Pure RSSI Based Low-Cost Self-localization System for ZigBee WSN

by

Philip Lin
B.A.Sc., Simon Fraser University, 2011

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

In the
School of Engineering Science
Faculty of Applied Science

Simon Fraser University

Summer 2011

Approval

Name: **Philip Lin**

Degree: **Master of Applied Science**

Title of Thesis: **Pure RSSI based Low-cost Self-localization System for ZigBee WSN**

Examining Committee:

Chair: **Dr. Parvaneh Saeedi**

Assistant Professor, School of Engineering Science

_____

Dr. Bozena Kaminska

Senior Supervisor

Professor, School of Engineering Science

_____

Dr. Mehrdad Moallem

Supervisor

Associate Professor, School of Engineering Science

_____

Dr. Kamal Gupta

Internal Examiner

Associate Professor, School of Engineering Science

Date Defended/Approved: August 2, 2011

Abstract

In modern wireless sensor networks (WSN) applications, location awareness has been one of the features that attracted many research interests. Various applications utilize location information for surveillance and asset tracking purposes. Common WSN localization systems use radio frequency (RF), ultra-sound, or laser devices to provide range information, whose node positions are to be determined by various algorithms accordingly. Multi-dimensional scaling (MDS) is one of the most common algorithms for transforming inter-node distances into node positions in Cartesian coordinates. However, MDS algorithm, by nature, has a cubic computational complexity. Also, the algorithm's ability to localize is restricted to fully connected WSNs, where every node sees every other node. This thesis proposes a low-cost pure RF based localization system, implemented with a novel clustering MDS algorithm. Its most attractive feature is its ability to localize a partially connected WSN with a linear computation complexity without sacrificing the localization accuracy. In this thesis, we review various localization techniques and conduct experiments to compare the clustering MDS' performance against the classical MDS' and GPS'. The localization with a commercial GPS, although, better than the above two methods, has also introduced significant discrepancy. At the end, we have demonstrated that RF localization in our low-cost system does not deliver GPS-grade accuracy, but its ability to localize partially-connected WSN and low computation complexity have outperformed the classical MDS approach.

Acknowledgement

I would like to thank the following people for the contribution to this thesis. Throughout my master's study, with the help of these individuals, I have obtained valuable knowledge and skills to advance to the next stage in my life.

I would like to thank my supervisor, Dr. Bozena Kaminska for accepting me into such a strong and renowned research lab and for providing me with abundant lab resources and knowledgeable research fellows, Dr. Jens Wawerla and Dr. Marcin Marzencki. With your valuable guidance and advice, I have had the best school years in my life.

I would also like to thank my examining committee, Dr. Mehrdad Moallem and Dr. Kamal Gupta, for taking part in my thesis defense and for providing valuable suggestions.

My special appreciation goes to my research colleagues, Jian Guo and Thomas Cho, who have contributed their personal time to help me conduct field test, gathering data for my research.

Last, my sincere appreciation will go to my all my family member and my girl friend, Christine Tseng, who have been supporting me throughout my studies. Without them, I will not be able to enjoy my research years as much as I have.

Table of Content

List of Tables

List of Figures

List of Abbreviations

AF	Application Framework
AOA	Angle of Arrival
APIT	Approximate Point-In-triangulation Test
APS	Ad-hoc Positioning System
BS	Base Station
CID	Cluster ID
CMDS	Classical Multi-dimensional Scaling
COG	Centre of Gravity
DV	Distance Vector
dwMDS	Distributed Weighted MDS
EM	Electromagnetic
GPS	Global Positioning System
HMM	Hidden Markov Model
ICP	Iterative Closest Point
LOS	Line of Sight
LS	Least Squared
MAP	Maximum a Posteriori
MDS	Multi-dimensional Scaling
MLE	Maximum Likelihood Estimator
MS	Mobile Station
MSREs	Square Range Errors
OSAL	OS Abstraction Layer
PAN	Personal Area Network
PDF	Probability Distribution Function
PDM	Proximity Distance Map
RF	Radio Frequency
RMU	Range Modeling Unit
RSSI	Received Signal Strength Indicator
SOC	System-on-a-chip
SOSC	Self-organizing and Self-calibrating
SPE	Stochastic Proximity Embedding
SVD	Singular Value Decomposition
TDOA	Time Difference of Arrival
TOA	Time of Arrival
TSVD	Truncated Singular Value Decomposition
WLS	Weighted Least Squared
WSN	Wireless Sensor Networks

Chapter 1 - Introduction

1.1 Ranging Problems in Localization Systems

With the advancement in wireless technologies, wireless sensor networks (WSN) have recently been widely researched in areas of context-aware and location-aware applications. Context-awareness and location-awareness for a WSN provide the network with the ability to determine ever-changing conditions in a real-time environment. One of the most discussed and researched area in the above two types of WSN applications is localization. The sole purpose of localization in WSN is to facilitate network installation and to perform network monitoring at any given area of interest within the network perimeter. The localization ability granted to sensor nodes within the network allows the flexibility of reconfiguring node layouts to accommodate for any environmental settings without manually updating the new node locations.

Localization is the technique that determines object positions from any given distance or proximity information with respect to pre-defined reference points. Prior to providing meaningful location information of an object, a global reference has to be defined. For example, the most common reference used in modern localization application is global positioning system (GPS) coordinates. However, for low-cost localization systems, like [1], that do not have the luxury of equipping GPS, it is common to define the location reference in 2-D Cartesian space. Later in the related work section, we will discover that in order to localize nodes in WSN, most localization techniques are required to have knowledge of

the inter-node distances. The objects' positions can then be computed with a distance to position coordinate transformation algorithm, such as multi-dimensional scaling (MDS) algorithm [2], which will be discussed later in this thesis.

In WSN localization applications, obtaining inter-node distance information is essential prior to performing node localization. Some of the most common distance measuring techniques, described in [3], utilize ultra-sound or laser. Range measurement with the above sources can be determined by the time a signal travels from one node to another. This ranging technique is known as time of arrival (TOA) and is also a commonly used technique in tracking applications, such as Radar [4]. However, WSN localization systems utilizing TOA can be costly and complex, due to additional source generation hardware and algorithm to perform the processing. Also, TOA requires line of sight (LOS) communications between nodes, making this technique limited to outdoor or obstacle-free environments to provide accurate localization.

Without introducing additional ranging devices to localize WSN nodes, ranging with RF signals are also widely researched and discussed. Some of the common RF ranging techniques are time difference of arrival (TDOA), angle of arrival (AOA), and received signal strength indicator (RSSI) ranging. TDOA measures distance between two nodes by multiplying the difference of time traveled from one node to another with the speed of light. This technique requires clocks on both nodes to be synchronized and to be precise enough to measure the subtle difference in time of travel. However, integrating an on-board clock or

timer with high precision can be very costly. AOA incorporates directional antennas on nodes to detect the angle of the arriving signal, and it localizes nodes based on the direction of RF signal propagation incident on the antennas. Same as TDOA, AOA requires additional antennas to localize, and therefore not only increasing the cost, but also increasing system complexity. The RSSI ranging technique is the simplest ranging technique, in terms of system complexity and cost, among all the techniques. RSSI ranging is done by modeling the RF signal propagation over distance. In theory, the RF propagation model demonstrates the monotonic degradation of RF signal strength over distance. However, in practice, as many researches, [5] and [6] have discovered, due to variety of electromagnetic (EM) interferences and reflection, RSSI values do not always reflect its corresponding ground-truth distances.

The ranging techniques discussed above are known as the range based measurement techniques. Techniques, known as the range-free based measurement techniques, are also common implementations in WSN localization systems. Most range-free localization is designed to provide proximity detection based on hop counts. Ranging based on hop-counts only depends on communication topology of the WSN and does not require additional ranging devices. Range free localization systems are low in system complexity, but they provide poor localization resolution.

Localization accuracy is highly dependent on range measurements. A fine ranging technique results in high reliability in position computation. However, every ranging technique exhibits limitations depending on the environment

settings and interferences. As a result, the objective of this thesis is to provide a versatile, low-cost, low-power localization system. This thesis proposes a pure RSSI based localization system that provides better localization performance and a more cost-effective approach than the conventional RSSI based localization methodology in situations where distance information is faulty and missing due to disconnected communication between node pairs.

In later sections, we will discuss in detail the design implementation of our localization system. Unlike many of the related works, we propose a localization system that requires no additional hardware and utilizes only information provided by RF signals to do both ranging and localization with reduced computational complexity to accommodate for low-cost and low-power applications.

1.2 The Self-Organizing and Self-Calibrating (SOSC) WSN

Accuracy of range measurement between each node pair within a WSN poses an essential problem to the accuracy of node localization. To address the instability issue of the RSSI, caused by EM interferences, the localization system has to have the ability to reduce the influence of the faulty inter-node RSSI values and to localize the whole network based on the relatively reliable RSSI values. The solution to the above problem is called the SOSC WSN, which localizes the network nodes with the following features

- Real-time modeling of the RSSI to distance relationship; it is designed to self-calibrate to accommodate for EM interference and noise

- A self-organized system and network architecture that allows the nodes within the WSN to communicate to retrieve inter-node distance data
- MDS-based position computation and mapping algorithm that is used to translate the inter-node RSSI values into their corresponding 2-D positions

The SOSC WSN consists of a range modeling unit (RMU), which in real-time, computes parameters of a radio propagation model, which translates the inter-node RSSI values into a distance estimate. The RMU provides a real-time distance metric for RSSI in the environment to cope with the ever changing random noise and interference. With the distance metric defined, the SOSC system utilizes the novel MDS-based algorithm, proposed in this thesis, for commutating and reconstructing the ground truth network layout. The above components of SOSC WSN allow the nodes to localize themselves with simply a radio. The radio, in this case, is a 2.4 GHz on-chip radio, provided by the Texas Instrument (TI) CC2530 system-on-a-chip (SOC). This radio is designed to be compatible with the ZigBee standards. This choice of hardware allows SOSC WSN to be a ZigBee based ad-hoc network, where each node within the network participates in routing and forwarding data for other nodes by both single hop and multi-hop communications. The ZigBee network topology allows the SOSC nodes to communicate and localize one another even in regions with partial connectivity.

1.3 Scope and Outline of Thesis

The primary focus of this thesis is the design implementation of a low-cost, low-power, and low system-complexity localization system based on a ZigBee ad-hoc

mesh network [7]. This thesis also discusses the merits and weaknesses of several popular WSN localization systems. Due to the inaccuracy of RSSI ranging measurement in nature, this thesis is not concerned with comparing the localization accuracy with systems utilizing more sophisticated and refined distance measurement techniques, such as ultra-sound or laser ranging. The design approach of the localization and mapping algorithm is derived from RSSI to distance relationship experimented and realized by several literatures. Therefore, this thesis is not concerned with reproducing works of characterizing RSSI to distance relationship; instead, it builds on results from the literature to support the algorithm design and to demonstrate the validity of the approach with an extensive experiment.

Chapter 2 provides brief background information on ZigBee standard and its communication topology that allows the sensor nodes to obtain the distance information from one another; the background also covers the model and techniques necessary to translate the inter-node distance information and to reconstruct the physical network layout respectively. Chapter 3 is a review of the related work done on WSN localization. This chapter categorizes localization into two categories, range-based and range-free localizations. This chapter discusses the localization performance and implementation practicality among the above various localization algorithms and systems. Chapter 4 covers the design of the SOSC system, in terms of the network, system architecture. Chapter 5 discusses the localization and mapping algorithm that allow the SOSC WSN to self-localize with sufficient amount of accuracy. Chapter 6 describes the SOSC system's self-

localizing ability in real-life scenarios to demonstrate that this novel localization approach is able to reduce the inaccuracy generated by RSSI ranging to achieve an equally good accuracy that enables localization in partially connected network with less computation complexity. Chapter 7 will conclude and discuss the future work for this thesis.

Chapter 2 - Background information

The SOSC WSN is designed based on wireless mesh network topology. With the use of novel MDS based algorithm along with the network clustering technique, the SOSC WSN is capable of computing its node locations under various environmental settings. The above localization technique proposed in this thesis is called the clustering MDS localization algorithm. In this background information overview, three major components underlying the design and implementation of the SOSC WSN will be discussed, and they are

- ZigBee mesh network topology and communication
- Mapping technique

  Distance measuring technique

2.1 ZigBee Communications Overview

ZigBee, [8], is a specification for low-cost and low-power network protocol based on IEEE 802.15.4 personal area network (PAN) protocol standard. The low-cost and low-power features make ZigBee to a suitable protocol for WSN applications, requiring high density of wireless sensor coverage and long operating time on sensor devices running on small batteries. A ZigBee network consists of exactly one coordinator, several routers, and end devices. A ZigBee coordinator initiates the network and defines the network PAN-ID, acting as a trust centre and repository for security keys; a ZigBee router can be configured to run applications and to also pass messages to other router devices within the same network; a ZigBee end device can act as a sensing device that communicates with routers

or coordinator, however it has no routing capability. ZigBee's network communication is based on a mesh network topology. With mesh networking, ZigBee achieves high reliability in data communication. A typical ZigBee mesh network is illustrated in the following figure

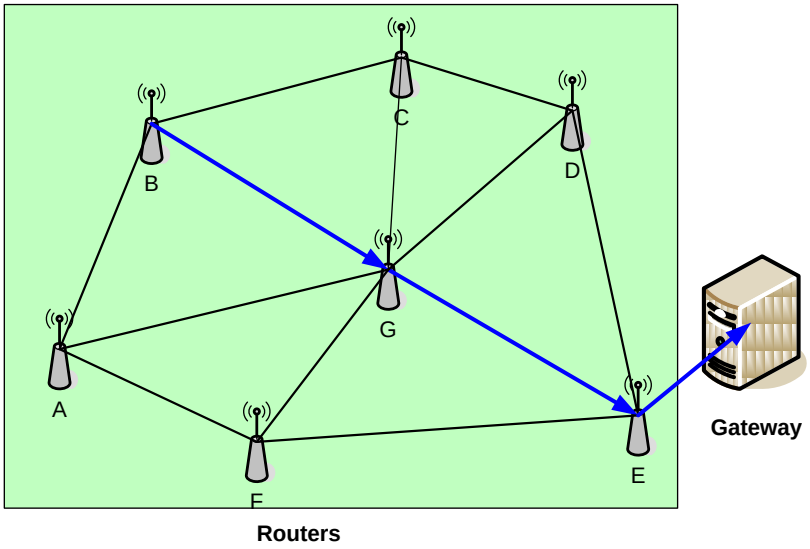

Figure 2-1: ZigBee Mesh Networking

In a ZigBee mesh network, all the wireless sensors are able to communicate with one another. Their messages can multi-hop and be routed through their neighboring router nodes to reach the destination. In ordinary situations, ZigBee routers in the network discover the "shortest path" to relay the message to its destination, which in most cases, being the gateway or the devices that are capable of storing and computing network information.

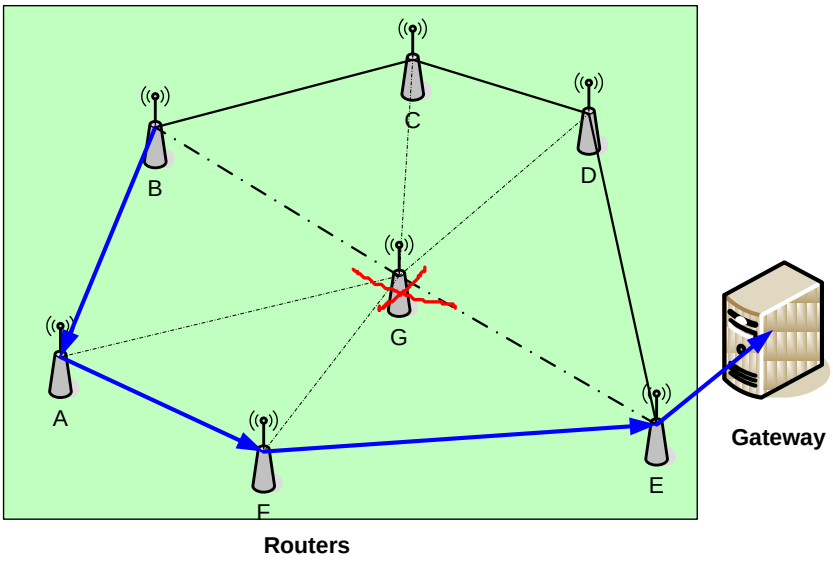

Figure 2-2: ZigBee Mesh Networking Discovering Path

ZigBee, being a mesh network, is robust in terms of data communication because messages can be routed through any available router within the network. In cases where any of the routers loses connectivity to the network, shown in Figure 2-2, the message will be detoured to the next available “shortest path”, the path with the least communication cost, to reach the gateway. As a result, communication within a ZigBee network can still allow messages to reach the gateway in a partially connected network.

In a ZigBee network, all devices can communicate with other devices by broadcasting or by uni-casting. Establishing successful communications between node pairs requires the following addressing components

Table 2-1: ZigBee Network Addressing Components

Name	Description
Channel	ZigBee contains 16 channels in the physical RF spectrum
PAN ID	The address of a network within a particular channel
NWK Address	The address of a sensor node within the network
Application Endpoint	The address of an application within the network
Cluster ID (CID)	An Message identifier for exchanging information within an application

From an application's point of view, one may deploy different ZigBee networks at various locations for separate monitoring or sensing purposes. To differentiate one network from the other, network channels and PAN IDs are designated to each network to allow independent network operation. Within each individual network, each wireless node communicates with one another by application endpoints, specifying the application which the information belongs to, and by CID, specifying the action to execute within a particular application. The network address (NWK address), identifies the destination address of the wireless node a particular message is directed to.

Any ZigBee node within a network can perform two different addressing modes to serve different application purposes and to exchange information with other nodes. The two addressing modes are

- **Broadcast:** A broadcast message is intended to be received by all the wireless modules within the network. Broadcasting is initiated by any of the nodes transmitting a message to its neighboring nodes, and the message is repeated and forwarded to the next nearby node until every node within the network receives the message.

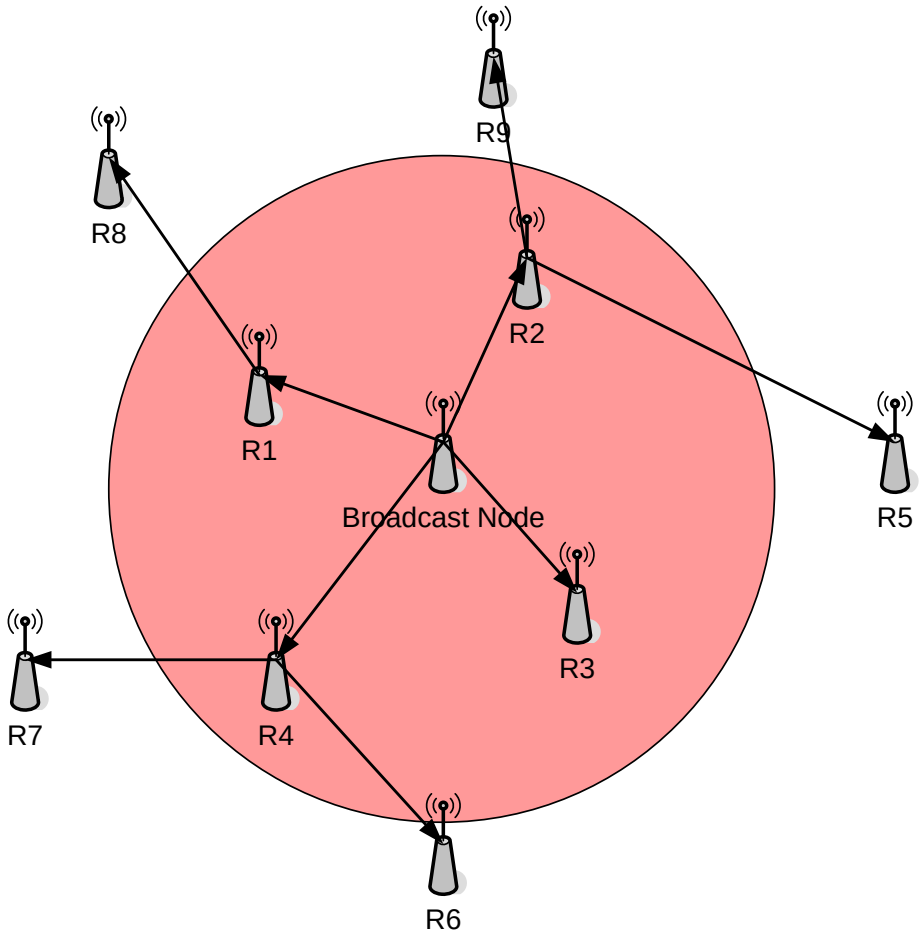

Figure 2-3: ZigBee Broadcasting Mode

In Figure 2-3, the red shaded circle is the transmission range of the broadcast node. The multi-hop nature of ZigBee protocol allows the broadcast message to reach devices outside of the range. The degree of message forwarding, however, can be configured by controlling the number of hops allowed to cover the network. This regional broadcasting scheme is useful in applications where nodes are only interested in exchanging information with its neighboring nodes.

- **Uni-cast:** Unlike broadcasting, uni-casting is intended for a device-to-device direct communication. Uni-casting is an effective way of communication, because this messaging scheme does not occupy

communication channel of every node within the network. Figure 2-4 is an example of uni-casting, where the node performing uni-casting is to transmit the message directly to the node, R5's address. R5, however, is out of the transmission range of the uni-cast node, and thus requiring an intermediate router node, R2 to forward the message.

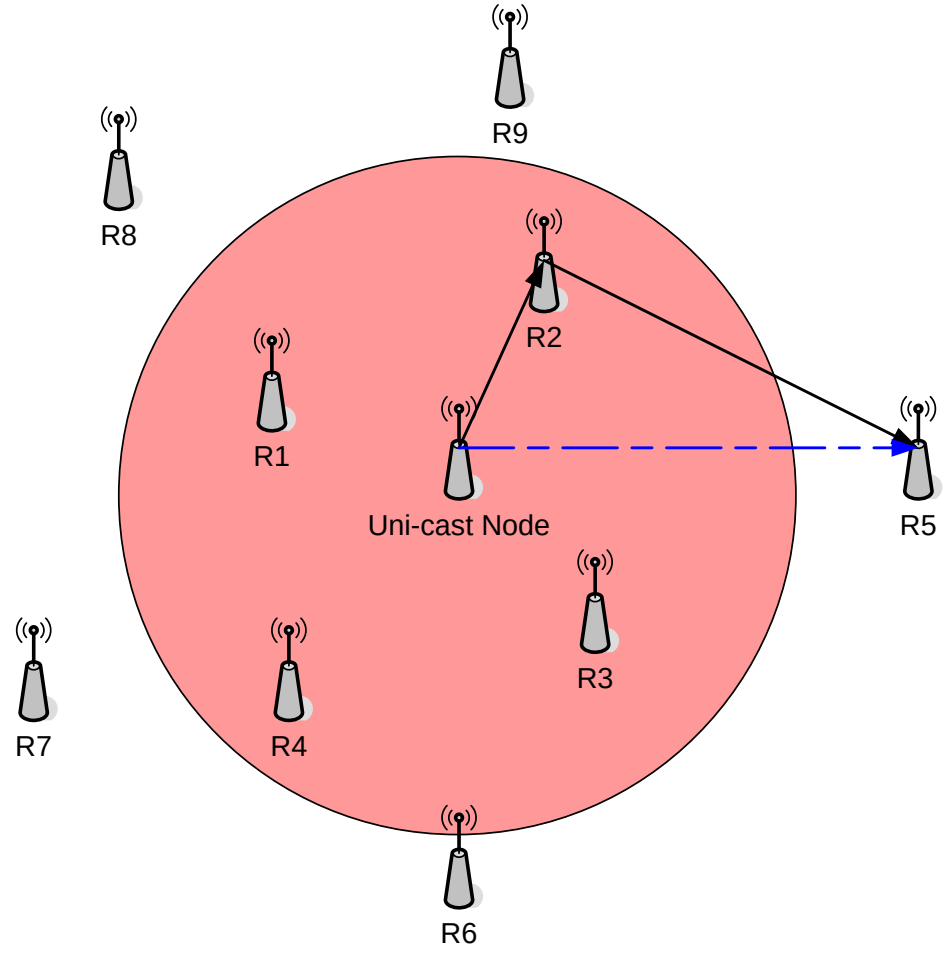

Figure 2-4: ZigBee Uni-casting Mode

In the above case, only R2 is occupied for routing the message to R5. This implies that the rest of the nodes are still able to communicate with one another to exchange information without being interrupted.

2.2 MDS Algorithm

As described in [2], multidimensional scaling (MDS) is a method that transforms the similarity or dissimilarity, represented in Euclidean distances among pairs of objects, into lower dimensional multidimensional space. In our particular application, the objects are the wireless sensor nodes, and the dissimilarity of interest is the physical distance among every pair of nodes within a SOSC WSN. One of main purposes of utilizing MDS algorithm for data analysis is to provide visualization of object distribution in space. For example, one may want to visualize the locations of individual cities in Vancouver, BC, on a 2-D map from the given distances between those cities. Table 2-2 shows the table of dissimilarity between each city pair in metro Vancouver, and the plot in Figure 2-5 represents the two-dimensional MDS spatial distribution between the cities.

Table 2-2: Inter-city Distance Matrix

City	Vancouver	Burnaby	Coquitlam	Richmond	Delta
Vancouver	0	10	26	10	27
Burnaby	10	0	8	24	17
Coquitlam	26	8	0	23	15
Richmond	10	24	23	0	16
Delta	27	17	15	16	0

Figure 2-5: Vancouver City Layout

The objective of utilizing MDS is to transform the complex dissimilarity into a simple visual layout, like Figure 2-5. In two-dimensional space and to also represent object locations in Cartesian coordinates which can be universally converted into other coordinate standards, such as GPS coordinates. To understand the underlying theory of MDS, the following terminologies have to be defined

2.2.1 **Dissimilarity Matrix**

As mentioned earlier, dissimilarity or similarity of any pairs of objects indicates their likelihood. This likelihood is expressed in forms of distance metric. In classical MDS (CMDS), this distance is calculated with Euclidian distance formula

$$d_{ij} = \sqrt{\sum (x_i - x_j)^2} \quad (1)$$

Where d_{ij} is the inter-object distance between i^{th} and j^{th} object $x_1,\ldots,x_i \in \mathbb{R}^N$. The dissimilarity matrix is the table of all the inter-object distances, calculated for all pairs of $i,j \in I$. The resulting dissimilarity has the form

$$\Delta := \begin{pmatrix} d_{1,1} & \cdots & d_{1,I} \\ \vdots & \ddots & \vdots \\ d_{I,1} & \cdots & d_{I,I} \end{pmatrix} \tag{2}$$

Having the dissimilarity matrix, the question remains how to transform these inter-object distances into a coordinate standard to provide a visual representation of the objects' corresponding spatial distribution. A common method to solve the above problem is the following optimization

$$\min_{x1,\ldots,xI} \sum_{i<j} (\|x_i - x_j\| - d_{ij})^2 \tag{3}$$

The solution can be found by performing Eigen-decompositions on the matrix.

2.2.2 **Eigen-decomposition for MDS**

Eigen-decomposition is a technique in linear algebra to factorize a matrix into its Eigen vector and Eigen values. For a *NxN* square matrix *A*, using Eigen-decomposition, can be expressed as

$$Av = \lambda v \tag{4}$$

Where $\boldsymbol{v}$ and $\boldsymbol{\lambda}$ are the Eigen vector and Eigen value of matrix *A* respectively. **Error! Reference source not found.** can be rewritten with Eigen vector q_i, for (*i* 1,….,N). The new expression is shown below

$$A = Q\Lambda Q^{-1} \tag{5}$$

In MDS algorithm, matrix A is the dissimilarity matrix, whose distances are calculated from the set of vector of coordinates, ***X***. To perform the Eigen-decomposition for dissimilarity matrix, there are several steps to follow,

- **Calculate the squared dissimilarity matrix:** The square of a dissimilarity matrix can be expressed as $\mathbf{D}^{(2)}$, where

$$\boldsymbol{D}^{(2)} = \boldsymbol{c}\mathbf{1}' + \mathbf{1}\boldsymbol{c}' - 2\boldsymbol{X}\boldsymbol{X}' \tag{6}$$

The vector ***c*** is the vector containing the diagonal matrix elements of ***XX'***, which can be expressed as vector ***B***, also referred to as the matrix of scalar products.

- **Apply double centering to the squared dissimilarity matrix:** Double centering is a method to subtract row and column means of a matrix from its elements and to add back the grand mean. After applying double centering by multiplying the centering matrix ***J*** on both side of matrix $\boldsymbol{D}^{(2)}$ and multiplying the whole vector by a factor of $-\frac{1}{2}$, the squared dissimilarity matrix $\boldsymbol{D}^{(2)}$ becomes,

$$\begin{aligned} -\frac{\mathbf{1}}{\mathbf{2}}\mathbf{J}\mathbf{D}^{(2)}\mathbf{J} &= -\frac{\mathbf{1}}{\mathbf{2}}\mathbf{J}(\mathbf{c}\mathbf{1}' + \mathbf{1}\mathbf{c}' - 2\mathbf{X}\mathbf{X}')\mathbf{J} \\ &= -\tfrac{\mathbf{1}}{\mathbf{2}}\mathbf{J}\mathbf{c}\mathbf{1}'\mathbf{J} - \tfrac{\mathbf{1}}{\mathbf{2}}\mathbf{J}\mathbf{1}\mathbf{c}'\mathbf{J} + \tfrac{\mathbf{1}}{\mathbf{2}}\mathbf{J}(2\mathbf{B})\mathbf{J} \\ &= -\tfrac{\mathbf{1}}{\mathbf{2}}\mathbf{J}\mathbf{c}\mathbf{0}' - \tfrac{\mathbf{1}}{\mathbf{2}}\mathbf{0}\mathbf{c}'\mathbf{J} + \mathbf{J}\mathbf{B}\mathbf{J} = \mathbf{B} \end{aligned} \tag{7}$$

- **Calculate the Eigen-decomposition for matrix *B*:** To compute the MDS coordinates of matrix ***B***, the same factorization method stated in (5) is

applied to obtain the corresponding Eigen vectors and values. **Error! eference source not found.** can be rewritten as follows

$$\mathbf{B} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}' \tag{8}$$

$$= \left(\mathbf{Q}\boldsymbol{\Lambda}^{\frac{1}{2}}\right)\left(\mathbf{Q}\boldsymbol{\Lambda}^{\frac{1}{2}}\right)' = \mathbf{X}\mathbf{X}'$$

The final solution is determined by multiplying the first *m* positive Eigen values Λ_+ with the first m columns of the Eigen vectors $\boldsymbol{Q}_+$. The MDS coordinates, $\boldsymbol{X}$, becomes

$$\mathbf{X} = \mathbf{Q}_+\boldsymbol{\Lambda}_+^{\frac{1}{2}} \tag{9}$$

The above algorithm describes the procedure to factorize the inter-object distance matrix into their corresponding Eigen vectors and values. This solution is the best-fit, however, not unique, and it consists of rotational and translational offsets from the actual objects' spatial layout. The following section describes the procedure to transform the best-fit MDS solution into the solution that matches the actual inter-object layout in space.

