MOBILE ASSISTED RANGE-BASED LOCALIZATION IN MULTI-STOREY BUILDINGS

ALİ TEOMAN UNAY

JANUARY 2015

MOBILE ASSISTED RANGE-BASED LOCALIZATION IN MULTI-STOREY BUILDINGS

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF

NATURAL AND APPLIED SCIENCES OF

IZMIR UNIVERSITY OF ECONOMICS

BY

ALİ TEOMAN UNAY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

IN THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

JANUARY 2015

M.S. THESIS EXAMINATION RESULT FORM

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Cüneyt Güzeliş
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. E. Turhan Tunalı
Head of Department

We have read the thesis entitled **"MOBILE ASSISTED RANGE-BASED LOCALIZATION IN MULTI-STOREY BUILDINGS"** completed by **ALİ TEOMAN UNAY** under supervision of **Assoc. Prof. Dr. HÜSEYİN AKCAN** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Cem EVRENDİLEK
Co-Supervisor

Assoc. Prof. Dr. HÜSEYİN AKCAN
Supervisor

Examining Committee Members

Date: 23.01.2015

Assoc. Prof. Dr. HÜSEYİN AKCAN
Dept. of Software Engineering, İUE

Assoc. Prof. Dr. Cem Evrendilek
Dept. of Computer Engineering, İUE

Prof. Dr. Brahim Hnich
Dept. of Computer Engineering, İUE

Asst. Prof. Dr. Ufuk Celikkan
Dept. of Software Engineering, İUE

Asst. Prof. Dr. Burkay Genc
Instutitute of Population Studies, Hacettepe University

ABSTRACT

MOBILE ASSISTED RANGE-BASED LOCALIZATION IN MULTI-STOREY BUILDINGS

ALİ TEOMAN UNAY
M.S.in Intelligent Computing Systems.
Graduate School of Natural and Applied Sciences
Supervisor: Assoc. Prof. Dr. HÜSEYİN AKCAN
January 2015

Localization is used as a service required in many other application areas ranging from health care to vehicular networks, habitat monitoring, and to search-and-rescue missions. Location information in these applications are used for various purposes such as classification of data gathered, construction of movement plans or evacuation plans, etc. Although global positioning systems (GPS) are a general choice for localization, these systems do not work in indoors. Yet, indoor environments pose many challenges to localization algorithms such as quadrilateration. In this work, we analyze the weaknesses of quadrilateration for multi-storey indoor localization scenarios, and propose novel methods to improve the quadrilateration algorithm with the help from a mobile node, if necessary. We also aim to minimize the mobile node's travel time while maximizing the number of nodes localized. In this thesis, we propose a novel mobile assisted range-based indoor localization algorithm specifically designed for multi-storey buildings. Our algorithm is composed of the Passive Localization and the Mobile Assisted Localization phases with methods to minimize the interference of the mobile as much as possible.

Keywords: trilateration, quadrilateration, 3D localization, indoor localization, NP-hardness, wireless sensor networks..

ÖZ

GEZGİN YARDIMIYLA ÇOK KATLI BİNALARDA UZAKLIK ÖLÇÜMÜNE DAYALI KONUMLAMA

ALİ TEOMAN UNAY
Akıllı Mühendislik Sistemleri, Master
Fen Bilimleri Enstitüsü
Tez Danışmanı: Doç. Dr. HÜSEYİN AKCAN
Ocak 2015

