Rearrange (AB.1), we have

$$\begin{aligned} \mathbf{w}(i+1) &= \Phi(i+1) - \Phi(i) - \mathbf{u}(i+1) \\ &= [v_{i+1} \cos(\varphi_{i+1}) - \exp(-\frac{\sigma_\varphi^2}{2}) \tilde{v}_{i+1} \cos(\tilde{\varphi}_{i+1}) \quad v_{i+1} \sin(\varphi_{i+1}) - \exp(-\frac{\sigma_\varphi^2}{2}) \tilde{v}_{i+1} \sin(\tilde{\varphi}_{i+1})]^T \end{aligned} \quad (\text{AB.2})$$

Let $\tilde{\mathbf{v}}(i+1) = [\tilde{v}_{i+1} \quad \tilde{\varphi}_{i+1}]^T$, and

$$\mathbf{w}(i+1) = \tilde{\mathbf{f}}_{i+1}(\tilde{\mathbf{v}}(i+1)) \quad (\text{AB.3})$$

By the Taylor expansions for the second moment of function of random variables,

$$\text{cov}(\mathbf{w}(i+1)) \approx \tilde{\mathbf{F}}_{i+1} \text{cov}(\tilde{\mathbf{v}}(i+1)) \tilde{\mathbf{F}}_{i+1}^T \quad (\text{AB.4})$$

where $\tilde{\mathbf{F}}_{i+1}$ is the Jacobian matrix of $\tilde{\mathbf{f}}_{i+1}(\tilde{\mathbf{v}}(i+1))$ with respect to $\tilde{\mathbf{v}}(i+1)$,

$$\tilde{\mathbf{F}}_{i+1} = \exp(-\frac{\sigma_\varphi^2}{2}) \begin{bmatrix} -\cos(\tilde{\varphi}_{i+1}) & \tilde{v}_{i+1} \cdot \sin(\tilde{\varphi}_{i+1}) \\ -\sin(\tilde{\varphi}_{i+1}) & -\tilde{v}_{i+1} \cdot \cos(\tilde{\varphi}_{i+1}) \end{bmatrix} \quad (\text{AB.5})$$

Since the measurements of the pedometer and compass are uncorrelated, the covariance the DA measurement can be found as:

$$\text{cov}(\tilde{\mathbf{v}}(i+1)) = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\varphi^2 \end{bmatrix} \quad (\text{AB.6})$$

Substitute (AB.5) and (AB.6) into (AB.4), we can find the covariance matrix of $\mathbf{w}(i+1)$ as in (5.12)

$$\mathbf{Q}(i+1) = \exp(-\sigma_\varphi^2) \cdot \begin{bmatrix} \sigma_v^2 \cos^2(\tilde{\varphi}_{i+1}) + \sigma_\varphi^2 \tilde{v}_{i+1}^2 \sin^2(\tilde{\varphi}_{i+1}) & \sigma_v^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) - \sigma_\varphi^2 \tilde{v}_{i+1}^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) \\ \sigma_v^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) - \sigma_\varphi^2 \tilde{v}_{i+1}^2 \cos(\tilde{\varphi}_{i+1}) \sin(\tilde{\varphi}_{i+1}) & \sigma_v^2 \sin^2(\tilde{\varphi}_{i+1}) + \sigma_\varphi^2 \tilde{v}_{i+1}^2 \cos^2(\tilde{\varphi}_{i+1}) \end{bmatrix} \quad (\text{AB.7})$$

Appendix C

Information Matrix with Negative Constraint Information

The log likelihood function of sensor nodes' positions with both positive and negative constraint information is given as:

$$l(\mathbf{X}) = -\sum_{i=1}^{N_1} \sum_{j=i+1}^N (w_{ij} \left(\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2} - \ln(b_{ij}) \right) + (w_{ij} - 1) \ln \left(\frac{1}{2} + \frac{1}{2} \operatorname{erf} \left(\frac{\ln d_{ij}(\mathbf{X}) - \ln \bar{R}}{\sqrt{2}\sigma} \right) \right)) \quad (\text{AC.1})$$