2.3 Iterative Closest Point (ICP) Algorithm

MDS algorithm defines the best-fit solution only with respect to the dissimilarities among nodes within a network. However, visually speaking, the best-fit solution is meant to be fitted to the global ground truth reference. In real-world scenario, this ground truth reference is referred to as anchor points. The best-fit solution is produced by determining the optimal positions to register the MDS-computed node positions to match with the anchor points. This mapping optimization

procedure can be achieved by applying the ICP algorithm, developed by authors in [9]. ICP is performed by minimizing the following equation

$$\mathbf{E}(\mathbf{R},\mathbf{t}) = \sum_{\mathbf{i=1}}^{\mathbf{N_m}} \sum_{\mathbf{j=1}}^{\mathbf{N_d}} \mathbf{w_{i,j}} \left\| \mathbf{m_i} - (\mathbf{R d_j} + \mathbf{t}) \right\|^2 \quad (10)$$

Where $(\mathbf{R}, \mathbf{t})$ are rotation and translation required to be done to register the nodes into the most optimal positions. $\mathbf{w_{i,j}}$ is the weight, equal to 1, if $\mathbf{m_i}$ is the closest point to $\mathbf{d_j}$, otherwise $\mathbf{w_{i,j}} = \mathbf{0}$. This algorithm iteratively searches through the number of points in the model set $\mathbf{M}$ until the closest points are found. The final result produces the rotational and the translation matrix, whose theoretical backgrounds are described in the following section.

2.3.1 Rotational Matrix

As mentioned previously, MDS algorithm provides the best-fit coordinates, ***X***, based on the dissimilarity matrix ***B***. This best-fit solution, however, can contain rotational or translational ambiguity in space.

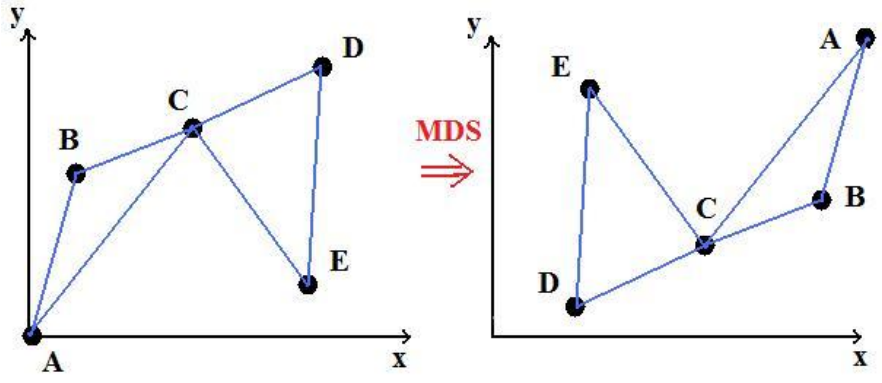

Figure 2-6: Rotational and Translational MDS Solution

The plot on the right in Figure 2-6 is the example of a best-fit solution computed by MDS algorithm. The plot on the left is the actual node layout in physical

space. It can be shown that after performing MDS on the dissimilarity matrix from the actual layout, the best-fit solution is rotated and translated. To obtain the actual layout from the solution provided by MDS algorithm, two or three anchor points have to be pre-defined. Anchor points provide fixed coordinates that limit the degree of freedom for the MDS solutions, so its solution can be correctly oriented to fit the actual object layout in physical space. A rotational matrix, ***R***, is generally defined as

$$\mathbf{R}(\boldsymbol{\theta}) = \begin{bmatrix} \cos\boldsymbol{\theta} & -\sin\boldsymbol{\theta} \\ \sin\boldsymbol{\theta} & \cos\boldsymbol{\theta} \end{bmatrix} \tag{11}$$

The above equation can rotate a set of object coordinates by an angle $\boldsymbol{\theta}$ in the clock-wise direction. To perform rotation in the counter clock-wise direction, (11) becomes

$$\mathbf{R}(\boldsymbol{\theta}) = \begin{bmatrix} \cos\boldsymbol{\theta} & \sin\boldsymbol{\theta} \\ -\sin\boldsymbol{\theta} & \cos\boldsymbol{\theta} \end{bmatrix} \tag{12}$$

The angle $\boldsymbol{\theta}$ is defined by the angle separation between the two anchor points and their corresponding MDS solutions. The angle can be calculated from the following equation

$$\boldsymbol{\theta} = \cos^{-1}\left(\frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|}\right) \tag{13}$$

Where vectors **a** and **b** are defined by the actual two anchor positions and their corresponding MDS solutions respectively. To calculate the angle of separation, the MDS solution in Figure 2-6 is first translated to overlap the actual plot at the origin, defined by point A. The new plot becomes

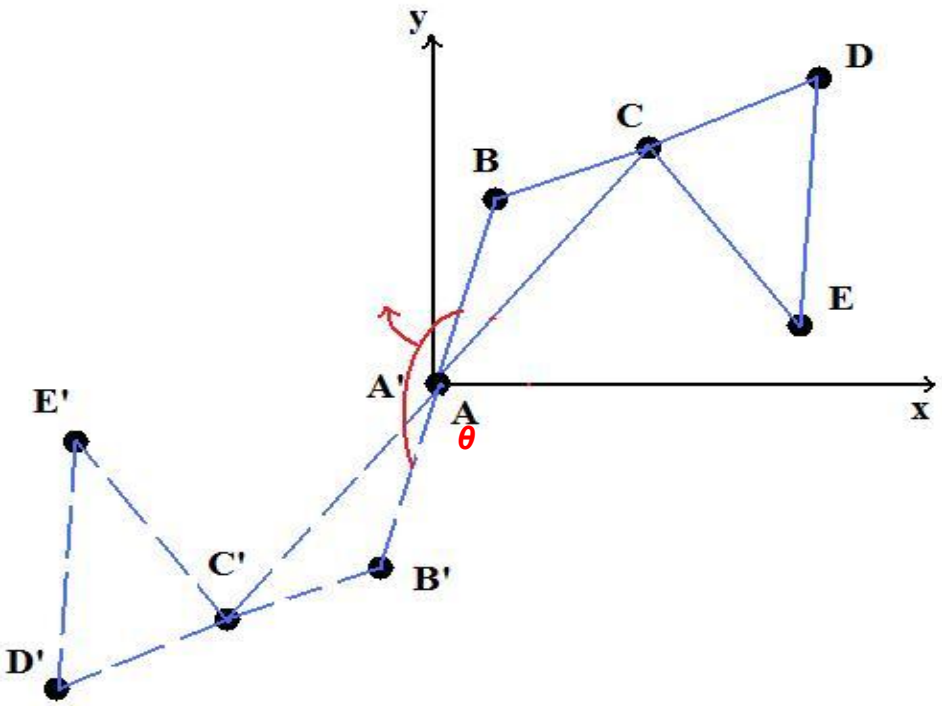

Figure 2-7: Translated MDS Solution and its Angle separation with the Actual Layout

From (13), the angle of separation between the actual layout and the MDS solutions is calculated, and its value is expressed in the range, where $0 \leq \theta \leq \pi$, in either the clock-wise or the counter clock-wise direction. To rotate the whole set of objects by angle, $\boldsymbol{\theta}$, in vector space, the rotated object coordinates are expressed as follows

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{14}$$

2.4 Radio Frequency Ranging Techniques

In literatures, WSN localization applications are usually used to compare with GPS localization, because of their low-power and low-cost advantages over GPS localization. Bulusu in [1] describes the design of low-cost GPS-less localization system with the use of only RF communication capabilities. As previously

discussed, most localization methods require distance or proximity information between nodes to compute node positions. This inter-node distances can be determined by various methods, such as ultra-sound or laser. However, ultra-sound waves and laser are easily absorbed and reflected by obstacles. As a result, ultra-sound and laser ranging techniques become limited to perfect line-of-sight (LOS) environments.

To reduce the system cost and complexity and to be able to localize in a non-LOS environment, RF wave is also often used for measuring pair-wise distances between radio devices, [10]. The most common RF ranging techniques are

2.4.1 **Time Difference of Arrival (TDOA)**

According to Xu, [11], TDOA is a commonly used ranging technique. In the context of range measurements implemented by Xu, UWB radio technology is used. Radio waves are forms of electromagnetic radiation, traveling at the speed of light, $\boldsymbol{C} \cong 3 \times 10^8\ \mathrm{m/s}$. The inter-node distances between a pair of radio devices can be calculated from a radio signal's traveling time from one device to the other. The distance traveled between radio devices a and b can be approximated by the following equation

$$d_{ab} = T_{ab} \times c \tag{15}$$

T_{ab} is the time taken by the radio signal to travel from device a to device b. To measure this time difference, the clocks on both radio devices are required to be synchronized. Any mismatch in the timer synchronization on both sides will result

in imprecision in distance approximation. The following illustration is the typical wireless sensor localization network based on TDOA.

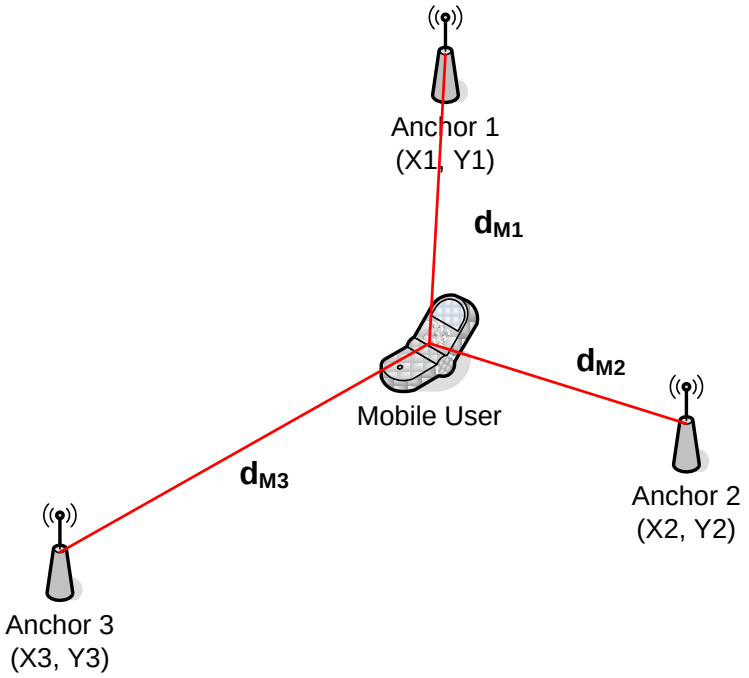

Figure 2-8: TDOA Triangulation based Localization

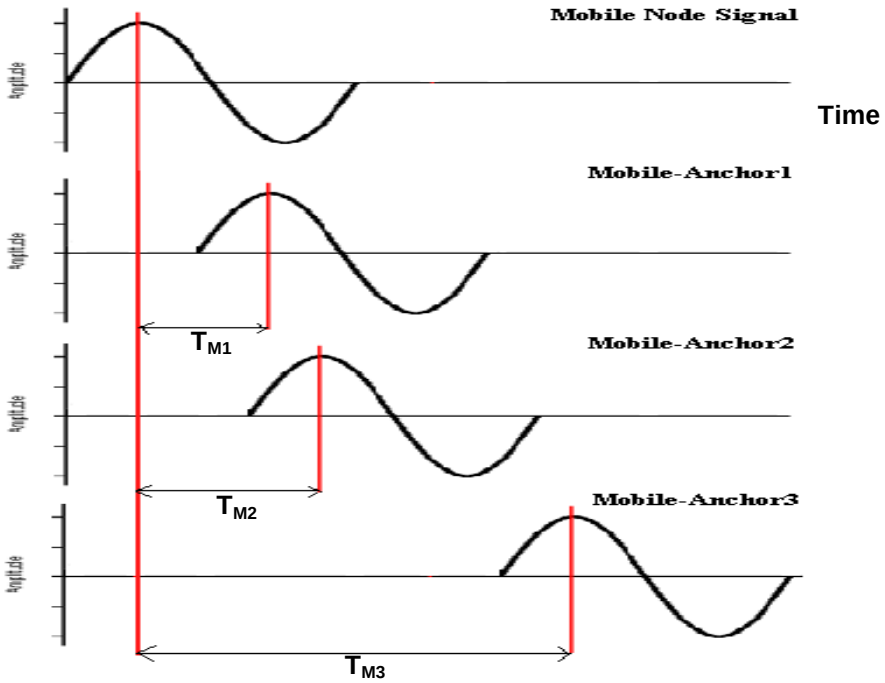

Figure 2-9: Time Traveled between Mobile Node and its Neighbouring Anchors

Figure 2-9 demonstrates the time taken for mobile node's radio signal to reach its three neighboring anchor nodes. Using (15), the corresponding distances of the time difference of RF signal arrival can be computed.

2.4.2 **Receiver Signal Strength Indicator (RSSI)**

Another common ranging method is utilizing the RF signal propagation characteristics. In a typical radio communication system shown in Figure 2-10, the receiver signal strength, $\boldsymbol{P_r}$, is modeled by Friis equation, [12], in (16)

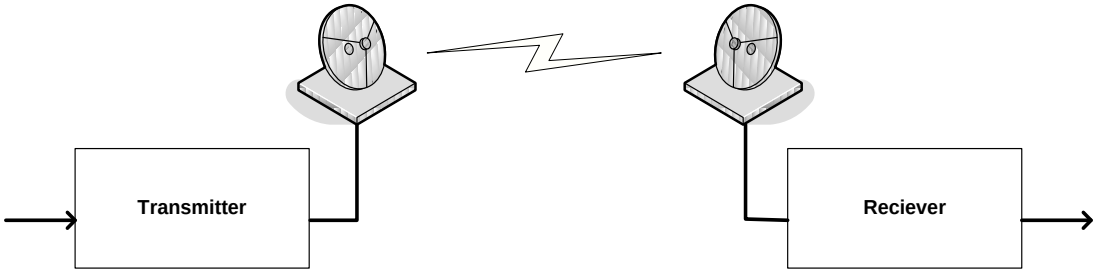

Figure 2-10: Basic Radio Communication System

$$P_r = P_t G_t G_r \left(\frac{\lambda}{4\pi R}\right)^2 \tag{16}$$

Where $\boldsymbol{P_t}$ is the transmission power; $\boldsymbol{G_t}$ and $\boldsymbol{G_r}$ are the transmitter and receiver antenna gains respectively; $\boldsymbol{\lambda}$ is the wavelength of the RF signal; $\boldsymbol{R}$ is the distance between the two radio devices. It is obvious that the receiver power is mainly governed by its distance from the transmitter and its power decays in an inversely proportional relationship with the squared distance. The characteristic curve is shown in Figure 2-11. When both the transmitter power and receiver power are known, the distance between the two radio devices can be approximated.

Figure 2-11: RSSI vs. Distance Characteristic Curve

However, it is difficult to observe accurate receiver and transmitter signal strength without a proper measuring device. In modern radio receivers, their on-board controllers are able to detect the power present in the received signal. This measurement of received signal power is referred to as RSSI. According to Texas Instruments [10], the relationship between RSSI and distance can be modeled by the following equation

$$RSSI = -(10n \log_{10} d + A) \tag{17}$$

Where ***n*** is the propagation constant; ***d*** is the distance; ***A*** is defined as the received signal strength at distance of one meter.

Chapter 3 - Related Work

As discussed in Chapter 2 -, to develop a SOSC WSN, the building blocks required are, a wireless network capable of performing single hop and multi-hop communication between the sensor nodes; a robust sensor position computation technique; a reliable distance measuring mechanism to maintain the accuracy and consistency of the localization result. This section describes the related work utilizing various distance measuring techniques, network topologies, and algorithms to perform localization in WSNs. Unlike [3], treating the ranging techniques, position computations, localization algorithms as separate entities of localization systems, all the related works in this section are categorized into two main groups, range based and range-free localization systems

3.1 Range Based Localization Systems

Range based localization systems take advantage of distance measuring devices to obtain an overall distance measure between a pair of nodes within a WSN. With the use of various localization algorithms, the positions of the sensor nodes can be computed from the inter-node distances. The following are examples of related works on range based localization systems. The authors of, [13], [14], and [15], introduce modern range-based localization techniques and implementations for WSNs. Various measurement techniques mentioned in their works, such as TDOA, RSSI profiling, GPS ranging, and AOA [16], are common approaches to obtain inter-node distance prior to performing localization. Given the inter-node distance, range based localization systems utilize stochastic

methods or geometric triangulation to perform the conversion from inter-node distances to their corresponding Euclidean coordinates. The following examples describe the details of various implementations

3.1.1 **RSSI Based Localization Systems**

RSSI based localization systems are the simplest and cheapest, because they utilize only radios to perform distance estimation and do not require additional distance measuring devices that increase the overall complexity of individual sensor devices. The localization technique using RSSI was first adapted in mobile networks in [17]. This author proposed a weighted least squared (WLS) method to compute the positions of mobile stations (MS), having true location of $[x_s, y_s]^T$, from their distances to each base station (BS) with coordinates $[x_i, y_i]^T, i = 1,2, \dots, N$. The RSS to distance relationship is given by the model, $P_i^r = K_i \frac{P_i^t}{d_i^a}$, $i = 1,2, \dots, N$, where P_i^t is the transmission power in dBm, K_i is the factor influencing the signal strength, and a is the propagation constant. The RSS, denoted by r_i, can be represented in 2-D space by the following equation, $r_i = K_i \frac{P_i^t}{P_i^r} + n_i = [(x_s - x_i)^2 + (y_s - y_i)^2]^{\frac{a}{2}} + n_i, i = 1,2, \dots N$. The WLS method essentially solves the system of linear equations of each MS to BS pair distance; the system of equations can be generalized by the matrix form, $\mathbf{A}\theta = \mathbf{b}$, where $\mathbf{A} = \begin{bmatrix} x_1 & y_1 & -0.5 \\ \vdots & \vdots & \vdots \\ x_N & y_N & -0.5 \end{bmatrix}$, $\theta = \begin{bmatrix} x_s \\ y_s \\ R_s^2 \end{bmatrix}$, and $\mathbf{b} = \frac{1}{2}\begin{bmatrix} x_1^2 + y_1^2 - r_1^{\frac{2}{a}} \\ \vdots \\ x_N^2 + y_N^2 - r_N^{\frac{2}{a}} \end{bmatrix}$. With WLS method, the localization simulation results have shown the mean square range errors (MSREs) lower than the standard least squared (LS) method. The LS approach

is also known as a dimensionality reduction method. Some similar approaches based on dimensionality reduction are introduced in [18], [19], [20], and [21] where the most common localization approaches use the MDS algorithm. In [21], Aloor and Jacob introduce non-MDS based algorithm, called the stochastic proximity embedding (SPE), to provide a comparable localization accuracy with less computation complexity than the classical MDS methods. As concluded by Aloor and Jacob, the competitive advantage of SPE over other MDS-variants is its comparable localization performance, lower computation complexity, and its ability to localize in irregular network topologies; however, being highly dependent on distance information, the localization accuracy of the above algorithms are all subject to error in noisy environments. This thesis proposes a MDS-based localization approach, which not only has a linear scaling property in complexity, but also has the capability to reduce the influence of range error to produce lower localization error than the classical MDS method.

Range measurement is a major component in range-based localization. As for RSSI-based ranging, mathematical models for converting RSSI to distance follow the inverse squared relationship, as illustrated earlier in Figure 2-11. However, as observed in [22], interferences in the RF channel can cause the inter-node distance to enlarge, causing high localization error. Many works have been done to model various RF channel interferences. Depending on the particular RF interference the equation attempts to model, the complexity of the equations may vary. For example, in [23], the RSSI to distance relationship in a RF shadowing model is expressed by the equation, $RSS(d) = P_T - PL(d_o) -$

$10\eta \log_{10} \frac{d}{d_o} + X_\sigma$. The path loss exponent, η, as experimentally determined, can vary between 2 and 6, depending on the terrain and the surroundings in pair-wise LOS communication. However, for simplicity, not considering the RF channel fading, Wu utilizes (17) to obtain distance information. To compute the corresponding positions from the inter-node distances, Wu proposed mean filtering to localize nodes. Mean filtering is a low pass filter that reduces the influence of impulse noises or sudden change in RSS, smoothing the signal. In the 12x12 m^2 indoor environment, this particular localization system introduces an average error of 1.6 meters.

The previous two works perform localization based on distance extracted from theoretical models of RF wave propagation. A famous research, RADAR [24], done by Bahl and Padmanabhan, proposed an empirical approach by constructing a map of overlapping signal strength coverage of the BSs that define the boundaries of the WSN. The map is determined by recording the signal strength information in various locations within the network. As a result, each location will present a different combination of signal strengths transmitted by the neighbouring BSs. From comparing the empirical method with the theoretical method, using the propagation models, the author claimed that RADAR system can achieve a median resolution of 2 to 3 meters, as opposed to a median resolution of 4 to 5 meters in the theoretical methods.

The empirical localization method is always done off-line, as the system is required to have knowledge of the environment prior to tracking sensor nodes within the areas of interest. For the localization system, requiring such an off-line

training, [25], [26], and [27] proposed stochastic methods to design rule-based localization algorithms that determines the most probable node positions from the RF signal strength information gathered. The stochastic localization algorithms from the above three references utilizes methods like maximum likelihood estimator (MLE), maximum a posteriori probability (MAP), and the hidden Markov model (HMM). One characteristic that all stochastic localization systems have in common is that all systems can be trained to have high accuracy and high reliability in noisy environments. However, they require extensive off-line environment data collection and thus are not suitable for applications requiring rapid network deployment.

3.1.2 Non-RSSI Range Based Localization Systems

Although, RSSI plays an important role as a ranging method in localization systems, many works have been done on using other techniques to obtain distance measurements. Those techniques can be categorized as follows

- **TOA/TDOA Based Localization:** As previously discussed, radio signal strength is a commonly used ranging technique in WSN localization applications. Time can also be used to interpret distance between two nodes in a WSN. TOA and TDOA are widely applied time-based ranging and tracking techniques. TOA requires all transmitters to be synchronized with the anchors, so the travelling time from a transmitter signal to each anchor will determine transmitter-to-anchor distances, which allow the transmitter's position to be trilaterated. TDOA, on the other hand, only concerns with the difference between arrival times from one transmitter to

the anchors. The advantage of TDOA is that it requires only the anchors to be synchronized. Common WSNs have more transmitters than anchors. Using TDOA localization can greatly reduces system complexity and network overhead for transmitter-to-anchor synchronizations. Many variants of the TDOA based localization algorithms, like [28], [29], and [30], are introduced to deliver better accuracy and reliability under noisy environment. However, the above localization algorithms rely on solving the systems of squared distance, $r_i{}^2 = (x - x_i)^2 + (y - y_i)^2, i = 1, 2, \dots N$, where $r_i{}^2$ is the TDOA measured distance from different reference nodes to the target node. Although, TDOA requires much less synchronization than TOA localization. Synchronization is still a difficult procedure, for it requires highly precise clocks and sophisticated inter-anchor communication scheme to synchronize their clocks. Besides RF waves, sound waves are also common ranging sources in TDOA localization systems, such as the Cricket localization system, [31] and [32]. TDOA using ultrasound sources does not require clocks to be as precise as those used for synchronizing RF waves. However, TDOA methods will always induce network overhead for synchronization and thus is not suitable for low-cost and low-power applications, whose network bandwidth and device clock precision are limited.

- **Angle of Arrival (AOA) Based Localization System:** a standard AOA method is initially designed to determine the direction of the signal propagation. The receivers of an AOA based localization systems are

often equipped with antenna arrays. The angle of the arriving signal is determined by calculating the time delay of signal at each antenna. In [33], the author used a probabilistic approach to determine the position of nodes within the network by using only the AOA information. This is done by first assuming all the transmission between nodes are bounded by a maximum transmission range of d_{max}, and any information beyond the transmission range will be neglected. Second, all the nodes are assumed to be in LOS communication with one another without any obstruction within the network. Having defined the above two assumptions, the AOA information received by a target node j, is modeled with a Gaussian distribution, $f_{\Theta ij}(\theta) = \frac{1}{\sqrt{2\pi}\sigma_\theta} e^{-\frac{(\theta-\tilde{\theta}_{ij})^2}{2\sigma_\theta^2}}$. Translating the AOA information to the corresponding Cartesian coordinates, the author proposes using joint pdf between distance, d_{ij} and angle, $f_{\Theta_{ij},d_{ij}}(\theta, d)$. The Cartesian coordinate can be realized by transforming this joint pdf. The transformed pdf is

$$f_{X_j,Y_j}(x_j, y_j) = \frac{1}{|d_j|} f_{D_j,\Theta_j}(d_j, \theta_j)\Big|_{\substack{d_j=\sqrt{x_j{}^2+y_j{}^2} \\ \theta_j=\upsilon(x_j,y_j)}}$$

Where $\upsilon(x_j, y_j)$ is a polar angular function of x and y:

$$\upsilon(x_j, y_j) = \begin{cases} \frac{\pi}{2} - \tan^{-1}\frac{y}{x}, & x \geq 0, \\ \frac{3\pi}{2}\tan^{-1}\frac{y}{x}(mod(2\pi)), & x < 0. \end{cases}$$

By calculating the probability distribution intersecting at unknown node j from its neighbouring anchors, the position can be determined. Other AOA

based localization systems such as [34] and [16] proposed triangulation algorithms to demonstrate comparable localization accuracy.

3.1.3 **Hybrid Range Based Localization System**

Being fading and interference-prone, RSSI ranging can be unreliable and erroneous. To avoid compromising the localization accuracy due to RSSI range measurement error, two approaches can be applied. One approach is to train the system to determine the trust-worthiness based on prior knowledge of range measurement distribution, where higher weights are assigned to the measurements believed to be more accurate. The second approach is to utilize a more reliable ranging device, which is less prone to noise than RSSI. Like the Cricket system, ultra-sound is used in conjunction with RSSI to compute inter-node distances. In [35], the author combines the above two approaches to address the ranging error issue caused by RSSI. It is important to remark that this literature has the most similar approach as the SOSC system proposed in this thesis and will be used as comparison throughout the thesis.

The algorithm used in [35] is called distributed weighted MDS (dwMDS) algorithm. This algorithm is designed to operate with a broad range of ranging methods, such as RSSI or TOA. Based on the range measurements obtained, a cost function is defined as

$$S = 2 \sum_{1 \le i \le n} \sum_{i < j \le n+m} \sum_{1 \le t \le K} w_{ij}^{(t)} \left((\delta_{ij}^{(t)} - d_{ij}(X) \right)^2 + \sum_{1 \le i \le n} r_i \|x_i - \bar{x}_i\|^2)$$

Where $w_{ij}^{(t)}$ is the weight, equal to 0 when the measurement $\delta_{ij}^{(t)}$ is not available between i and j. $w_{ij}^{(t)} \geq 0$ when the measurement is believed to be more

accurate. However, this literature does not elaborate on the methodology for differentiating an accurate range measurement from many other measurements. The above assumption has allowed the algorithm to be non-parametric, but its adaptive nature for searching for the more accurate range measurement results in a $O(n^2)$ computation complexity.

3.2 Range-free Localization Systems

Range based localization algorithms can achieve high accuracy in position estimation. Due to additional ranging devices required to provide distance information, the complexity and cost of the overall system can be high. Range-free localization systems do not require metric distance information to perform location calculation. Range-free localization utilizes connectivity information, such as, “who is within the communications range of whom” [36], to approximate the position of nodes. As a result, the systems are simpler and more cost effective while sacrificing localization accuracy. Many of the range-free localization algorithms are developed based on connectivity information as discussed by [37]. The most popular range-free localization algorithm, utilizing the distance vector (DV) hop approach, is proposed by [38] and is further modified and improved by [39]. There are also other range-free approaches like the approximate point-in-triangulation test (APIT) approach in [40], and the proximity distance map (PDM) algorithm in [41]. These algorithms will be discussed in the following sections.

3.2.1 **DV-Hop Localization Algorithm:**

The first DV-hop algorithm was proposed by Niculescu and Nath [38] specifically for ad-hoc networks. An ad-hoc network, by nature, transmits

information throughout the whole network through hopping. Such an ad-hoc based localization system is also known as an ad-hoc positioning system (APS). APS performs localization based on the hop-by-hop characteristic in a distributed fashion and is an effective localization system when the network coverage is large, where not every node is within the communication range of every other node. In a DV-hop based APS, the network is bound by a few anchor nodes with known inter-node distances from one another. These anchor nodes are known as the landmarks i, where $i = 1, 2, \dots, N$. The proximity information is governed by the correction function, C_i, where

$$C_i = \frac{\sum \sqrt{\left(X_i - X_j\right)^2 + (Y_i - Y_j)^2}}{\sum h_i}, \qquad i \neq j$$

Each landmark node has its correction function. The function is defined by the ratio of inter-node distances, represented with two-dimensional Cartesian coordinates, X_i and Y_i, and the sum of hops, h_i. The localization algorithm based on DV-hops uses triangulation. This particular APS has the knowledge of the number of hops that each node is required to reach each landmark node. The proximity information is obtained by multiplying the landmark-specific correction function to the number of hops the target node is required to reach it. The proximity information of the landmark nodes to the unknown node is triangulated, and its position is then estimated. One advantage of DV-hop localization is that the localization result is not affected by distance measurement error. However, in real-life

scenario, hop sizes from node to node may differ, and thus reducing the localization resolution. To refine the accuracy of the localization results, the author proposed a method, called DV-distance, where instead of using hop counts as the proximity measurement, the localization system uses RF signal strength. This method is also proposed in [42]. However, this method, although is theoretically more accurate than the regular DV-hop localization algorithm, in reality, raw RSSI values do not always reflect the actual distances, due to RF interference and noise. Unless, a careful RSSI-distance model is considered, the assumption of distance directly based on RSSI can be erroneous in real-world environments. The author in [43] mentions that ordinary DV-hop based localization's accuracy suffers from the hop-count shift problem, which is mainly caused by the assumption of ideal radio communication models. In this paper, the author proposed an improvement to the ordinary DV-hop localization by introducing a more complex localization mechanism that considers various radio channel behavior, such as shadowing.

3.2.2 The Approximate Point-In-triangulation Test (APIT) Approach

Although, the DV-hop localization technique is a simple and an effective method to provide a rough location estimate for nodes in large area networks, the technique is sensitive to localization error when the network is not isotropic. In real-life scenario, radio patterns and node placements are usually random due to the nature of radio propagation and the presence of obstacles, disrupting the LOS between any pairs of nodes

within a network. An APIT based localization system solves the irregular node placement problem as discussed above. An APIT localization algorithm utilizes anchor nodes to form an enclosed region to determine if a target node is within or outside of the centroid, defined by the enclosed region. The enclosed region is a triangle, formed by any of the three anchor nodes within the network. In any APIT localization system with N anchors, there will exist $\binom{N}{3}$ triangular regions. The APIT algorithm computes the set of centroids for the triangles, namely the centre of gravity (COG) [40], where the target node belongs to. The algorithm searches through all the triangles to locate and to finalize the target node's position when an accuracy threshold is reached. APIT algorithm, although, does not rely on the conversion of RF signal strength to distance to perform localization, it relies on RF signal strengths between the target node and its neighboring anchors to determine whether the particular target node belongs to the centroid of the corresponding triangle. As a consequence, the localization result can be faulty if any of the anchor node reports incorrect anchor-to-target signal strength, positioning the target node in an incorrect location. However, the author has proven that APIT has comparable accuracy and a reasonably small computation overhead against other range-free localization schemes, such as the DV-hop localization algorithm.