Konumlama; sağlık hizmetleri, habitat gözlemleme, arama-kurtarma çalışmaları gibi bir çok uygulamada kullanılan bir hizmettir. Bu uygulamalarda konum bilgisi; toplanılan verinin sınıflandırılması, bina içi hareket planlarının ya da tahliye planlarının kurgulanması gibi amaçlar için kullanılır. Küresel konumlandırma sistemleri (GPS) konumlama hizmeti için genel tercih olsa da bu sistemler kapalı alanlarda çalışmaz. Bununla birlikte, kapalı ortam, quadrilaterasyon gibi konumlama algoritmaları için de pek çok sorun teşkil eder. Bu çalışmada, quadrilaterasyonun çok katlı binalardaki zayıflıklarının analizini yaptık ve quadrilaterasyon algoritmasını iyileştirmek için özgün metodlar geliştirdik. Bu metodlarda, ihtiyaç duyulan zamanlarda bir gezgin birimin yardımını aldık. Ayrıca, bu gezgin birimin kullanımını en aza indirirken, konumlanan birim sayısını maksimuma çıkarmayı da amaçladık. Bu tezde, spesifik olarak çok katlı binalar için tasarlanmış, özgün , uzaklık ölçümüne dayalı bir konumlama algoritması sunuyoruz. Algoritmamız, Pasif Konumlama ve Gezgin Yardımlı Konumlama safhalarından oluşuyor ve gezgin müdahalesinin gereksinimini olabildiğince en aza indirmeyi hedefliyor.

Anahtar Kelimeler: trilaterasyon, quadrilaterasyon, 3B konumlama, kapalı alanda konumlama, NP-hardlık, kablosuz algılayıcı ağlar.

ACKNOWLEDGEMENT

This work is supported by Scientific and Technological Research Council of Turkey (TUBITAK) Career Grant No: 112E099.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

Localization is a general term for determining position information of the sensor nodes in a wireless sensor network (WSN) on an environment either in 2D or 3D space. The reference of origin for localization can be either global or local (in a building, etc.) [1] [2] [4] [5] [6] [7] [8] [9]. Generally, GPS is the common choice for the localization process yet it may not be available indoors such as buildings or tunnels [10]. Also GPS is not accurate enough for scenarios that depend on highly precise localization. In this case, a well-known GPS-free localization technique, range-based localization, is widely used [4] [5] [6] [11].

In ranged-based localization, distance values between neighbor nodes are measured in a certain amount of measurement errors. The goal is finding positions of the sensor nodes in the WSN relative to each other. However, using only distance measurements among nodes, turns the localization problem into a graph realization problem which is strongly NP-Hard [12] [13]. A version of the problem restricted to 2D lends itself to a polynomial time solution through trilateration [14] when the underlying graph has a trilateration ordering. However, the existence of noise in measurements turns the localization via trilateration into an NP-complete problem [15] yet again. Trilateration algorithm in 2D, starts with three beacon nodes with known positions, and localizes a new node using distances to three non-collinear nodes already localized at each subsequent iteration. In this paper, we generalize trilateration in 2D to 3D called *quadrilateration*, which uses four non-coplanar nodes to localize a fifth.

We address the problem of localization in 3D for multi-storey indoor environments in this paper. This particular 3D localization problem is reduced to a sequence of 2D subproblems corresponding to floors in a building. To this end, a novel localization algorithm in 3D, called Mobile Assisted Range-Based Indoor Localization (MARIL) is proposed. MARIL extends the trivial quadrilateration algorithm with new techniques to operate in multi-storey environments with the help of a mobile node. Although the employment of a mobile node ensures the localization of the entire network in finite time, MARIL is specifically designed to minimize the intervention of the mobile node. Our algorithm is based on the observation that sensors deployed onto parallel floors in multi-storey buildings form parallel planes, and we exploit this planar deployment in localizing the nodes. When a node's position information is calculated, that node is called as a *localized node* while a node with no position information is called as an *unlocalized node*.

In MARIL algorithm, localization starts with figuring out the relative positions of unknown neighbors using the position and distance informations of three apriori known nodes as seeds. If available distance measurements or position informations are not enough, quadrilateration cannot localize all of the sensor nodes due to ambiguities. MARIL algorithm gets a mobile nodes help, for these cases, to finish the localization. Yet, since it is costly to use a mobile node in terms of time and energy, it only interferes to fix the ambiguities via localizing an unlocalized sensor node or a group of unlocalized sensor nodes. After enough distance and position information gathered by the mobile node to continue localization, the mobile node stops until it is commanded otherwise by the algorithm again.