The likelihood function can be written into two parts, the positive constraint part and negative constraint part.

$$l(\mathbf{X}) = pl(\mathbf{X}) + nl(\mathbf{X}) \quad (\text{AC.2})$$

where

$$pl(\mathbf{X}) = -\sum_{i=1}^{N_1} \sum_{j=i+1}^N (w_{ij} \left(\frac{(\ln d_{ij}(\mathbf{X}) - \ln \tilde{d}_{ij})^2}{2\sigma^2} - \ln(b_{ij}) \right)) \quad (\text{AC.3})$$

$$nl(\mathbf{X}) = -\sum_{i=1}^{N_1} \sum_{j=i+1}^N (w_{ij} - 1) \ln \left(\frac{1}{2} + \frac{1}{2} \operatorname{erf} \left(\frac{\ln d_{ij}(\mathbf{X}) - \ln \bar{R}}{\sqrt{2}\sigma} \right) \right) \quad (\text{AC.4})$$

The Fisher information matrix (FIM) $\mathbf{F}_X$ is defined as:

$$\begin{aligned}\mathbf{F}_{\mathbf{X}} &= -E(\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}l(\mathbf{X})))^T \\ &= -E(\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}pl(\mathbf{X})))^T - E(\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}nl(\mathbf{X})))^T\end{aligned}\quad (\text{AC.5})$$

The positive constraint part $-E(\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}pl(\mathbf{X})))^T$ is the FIM without negative constraint information that can be found in [36]. Here, we will only focus on the negative constraint part. Since there is no random variables in $nl(\mathbf{X})$, $-E(\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}nl(\mathbf{X})))^T = -\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}nl(\mathbf{X}))^T$.

Let,

$$f_{ij}(\mathbf{X}) = \ln\left(\frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{\ln\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - \ln \bar{R}}{\sqrt{2}\sigma}\right)\right) \quad (\text{AC.6})$$

$-\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}nl(\mathbf{X}))^T$ can be found as follows,

$$-\nabla_{\mathbf{X}}(\nabla_{\mathbf{X}}nl(\mathbf{X}))^T = \begin{bmatrix} \mathbf{NF}_{xx} & \mathbf{NF}_{xy} \\ \mathbf{NF}_{xy}^T & \mathbf{NF}_{yy} \end{bmatrix} \quad (\text{AC.7})$$

where,

$$\begin{aligned}\mathbf{NF}_{xx} &= \begin{bmatrix} \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_1^2} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_1 \partial x_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_1 \partial x_{N_1}} \\ \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_2 \partial x_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_2^2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_2 \partial x_{N_1}} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_{N_1} \partial x_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_{N_1} \partial x_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_{N_1}^2} \end{bmatrix} \\ \mathbf{NF}_{xy} &= \begin{bmatrix} \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_1 \partial y_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_1 \partial y_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_1 \partial y_{N_1}} \\ \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_2 \partial x_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_2 \partial y_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_2 \partial y_{N_1}} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_{N_1} \partial y_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_{N_1} \partial y_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial x_{N_1} \partial y_{N_1}} \end{bmatrix}\end{aligned}$$

$$\mathbf{NF}_{yy} = \begin{bmatrix} \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_1^2} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_1 \partial y_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_1 \partial y_{N_1}} \\ \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_2 \partial y_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_2^2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_2 \partial y_{N_1}} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_{N_1} \partial y_1} & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_{N_1} \partial y_2} & \cdots & \sum_{i=1}^{N_1} \sum_{j=i+1}^N (1-w_{ij}) \frac{\partial^2 f_{ij}}{\partial y_{N_1}^2} \end{bmatrix} \quad (\text{AC.8})$$

The first derivatives of f_{ij} are

$$\begin{aligned} \frac{\partial f_{ij}}{\partial x_i} &= \frac{2}{\sqrt{2\pi}\sigma} e^{-t_{ij}^2} \frac{1}{1+\text{erf}(t_{ij})} \frac{x_i - x_j}{d_{ij}^2} \\ \frac{\partial f_{ij}}{\partial y_i} &= \frac{2}{\sqrt{2\pi}\sigma} e^{-t_{ij}^2} \frac{1}{1+\text{erf}(t_{ij})} \frac{y_i - y_j}{d_{ij}^2} \end{aligned} \quad (\text{AC.9})$$

where $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ and $t_{ij} = \frac{\ln(d_{ij}) - \ln(\bar{R})}{\sqrt{2}\sigma}$.