3.2.3 **The PDM Localization:**

As mentioned earlier, the anisotropic characteristic of a typical WSN is mainly caused by irregular terrain of the network, anisotropic RF radiation pattern, and a random placement of the sensor nodes. This characteristic greatly degrades the accuracy and reliability of the range-free hop-based localization algorithms, such as the DV-hop algorithm discussed previously. To address the anisotropic nature that pure hop-based localization schemes are unable to solve, [41] proposed applying the transformation of the proximity measurement between sensor nodes into a geographic distance embedding space by utilizing the truncated singular value decomposition based (TSVD-based) technique, called PDM. PDM is defined by a linear transformation T, consisting of two major components: the M-by-M proximity matrix $P = [p_1, \dots, p_M]$ and the geographic distance matrix $L = [l_1, \dots, l_M]$. The linear transformation T is represented by $T = LP^T(PP^T)^{-1}$. To estimate the geographic distance L, the author applied the singular value decomposition (SVD) to derive a truncated pseudo-inverse of P, P_γ^+. PDM can reconstruct the embedding space for geographic distance using the proximity measurements. The estimate of the geographic distances is $\tilde{l}_s = Tp_s = LP_\gamma^+ p_s$, where p_s is the proximity vector obtained from the hop count to the anchor nodes. This PDM based localization method's performance is compared with methods like the DV-hop and MDS-MAP method and is shown to outperform the two methods both under isotropic and anisotropic conditions.

3.3 Chapter Summary

The objective to perform localization in a WSN is to allow the sensor nodes to realize their geographical positions among themselves without user intervention. The most essential component for position computation is the distance information between each pair of nodes within the network. This distance information can be expressed in form of absolute distance (range-based) or proximity (range-free). Based on these two types of distance information, various localization algorithms have been developed. However, each type of localization algorithm exhibits its weakness. For range-based localization systems relying heavily on prior knowledge of inter-node distance distribution, off-line training and data collection are required. Although, their localization performance is highly reliable under noisy conditions, it is inefficient and inconvenient having to train the system for different environment settings. Systems that do not require off-line training rely mostly on linear transformation to translate the inter-node distance information into corresponding coordinates. In general, range-based localization has a finer resolution than range-free localization. However, ranging techniques, using radio or ultra-sound, are prone to error in environments with reflective media and multi-path effects. As a result, the localization algorithms often incorporate statistical approaches to cope with these conditions.

Range-free localization schemes, on the other hand, are more reliable, due to a low requirement in resolution demanded by its applications; it is often sufficient to localize sensor nodes in terms of proximity, instead of the absolute position by coordinates. However, range-free localization systems' performance

can be affected by their network topology and network characteristics. For example, most of the range-free localization algorithms perform computation based on hop-counts between nodes and the anchors. The network topology and the degree of isotropy define the routing table, which also defines the hop range. In an anisotropic network, the physical distance, requiring one hop to communicate between two nodes, can require more than one hop for other nodes, depending on the communication cost. This disadvantage is demonstrated by the DV-hop localization scheme. However, it can be improved and made more robust in anisotropic networks by increasing the density of anchor nodes or by utilizing the APIT and PDM algorithms.

Chapter 4 - SOSC System Architecture

SOSC is a ZigBee based WSN localization system, whose network topology is inherited from the ZigBee topology. The SOSC WSN is intended for sensing and monitoring applications, and self-localizing capability is the feature that facilitates the deployment of the WSN. SOSC WSN consists of only one gateway, which is also a ZigBee coordinator, numerous routers and end devices. In this section we describe the SOSC system architecture, the network, hardware, and also the firmware architecture.

4.1 System / Network Architecture

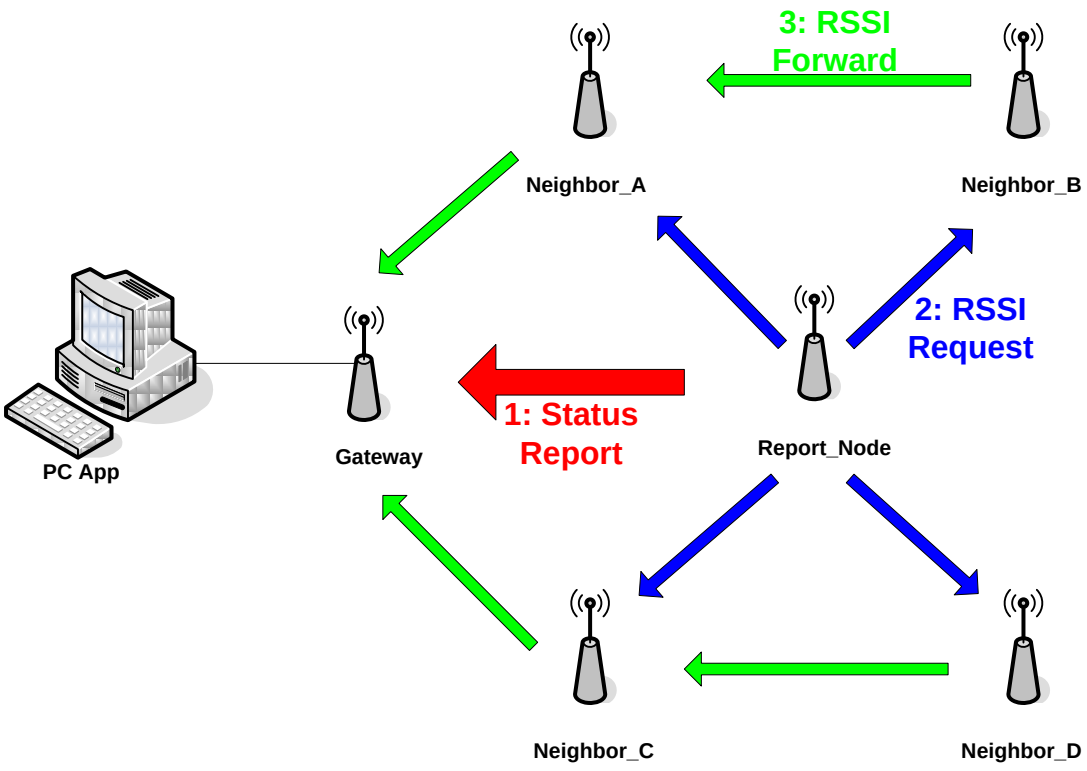

Figure 4-1: System and Network Architecture

In the context of this thesis, we will focus on the localization component of the SOSC system. Prior to performing localization, the most essential component required for computation is the inter-node distance information. In a SOSC WSN, Figure 4-1, the gateway is responsible for receiving the inter-node distance from the devices to be localized. The gateway, interfaced to a PC, will forward the inter-node RSSIs through serial interface for further position computation. The gateway is only responsible for receiving inter-node distances and plays no role in the localization of the SOSC WSN. On the other hand, the routers and end devices are the ones responsible for establishing the communication among one another to report the inter-node RSSI to the gateway in an orderly fashion.

The reporting of inter-node RSSI values is initiated by a periodic and asynchronous node status report. This status report denoted by ‘1’ in Figure 4-1, developed by a former CiBER colleague, Benny Hung [44], is used to reduce the congestion of the channel traffic in a large scale network. Each status report consists of the device ID of the reporting node and its current status, such as the on-board battery level or the temperature in the surroundings. Only one node is allowed to report status at each allocated time slot. Upon each status report, the status reporting node broadcasts a request, denoted by ‘2’, to its neighboring nodes within one hop distance to forward the RSSI, denoted by ‘3’, received from the reporting node to the gateway. The inter-node RSSI forwarded to the gateway is a packet that contains the device ID of the reporting node, the device ID of the node receiving the request, and the RSSI value received.

The computing unit, in this case, is a PC, interfaced to a serial port of the gateway device, obtains the inter-node RSSI packets forwarded directly from the gateway upon reception. The PC is responsible for storing and organizing the inter-node RSSIs into a square dissimilarity matrix with its RSSI values indexed with the reporting node's and the receiving node's device IDs. The PC computes the nodes' positions with a centralized MDS-based algorithm. This algorithm computes and reconstructs the map of the node distribution in 2-D space from the dissimilarity matrix. The PC also provides a visual realization of the node layout in a graphical user interface (GUI), which allows the users to easily understand the spatial distribution of the nodes within the SOSC WSN.

4.2 SOSC System Hardware Architecture

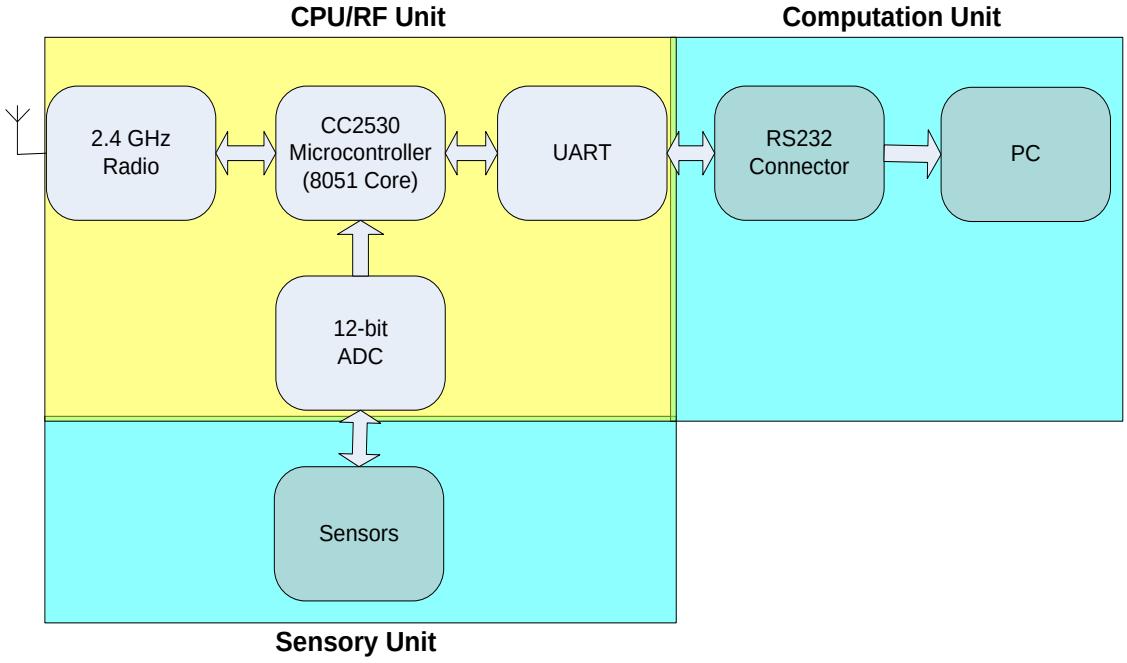

Figure 4-2: Hardware Architecture of a Gateway Device

It is important to note that the underlying hardware devices of the SOSC WSN are engineered by other CiBER researchers. Since, this thesis only concerns the localization feature of the SOSC devices, the discussion on hardware architecture will only focus on device power consumption, data collection interfacing, and communication.

The hardware architecture of the gateway, routers, and end devices in a SOSC WSN share similar design and are all based on the TI CC2530 system on a chip (SOC). Figure 4-2 is an illustration of the major SOSC system building blocks. The CPU/RF unit governs the inter-node communication and sensing of the inter-node RSSI. The sensory unit has a 14-bit ADC on the CC2530 SOC. When equipped with variety of sensors, it is capable of sensing the surrounding temperature, on-board battery level, or vibration voltage. This unit provides necessary information on each device's operating status upon each status report. The computation unit is interfaced directly to the on-chip UART port. Through this serial connection, a PC is able to obtain the status information of individual node and the inter-node RSSIs with its neighboring nodes. Depending on the type of devices and the role it plays in the SOSC WSN, SOSC nodes do not require all functional units. The table below shows the functional units required for each type of devices.

Table 4-1: Table of Required Functional Units on each Type of Device

	CPU/RF Unit	Sensory Unit	Computation Unit
Gateway/PC	Yes	No	Yes
Router	Yes	Yes	No
End Device	Yes	Yes	No

The gateway's major function is to gather and forward the network information to the PC; therefore, it does not require additional hardware on-board. On the other hand, router and end devices are the sensing and tracking units within the WSN. Therefore, routers and end devices are often equipped with sensors, capable of conducting environmental or physiological monitoring. Figure 4-3, Figure 4-4, and Figure 4-5 are the actual ZigBee based wireless nodes used in the SOSC WSN.

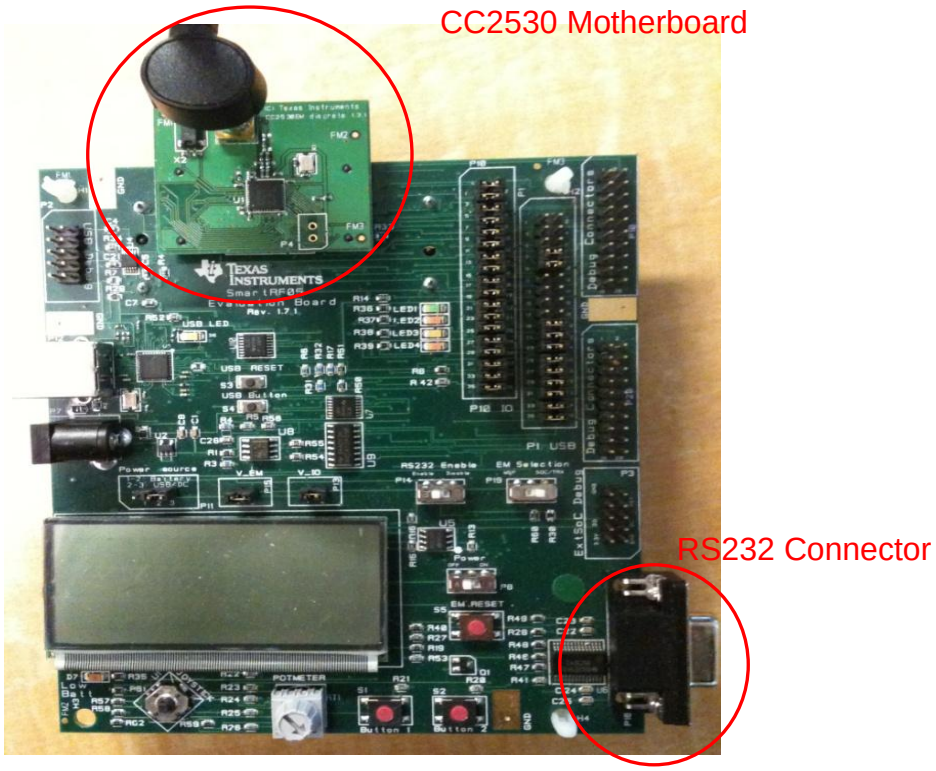

Figure 4-3: SOSC Gateway Device

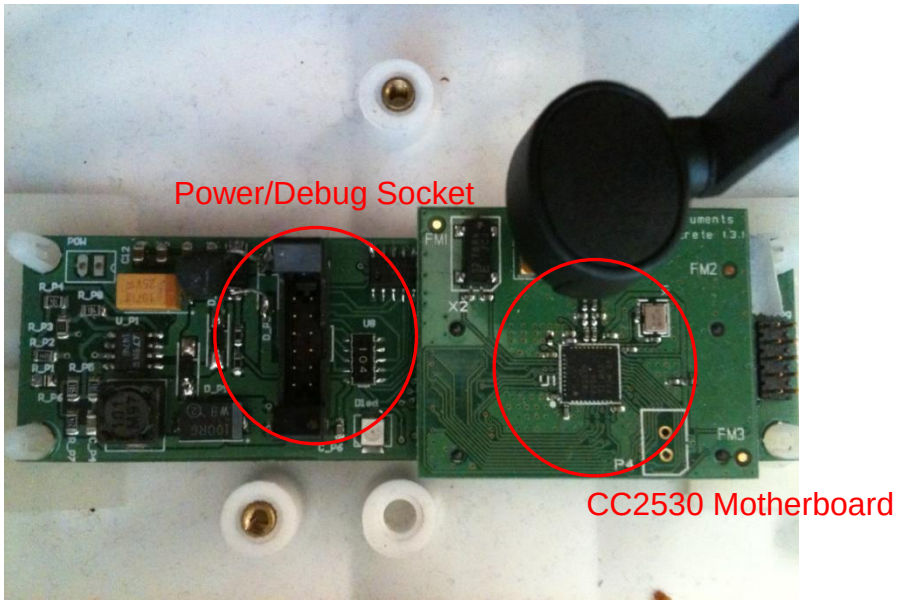

Figure 4-4: SOSC Router Device

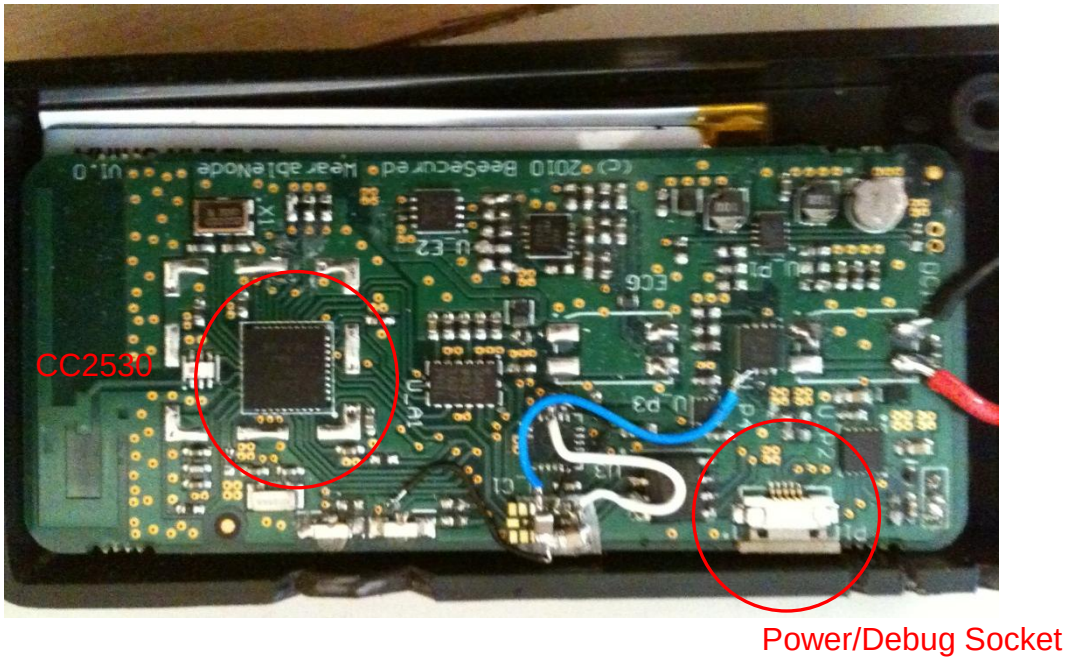

Figure 4-5: SOSC End Device

The gateway device is configured on a CC2530 evaluation board (EB) in Figure 4-3, which has the on-board RS232 connector interface and can easily be programmed to receive data from nodes in the WSN; the data is communicated directly with a PC through UART. The routers and end devices are designed to

consume minimal power and therefore few I/O peripherals and a limited number of application-specific sensors are required. For the localization purpose in this thesis, we utilize the on-board components listed below

- **Microcontroller:** SOSC nodes use TI's CC2530, consisting of an IEEE 802.15.4-compliant 2.4 GHz on-chip radio transceiver. During the on-time, the radio component consumes 28.7mA in Tx mode and 20.5mA of current in the Rx mode respectively. For serial data communication, CC2530 contains universal synchronous/asynchronous receiver/transmitter (USART), which can be configured as either serial peripheral interface (SPI) or as an UART. CC2530 is a low power microcontroller that consumes 3.4mA of current under normal CPU operation without radio activity and 1μA under power saving mode.
- **Debug/Power Connector:** the debug and power connector on SOSC routers and end devices serve two purposes. The most important purpose is to power the devices and the second purpose is to provide an interface for programming the on-board microcontroller. This connector could also be customized to provide interface to external sensors connected to the SOSC devices, such as intrusion sensors for perimeter surveillance applications.
- **RS232 Interface:** the RS232 interface provides a direct data link to external host devices such as a PC. Among the three types of SOSC devices, only the gateway consists of this interface and is allowed to exchange information with a PC directly.

- **Antenna:** antenna provides gain for the radio signal to propagate afar, exchanging data and conducting localization among all the SOSC nodes. SOSC WSN utilizes omni-directional antennas to communicate and collect inter-node RSSI information.

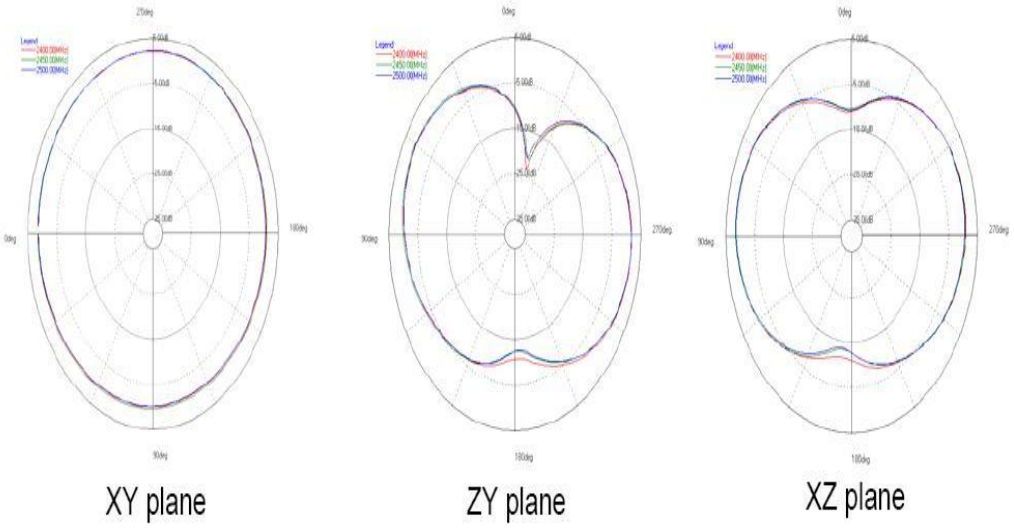

Figure 4-6: SOSC Antenna Radiation Pattern [45]

The localization approach proposed in this thesis converts RSSI into distance, based on the radio propagation model discussed in (17). Antenna property is a major influence on the signal strength polar distribution in the antenna's LOS communication with others. This property is realized with the radiation pattern, which provides the visualization of directionality of the antenna gain. The antenna used for SOSC WSN is a 2.4 GHz omni-directional antenna from Antenova Ltd. It has a radiation pattern as illustrated in Figure 4-6 and demonstrates equivalent antenna gain in all directions in the XY plane; the pattern is less omni-directional in the other two planes, and yet it still provides similar gains in all directions.

4.3 SOSC Software Architecture

The SOSC software consists of two major components: the first one is the on-chip firmware, which provides functions for the gateway to receive and to interpret inter-node RSSI packets and for the routers and end devices to request the inter-node RSSIs from their neighbors on an asynchronous scheduling mechanism; the second component is the software GUI on a client PC, which is responsible for computing the localization algorithm based on the inter-node RSSI and for displaying the estimated positions.

4.3.1 **SOSC Firmware:**

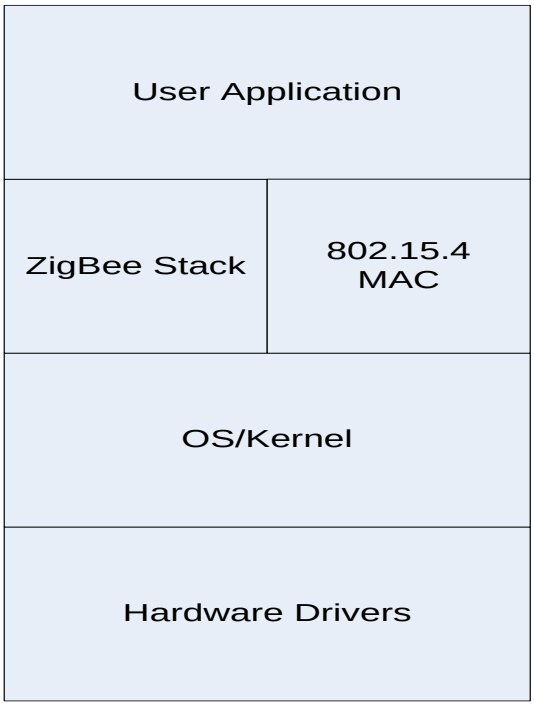

Figure 4-7: Firmware Architecture for SOSC Devices

Figure 4-7 shows the firmware architecture functional blocks for designing the application running on the SOSC devices. Besides the user application layer, the lower layers are proprietary and are provided by TI's Z-Stack. The OS kernel is

responsible for managing event loops and interrupts. The ZigBee stack and the 802.15.4 MAC layer govern the ZigBee protocol and the radio transmission respectively. It is important to note that all the layers in the Z-Stack are implemented in C.

The user application contains the processes allowing each node to communicate with the rest of the network to gather the inter-node RSSI. As mentioned previously, the node reporting the status initiates the request for RSSI from its neighboring nodes. The initiation of the request and its interaction with other layers prior to broadcasting is as follows

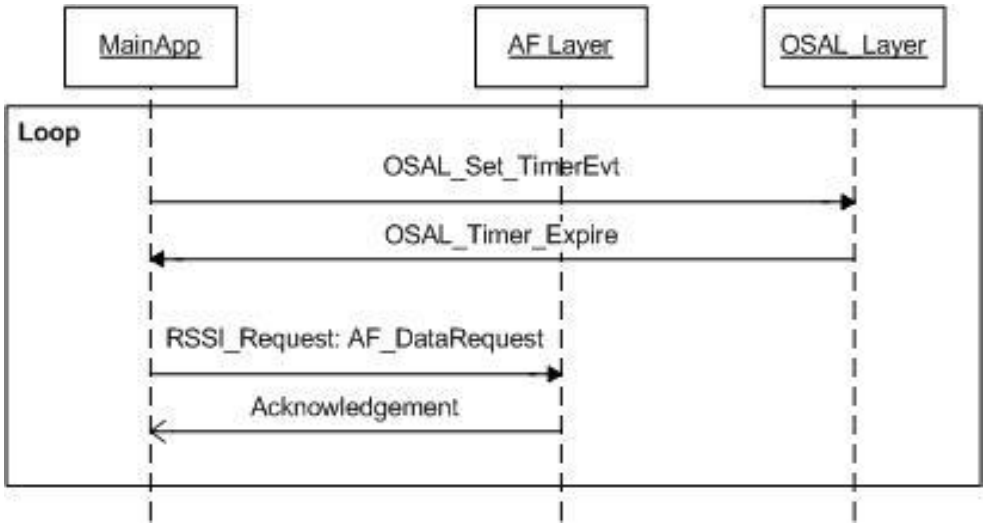

Figure 4-8: RSSI Request MSC between Z-Stack Layers

The status report is generated with a timer interrupt in each node in an asynchronous and periodic manner. This interrupt is handled by the OS abstraction layer (OSAL). Upon each timer interrupt, OSAL requests the application layer to initiate the RSSI request to the neighboring nodes via broadcasting, handled by the application framework interface layer (AF layer). The broadcast message to the neighboring nodes has the following format

DESTINATION_ADDR	CID	REPORT_NODE_ID	BROADCAST_RADIUS
2 Bytes	2 Bytes	2 Bytes	1 Byte

Figure 4-9: RSSI Request Broadcast Packet

The destination address indicates where the message should be directed to. This address is often referred to as the IEEE 16-bit short address. For uni-casting, this address is required to contain the address of the target device. For broadcasting, this address should be coded with the predefined value by the Z-Stack. The cluster ID, as specified in Table 2-1, is an ID that specifies the exchange of information within an application. This CID for broadcasting is named “PEG_ID_CID” and is to be identified once the over the air (OTA) broadcast is received by the neighboring nodes. The broadcast radius indicates the number of hops allowing the OTA message to transmit to reach its end-point. For RSSI request broadcasts, the node performing the status report only requests the RSSI values from its one-hop neighbors.

The message interaction between Z-Stack layers upon reception of the RSSI request broadcast is illustrated in Figure 4-10

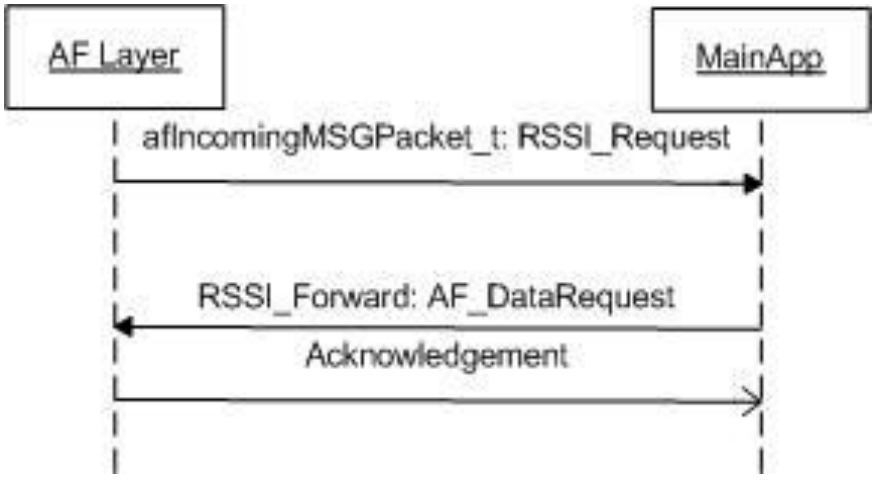

Figure 4-10: Receiving Node MSC between Z-Stack Layers

The OSAL layer acknowledges the application layer of the packet arrival, and the application layer unpacks the message, whose content is shown previously in Figure 4-9, where the CID acknowledges the application layer to obtain the RSSI value from the broadcasted message and to create and to forward to the gateway the new packet, containing the destination address, broadcaster's ID, its RSSI, and the receiver's ID. The new packet has the following format

DESTINATION_ ADDR	CID	REPORT_NODE_ID	RSSI	RECEIVER_NODE_ID	BROADCAST _RADIUS
2 Bytes	2 Bytes	2 Bytes	1 Byte	2 Bytes	1 Byte

Figure 4-11: RSSI Forward Packet

This packet containing the RSSI information is directed to the gateway via a multi-hop uni-casting mechanism. The packet specifies the address of the ZigBee coordinator, gateway, to be always 0x0000. The CID that allows the gateway to identify the process for retrieving the RSSI values following the identifier, "NODE_RSSI_CID".