In MARIL, we assume the floor plan of the building is known apriori, and since all the nodes in a single storey is assumed to be on the same plane, we require all sensors to be stationed on the floor level position. Also, the mobile node always tracks the floor number that it belongs.

Indoor localization has many application areas ranging from search-and-rescue missions [16] to locating people or items in hospitals; from locating goods in large depots to tracking customers in shopping malls. In most of these scenarios either a mobile node is being tracked, or a mobile node is seeking to find the positions of the static nodes. An example scenario is presented in Figure 1.1, where there are stationary nodes in a multi-storey building, and a mobile node travels to locate the positions of these stationary nodes. Stationary nodes can be assumed to correspond to people in a building carrying wireless radios, and do not move (due to fire conditions, etc.). Even when all the nodes are mobile, a snapshot of the network is considered feasible to be a static input to our algorithm.

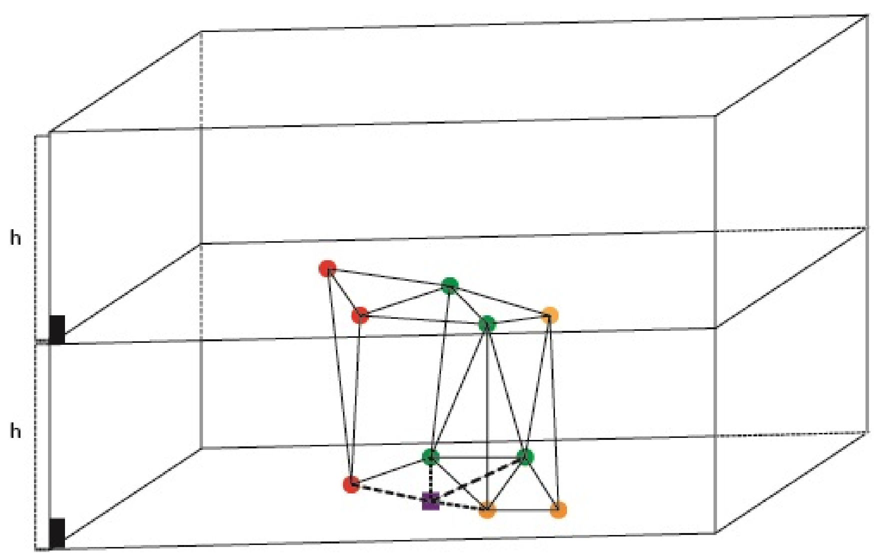

Figure 1.1: An example multi-storey building with stationary nodes deployed and a mobile node. For the ease of representation, only two floors with equal heights to the ceilings are illustrated. Purple square represents the mobile node, red circles represent unlocalized nodes, green circles represent localized nodes, and yellow circles represent beacon nodes with apriori known positions. Lines between circles represent distance measurements. Black rectangles on the left side of each floor represent the entrance points to the respective floors.

MARIL algorithm that we propose in this study is applicable to various real life scenarios depicted as:

- In disaster situations like fire, storm, earthquake, etc. accurate localization of civilians that are stuck in a building is a critical information for a search-and-rescue mission. In a search-and-rescue mission, the mobile node (firefighter, etc.) is expected to locate and rescue the people, and also aid in generating an evacuation plan. As time is a very critical factor here, the mobile node is required to travel along a path with the minimum total distance until all the nodes are localized. Using MARIL algorithm, we can generate a mobility path plan for the mobile node in advance.

- Locating specific goods in a huge multi-storey depot. If every good in the store has a sensor node placed on the bottom of it, their positions can be calculated.

Multi-storey buildings, where nodes are positioned on parallel floors, pose a challenge for 3D localization algorithms such as quadrilateration, as quadrilateration requires distance measurements to four non-coplanar nodes with apriori known positions to operate. The problem becomes more apparent in a realistic real-world scenario when the wireless range is small, and the nodes can communicate with neighbors as far as one floor above or below only. For such low ranges, quadrilateration in general gets stuck in the first two floors of the building. Therefore, in this study, we propose a novel range-based localization algorithm specifically designed for multi-storey indoor environments. We extend the quadrilateration algorithm to take advantage of the parallel floors, and introduce also a mobile node to aid the localization when only necessary. In doing so, we propose methods to reduce the distance traveled, and hence minimize the time and the energy spent by the mobile.