The second derivatives of f_{ij} are,

$$\begin{aligned} \frac{\partial f_{ij}}{\partial x_i^2} &= n_{ij}^{[xx]} \\ \frac{\partial f_{ij}}{\partial x_i \partial x_j} &= \frac{\partial f_{ij}}{\partial x_j \partial x_i} = -n_{ij}^{[xy]} \\ \frac{\partial f_{ij}}{\partial y_i^2} &= n_{ij}^{[yy]} \\ \frac{\partial f_{ij}}{\partial y_i \partial y_j} &= \frac{\partial f_{ij}}{\partial y_j \partial y_i} = -n_{ij}^{[yx]} \end{aligned} \quad (\text{AC.10})$$

$$\frac{\partial f_{ij}}{\partial x_i \partial y_i} = \frac{\partial f_{ij}}{\partial y_i \partial x_i} = n_{ij}^{[xy]}$$

$$\frac{\partial f_{ij}}{\partial x_i \partial y_j} = \frac{\partial f_{ij}}{\partial y_j \partial x_i} = -n_{ij}^{[xy]}$$

$$\frac{\partial f_{ij}}{\partial x_k \partial x_l} = \frac{\partial f_{ij}}{\partial y_k \partial y_l} = 0, \quad \text{if } k \neq i, k \neq j \text{ or } l \neq i, l \neq j$$

where,

$$n_{ij}^{[xx]} = c_{ij} \left(1 - \frac{(x_i - x_j)^2}{d_{ij}^2} \right) m_{ij}$$

$$n_{ij}^{[yy]} = c_{ij} \left(1 - \frac{(y_i - y_j)^2}{d_{ij}^2} \right) m_{ij}$$

$$n_{ij}^{[xy]} = -c_{ij} \frac{(x_i - x_j)(y_i - y_j)}{d_{ij}^2} m_{ij}$$

$$m_{ij} = \frac{\ln(d_{ij}) - \ln(\bar{R})}{\sigma^2} + 2 + \frac{2e^{-t_{ij}^2}}{\sqrt{2\pi}\sigma(1 + \text{erf}(t_{ij}))} \quad \text{and} \quad c_{ij} = \frac{e^{-t_{ij}^2}}{\pi\sigma d_{ij}^2(1 + \text{erf}(t_{ij}))}$$

Let $[\mathbf{NF}_{xx}]_{k,l}$ be the k -th row and l -th column element of matrix $\mathbf{NF}_{xx}$. Substitute (AC.10)

to (AC.8), we have:

$$[\mathbf{NF}_{xx}]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (1 - w_{kj}) n_{kj}^{[xx]}, & k = l \\ (w_{kl} - 1) n_{kl}^{[xx]}, & k \neq l \end{cases}$$

$$[\mathbf{NF}_{\mathbf{xy}}]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (1 - w_{kj}) n_{kj}^{[xy]}, & k = l \\ (w_{kl} - 1) n_{kl}^{[xy]}, & k \neq l \end{cases} \quad (\text{AC.11})$$

$$[\mathbf{NF}_{yy}]_{k,l} = \begin{cases} \sum_{j=1, j \neq k}^N (1 - w_{kj}) n_{kj}^{[yy]}, & k = l \\ (w_{kl} - 1) n_{kl}^{[yy]}, & k \neq l \end{cases}$$

FIM with NC information is found as the summation of the FIM without NC and $\mathbf{NF}$,
i.e.:

$$\mathbf{F}_{\mathbf{X}} = \mathbf{F}_{\mathbf{X}}^p + \mathbf{NF} \quad (\text{AC.12})$$

The substitution of (6.20) and (AC.11) into (AC.12), will result the $\mathbf{F}_{\mathbf{X}}$'s expression in
(6.21).