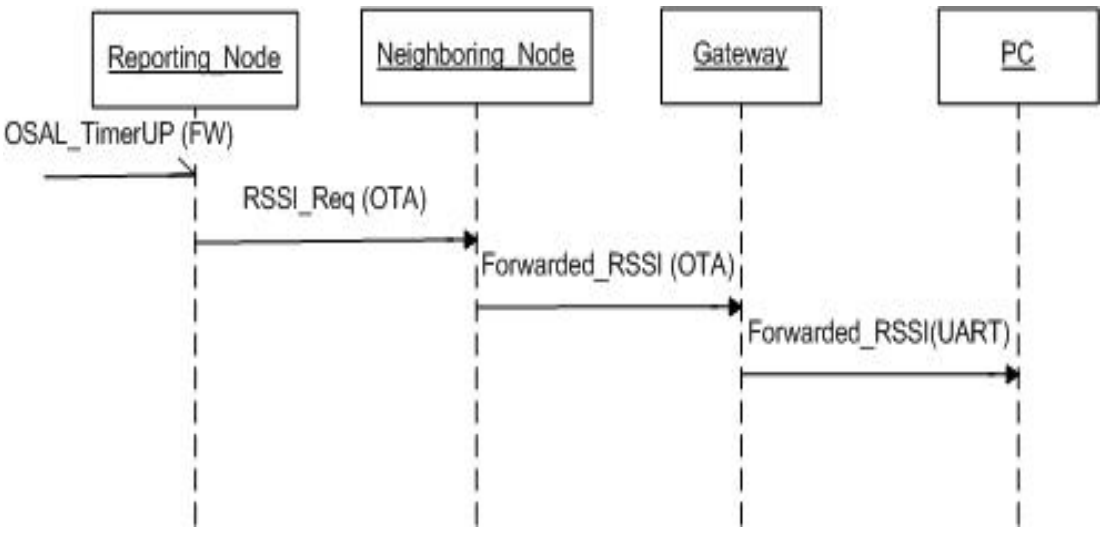

Figure 4-12: Network Message Flow Sequence

Figure 4-12 demonstrates the process and the messages required to obtain the inter-node RSSI throughout the SOSC network, containing the router nodes, end devices, gateway, and the PC.

The topological structure and message sequence described previously defines the communication network for obtaining the inter-node RSSI from every SOSC node. Having described the messages and commands required to initiate the SOSC devices to report the inter-node RSSI, the underlying flow of the SOSC device firmware is summarized in the sequence illustrated in Figure 4-13

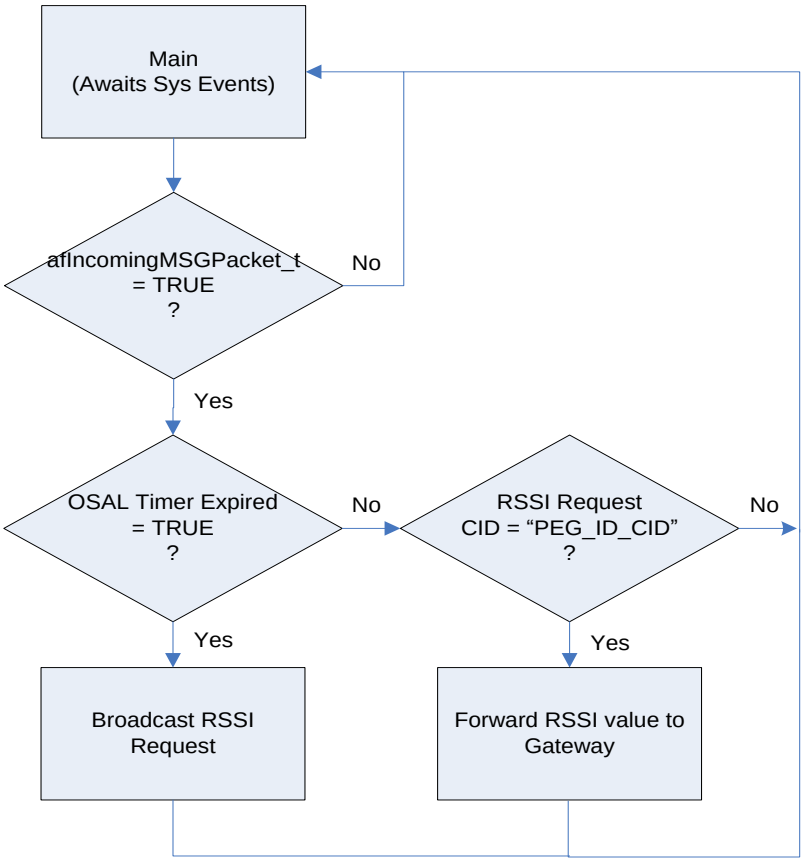

Figure 4-13: Flow Chart of SOSC Router/End Devices

This main program is an infinite loop that constantly awaits the system event interrupts passed from the OSAL layer. The occurrence of system events of

interest in this case are all indicated by the data buffer in “afIncomingMSGPacket_t” structure. When this structure is found to contain the system event, the main program continues to unpack and to check the system event type, determining the process to be executed. To obtain the inter-node RSSIs from the SOSC nodes, the two system event types of interest are the timer event and the OTA message event. The timer event triggers the process to broadcast the RSSI request and the OTA message event prompts the receiver device to check the CID that initiates the forwarding of the RSSI value to the gateway device.

The gateway device, being the listener that forwards the PC inter-node RSSI, constantly awaits the OTA messages from the nodes and generates a packet to be transmitted through UART. The UART packet follows the format below

HEADER	CID	REPORT_NODE_ID	RSSI	RECEIVER_NODE_ID
2 Bytes	2 Bytes	2 Bytes	1 Byte	2 Bytes

Figure 4-14: UART Packet Format

The above packet is built and transmitted through UART on firmware level by the gateway device. The flow of the gateway application is designed as show in Figure 4-15

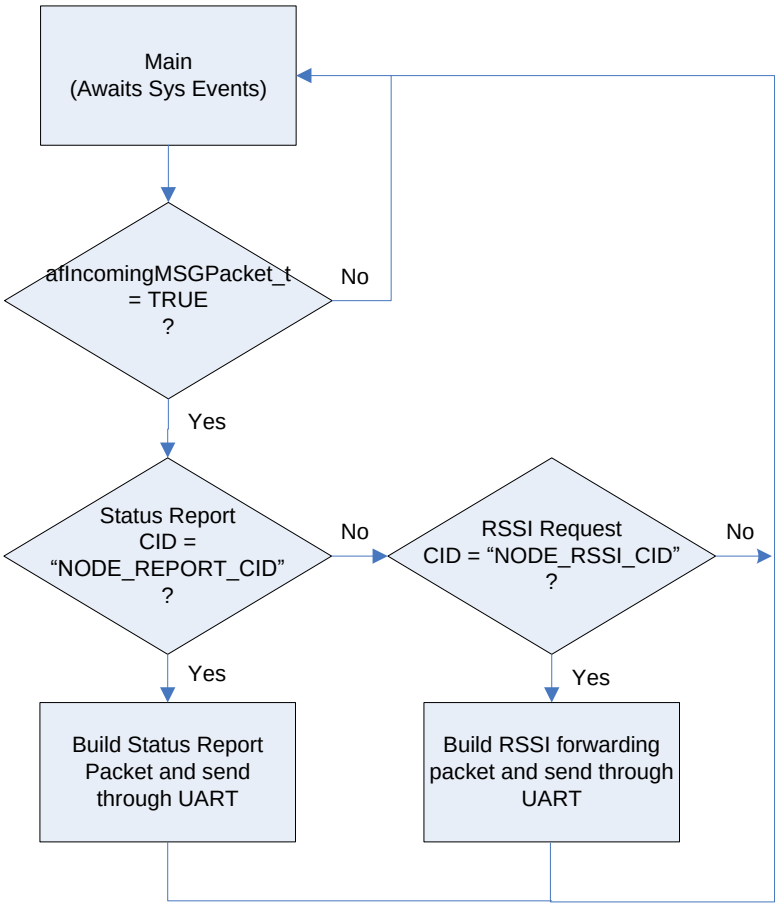

Figure 4-15: Flow Chart of Gateway Firmware

Similar to the firmware on routers and end devices, the gateway's application is an infinite loop that awaits the OSAL system events, which in this case only the OTA messages from the SOSC nodes. Upon the reception of the OTA messages, the gateway identifies the two CIDs as either the status report message or the inter-node RSSI message. The two CIDs define the two processes that create the packet to be forwarded to the PC, based on the type of message received.

The above concludes the design and description of the message structure and communication topology in firmware level required for the SOSC WSN to

output the inter-node RSSI through the gateway device to the PC. The data manipulation and the position computation from the localization algorithm on the PC will be discussed in the next section.

4.3.2 **PC Software Architecture:**

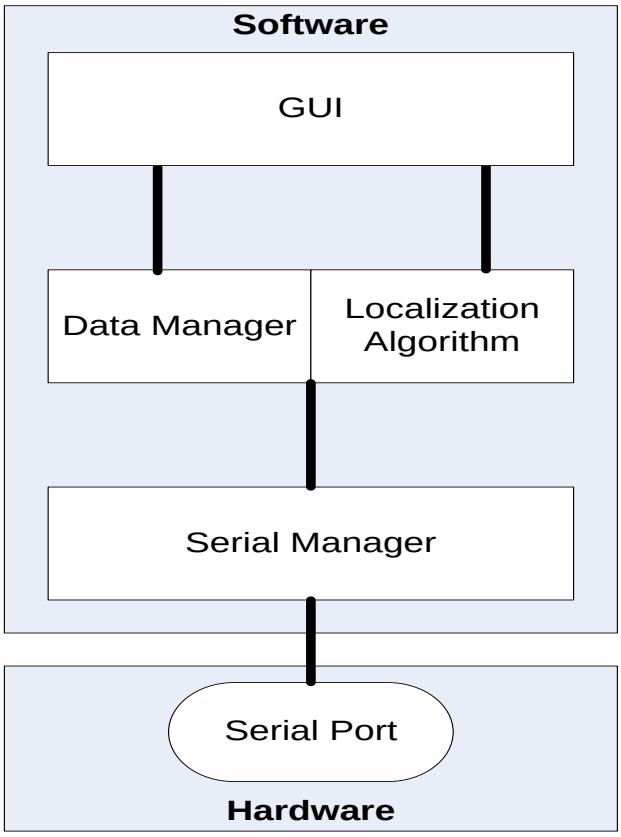

Figure 4-16: PC Application Software Architecture

The PC application is written in C# and utilizes the Windows API to handle the serial communication, such as USB and UART. This application's architecture layout is shown in Figure 4-16, where the serial manager, using the predefined function-calls from the Windows API, retrieves the RSSI and node information data received in the data buffer. The RSSI and node information are to be organized into the dissimilarity matrix table for the localization algorithm to

perform node position computation. The node positions ultimately will be displayed with the GUI.

The data manager, upon each data reception event passed from the serial manager, organizes the inter-node RSSI values into a matrix, indexed with the RSSI's corresponding broadcasting node ID and the receiver node ID. The algorithm for the data manager is shown as follows

Algorithm RSSI_Sorting(*$brctID_m$, $rcvID_m$, $RSSI_m$**)***

$SOSC\_node_{list}[] = \emptyset$
$Dissimilar_{matrix} = \emptyset$
i = 0
j = 0
for all m in M do
$brctID = brctID_m$
$rcvID = rcvID_m$
$RSSI = RSSI_m$
if $SOSC\_node_{list}[] = \emptyset$ then
$SOSC\_node_{list}[m]$ = ***Add_ID(****brctID**)***
Obtain index of brctID in $SOSC\_node_{list}[]$
endif
if $SOSC\_node_{list}[] \cap rcvID = \emptyset$ then
$SOSC\_node_{list}[m+1]$ = ***Add_ID(****rcvID**)***
Obtain index of rcvID in $SOSC\_node_{list}[]$
endif
i = ***Index_of(****brctID in $SOSC\_node_{list}[]$**)***
j = ***Index_of(****rcvID in $SOSC\_node_{list}[]$**)***
$Dissimilar_{matrix}[i,j] = RSSI_m$
endfor
return $Dissimilar_{matrix}$

Figure 4-17: Data Manager Algorithm in the PC Application

The algorithm starts with initializing the array, $SOSC\_node_{list}[]$, that keeps track of the node devices joining the network; it then initializes a square matrix, $Dissimilar_{matrix,}$ to store the RSSI values. As mentioned earlier, with each

incoming data from the serial port, the algorithm checks if $SOSC\_node_{list}[]$ contains the broadcast node ID, brctID, and the receiver node ID, rcvID. New nodes will be added to $SOSC\_node_{list}[]$ if the corresponding IDs are not found in the list. At the end, the RSSI value will be added to 2-dimensional $Dissimilar_{matrix}$ according to the indices of the brctID and rcvID in $SOSC\_node_{list}[]$ respectively. The process of storing RSSI values from the $SOSC\_node_{list}[]$ indices into $Dissimilar_{matrix}$ is illustrated as follows

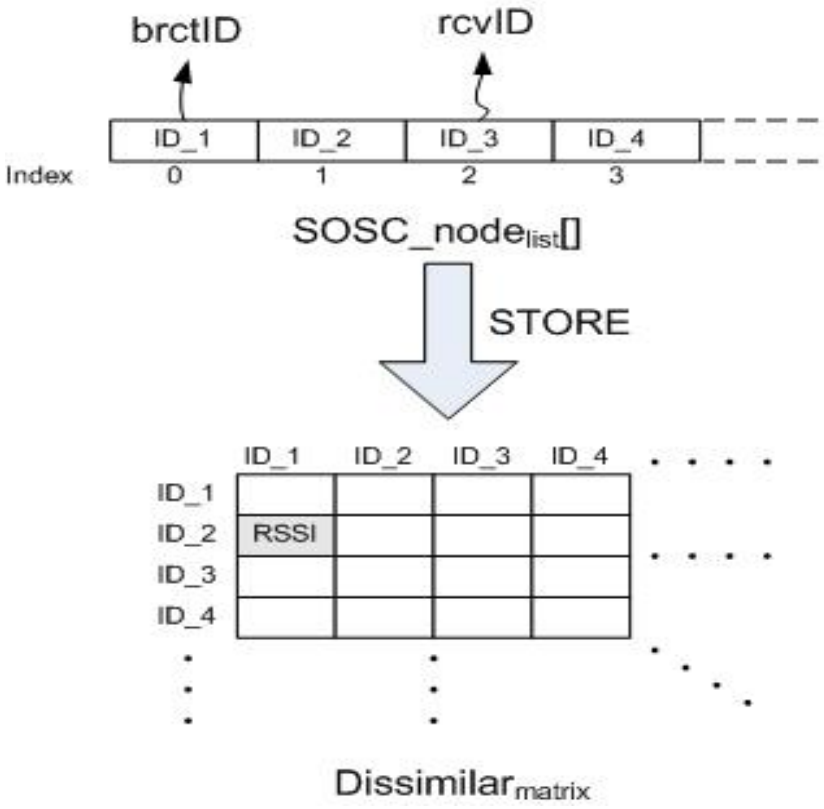

Figure 4-18: Data Manager Process for Constructing the Dissimilarity Matrix

The process illustrated in Figure 4-18 continues until all the inter-node RSSI values from the SOSC WSN have been obtained to complete constructing $Dissimilar_{matrix}$. This data manager not only allows the data to be stored in an orderly fashion, but also allows the SOSC WSN to have the real-time knowledge

of the number of nodes currently in the network. Having had the dissimilarity matrix for SOSC WSN with N nodes, the localization algorithm will determine the node positions and store the position estimates in a 2xN matrix. The algorithm for position computation is shown below

Algorithm MDS_Algo(Dissimilar$_{matrix}$[])

P = I_n – tileArray($\frac{1}{n}$)

B = P x (-0.5(Dissimilar$_{matrix}$[])^2) x P

Dissimilar$_{matrix}$[i,j] *= $\frac{Dissimilar_{matrix}[i,j]^T + Dissimilar_{matrix}[i,j]}{2}$*

[E,V] = Find_EigenVect(B)

V = Sort(V)

for each n in N do

If V_n < 0

V_n = empty

else

V_n = V_n

endif

endfor

X = V $\times$ $\boldsymbol{E^{1/2}}$

return ***X***

Figure 4-19: MDS Algorithm for Position Computation

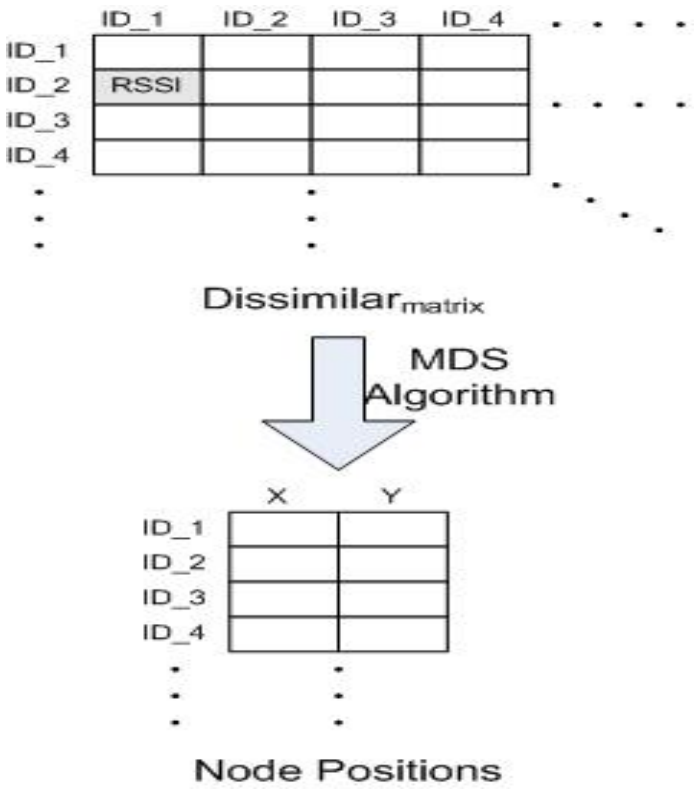

Figure 4-20: Position Transformation Process

The algorithm shown in Figure 4-19 is the summary of **Error! Reference source ot found.**, (8), and (9). The details of matrix ***P*** and ***B*** are described in the equations respectively. This algorithm is the portion of the localization algorithm process, illustrated in Figure 4-20, and is responsible for transforming the inter-node RSSI into the Cartesian-coordinate-based positions. The other portion of the localization algorithm handles the mapping of those node positions and acquires the correct layout in physical space. The design of the overall localization algorithm will be discussed further in the next chapter.

To display the final mapping of the SOSC node distribution in space, a GUI is designed to provide not only a visual understanding of the layout in space, but also the real-time status of the SOSC WSN, such as the number of sensor

nodes within the network and their operating status. The architecture design for the GUI is shown as follows

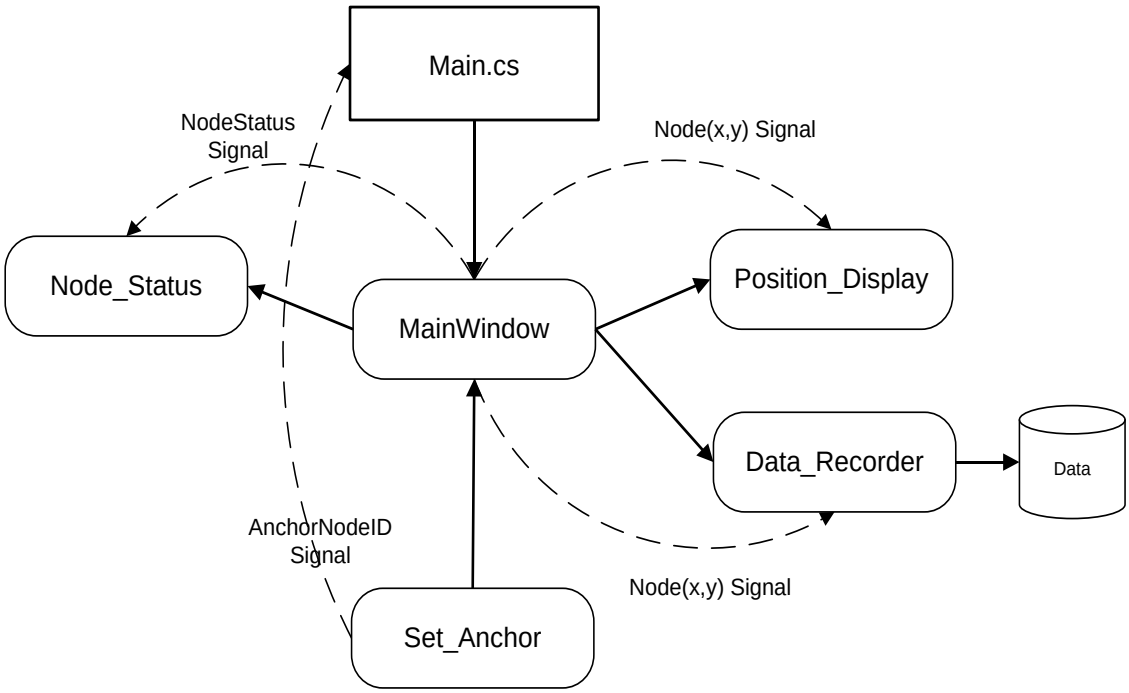

Figure 4-21: SOSC GUI Architecture

The main program contains the localization algorithm and the I/O device handlers built from the Windows API. The architecture shown in Figure 4-21 assumes the incoming UART data has been received and processed in the background, therefore the interconnection with the serial port is not shown. The “MainWindow” is responsible for the visual display that handles the display of node position layout, the data recording of the real-time localization computation results, and the display of the real-time status of the SOSC nodes. Also, as part of the localization computation, this GUI allows the user to choose the two anchor nodes of interest to provide the correct visual layout of the SOSC nodes. The GUI has the following look

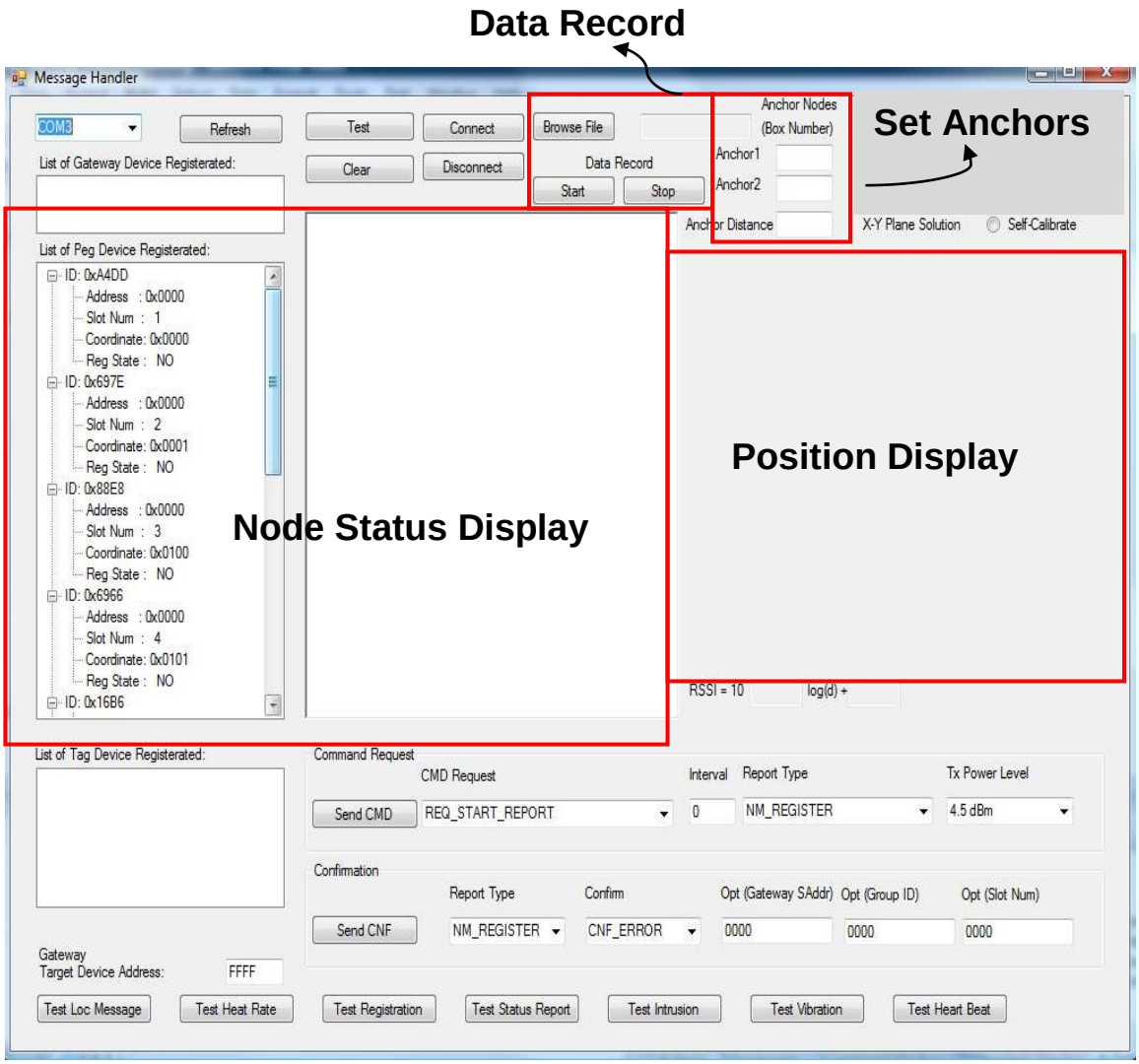

Figure 4-22: The Actual GUI Implemented

Figure 4-22 illustrates the actual GUI used to display the result in localizing the SOSC node positions. The GUI components highlighted with the red box are the ones used for the purpose of localization and SOSC network related information display.

4.4 Chapter Summary

This chapter describes the overall SOSC devices' network, hardware, firmware architecture design, and the back-bone MDS algorithm design on a PC. As for

hardware architecture, two major components, the microcontroller unit and the antenna are carefully chosen based on its functionality, such as the I/O capability and antenna directionality, to reliably obtain distance data for localization computation. The firmware architecture establishes the communication between SOSC devices for the centralized monitor or computer, in this case, being the PC, to obtain the inter-node distance information, enabling the PC application to compute and to display the localization result. In the next chapter, we will discuss the mapping algorithm to obtain the final localization result. The mapping algorithm proposed in this thesis has a distributed nature. However, this thesis concentrates on analyzing and demonstrating the localization performance and not on the network performance. As a result, the algorithms are made centralized on PC, consisting of the MDS based position computation algorithm, the RMU, and the mapping algorithm.

Chapter 5 - Localization and Mapping Design

Previously, the software architecture of the MDS based position computation algorithm was discussed. The proper RSSI-to-distance conversion model and reconstruction of the SOSC node layout are yet to be described. Two of the most important features of the SOSC WSN are the real-time ranging and the clustering localization features. The real-time ranging is done through a range modeling unit (RMU), which provides a simple ranging for position estimate. The clustering localization allows the centralized localization algorithm to successfully and thoroughly compute the location of every node within the network, regardless of the scale and shape of the network layout. The detail of RMU and the clustering mapping reconstruction features will be discussed in the following section

5.1 Range Modeling Unit (RMU) Distance Conversion

In real-world environments random electromagnetic interference and noise are inevitable in radio transmission. These interferences introduce additional degradation in signal strength, causing the RSSI values to degrade inversely proportional to distance in a non-monotonic fashion. To illustrate this non-monotonic inversely proportional RSSI to distance relationship, an experiment was conducted by using two nodes to broadcast periodically to each other and to obtain the RSSI of the broadcast packet at different distance away from each other. The measurement starts at 2.5 meters separating the two nodes, and for every 2.5 meter separation increments, 1000 RSSI samples are recorded over time up to 40 meters.

The RSSI samples are averaged over the 1000 samples taken at each distance and the final results are shown in Figure 5-1. It is obvious that, although in general, the RSSI degrades in an inversely proportional fashion; at 27.5 and at 30 meters, the signals appear to be much weaker than that at further distances. According to the literature [5] and [6], discussed earlier in Chapter 3 -, the common cause for this weaker signal at 27.5 and 30 meters is due to fading or various reflections from the surroundings.

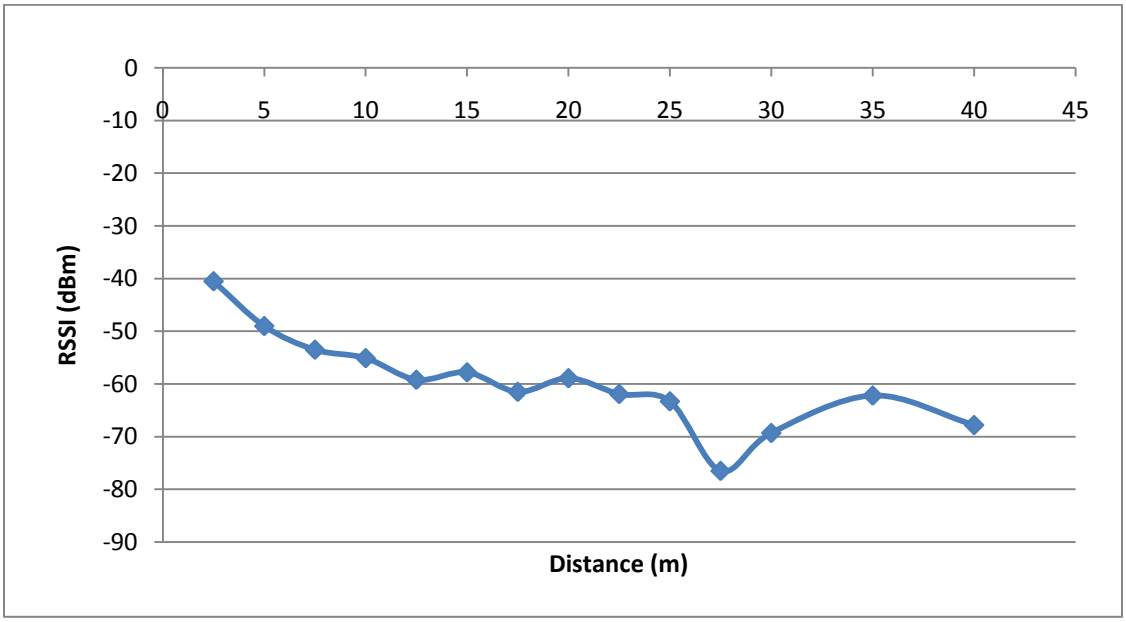

Figure 5-1: Experimental RSSI to Distance Relationship

Regardless of the EM interference and fading that introduces irregularity in signal strength degradation; there is still a clear inversely proportional relationship between RSSI and distance according to Figure 5-1. Therefore, it is possible to model this characteristic to allow the interpretation of distance from RSSI in an ideal environment. However, in reality, without a proper radio channel model to convert RSSI to distance, the accuracy of conversion will be easily influenced by

the ever-changing channel noise and interference. The loss in distance conversion accuracy can result in a catastrophic localization estimation error. Two common methods can be applied to conduct the RSSI-to-distance conversion; the first one is the empirical method, where the system obtains knowledge of the environment by collecting a sufficient amount of offline data, such as the propagation property between any two nodes with known distance. The empirical approach gathers the radio channel data for every link within the network; the collection of data allows every link within the network to calculate a probability density function (pdf) for channel properties, which allows the links to determine the most probable propagation model to convert RSSI into the corresponding distance. However, determining the radio propagation model for every link requires the knowledge of all the inter-node distances, which defeats the purpose of building a self-localizing and maintenance-free WSN.