The main contributions of this thesis can be listed as:

- We propose a range-based, GPS-free indoor localization algorithms for multi-storey buildings that we call as MARIL.

- MARIL algorithm exploits the parallel positioning of the floors to turn the 3D localization problem into several 2D localization problems, and proposes novel ways to reconstruct the solution of the 3D localization problem from the solutions of these 2D sub-problems.

- In cases of ambiguities in the WSN graph structure that prevents the localization to continue, MARIL algorithm uses a mobile node to solve the ambiguity. We propose methods to minimize the intervention of the mobile node.

- MARIL algorithm is designed to work in environments with measurement errors.

The remainder of this study is structured as follows. In Chapter 2, related works are discussed. In Chapter 3, background and technical information for this study is given. In Chapter 4, MARIL algorithm is discussed in detail. In Chapter 5 simulation results is presented with various experiment scenarios. In Chapter 6 conclusion of the study is discussed.

Chapter 2

Related Work

Localization is the technique that calculates the positions of the wireless sensors in a WSN. Traditionally, GPS is used for localization but GPS cannot provide the accuracy that is required for many applications [17][18][19].

Range based localization [1][3] is the technique that is used in applications where GPS fails to provide the accuracy that is needed. Also, it is not possible to obtain indoor location data using GPS. In range based localization, distance information between sensor nodes can be presumed to be known apriori or can be measured dynamically using several well known ranging techniques like time of arrival (TOA), time difference of arrival (TDOA) [20], or ultra wide band (UWB) [21].

Using only range values among nodes, turns the localization problem into a graph realization problem which is NP-hard [12]. The problem is still NP-hard even for the rigid graphs that are known to be localizable [22] [23].

Previous research on localization in wireless sensor networks emphasize on stationary sensor networks [1] [24] and mobile sensor networks [7][3]. Also, there are works which use both stationary and mobile networks [25] [26] [27].

In addition to localization, enhancing or repairing initial WSN graph structures are also stated on various studies [27]. [25], is a study that works on

GPS-free range-based localization and makes the network structure more rigid via a mobile nodes aid. [26] also gets help from virtual nodes but mostly focuses on mapping using SLAM algorithm and uses a sonic device to detect obstacles and walls.

GDL algorithm is proposed in [3]. In GDL, every sensor node in the network has a compass and all nodes can move individually. GCDL algorithm, in the same work, is the compass free version of GDL but it requires additional movement to work. Both algorithms exploits node mobility to calculate neighbor positions without the use of any global information. Sequential Monte Carlo method is introduced in [28] and mobility of nodes is used to improve the accuracy and precision of localization for partially or completely mobile sensor networks. Robomote [27], uses a control law in the context of sensor-based path planning [29]. The mobile node queries static nodes in its neighborhood and computes the optimal location velocity using field gradient information from each neighbor. Movement action of Robomote uses an algorithm based on biased random walk, modeled on taxis in bacteria, for tracking gradient sources.

Chapter 3

Background and Technical Information

In this chapter, notation and background information is presented which is used in the rest of the study.

3.1 General Notation

Let G be an undirected connected WSN graph; $G = (V, E)$ in an environment R, $V = \{v_1, v_2, ..., v_n\}$ is the set of vertices, $E = \{e_1, e_2, ...e_m\}$ is the set of edges where, n is the number of sensor nodes in the WSN and m is the number of calculated distance measurements between sensor nodes. d is the distance function that has the value for each $(v_i, v_j) \in V$ is the distance between v_i and v_j. Point formation is a set of points $P = \{p_1, p_2, ..., p_n\}$ in 2 or 3 dimensional space. The point p_i represent the position of node v_i in G. Depending on the range of the nodes, each sensor node $n_i \in R$ has a specific set of neighbors (adjacent nodes) Adj_i. All sensor nodes has a certain amount of measurement distance limit. That limit will be called as the *range* of a node.

Let $F = F_1, F_2, ...F_k$ be the set of floors of the building in R where G is present and k is the number of floors. $H = \{h_1, h_2, h_3, ...h_{k-1}\}$ is the set of floor heights. Every $v_i \in V(G)$ is assigned to one floor F_t where $1 \leq t \leq k$. Initially, there are three nodes, $B = \{v_{b_1}, v_{b_2}, v_{b_3}\}$ and they are placed in two different floors, F_a, F_b. Coordinate informations of nodes in B are known apriori and they are called as *beacon nodes*. All three nodes must have a common unlocalized neighbor and both F_a and F_b need to have at least one beacon node in order to obtain the required positioning of beacon nodes for our algorithm to start. Details of these requirements are described in Section 4.1.3.

3.2 Network Localization & Rigidity

Network localization is the operation of determining sensor nodes' relative positions to each other in a network. All sensor nodes have a particular radio communication distance limit, which is called as the *range* of the node. Nodes can only detect other nodes if they are within their range and two nodes that detects each other are called as *neighbor nodes*. The operation of determining P using given edge lengths is called *range based localization*.

3.2.1 Rigidity

Let G' be a graph with the distance function d'. If d' is the same as d for any point formation of G' then G and G' are *congruent*. G is *globally rigid* if it is congruent with any point formation of G that has the same distance function. G is *generically rigid* in a two dimensional environment if and only if E contains a subset E' consisting of $2n - 3$ edges with the property that for any nonempty subset $E'' \subset E'$, the number of edges in E' cannot exceed $2n' - 3$ where n' number of vertices of G which are endpoints of edges in E' (Laman [30]). After removing any single edge from G if it still remains generically rigid, then G can be called as *redundantly rigid*. If G is redundantly rigid and $z + 1$ connected, where z is the dimension of the space, G is called as *generically globally rigid* [23].

It is proven, if the graph structure is *generically globally rigid*, then it is localizable, however, the problem is still NP-hard [22] [23]. If the graph structure of the network is not generically globally rigid, some nodes may appear on two possible spots. That ambiguous situation is called as *flip ambiguity* (Figure 3.1) [31].