The second method for RSSI-to-distance conversion is by utilizing the theoretical radio propagation models. The advantage of using a theoretical model is that it requires no system training and is easy to implement. However, one theoretical model is unable to cope with all the random interference and reflections in every environment. The theoretical models are only suitable in a nearly ideal environment, where every node has perfect LOS communication in a perfectly flat terrain without obstructions.

Apparently, the two conversion methods demonstrate different strengths in different system design criteria. The former is suitable for a complex system with good computation and storage capability for providing the most probable

conversion result from the large collection of off-line data. The second method is simple, but it lacks the flexibility to adapt to random changes caused by various interferences. As a result, this thesis proposes a hybrid solution that produces a nearly-empirical and theoretical RSSI-to-distance conversion model in a run-time environment. The nearly-empirical character of the self-calibrating distance conversion module gathers the run-time inter-node RSSI and calculates the propagation model in real-time. The propagation model is theoretical and is based on (17), whose propagation constant, n, and constant, A, are calculated upon each reception of each RSSI value. In the SOSC WSN, incorporating this real-time RMU allows the system to interpret the RSSI values more reliably under noisy conditions that cause the RSSI to appear weaker than predicted by the theoretical model at a given distance. The following sections will describe the RMU design and its performance compared to the empirical method.

5.2 SOSC RMU Architecture

The RMU's architecture is inspired by (17). The definitions of the two constants in the equation define the communication scheme, facilitating the real-time radio propagation modeling. The system architecture is illustrated as follows

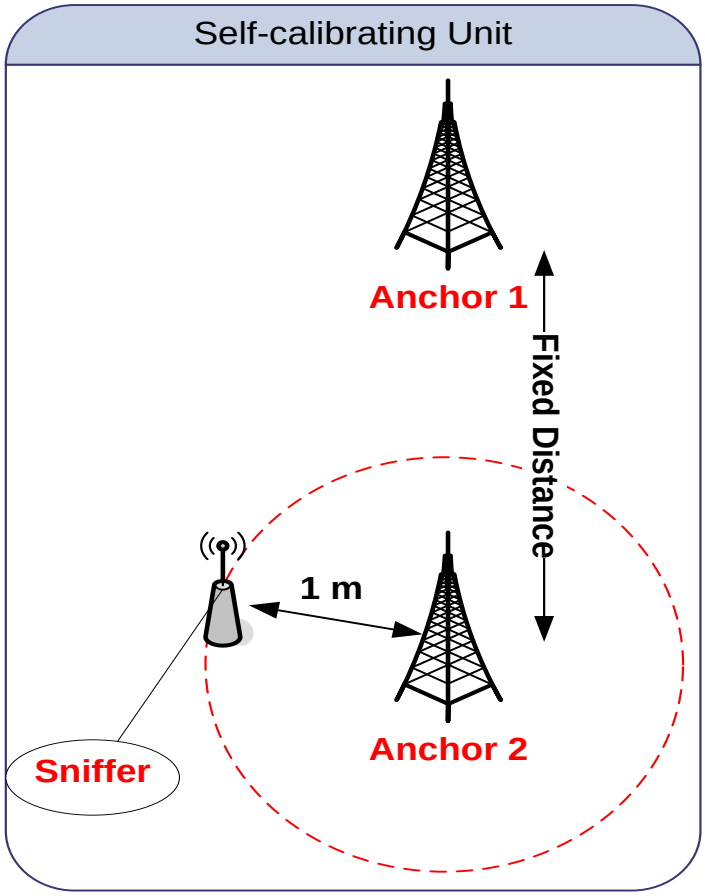

Figure 5-2: SOSC Self-Calibrating Distance Conversion Unit Architecture

The RMU shown in Figure 5-2 consists of one “sniffer”, which is responsible for retrieving the RSSI at 1 meter away from the neighboring anchor node. This RSSI value represents the constant, A, in (17). The other components included by the RMU are the three anchor nodes. These three anchor nodes not only are responsible for determining the correct mapping orientation and layout, discussed later in the section, but also are responsible for determining the propagation constant, n. The three anchor nodes are fixed at some known distance away from each other; the propagation constant is calculated from the inter-anchor RSSI in the run-time environment. The reporting of RSSI values to determine the propagation and the A-constants are based on the asynchronous

periodic reporting scheme described previously in section 4.3. The flow for calculating the real-time propagation model is shown below

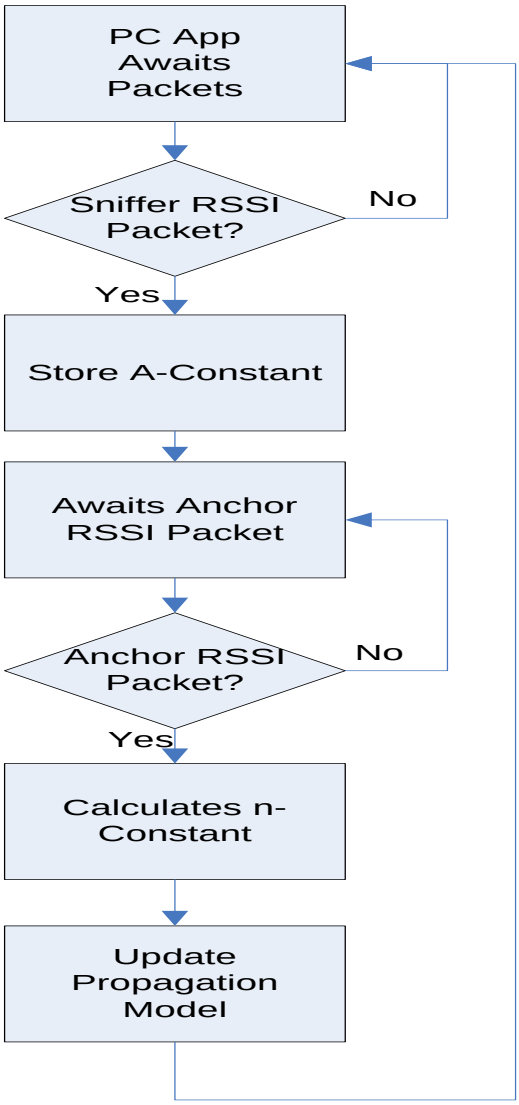

Figure 5-3: RMU Radio Propagation Model Forming Process

The process for determining the real-time radio propagation model, shown in Figure 5-3, is a stand-alone process that operates in parallel with the process that gathers the inter-node RSSI values from all SOSC nodes. This process of RMU converts all the inter-node RSSIs into their corresponding distances in real-

time. It adapts to any interferences by updating the propagation model constantly to ensure a correct distance conversion for each inter-node RSSI. However, because noise distribution is uneven across space, it is difficult to characterize a large network with just a single RMU. The performance of the RMU is demonstrated in the following section.

5.3 Performance Comparison of the RMU

This section is to demonstrate how RMU is resistant to interference and to successfully convert RSSI into the corresponding distance with minimal error. This experiment is conducted with the set-up shown in Figure 5-2 in an indoor environment, and the converted distances are recorded through the PC application GUI. The set-up is illustrated below with a floor plan of the cubicle

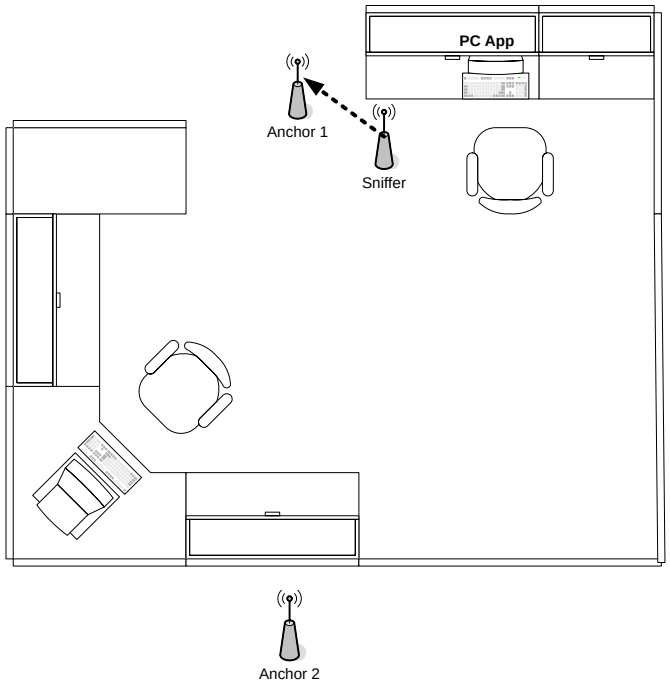

Figure 5-4: Indoor Experimental Layout for RMU

This experiment is arranged in a way, shown in Figure 5-4 the two anchor nodes are separated with the cubicle divider which can cause reflection and further degrades the transmission strength at any given distance. The experiment is conducted at 2 meters and at 7 meters separating the two anchors respectively. During the experiment, the propagation constant and the A-constant are recorded. The average of the collection of propagation constants and the A-constants are used to define the empirical radio propagation model. Results of the RMU conversion and the empirical conversion are compared and shown below

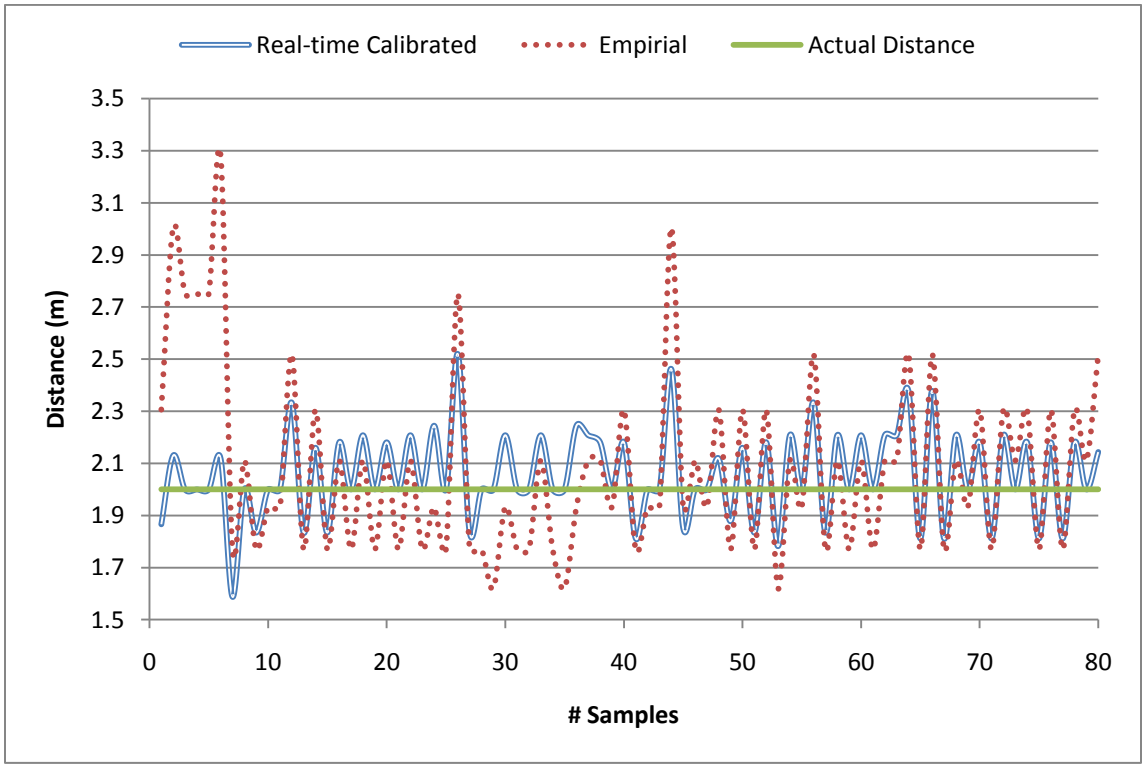

Figure 5-5: Self-calibration vs. Empirical RSSI-Distance Conversion at 2m Apart

The objective for the two conversion methods is to provide the conversion result as close to the physical separation, in this case, 2 meters, as possible. It is

obvious in Figure 5-5 that the conversion by RMU can estimate the actual distance better and more reliably than the empirical method. A detailed analysis on the percentage error and variance of the conversion are shown in the following table

Table 5-1: Average % Error and Variance of Distance Conversion with 2-meter separation

	Average Error (%)	Variance(m^2)
RMU	6.9	0.03
Empirical Conversion	13.6	0.12

Table 5-1 shows that from numerous conversions performed at a fixed distance of 2 meters, RMU almost doubles the accuracy, as compared to the empirical conversion method. The variance of the conversion results obtained with RMU is very small compared to that of the empirical conversion. These two indicators demonstrate that RMU not only has a better RSSI-distance conversion accuracy, but also has a much more consistent and a more repeatable conversion, due to its lower variance. To further show RMU's superior capability in providing a reliable RSSI-distance conversion, the same setup is used with separation of 7 meters. The results are shown in Figure 5-6

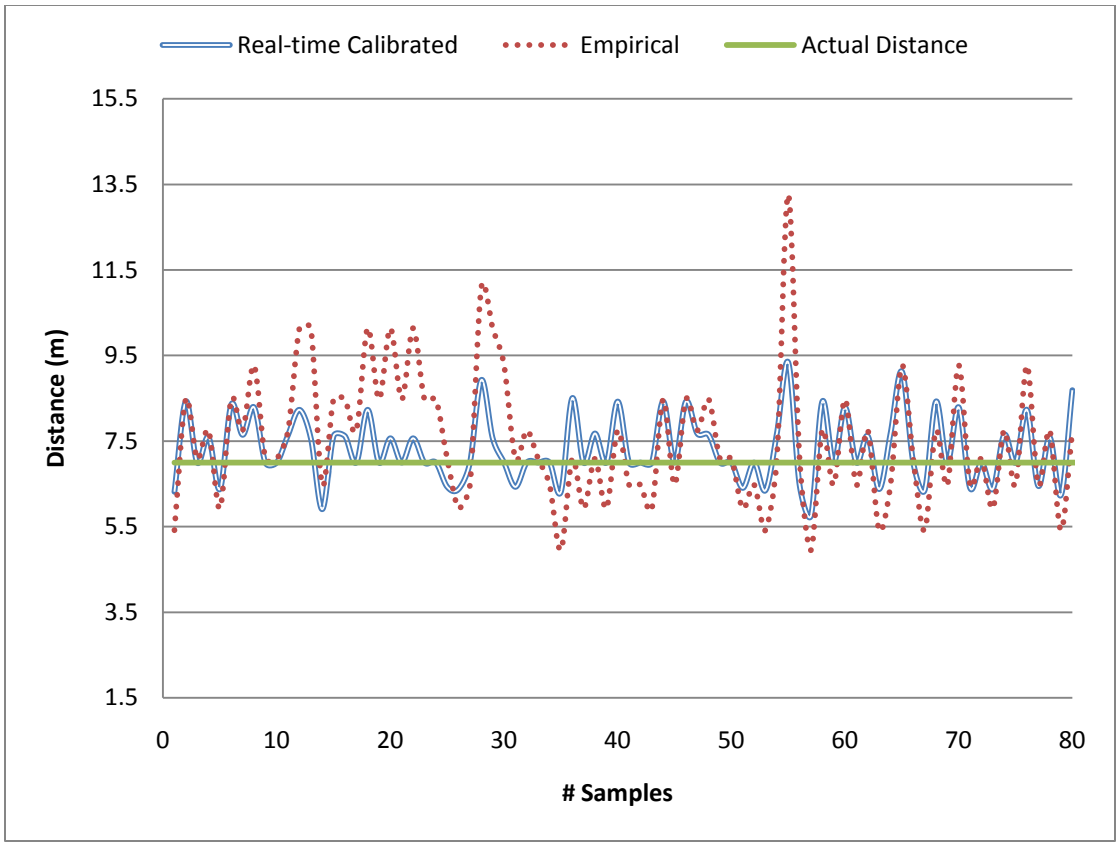

Figure 5-6: Self-calibration vs. Empirical RSSI-Distance Conversion at 7m Apart

Table 5-2: Average % Error and Variance of Distance Conversion with 7-meter separation

	Average Error (%)	Variance(m^2)
RMU	8.8	0.58
Empirical Conversion	16.4	2.17

Table 5-2 shows the consistency in RMU's superior performance over the empirical conversion. From conducting the above experiments in an indoor environment with numerous obstructions, it is obvious that RMU is capable of minimizing the influence of electromagnetic interference and noise to convert from RSSI to distance with sufficient amount of accuracy. RMU, unlike the empirical conversion, does not require off-line training and is able to adapt to the environmental condition in real-time.

5.4 SOSC Mapping Technique Design

As previously described, the procedure of a localization algorithm follows three major steps; the first step is to gather distance information between each pair of nodes within the network; secondly, the position computation algorithm calculates the corresponding position in a 2D-Euclidean space; finally, the most important step is to determine the actual layout for the node positions computed in the former step. In [46], the author proposes a novel mapping technique by partitioning WSN into quadrilaterals, formed by 4 nodes. The layout of the nodes can be realized by finding the overlapping nodes between the quadrilaterals. However, two of the most common problems in map reconstruction, also noted by the author, are the translational and orientation ambiguity. As for orientation ambiguity, it consists of rotational and flipping ambiguities. To reduce the number of degrees of freedom, reference points or anchors are required to provide the origin and direction of the map layout. In most cases, at least three anchors are required to determine the layout. For simplicity, the following section will assume the direction of the network layout and uses three anchor nodes to demonstrate the procedure to overcome the above ambiguities.

5.4.1 **SOSC Map Reconstruction**

The main purpose of the map reconstruction is to align the MDS-estimated node positions to the predefined anchor positions. This alignment procedure is an iterative minimization problem which ultimately allows the anchor nodes to be aligned to best fit their original configuration. As previously discussed, the ICP algorithm facilitates this minimization of discrepancy between two sets of

coordinates. With the use of ICP (see section 2.3), the map reconstruction algorithm is outlined in Figure 5-7

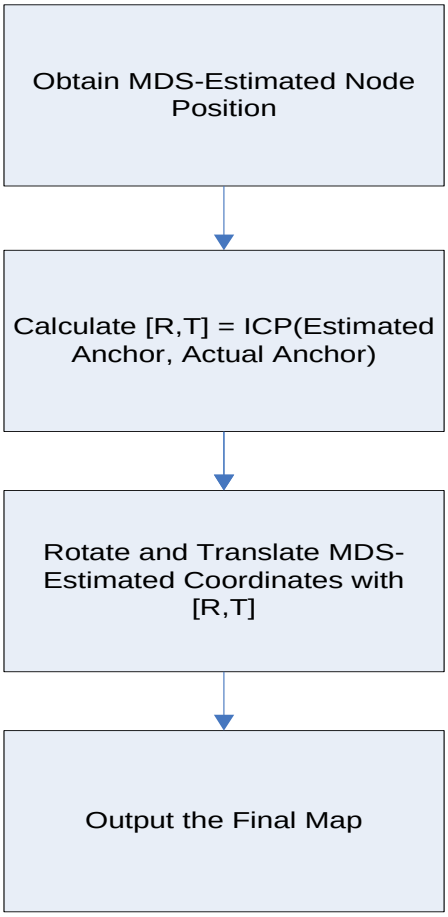

Figure 5-7: Map Reconstruction Procedure Flow

To demonstrate the reproducibility of the original mapping with the procedure shown in Figure 5-7 and to justify the use of ICP algorithm for minimizing mapping error, a simple scenario is created with the use of the following dissimilarity matrix.

Table 5-3: Dissimilarity Matrix of a Simple SOSC WSN

	Anchor_1	Anchor_2	Node 1	Node 2	Node 3	Node 4
Anchor_1	0	0.5	0.4717	0.6021	0.3202	0.4924
Anchor_2	0.5	0	0.2693	0.4610	0.3905	0.5408
Node 1	0.4717	0.2693	0	0.2	0.2	0.2828
Node 2	0.6021	0.4610	0.2	0	0.2828	0.2
Node 3	0.3202	0.3905	0.2	0.2828	0	0.2
Node 4	0.4924	0.5408	0.2828	0.2	0.2	0

It is important to note that the inter-node distance information from Table 5-3 is normalized for illustration purpose. The corresponding layout, shown with hollow dots, of the above distance matrix is plotted with the MATLAB

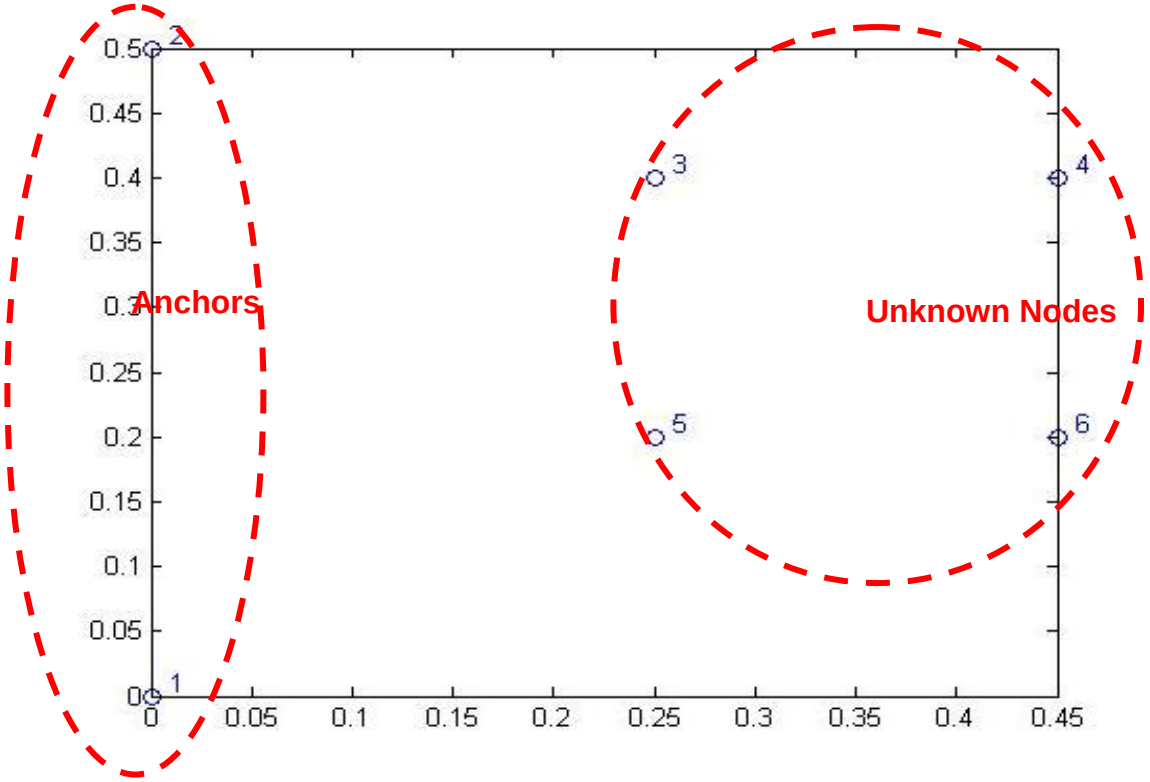

Figure 5-8: Original Layout of the SOSC Network

This demonstration assumes that the translation and rotation vectors have been calculated by ICP with respect to the two anchor nodes' original coordinates. The

steps taken by the SOSC mapping algorithm to reconstruct the mapping are illustrated below

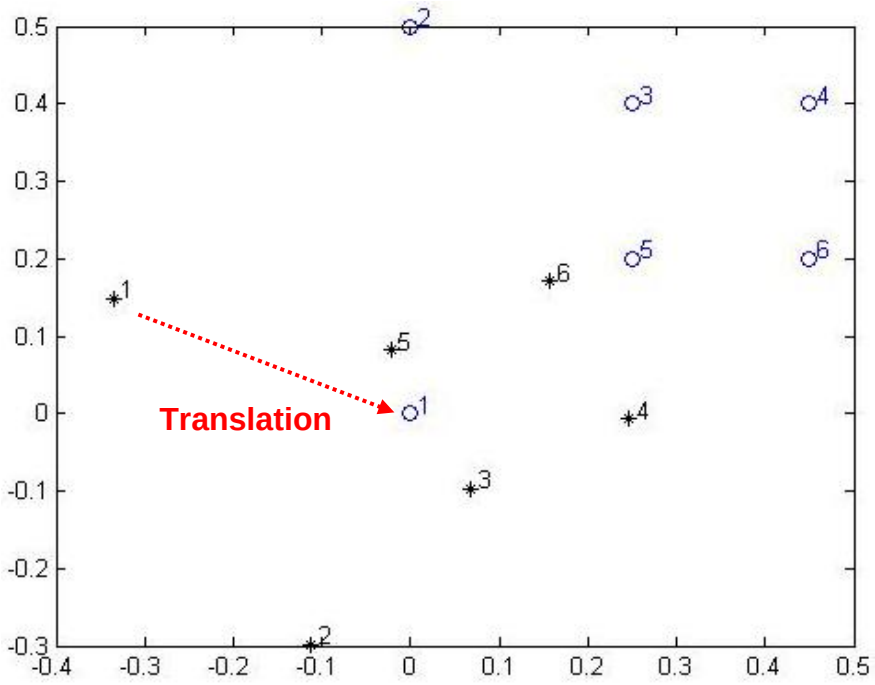

Figure 5-9: Map Reconstruction Step One - Translation

The MDS algorithm, as described previously, provides the best matched solution in terms of inter-node dissimilarity, but the corresponding estimated coordinates very often do not reflect their actual layout. Therefore, to reconstruct the map, the next step is to translate and rotate the MDS result to the designated anchor positions

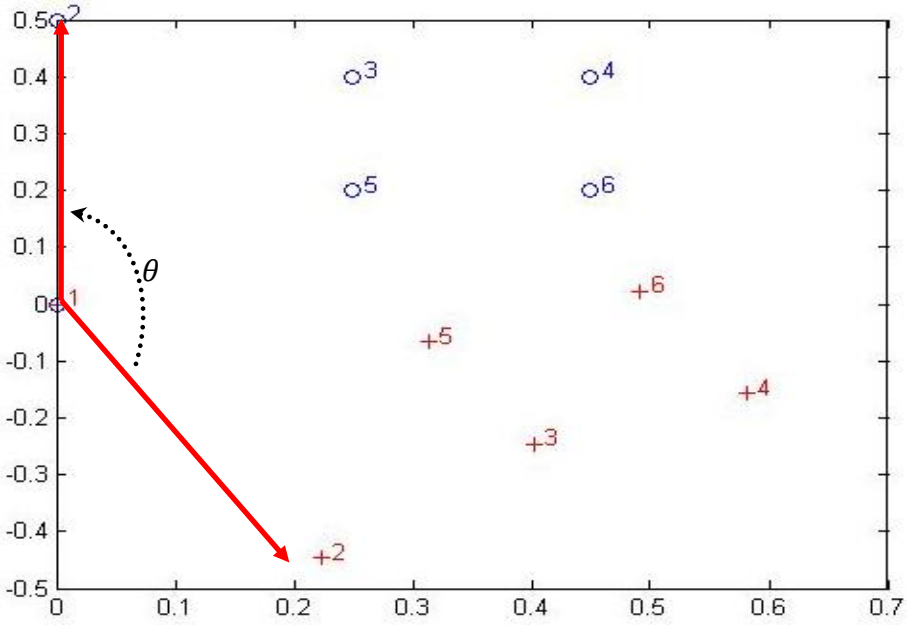

Figure 5-10: Reconstruction Step Two - Rotation

In Figure 5-10, it is easy to see that the current solution requires rotation to completely position the two anchor nodes at their designated coordinates.

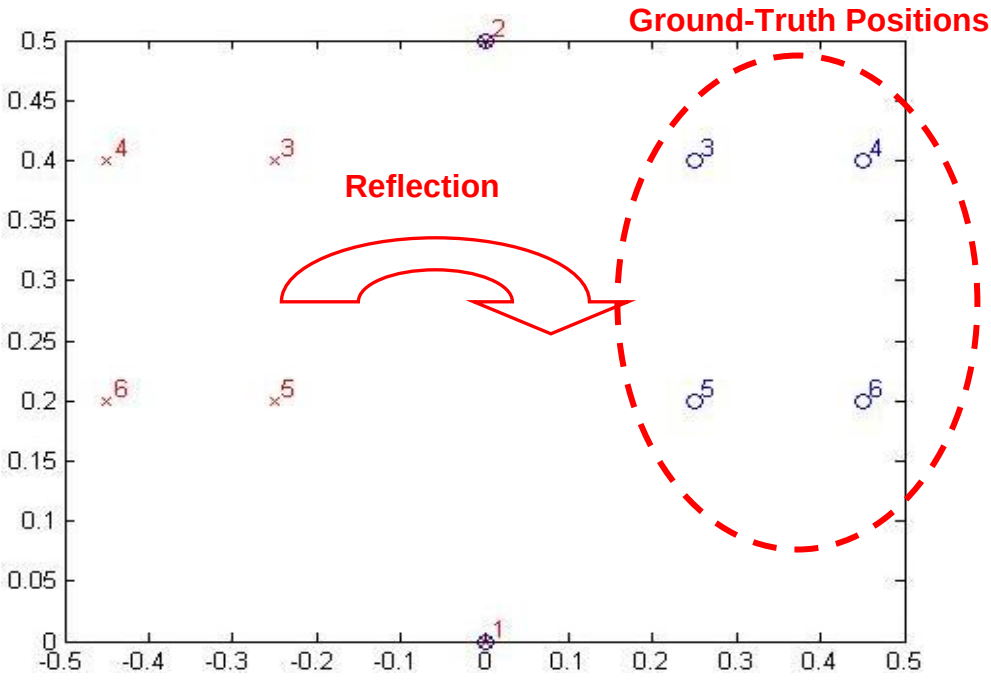

Figure 5-11: Reconstruction Step Three – Flipping

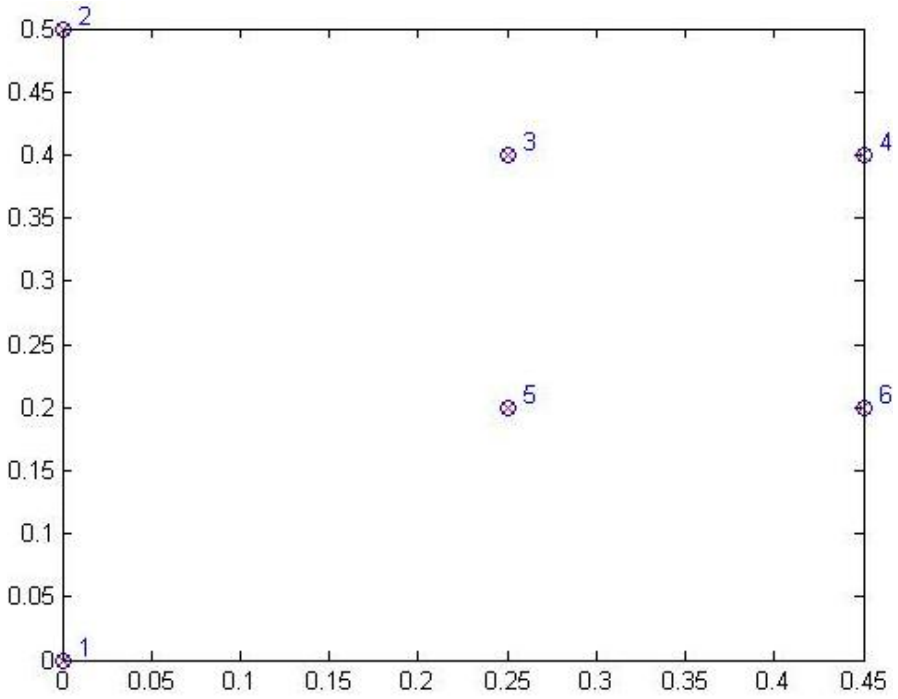

Figure 5-12: Reconstruction Step Four – Finalizing Mapping

Figure 5-11 demonstrates the duality of node layout. This duality occurs in a two-anchor WSN or in situation where anchor nodes are collinear with one another. Without prior knowledge of the network layout, it is difficult to determine whether the estimated coordinates require flipping. To overcome potential flip ambiguity, SOSC WSN utilizes three anchor nodes that are required to be not collinear.

This map reconstruction algorithm is independent of the number of nodes within the network. The above example has demonstrated that this algorithm is capable of reproducing the actual node layout from their inter-node distance information. However, the above example is assumed to be in an ideal and fully connected WSN without presence of EM interference that causes ranging error induced localization error. Localization error can also be induced by partially connectivity in a WSN, where a particular pair of nodes contains a broken link,

failing to produce distance information. This results in an empty entry in the dissimilarity matrix and prohibits the classical MDS algorithm to fail. To address the common range measurement error and connectivity issues that fail the classical MDS algorithm, this thesis proposes a novel MDS-based localization algorithm, called clustering MDS algorithm.

5.4.2 **Clustering MDS Localization Algorithm**

In ideal conditions, assuming perfect inter-node connectivity in the network without EM interference, it is possible to localize the whole network with the classical MDS algorithm from the inter-node distance information. However, in reality, wireless nodes have limited signal coverage and cannot obtain the RSSI from nodes outside of its coverage. This lack of connectivity between node-pairs can cause nodes not to be localized. As discussed earlier, related works attempt to use hop-count as their distance measuring metric; this technique can be performed with multi-hop communication scheme and is less influenced by the coverage. However, the disadvantage of hop-count based localization is its low resolution and inaccuracy. As a result, for RSSI based localization, in order to cope with the nodes' inability to obtain distance information from out-of-range nodes, this thesis introduces a novel localization method based on the concept of clustering.

The design of clustering MDS localization algorithm is inspired by the previous work done by Stefano Basagni [47] and Rajesh Krishnan [48]. Their works apply the concept of clustering to group the nodes in an ad-hoc network to create a hierarchical network organization to reduce the amount of data required to

maintain routing and control information in a mobile environment. Each cluster is governed by a cluster-head, defined by the node having the largest weight; the nodes within a one-hop distance away from the cluster-head will become members of the cluster. Based on the concept of clustering, in localization applications, many ideas from [49], [50], and [51] also apply local information to construct the global map of WSNs, whose nodes do not have perfect connectivity with every other node in the network. The classical MDS algorithm can be applied in the cluster by cluster fashion, so it does not require knowledge of the inter-node distance information of the global map. As a result, even without connectivity of certain node-pairs, the clustering MDS localization is still able to localize the particular node from its other neighboring nodes with good connectivity. The algorithm for the clustering MDS localization is

Algorithm Clustering_MDS(*Anchor_Pos{x_i,y_i} i=1,2,3 , Dissimilar$_{matrix}$[]****)***
Number_of_Network_Nodes = ***length_of(****Dissimilar$_{matrix}$[]****)***
node_to_clutDist = ∅
Clust_Index = {1,2,3}
If (N != 0)
for each n in N do
for each m in ***length_of(****Clust_Index[]****)*** *do*
node_to_clutDist[1,n] += Dissimilar$_{matrix}$[n, Clust_Index[1,m]]
endfor
endfor
Closest_Node = ***min(****node_to_clutDist****)***
Closest_ThreeAnchors =
Find_Three_MinAnchor(*Dissimilar$_{matrix}$[n, Clust_Index]****)***
Closest_ThreeAnchors = ***Add(****Closest_Node****)***
New_Dissimilar$_{matrix}$ = ∅
for j = 1 to 4 do
for k = 1 to 4 do
if j≠k then

```
                    New_Dissimilar_matrix[j,k] =
    Dissimilar_matrix[Closest_ThreeAnchors[1,j], Closest_ThreeAnchors[1,k]]
                endif
            endfor
        endfor
        Est_Node_Positions = MDS_Algo(New_Dissimilar_matrix)
        [R,T] = ICP(Anchor_Pos, Est_Node_Positions)
        Map Node_n with R and T
        Anchor_Pos = Add(Node_n)
        N = N-1;
        Clust_Index = Add(Index_of(Node_n in Dissimilar_matrix))
    else
        return Anchor_Pos
    endif
```

Figure 5-13: Clustering MDS Algorithm

At the start, the WSN contains N un-localized nodes and three anchor nodes with known physical locations. The three anchor nodes form the initial cluster for the network. During run-time, the algorithm, for each un-localized node, adds all the inter-node distances to its cluster nodes and determines the un-localized node that is closest to the current cluster. This closest node will be localized by its three closest cluster nodes with the classical MDS algorithm. Once the node is localized, ICP algorithm will be used to calculate the corresponding rotation and translation required to map the estimated location of the un-localized node with respect to the closest three cluster nodes. After conducting rotation and translation, the node will be added to the existing cluster and used to localize the rest of the un-localized nodes.

According to the RSSI range data collected with a pair of nodes, the result in Figure 5-1 has demonstrated a non-monotonic degradation in RSSI in regions between 25 and 40 meters. The propagation model in (17) shows that as

distance increases, RSSI value's rate of attenuation decreases. When interpreting distance with weak RSSI values can introduce a large and distance estimation error. Figure 5-14 is the illustration of ranging error induced by using weak RSSI signals as sources. It is clear to see that the same 30 dBm noise that degrades a stronger RSSI signal and a weaker RSSI signal results in completely different ranging errors. In this case, the stronger RSSI introduced only 3 meters of error, while the weaker RSSI introduces 30 meters of discrepancy. As a result, to reduce the error induced by EM interference or fading, the clustering MDS algorithm described earlier, specifically localizes the node that contains the strongest RSSIs to the cluster nodes. In other words, the localization algorithm proposed by this thesis performs localization based on the fact that weaker signal is more prone to error in distance estimate. For nodes that are closer together, the RSSI values tend to be stronger and their path loss characteristic is much more monotonic, and therefore providing a more accurate distance information and mitigating the overall localization error.

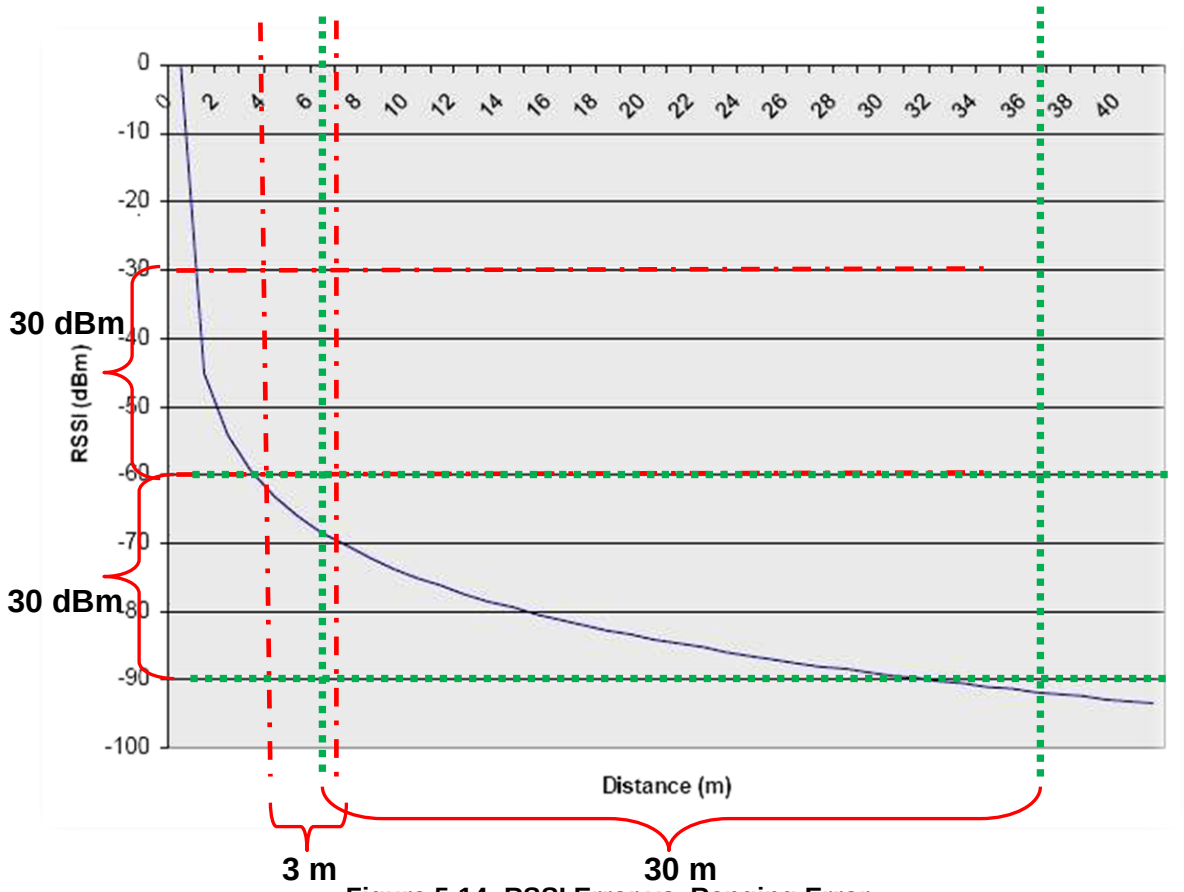

Figure 5-14: RSSI Error vs. Ranging Error

5.5 Chapter Summary

This chapter describes the design of the RSSI to distance conversion unit, RMU, the mapping methodology, and finally, the design of the clustering MDS localization algorithm. Since, the SOSC WSN is designed for low-cost and low-power applications; it does not require complex antenna and timing systems required for the AOA and TDOA ranging techniques. Instead, it adapts a RMU, which utilizes a generic propagation model to characterize the real-time distance conversion model in a real-time environment. Due to random EM interferences and ever changing climate conditions, the need for RMU is shown to be essential, because an empirical conversion model cannot dynamically adjust

parameters in the propagation model to account for the changes in the communication channel. Having had the distance information, this thesis demonstrates that it is possible to reconstruct the physical map by translation and rotation, with only three anchor nodes. Although, RMU is used to provide a real-time RSSI to distance interpretation, it is not capable of detecting the faulty RSSI values, induced by fading or interferences. As a result, clustering MDS algorithm is designed to discard the weaker RSSIs, which are prone to ranging error, from localization. Clustering MDS algorithm only considers the stronger and more reliable RSSIs to estimate node locations and so is able to mitigate localization error. In Chapter 6, simulations and experimental results will be shown to proof the validity of the above claim.

Chapter 6 - Results and Discussion

As previously discussed, due to limitation in radio signal coverage, nodes in a large scale WSN are not able to obtain distance information from all other nodes in the network. Also, due to EM interference and fading, the inter-node distance information obtained from network does not always reflect to the physical distance. The missing connectivity and faulty distance information can lead the classical MDS algorithm to provide poor position estimation, resulting in erroneous global map reconstruction. RSSI ranging is not known for its reliability and accuracy, but many literatures, such as [52], have attempted to perform localization with only radios so the overall system can be made inexpensive and energy efficient. Essentially, the accuracy of the localization is limited to the accuracy of the range measurement. Authors in [53] enhance this inaccuracy by performing an extensive data collection to build a statistical range model for various types of environments. All the literatures this thesis has discussed earlier attempts to achieve a good localization result by incorporating extensive number of anchor nodes, refined ranging techniques, or localization computation that rely greatly on the knowledge of the network environment. As for SOSC WSN, the system requires only three anchor nodes for mapping reference; a simple RMU that characterizes the real-time RSSI-to-distance model; a simple clustering MDS algorithm that can localize partially connected WSN with much less computation complexity than the classical MDS localization. In this chapter, simulations and real-world scenarios on localization and computation complexity will be demonstrated to show its superiority over the classical method.

6.1 Computation Complexity

The classical MDS algorithm is computationally expensive in nature. It is mainly due to Eigen-decomposition, whose computation complexity is $O(n^3)$. To localize a large scale network, using the classical MDS algorithm can be computationally expensive. The clustering MDS algorithm, on the other hand, performs the classical MDS algorithm in a cluster of 4 nodes for $n-3$ clusters, excluding the 3 anchor nodes. The resulting computational complexity without the mapping reconstruction algorithm is $O(c(n-3))$, where c is the constant time required to perform the classical MDS over 4 nodes. The map reconstruction algorithm, in this case being the ICP algorithm, whose computation complexity is $O(n)$, performs linear iteration over 4 nodes and repeats for $(n-3)$ clusters. This process iterates through each individual un-localized node across the whole network. As a result, the overall clustering MDS mapping algorithm has computational complexity of $O(c'(n-3))$, where c' is the constant time combining the computation time of 4-node MDS and 4-node ICP algorithms, which is simply $O(n)$.

The classical MDS algorithm exhibits a computation complexity of $O(n^3)$, while the clustering MDS algorithm only produces a complexity of $O(n)$. The clustering MDS algorithm, due to its linear computation complexity, is much more scalable for large network than the classical MDS algorithm. By examining the computation complexity of the two algorithms, the clustering approach has already established a great advantage over the classical method.

6.2 Localization Performance

In this section, simulations are conducted to compute localization performance of classical MDS and clustering MDS in both ideal and noisy environments. The simulation of an ideal environment assumes perfect range measurement and perfect node connectivity within the network. All localizations are performed with only two anchor nodes, placed along y-axis for mapping reference. The localization results are distributed in a 1x1 unit square space for illustration purposes. The simulation results for classical MDS and clustering MDS are shown in Figure 6-1

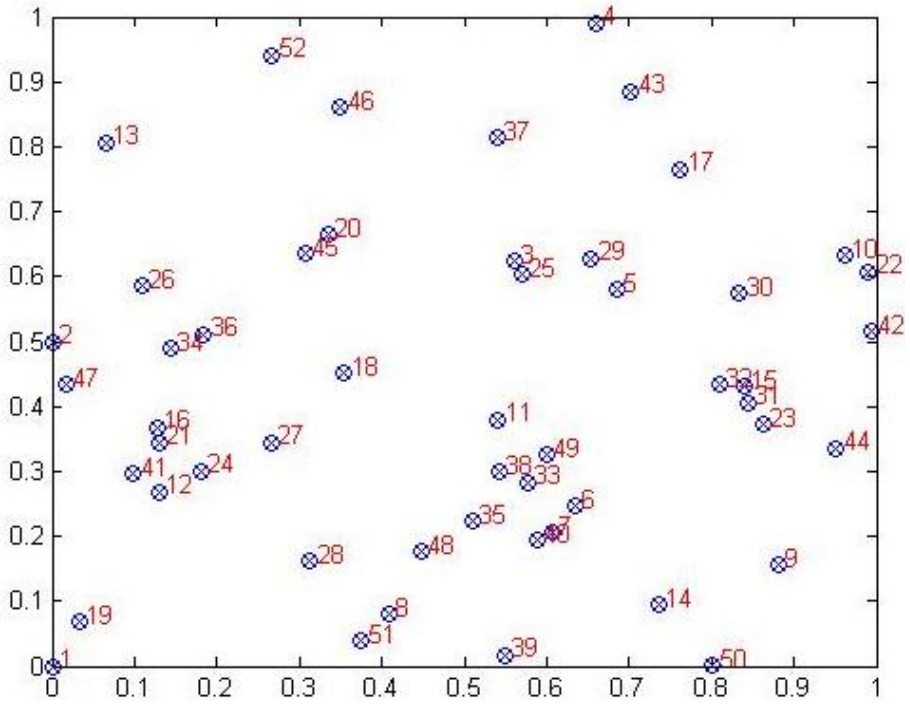

Figure 6-1: Classical MDS Map Reconstruction

The nodes marked with '○' are the actual layout and ones marked with 'x' are the estimations. This simulation is done by randomly generating the fifty node positions at start-up. The inter-node distances are calculated directly from the

generated coordinates, because this situation is assumed to be ideal, so the distance information should reflect its actual physical layout. The localization results are calculated with the MDS algorithm from the inter-node distance matrix. The objective of this simulation is to show the MDS algorithm's capability to reconstruct the original layout from just the distance information. It is easy to see that the estimation match with the original layout perfectly. Another simulation, shown below in Figure 6-2, uses clustering MDS localization algorithm.

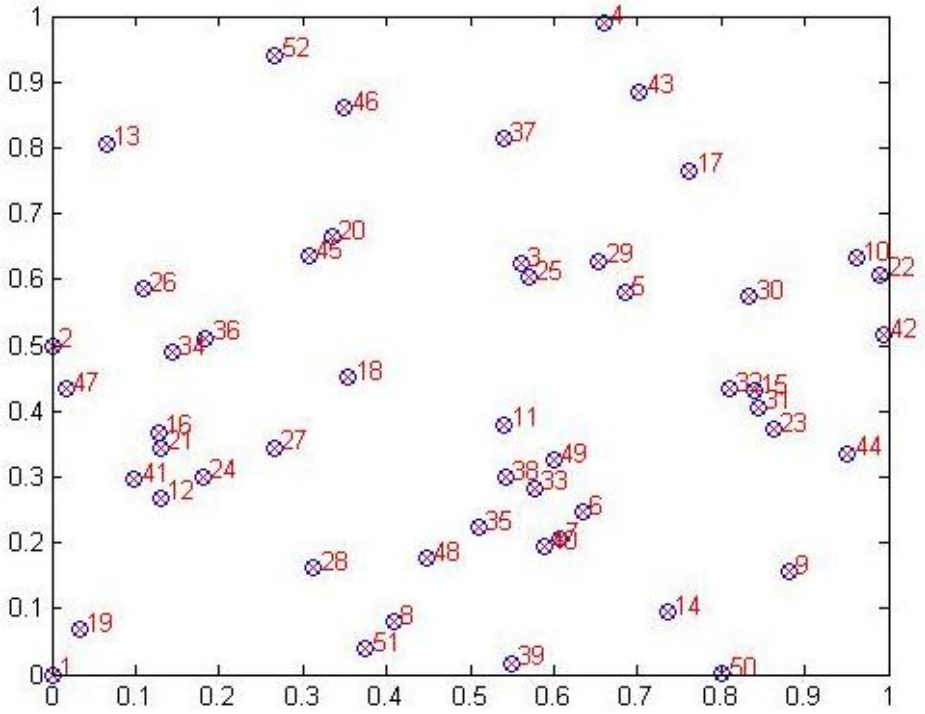

Figure 6-2: Clustering MDS Algorithm Map Reconstruction

From Figure 6-2, it is obvious to see that, with the clustering MDS algorithm, the localization result matches with the original layout. However, just by visualizing, it is difficult to evaluate the performance of the two algorithms. Therefore, a quantitative measure is required for evaluating the two algorithms. In this case,

mean error of position estimation is calculated for individual nodes within the network. The estimation error is defined by the following equation

$$\%Error = \sum_{i=1}^{N} \frac{\left\| X_{i(actual)} - X_{i(estimate)} \right\|}{\left\| X_{i(actual)} \right\|} \times 100 \tag{18}$$

Where i is the number of nodes within the network, and the Euclidean distance is measured with respect to the origin. This error essentially evaluates the "closeness" between the original point and the estimated point as a fraction of the original point's distance away from the origin. Utilizing this formula, the estimation accuracy of both the classical MDS and clustering MDS algorithms under perfect environment conditions are shown in Table 6-1

Table 6-1: Position Estimation Error Comparison

Algorithm Type	Estimation Accuracy (%)
Classical MDS Algorithm	1.44×10^{-12}
Clustering MDS Algorithm	2.14×10^{-11}

It is obvious that under perfect environmental and network conditions, both algorithms are able to reconstruct the original map layout with minimal error. However, the clustering MDS algorithm in this case exhibits a slightly worse accuracy then the classical method. This deviation in estimation accuracy is due to various round-off and precision errors during the rotational and translational map transformation processes. A larger propagation error can exist for a larger WSN. However, the round-off and precision error compared to the distance estimation error in real-world scenario is negligible. The following simulations demonstrate the estimation error of the two localization algorithms under more

realistic conditions, where the inter-node distance information is influenced by EM interference and induces error.

The experimental data in Figure 5-1 shows that as distance between nodes increases, the inter-node RSSI weakens, but the RSSI's rate of decrease is inversely proportional, resulting in poor resolution for interpreting far inter-node distances. Radio signals are prone to noise and fading in nature, and therefore, a slight variation in weak RSSI can introduce a large variation in distance conversion. To demonstrate the effect of signal noise on localization, a simple model on noisy distance information is defined as

$$d_{ij(noise)} = d_{ij(actual)} \times (1 + d_{ij(actual)} \times Err) \quad (19)$$

where i, $j = 1,2,..N$, and N is total number of nodes within the network. This simple model assumes that the noise in RSSI caused by interference or fading scales linearly with distance. The further an inter-node distance is, the larger the distance measurement error. In reality, this error relationship can be exponential according to the observation from Figure 5-1. However, to simplify calculations and for illustration purposes, it is sufficient to assume a linear distance estimation error model.

To demonstrate the above claim, two simulations are presented with ten unknown and randomly distributed nodes; maximum range error of 13% and 23% are introduced respectively to the nodes. The localization results calculated from classical MDS and clustering MDS algorithms are shown below

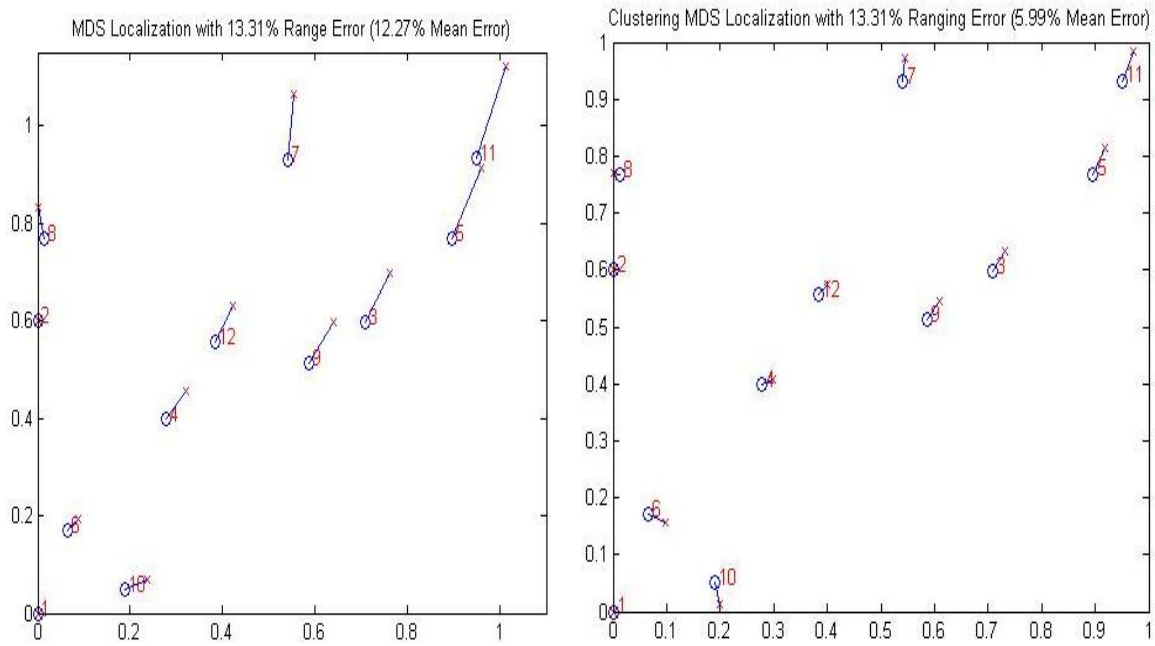

Figure 6-3: Classical vs. Clustering MDS Localization with 13% Range Error

The mean estimation error is the average of all the estimation error of every node in the network. It is not surprising to see that the classical MDS algorithm produces a larger positioning error than the clustering MDS algorithm. The classical MDS algorithm calculates the best-fit map solution based on all the inter-node distance information within the network. In the above simulation, where the range error is bounded by the maximum error of 13%, the classical MDS algorithm has to account for every error, and therefore its solution will approach an error of 13%. On the other hand, the clustering MDS algorithm always progressively localizes the closest node, whose distance information is less influenced by variation in RSSI, so its localization result is not dependant on the maximum error. In other words, because of the data-selective nature of clustering MDS, poor distance estimates are discarded from the localization calculation, and the overall positioning error can be further reduced.

The above observation is further verified with the following simulation with a larger range measurement error of 23%. The results are shown in Figure 6-4

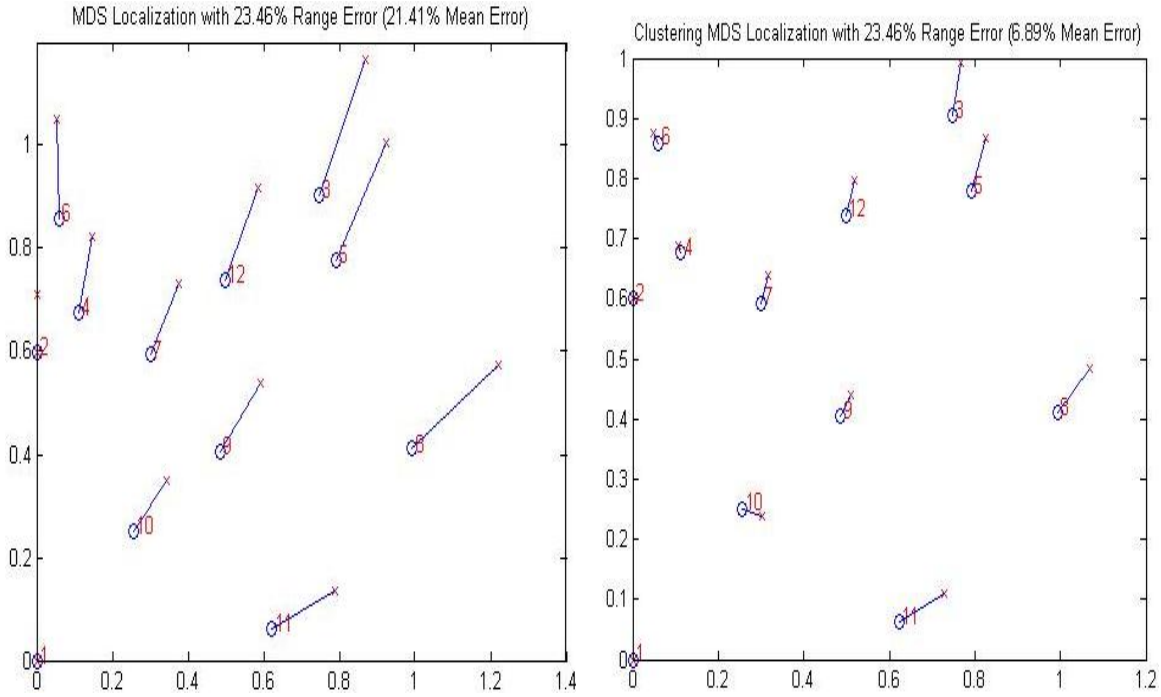

Figure 6-4: Classical vs. Clustering MDS Localization with 23% Range Error

It is easy to see that as the range error increases, the overall mean positioning error increases and approaches the maximum range error within the network. The same distance information is applied to the clustering MDS algorithm and its localization result is shown in the right side of Figure 6-4. The results for both algorithms demonstrates consistency in performance, where the classical MDS's positioning accuracy is always limited by the largest error in distance estimate; the clustering MDS's positioning accuracy tends to minimize the range error by always localizing nodes with less error relative to other nodes in the network.

6.3 Experimental Localization

6.3.1 **Indoor Localization**

In the previous section, a noisy distance model is defined in (19), based on observation from RSSI range measurement. The localization algorithms calculate the node positions from noisy distance information. From the localization results, it is clear to see that without the clustering MDS approach; the localization's accuracy is limited by the largest range measurement error of the dissimilarity. To support this claim further, a simple real-world scenario with 4 nodes is set up indoors with a node layout depicted in Figure 6-5

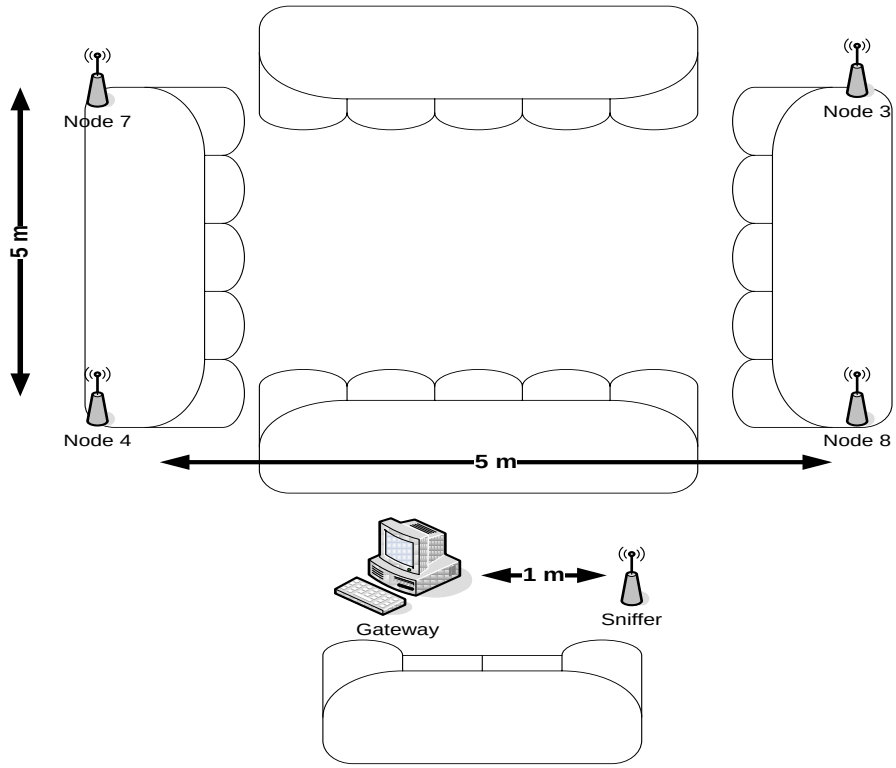

Figure 6-5: Indoor Localization Setup

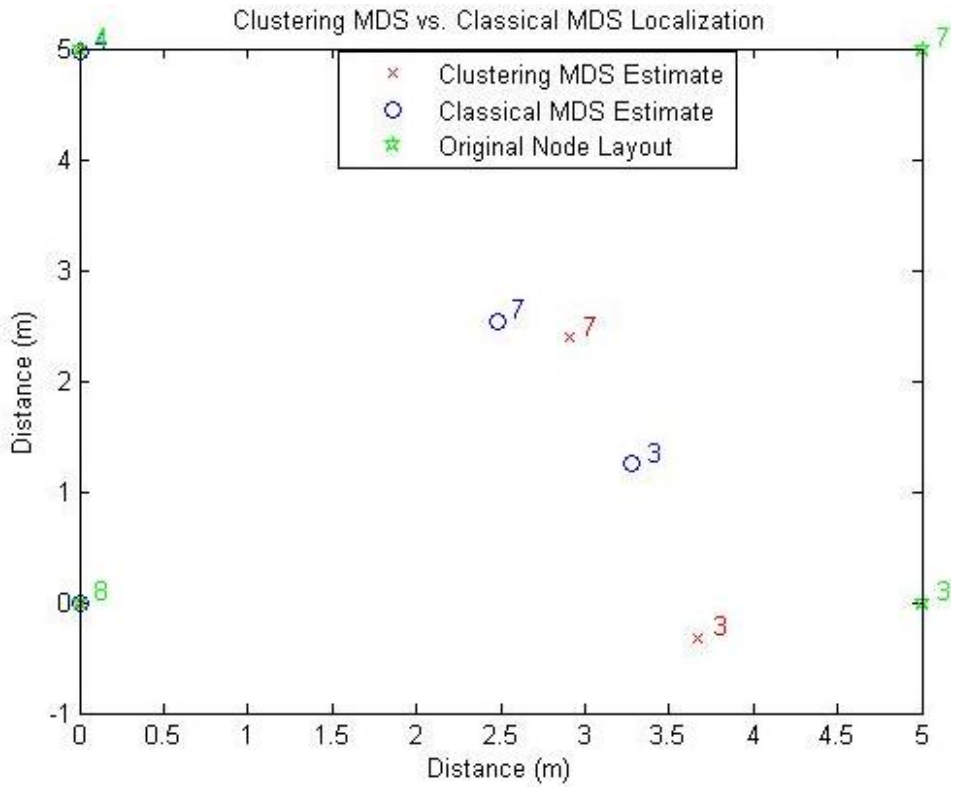

Figure 6-6: Localization Results from Clustering MDS and Classical MDS

Table 6-2: Positioning Error of Localization

Node	Classical MDS Estimated Position	Classical MDS Estimate Error (%)	Clustering MDS Estimated Positions	Clustering MDS Estimate Error (%)
Anchor Node #8	(0,0)	0	(0,0)	0
Anchor Node #4	(0, 4.97)	0.58	(0, 5)	0
Node #3	(3.28, 1.25)	42.5	(3.67, -0.32)	27.3
Node #7	(2.48, 2.54)	70.4	(2.91, 2.39)	66.9

The setup consists of 4 nodes, including 2 anchor nodes with known positions; the nodes are placed at four different corners of the couches, measuring 5 meter in length. Moreover and RMU is also present as part of the network to perform real-time RSSI to distance conversion. The result is shown in Figure 6-6 and the corresponding estimation error is listed in Table 6-2. Using the same set of dissimilarity collected, localization is further conducted and verified, using

different anchor nodes as the mapping reference. The result is shown in Figure 6-7 along with the position error in Table 6-3

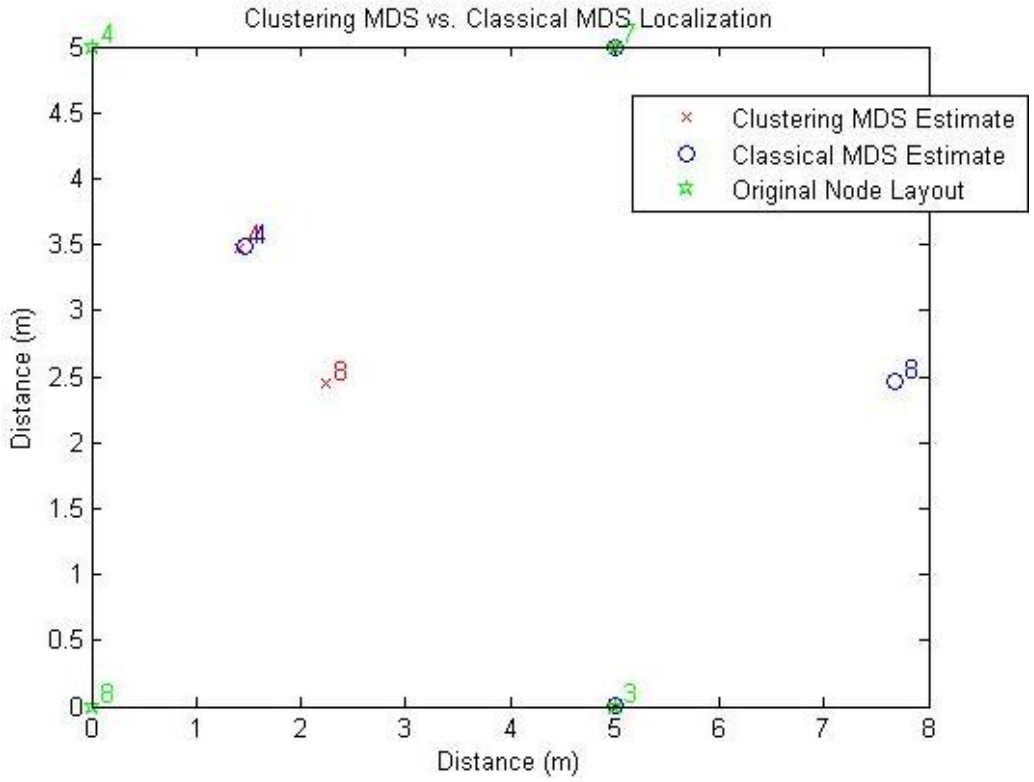

Figure 6-7: Localization Results from Clustering MDS and Classical MDS using Different Anchors

Table 6-3: Positioning Error for Different Anchor Nodes

Node	Classical MDS Estimated Position	Classical MDS Estimate Error (%)	Clustering MDS Estimated Positions	Clustering MDS Estimate Error (%)
Anchor Node #3	(5,0)	0	(5,0)	0
Anchor Node #7	(5, 5)	0	(5, 5)	0
Node #4	(1.46, 3.48)	42.1	(1.4, 3.47)	41.6
Node #8	(7.68, 2.46)	161.2	(2.24, 2.44)	66.3

From the above two scenarios, it can be seen that the results calculated with classical MDS method are very different under different network configuration. First, in Figure 6-6, applying the classical MDS algorithm with respect to anchors

Node 8 and 4, introduces a maximum positioning error of 70%. Because Node 7 has the worst estimation error of all, the overall localization is compensated for its error. On the other hand, the clustering MDS, although, is unable to improve the position estimate of Node 7 further, Node 3's location is calculated independently from the inter-node distances with Node 7 and 8, ignoring the more erroneous distance information of Node 4. The final positioning error for Node 3 is much less when localized with the clustering MDS algorithm than with the classical MDS algorithm. The layout in Figure 6-7 also demonstrates the superiority of clustering MDS localization and is consistent with the simulations shown previously. However, the clustering MDS localization cannot achieve high accuracy, because under random EM interference and reflections, it is extremely difficult to obtain stable RSSI values for reliable characterization with RMU.

From previous localization in indoor environment, it is obvious that the disadvantage of classical MDS localization is that it only provides one solution based on the dissimilarity provided. Therefore, when the dissimilarity contains range measurement error, the localization accuracy is always compromised by the largest error within the network. On the other hand, with the clustering MDS algorithm, nodes having larger error are often rejected from the position computation, and therefore resulting in a position estimate with better accuracy. In the next section, an outdoor distribution of SOSC network is deployed to further demonstrate clustering MDS algorithm's ability to reduce the localization error.

6.3.2 **Outdoor Localization**

Figure 6-8: Node Layout on the Field

In the previous section, due to limited indoor space for node distribution, only four nodes were placed and tested. This section, the SOSC system and its algorithm is further tested in a larger scale outdoor environment with total of seven nodes. There are a number of objectives to be achieved and verified; the objectives are

- To demonstrate the affect of range measurement error on localization accuracy
- To demonstrate the affect of network density on localization accuracy
- To compare the localization performance between classical MDS and clustering MDS algorithm

Figure 6-8 show a grass field at SFU, where the SOSC nodes were placed for the experiments. To achieve the objectives listed above, three layouts of different

network density, high, medium, and low density were used. In each density layout, inter-node distances are measured with a tape measure. Table 6-4 is an example of the inter-node distances from a high density node layout. In order to compare the localization performance at the end of each test, ground truth node positions have to be defined as reference. This ground truth is obtained from performing classical MDS algorithm on the tape measured dissimilarities.

Table 6-4: Tape Measured Dissimilarities of SOSC Nodes

Tape Measure Dissimilarities (m)							
Node ID	2	3	6	7	9	10	11
2	0	13.7	24	10.9	19.7	12.7	20.6
3	13.7	0	15.7	13.2	18.5	23.6	26.5
6	24	15.7	0	15	10.9	26.9	23.1
7	10.9	13.2	15	0	8.8	12.4	13.4
9	19.7	18.5	10.9	8.8	0	17.7	12.2
10	12.7	23.6	26.9	12.4	17.7	0	10.7
11	20.6	26.5	23.1	13.4	12.2	10.7	0

Given the transmission range of the node radio, the maximum inter-node distance was kept within 60m. The area coverage of the high, medium, and low density networks were designed to be 20mX25m, 50mX25m, and 60mX30m, respectively. To compute the ground truth coordinates, dissimilarities in Table 6-4 were computed with MDS algorithm. The network layouts are shown in Figure 6-9, Figure 6-10, and Figure 6-11

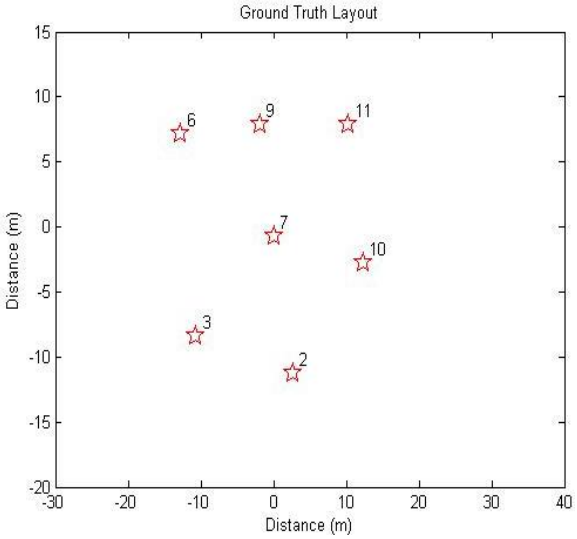

Figure 6-9: High Density Ground Truth Layout

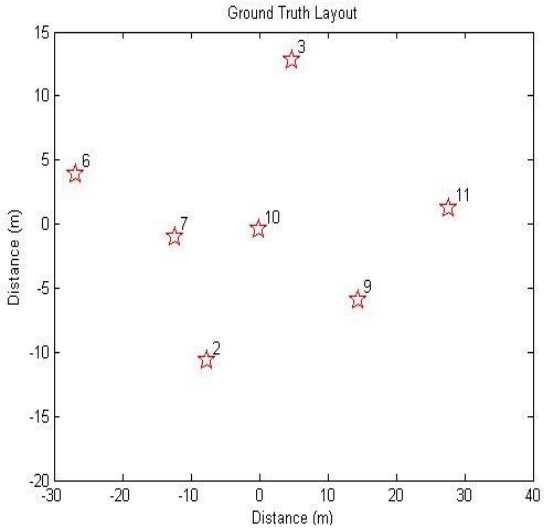

Figure 6-10: Medium Density Ground Truth Layout

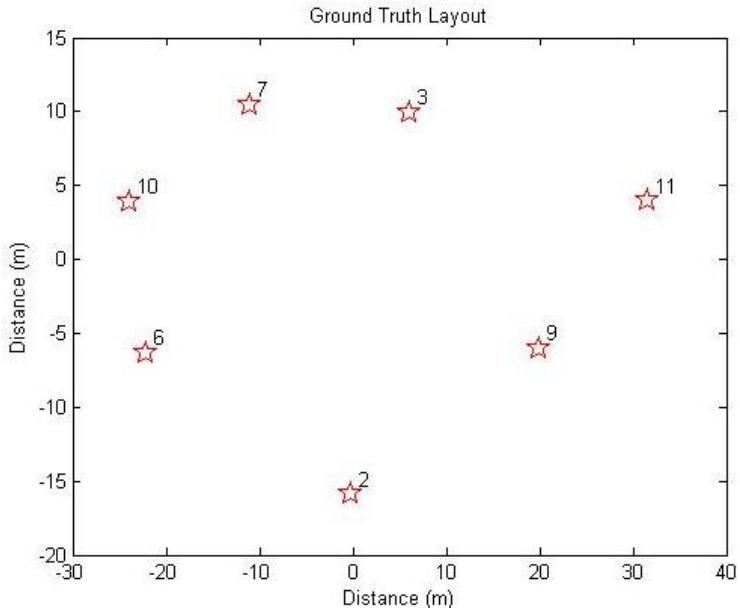

Figure 6-11: Low Density Ground Truth Layout

Once positioned, every node location is measured with a commercial handheld GPS device for comparison purpose. A GPS reports coordinates in terms of degrees of latitude and longitude. To represent the GPS coordinates in Cartesian coordinates, a standard WSG84 conversion, [54], is used. The resulting

Cartesian coordinates of the seven nodes, compared with the ground truth layout are shown in Figure 6-12, Figure 6-13, and Figure 6-14

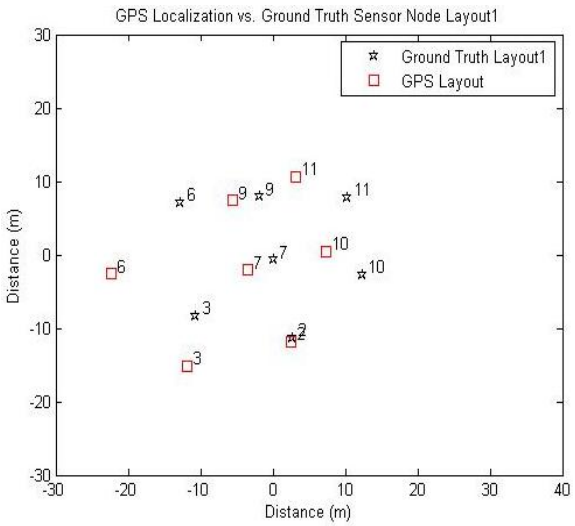

Figure 6-12: High Density GPS Localization Comparison

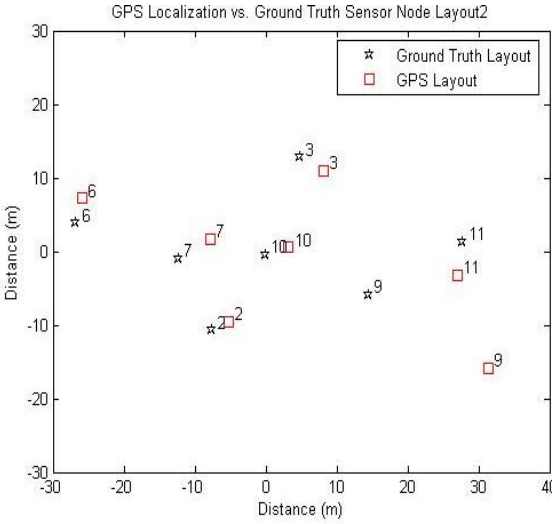

Figure 6-13: Medium Density GPS Localization Comparison

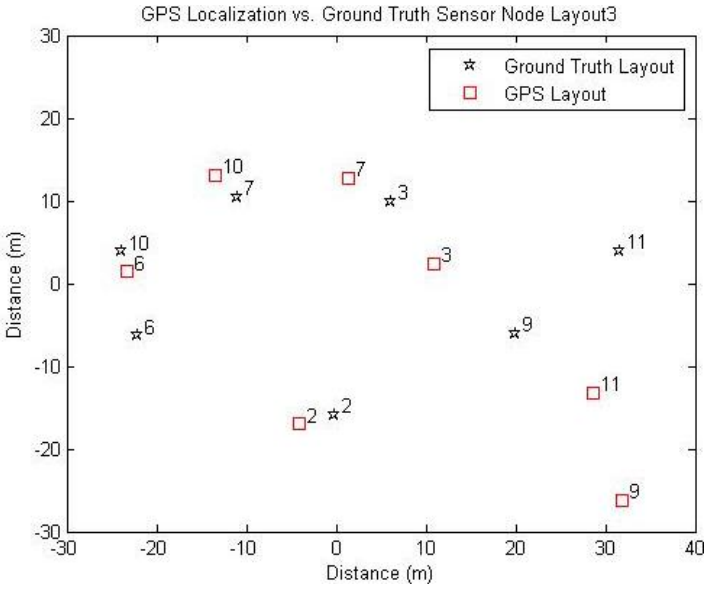

Figure 6-14: Low Density GPS Localization Comparison

It is important to note that the commercial GPS used for this test was not a differential GPS, and therefore it did not have the capability to correct its

measurement error due to the change in atmospheric conditions at the moment of measurement. This is one of the major factors causing the large discrepancy for node 6, 9, and 11 in the above three figures. Besides those localization errors, the overall GPS position estimation accuracy is listed in the table below

Table 6-5: Average GPS Localization Error

GPS Localization Error			
Density	**High**	**Medium**	**Low**
Average Distance Error (m)	6.08	6.1	12.63

It is obvious to see that even with state-of-art commercial GPS devices, it is still difficult to localize with high accuracy.

Having demonstrated the GPS localization performance, the experiment continued with RSSI-based localization with both classical and clustering MDS algorithms. The test utilized the RMU (see section 5.1) to convert real-time inter-node RSSI into corresponding inter-node distance. Each network layout is operated until a total of 1000 inter-node RSSI and distance have been received and converted. The RSSI-distance model is $RSSI = -(10 * (2.5)\log_{10} d + (32))$, where n is 2.5 and A is 32dBm. At each density, the RSSI measured at each inter-node link is compared with its corresponding ground truth distance. The resulting plots for each layout density are shown below

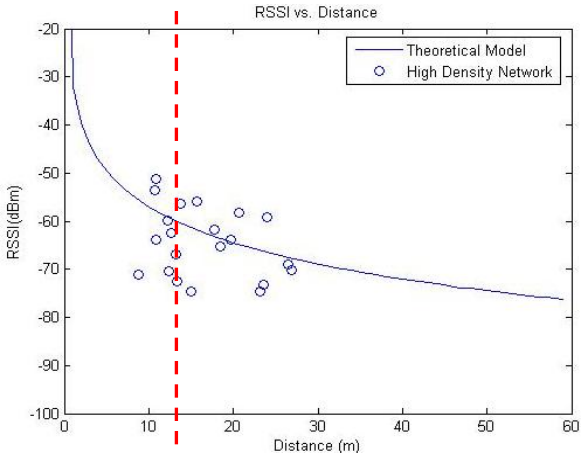

Figure 6-15: RSSI vs. Distance (High Density)

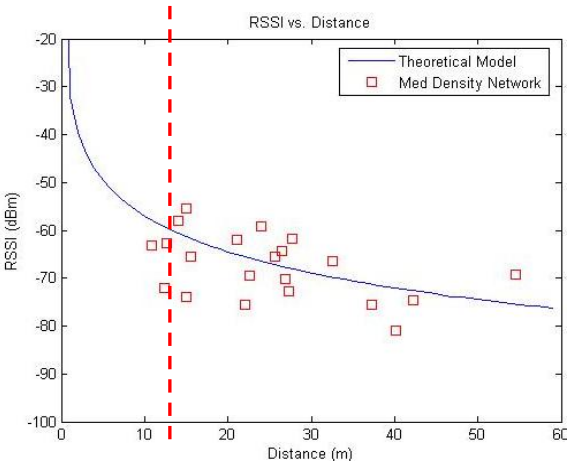

Figure 6-16: RSSI vs. Distance (Medium Density)

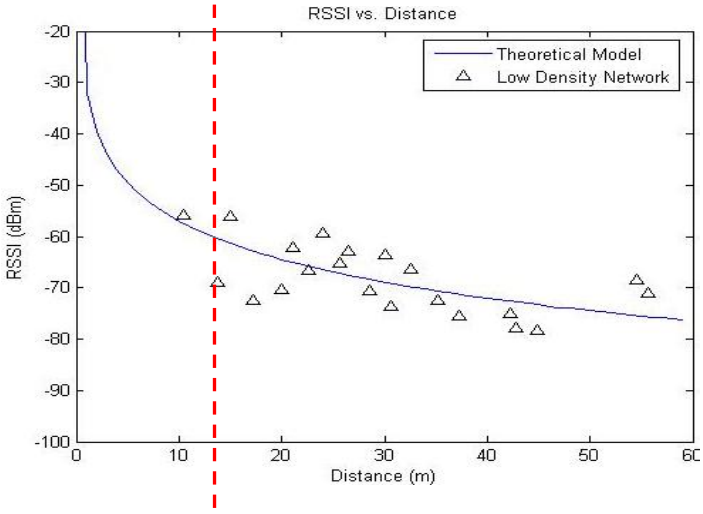

Figure 6-17: RSSI vs. Distance (Low Density)

From the above three figures, it is clear to see that RSSI values do not follow their theoretical values at each distance. The RSSI values in the high density case, due to noise and RF interference, spans 10's of dBm at the distance marked by the dotted line. As the networks became less dense, the inter-node RSSIs became more distinct at each distance and started to merge toward the RSSI-distance model more closely.

Using the above dissimilarities, the localization results computed by classical MDS and clustering MDS at each layout density are shown below

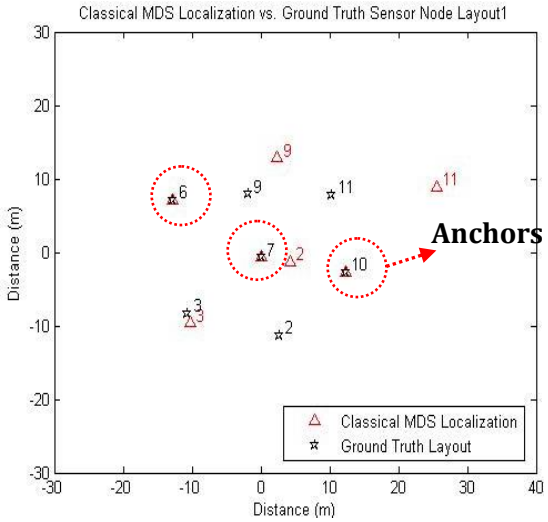

Figure 6-18: High Density Classical MDS Result

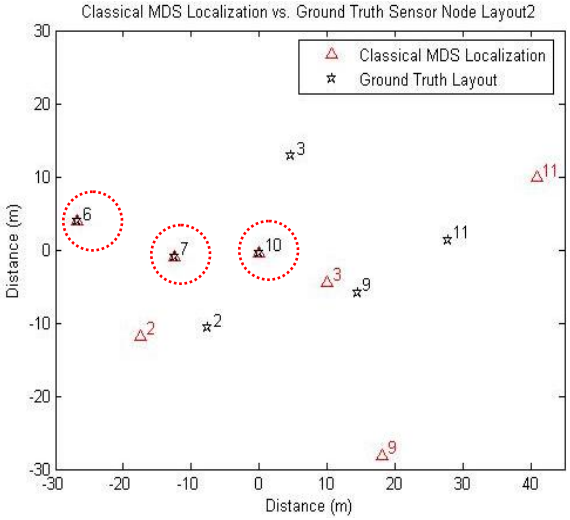

Figure 6-19: Medium Density Classical MDS Result

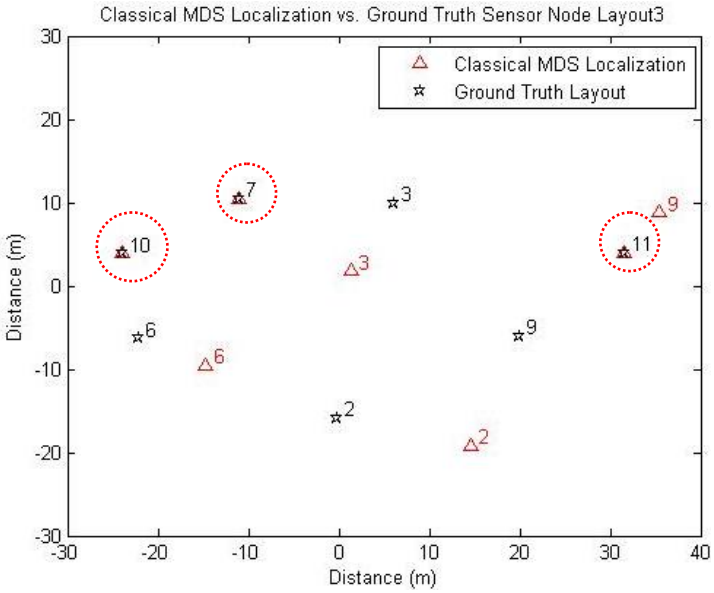

Figure 6-20: Low Density Classical MDS Result

Placing the nodes and obtaining ground-truth data via tape measure is very time consuming. Therefore every possible anchor configuration for each density was evaluated. This is done by choosing three anchor nodes out of the seven nodes, and it gives us $C_3^7 = 35$ possible network configurations for each density, making a total of 3x35 = 105 test configurations. The figures shown above are the best localization result from the 35 separate tests of each density. The table below is a summary of the average localization accuracy in each network density.

Table 6-6: Classical MDS Localization Accuracy

Classical MDS Localization Error			
Density	**High Density**	**Med Density**	**Low Density**
Average Error (m)	15.51	28.12	21.43
Min Error (m)	8.29	16.56	13.56
Max Error (m)	20.02	37.37	31.67
STD (m)	2.64	5.57	4.2

Notice that the localization error is larger in the medium density layout than in the low density layout. This is because the non-monotonic degrading behavior of RSSI with distance in the medium density, Figure 6-16, did not follow the theoretical model as closely as that in the low density scenario in Figure 6-17. This example demonstrates the fact that although, node density plays an important role in localization accuracy, localization accuracy is greatly dependent on range measurement accuracy.

Having had explored the localization performance of the classical MDS algorithm, the localization performance of the clustering MDS algorithm is tested and analyzed, following the same procedures described previously. The results are shown in Figure 6-21, Figure 6-22, and Figure 6-23

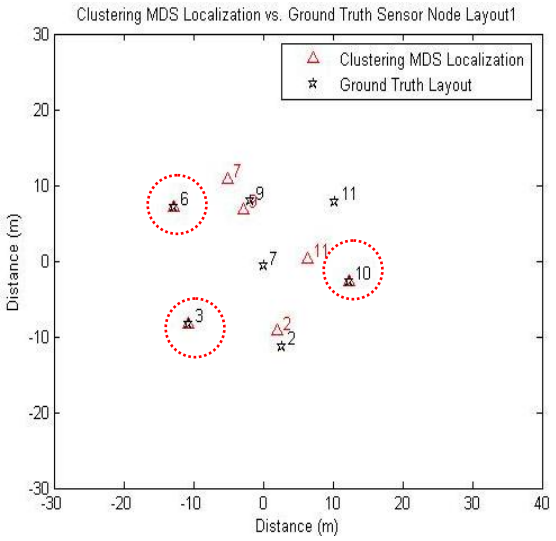

Figure 6-21: High Density Clustering MDS Result

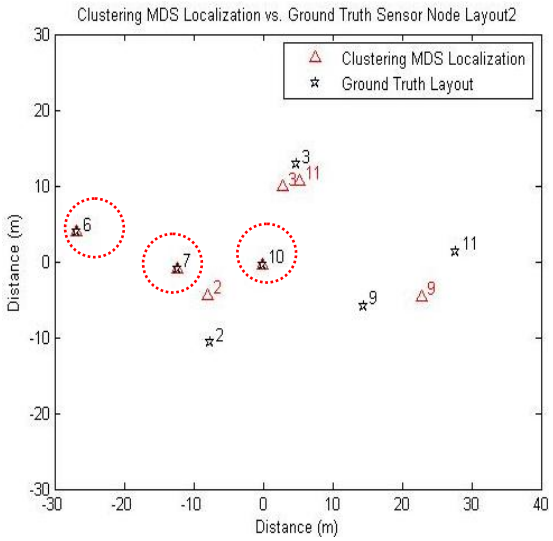

Figure 6-22: Med Density Clustering MDS Result

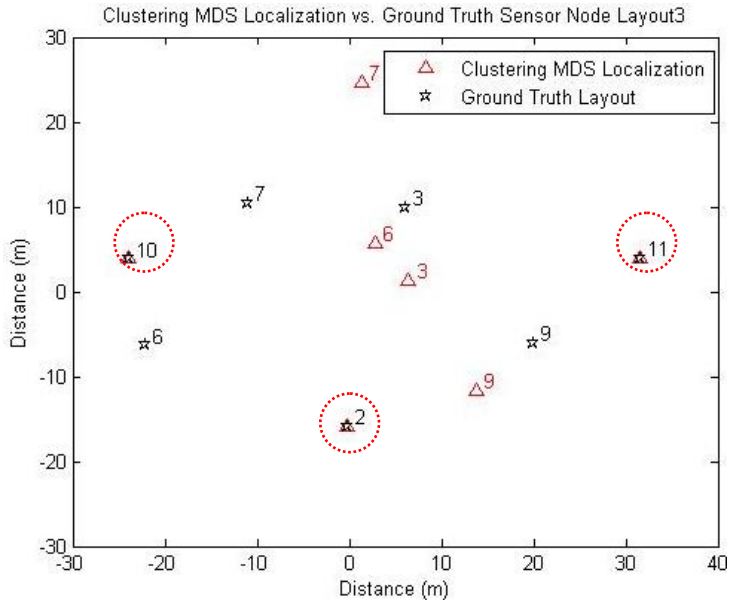

Figure 6-23: Low Density Clustering MDS Result

Table 6-7: Clustering MDS Localization Accuracy

Clustering MDS Localization Error			
Density	**High Density**	**Med Density**	**Low Density**
Average Error (m)	13.19	21.68	23.71
Min Error (m)	6.17	10.66	15.91
Max Error (m)	21.56	34.85	36.56
STD (m)	3.65	5.12	5.02

From the above three network densities, a total of 105 localization tests have been generated. To summarize the overall localization performance with respect to the node density, the average of the localization results under the same level of node density is calculate for every test case. The final localization error versus node density comparison is shown below

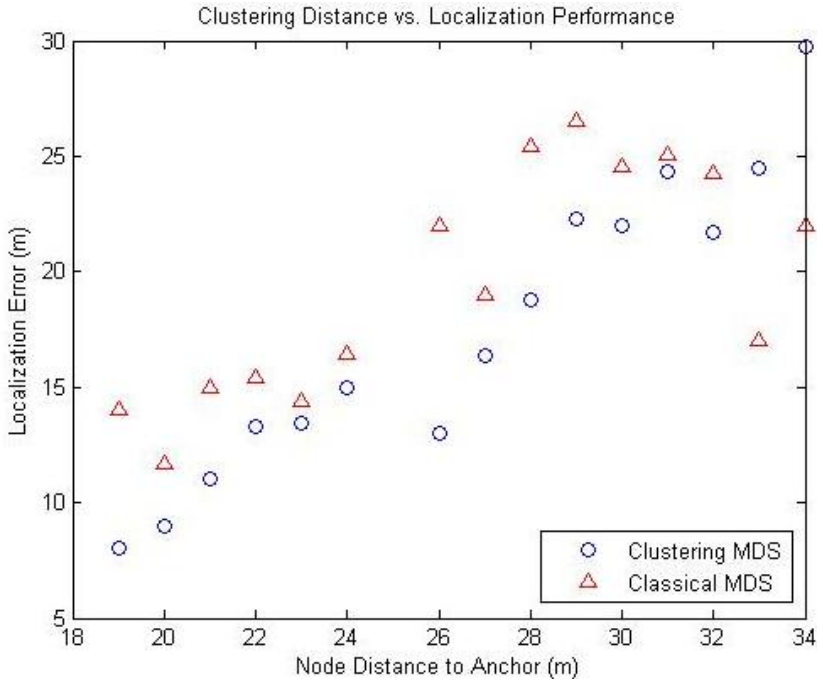

Figure 6-24: Localization Accuracy vs. Node Density

In general, Figure 6-24 shows that the localization accuracy is directly proportional to the network density and that clustering MDS exhibits an

improvement on the accuracy compared to the classical MDS. However, the clustering MDS algorithm, in situations where certain node pair's RSSIs are constructively interfered and appear to be closer to each other, the estimated location of that particular node pair can result in a large discrepancy, like node 6 in Figure 6-23. The nature of clustering MDS allows improvements on localization result in fading situations, like in the high and med-density scenario; it does not perform better than the classical MDS algorithm when RSSIs are constructively interfered in a lower density network, as illustrated in the lower density portion in Figure 6-24.

It is a well known fact that radio wave's signal strength is easily degraded by EM interference. As observed from the experiments, RSSIs can be either destructively or constructively interfered to produce values that do not correspond to their physical distance. This behavior has been shown in Figure 6-15, Figure 6-16, and Figure 6-17 and been shown to result in range measurement errors. To investigate the influence of range measurement errors on localization performance, an average range error is calculated from difference between the RSSI converted dissimilarity with the ground-truth dissimilarity. The localization error is then plotted against its corresponding range measurement error. The results from both the classical and clustering MDS are shown below

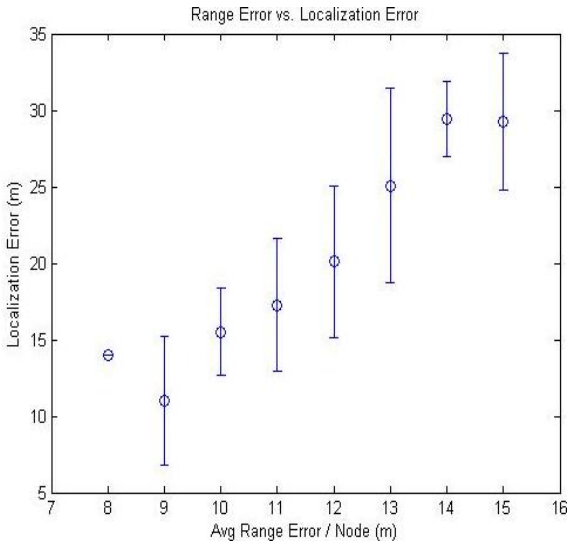

Figure 6-25: Localization Error vs. Range Error (Classical MDS)

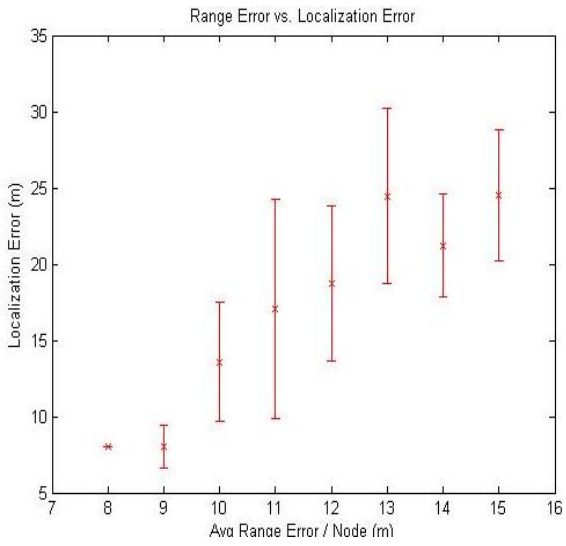

Figure 6-26: Localization Error vs. Range Error (Clustering MDS)

Being largely dependent on range measurement to perform localization, it is clear to see that in Figure 6-25 and Figure 6-26, the localization error distribution, represented by the error bars, of the two algorithms increases proportional with range error. However, due to the clustering MDS's selective nature on choosing distance information to localize nodes, the overall localization accuracy is less susceptible to erroneous distance information than the classical MDS algorithm. From the previous observations, the clustering MDS algorithm has an advantage over the classical MDS algorithm in medium and high density networks consisting of erroneous distance information. However, to justify one algorithm's superiority in localization performance over the other; it is important to further show the uniqueness by investigating the statistical distribution of the localization accuracy from the two algorithms.

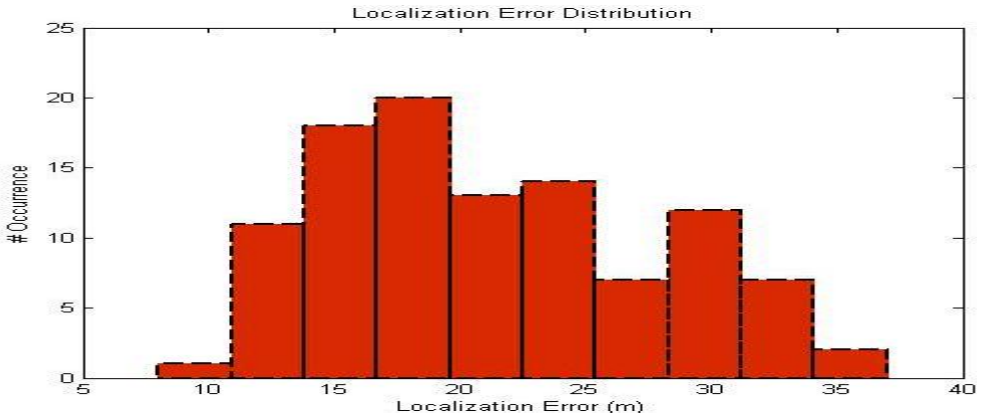

Figure 6-27: Classical MDS Localization Error Distribution

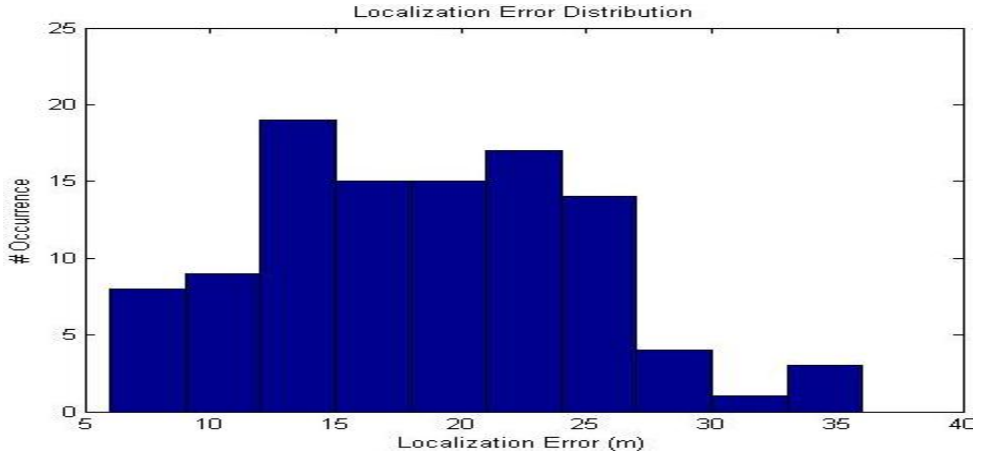

Figure 6-28: Clustering MDS Localization Error Distribution

From the 105 localization tests, the localization error of each algorithm is summarized in the two histograms, Figure 6-27 and Figure 6-28 respectively. To determine the similarity between the two statistical distributions before concluding on the uniqueness of clustering MDS algorithm, a two-sample t-test is conducted for each network density and for the overall case. The significance of each case is shown in the table below

Table 6-8: Two-sample T-Test of Statistical Significance

T-Test	
	Significance
High Density	3.30E-03
Medium Density	3.61E-06
Low Density	0.044
Overall	1.83E-02

By definition, statistical similarity between two distributions is indicated by the null hypothesis, which states that the two distributions are of the same statistical mean, can be rejected if the significance is below 5%. From Table 6-8, it is clear to see that in every scenario, the clustering MDS algorithm is indeed a different algorithm from the classical MDS.

6.3.2.1 Network Connectivity vs. Localization Error

In previous test cases, perfect connectivity of inter-node communication was assumed. In fact, in medium and low density network cases, there existed pairs of nodes that were not able to communicate with each other and therefore no RSSI values for these pairs were recorded. When one node fails to receive RSSI from the transmitting node, the dissimilarity matrix can result in an empty entry for that particular node pair, and therefore leaving the dissimilarity matrix unsymmetrical. Classical MDS algorithm cannot solve an unsymmetrical matrix by default. A proper scheme to handle the disconnected graph in a real network is expected, otherwise it is impossible to localize using the classical MDS algorithm.

Unlike classical MDS, which requires each individual inter-node distance to performing localization, the clustering MDS algorithm only requires the distance information of the closest neighbor to the anchors that are performing localization at the time instant. As a result, the algorithm ignores the disconnected link and utilizes other available links to localize. In this section, an example of disconnected link from the medium density case is re-created to demonstrate the localization performance of the clustering MDS algorithm over the classical MDS algorithm.

The following table is the dissimilarity collected from nodes in the medium density layout. The high-lighted node pair had a poor reception and was unable to produce a symmetrical matrix.

Table 6-9: Unsymmetrical Dissimilarity

Medium Density Dissimilarities (m)							
Node ID	2	3	6	7	9	10	11
2	0	19.46	12.24	17.59	31.79	16.87	55.51
3	19.46	0	23.66	55.36	15.92	10.87	22.01
6	12.24	23.66	0	47.52	50.33	33.84	30.76
7	17.59	55.36	47.52	0	42.75	40.23	0
9	31.79	15.92	50.33	42.75	0	21.80	8.60
10	16.87	10.87	33.84	40.23	21.80	0	15.57
11	55.51	22.01	30.76	91.63	8.60	15.57	0

In this case, a simple algorithm can be implemented to regenerate a symmetrical matrix by determining one of the missing entries and filling it with the distance information received by the other node. However, in a more extreme case, it is possible that this node pair cannot receive distance information in either

direction. To simulate this scenario, the high-lighted cells in Table 6-9 are both filled with 0 to indicate a complete broken link. This dissimilarity is then localized by both the classical and clustering MDS algorithms. The same test to produce a comparison between average node densities and localization error as shown in Figure 6-24. The test was conducted for both the fully connected case and the disconnected case; the results are shown as follows

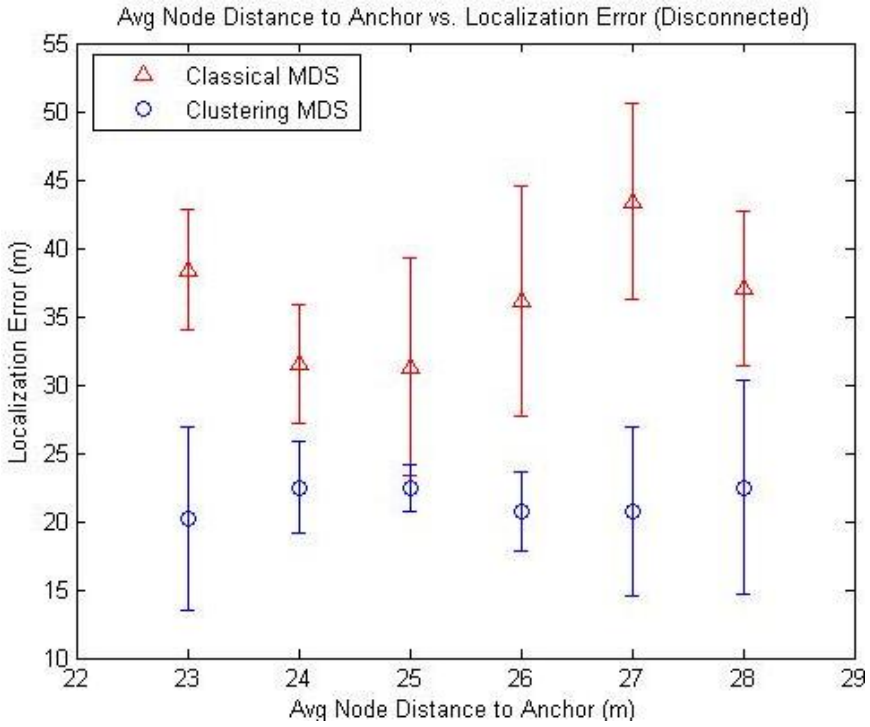

Figure 6-29: Localization Accuracy in Disconnected Network

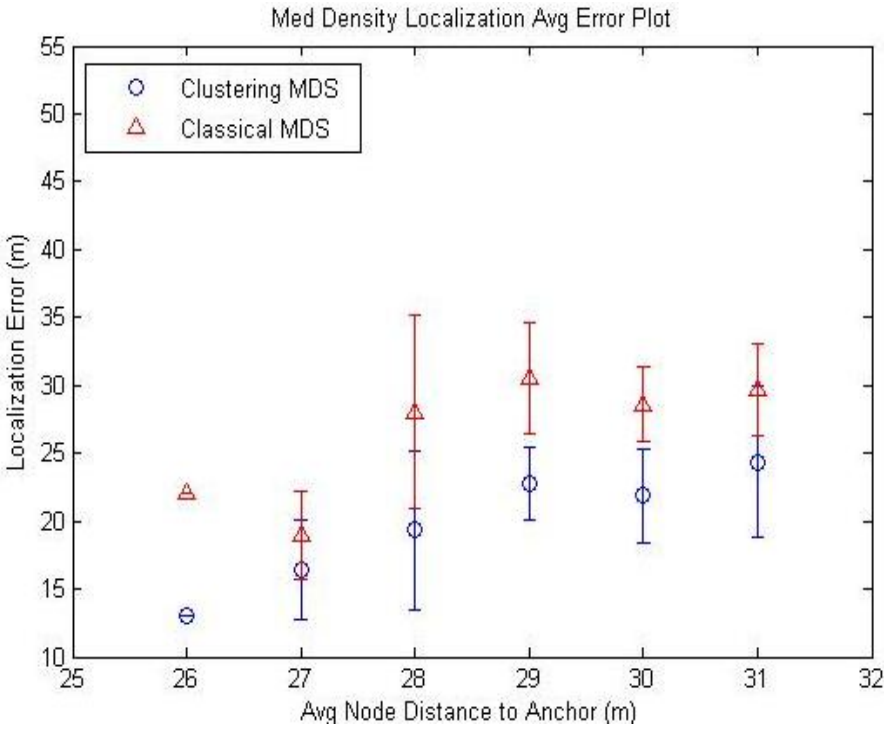

Figure 6-30: Localization Accuracy in Fully Connected Network

By removing one entry from the dissimilarity matrix, the average localization error in the disconnected scenario in Figure 6-29, has worsened by almost 15 meters on average. On the other hand, clustering MDS demonstrated only a slight degradation in performance. In this particular test, the WSN only consists of one broken link; in a larger scale and less dense network, the number of disconnected links can increase, causing the classical MDS algorithm to fully localize the network. By conducting this test, it can be verified that the clustering MDS is capable of withstanding the influence of missing inter-node distance information, sustaining the same level of accuracy as performing localization on a fully connected network.

6.4 Chapter Summary

In this chapter, based on computation complexity, simulation, and experimental results, we have outlined the advantages of localizing with the novel clustering MDS algorithm over the classical MDS algorithm. Conducting the tests with three different network density layouts, we were able to produce 105 individual network configurations. The 105 tests consist of different levels of node density and ranging errors. We summarized the results from these test results and analyze localization performance between the classical MDS approach and the clustering MDS approach against node density, ranging error, and connectivity of the network. The clustering MDS localization is shown to be capable of producing comparable and more resilient localization accuracy under noisy condition and poor network connectivity. Moreover, clustering MDS also demonstrates a much lower computation complexity, which ultimately makes this algorithm truly suitable for low-power and low-cost WSN localization applications.

Chapter 7 - Conclusion

In this thesis, we acknowledged the major difficulties in assuring performance in localization systems to be the hardship in defining the ground truth reference and the range measurement error. We have demonstrated the former by comparing the localization results between the GPS measured node positions and the MDS calculated node positions based on tape-measured distances. It is obvious that even with the commercial GPS device; the localization accuracy can still introduce discrepancies. The second difficulty is the most common in all range-based localization systems. Many previous works, discussed earlier, cope with the ranging error by off-line training or by developing complex models to minimize error in range measurement. It is also common to equip sensors with additional ranging devices to reduce the biasing effect, induced by using only one ranging device. Most of the solutions from related works are costly and complex to implement. As a result, in this thesis, we developed a low-cost and low-power SOSC system for ZigBee WSN to address the above problems and to also deliver comparable accuracy and much lower computation complexity than the classical MDS localization algorithm.

Having said ranging error is a major cause to degradation of localization accuracy, we introduced a real-time RMU to facilitate the distance conversion process. As opposed to the empirical approach, this RMU is adaptive to the ever-changing channel parameters in real-time and does not require training for the system. From experiments conducted in a small indoor environment, we were able to show that the RMU estimates the true distance better than the empirical

approach. However, because RMU converts RSSI to distance only based on three anchor nodes and one sniffer in an attempt to characterize the whole network. In reality, due to uneven noise distribution in space, it is challenging to generalize conversion models. However, to provide a more accurate conversion model to represent a large network, more than one RMU can be deployed to characterize individual region in the network.

Although, using only RSSI ranging for localization introduces limitation on producing highly accurate node position estimate, we proposed the clustering MDS along with the ICP algorithm to minimize mapping error. We also demonstrated the clustering MDS localization algorithm's ability to cope with the influence of ranging error and imperfect network connectivity; the clustering MDS is able to produce comparable localization results with much less computation complexity. Being linear to the scale of the network, the computation complexity of the clustering MDS allows the SOSC WSN to be cost-effective and attractive to large network applications.

7.1 Future Work

Future works to further improve this SOSC WSN include implementing the clustering MDS localization algorithm in firmware (in a distributed way) on low-cost and low-power microcontrollers; second, a more refined ranging technique and accurate distance conversion modeling are required to enhance the localization accuracy.

First, the clustering MDS algorithm is a distributed algorithm by nature; however, the focus of this thesis is on analyzing its improvements on localization

accuracy and computation complexity over the classical MDS localization. Therefore, the algorithm is implemented in a centralized manner and the benefit of reduced network overhead, delivered by using distributed algorithms in WSN is not discussed. For future work, this localization algorithm can be modified and implemented on the CC2530 SOC with an improved arithmetic and further reduced complexity to maintain low-power operation.

Second, as observed previously, RSSI ranging introduces large error, degrading the overall localization performance. To improve the current state, more distance information encapsulated by RF signals can be extracted to ensure the reliability of the range measurement without introducing additional ranging devices. For example, in most ad-hoc networks, hop-count is a popular feature and can be embedded in RF signals in inter-node communications. Hop counts are essentially a representation of communication cost; communication cost can be influenced by interference or natural signal degradation due to distance, but it is less sensitive than RSSI. Hop-count can be utilized in conjunction with RSSI for ranging purposes. By having this additional information, correctness of the range data can be determined more reliably, and the clustering MDS localization algorithm can be less biased by always localizing nodes with strongest RSSI.

References

[1] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *Personal Communications, IEEE*, vol. 7, no. 5, pp. 28-34, Oct 2000.

[2] Trevor F. Cox and Michael A.A. Cox, "Metric Multidimensional Scaling," in *Multidimensional Scaling*. USA: Chapman & Hall/CRC, 2001, ch. 2, pp. 32-44.

[3] A. Boukerche, H.A.B. Oliveira, E.F. Nakamura, and A.A.F. Loureiro, "Localization systems for wireless sensor networks," *Wireless Communications, IEEE* , vol. 14, no. 6, pp. 6-12, December 2007.

[4] Wikipedia. (2011, July) Radar - Wikipedia, the free encyclopedia. [Online]. http://en.wikipedia.org/wiki/Radar

[5] K. Whitehouse, C. Karlof, and D. Ceuller, "A practical evaluation of radio signal strength for ranging-based localization," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 1, January 2007.

[6] A. Awad, T. Frunzke, and F. Dressler, "Adaptive Distance Estimation and Localization," in *Digital System Design Architectures, Methods and Tools, 2007. DSD 2007. 10th Euromicro Conference*, Lubeck, 2007, pp. 471-478.

[7] HowStuffWorks. (2011, July) How Wireless Mesh Networks Work. [Online]. http://communication.howstuffworks.com/how-wireless-mesh-networks-work.htm

[8] ZigBee Alliance. (2011, June) ZigBee Specifications. [Online]. http://www.zigbee.org/Specifications.aspx

[9] P. Besl and N. McKay, "A method for Registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 14, no. 2, pp. 239 - 256, February 1992.

[10] K. Aamodt. (2010, Dec.) Texas Instruments. [Online]. http://focus.ti.com/lit/an/swra095/swra095.pdf

[11] J. Xu, M. Ma, and C.L. Law, "Position Estimation Using UWB TDOA Measurements," in *Ultra-Wideband, The 2006 IEEE 2006 International Conference on*, Waltham, MA, 2006, pp. 605-610.

[12] (2011, June) antenna-theory.com. [Online]. http://www.antenna-theory.com/basics/friis.php

[13] Guoqiang Mao, Barış Fidan, and Brian D. O. Anderson, "Wireless sensor network localization techniques, Computer Networks," *The International Journal of Computer and Telecommunications Networking*, vol. 51, no. 10, pp. 2529-2553, July 2007.

[14] M. Bal, M. Liu, W. Shen, and H. Ghenniwa, "Localization In Cooperative Wireless Sensor Network: A Review," in *International Conference on Computer Supported Cooperative Work in Design*, Santiago, Chile, 2009.

[15] R. Mautz, "Overview of Current Indoor Positioning Systems," in *Geodezija*

Ir Kartografija / Geodesy and Cartography, 2009 35(1), pp. 18-22.

[16] D. Niculescu and Badri Nath, "Ad hoc positioning system (APS) using AOA," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, 30 March-3 April 2003 , pp. 1734 - 1743 vol.3.

[17] K.W. Cheung, H.C. So, W.K. Ma, and Y.T. Chan, "Received signal strength based mobile positioning via constrained weighted least squares," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference*, 2003, pp. 137-140.

[18] Y. Kwon, K. Mechitov, S. Sundresh, W. Kim, and G. Agha, "Resilient localization for sensor networks in outdoor environments," Department of Computer Science, University of Illinois at Urbana Champaign, Technical Report UIUCDCS-R-2004-2449 March 2004.

[19] C.H. Wu, W. Sheng, and Y. Zhang, "Mobile Self-Localization using Multi-Dimensional Scaling in Robotic Sensor Networks," in *IEEE International Conference on Robotics and Automation*, 2007.

[20] Y. Shang and W. Ruml, "Improved MDS-Based Localization," in *Proc. IEEE Infocom*, 2004.

[21] Gopakumar Aloor and Lillykutty Jacob, "Distributed wireless sensor network localization using stochastic proximity embedding," *Computer Communications*, vol. 33, no. 6, pp. 745-755, April 2010.

[22] S. Capkun and J.P. Hubaux, "Secure positioning of wireless devices with application to sensor networks," in *In IEEE INFOCOM*, 2005.

[23] Lingfei Wu et al., "A practical evaluation of radio signal strength for mobile robot localization," in *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference*, Guilin, 19-23 Dec. 2009, pp. 516-522.

[24] P. Bahl and V.N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* , vol. 2, pp. 775-784, August 2002.

[25] M. Bertinato et al., "RF localization and tracking of mobile nodes in wireless sensors networks:Architectures, algorithms and experiments," University of Padova, Technical report 2008.

[26] Q. Chen, D.L. Lee, and W.C. Lee, "Rule-based WiFi localization methods," in *IEEE/IFIP international conference on embedded and ubiquitous computing*, 2008.

[27] S. Ray, W. Lai, and I.Ch. Paschalidis, "Deployment optimization of sensornet-based stochastic location-detection systems," *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE* , vol. 4, pp. 13-17, August 2005.

[28] Fuxiang Gong, Wang Qing, and Xiaoguo Zhang, "A new distance based algorithm for TDOA localization in cellular networks," in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International*

Conference, Chengdu, 2010, pp. 502-505.

[29] Ran Yu, Bin Xu, Guodong Sun, and Zheng Yang, "Whistle: synchronization-free TDOA for localization," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, 2010.

[30] Xing Wei, Ling Wang, and Jianwei Wan, "A New Localization Technique Based on Network TDOA Information," in *ITS Telecommunications Proceedings, 2006 6th International Conference*, Chengdu, 2007, pp. 127-130.

[31] Nissanka Bodhi Priyantha and Hari Balakrishnan, "The cricket indoor location system," Massachusetts Institute of Technology, Cambridge, MA, 2005.

[32] N.B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller, "Anchor-Free Distributed Localization in Sensor Networks," MIT Laboratory for Computer Science, Boston, Technical Report 2003.

[33] Rong Peng and Mihail L. Sichitiu, "Angle of Arrival Localization for Wireless Sensor Networks," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society*, Reston, VA, 2007, pp. 374 - 382.

[34] ChulYoung Park et al., "AoA Localization System Design and Implementation Based on Zigbee for Applying Greenhouse," in *Embedded and Multimedia Computing (EMC), 2010 5th International Conference*, Cebu, 2010, pp. 1 - 4.

[35] Jose A. Costa, Neal Patwari, and III, Alfred O. Hero, "Distributed weighted-multidimensional scaling for node localization," *ACM Transactions on Sensor Networks (TOSN)*, vol. 2, no. 1, February 2006.

[36] Yi Shang, Wheeler Ruml, Ying Zhang, and Markus Fromherz, "Localization from Connectivity in Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 11, pp. 961-974, November 2004.

[37] Jaehun Lee, WooYong Chung, and Euntai Kim, "Efficient range-free localization algorithm for wireless sensor network based on shortest path information," in *ICCAS-SICE, 2009*, Fukuoka, 2009, pp. 1962-1965.

[38] Dragoş Niculescu and Badri Nath, "DV Based Positioning in Ad Hoc Networks," in *Telecommunication Systems*, 2003, pp. 267-280.

[39] Hongyang Chen, Kaoru Sezaki, Ping Deng, and Hing Cheung So, "An improved DV-Hop localization algorithm for wireless sensor networks," in *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference*, 3-5 June, 2008, pp. 1557-1561.

[40] T. He, C. Huang, B.M. Blum, J.A. Stankovic, and T.F. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *International Conference on Mobile Computing and Networking*, San Diego, CA, USA, 2003, pp. 81-95.

[41] Hyuk Lim and J.C. Hou, "Localization for anisotropic sensor networks," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and*

Communications Societies. Proceedings IEEE, 2005, pp. 138-149.

[42] S. Tian, X.M. Zhang, P.X. Liu, P. Sun, and X.G. Wang, "A RSSI-based DV-hop algorithm for wireless sensor networks," in *Proc. of IEEE WICOM2007*, Sep 2007.

[43] Bang Wang, Hock Beng Lim, Di Ma, and Cheng Fu, "The hop count shift problem and its impacts on protocol design in wireless ad hoc networks," *Telecommunication Systems*, vol. 44, pp. 49-60, November 2009.

[44] Benny S. L. Hung, "Wireless Mesh Network System Design and Implementation For Surveillance, Health Monitoring and Tracking," Simon Fraser University, Burnaby, Masters Thesis 2010.

[45] Antenova. (2011, June) Antenova - Integrated Antenna Solutions. [Online]. http://pdf.eicom.ru/datasheets/antenova/titanis_24_2010byyyy/titanis_24_2010byyyy.pdf

[46] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proc. SenSys.*, 2004, pp. 50-61.

[47] S. Basagni, "Distributed Clustering Algorithm for Ad-Hoc Networks," in *Proc. Int'l Symp. Parallel Architectures, Algorithms, and Networks (I-SPAN)*, 1999.

[48] R. Krishnan and D. Starobinski, "Efficient clustering algorithms for self-organizing wireless sensor networks," *J. Ad Hoc Networks*, vol. 4, p. 36, 2006.

[49] Y. Zhou and L. Lamont, "An Optimal Local Map Registration Technique for Wireless Sensor Network Localization Problems," in *Fusion*, 2008.

[50] H. Chan, M. Luk, and A. Perrig, "Using clustering information for sensor network localization," in *Proc. IEEE Conf. Distrib. Comput. Sensor Syst.*, 2005.

[51] O.H. Kwon and H.J. Song, "Localization through map stitching in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 1, pp. 93-105, January 2008.

[52] R. Nagpal, "Organizing a global coordinate system from local information on an amorphous computer," MIT Department of Electrical Engineering and Computer Science, Boston, Technical Report 1999.

[53] H. Wymeersch, J. Lien, and M.Z. Win, "Cooperative localization in wireless networks," in *Proc. IEEE (Special Issue on UWB Technology and Emerging Applications)*, 2009.

[54] Wikipedia. (2011, July) World Geodetic System - Wikipedia, the free encyclopedia. [Online]. http://en.wikipedia.org/wiki/World_Geodetic_System