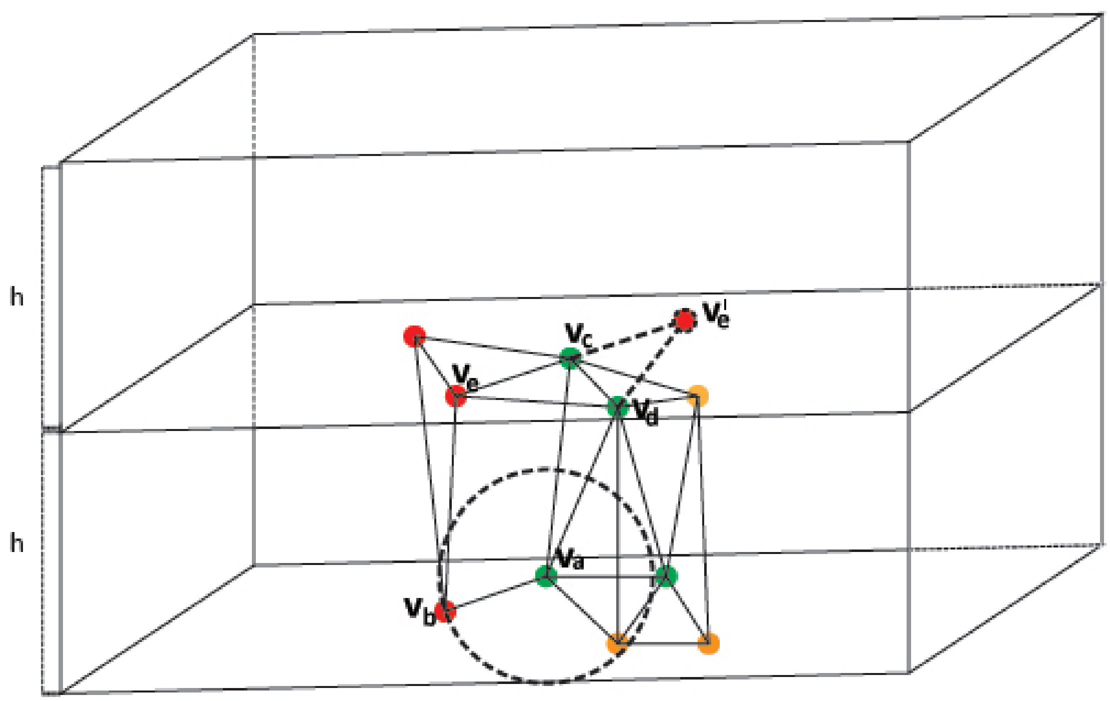

Figure 3.1: An example figure for ambiguous situations. Red circles represent unlocalized nodes, green circles represent localized nodes, and yellow circles represent beacon nodes with apriori known positions. Lines between circles represent distance measurements. Dotted lines and circles represent possible positions for respected nodes.

In Figure 3.1, v_e has measurements between two localized nodes v_c and v_d. Flip ambiguity occurs for this case as v_e appears on it's original position v_e and on flip position $v_e{}'$. Additional measurement information with a localized node is required to solve this ambiguity. v_b has only one measurement between a localized node v_a. In this case, an ambiguous situation occurs as v_b appears on a circle with radius $d(v_a, v_b)$. Additional measurements with two localized nodes are required in order to solve this ambiguity.

3.2.2 Trilateration & Quadrilateration

In z-dimensional geometry where $z \in \{2, 3\}$ trilateration and quadrilateration both are techniques of determining relative positions of unknown points via distance measurements, using circles or spheres.

In 2D, if a point lies on two concurrent circles, one can obtain two different position informations using circle centers and the two radii. Additional information such as an another concurrent circle which also contains the point may narrow the possibilities down to one unique position (Figure 3.2). Three center points are assumed to be non-collinear with each other. This technique is known as *trilateration* [14]. If there is enough connection between sensor nodes and range measurements are exact, trilateration function can solve the localization problem in polynomial time. Yet, in real world, range measurements have errors. Also, all nodes in the WSN may not find enough measurement data to apply trilateration. As mentioned before, Evrendilek and Akcan [15] recently showed that trilateration with range measurement errors is an NP-complete problem.

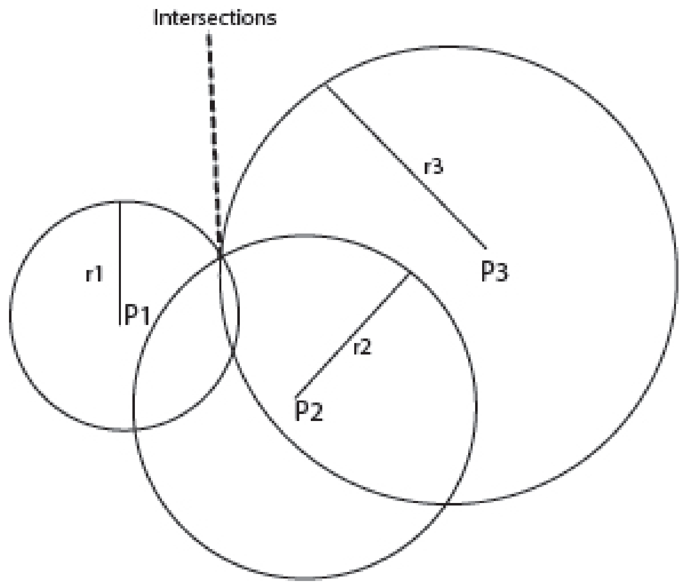

Figure 3.2: Trilateration - If a point lies on three circles, intersections of these circles provide a unique position information of the point, unless the centers lie on a straight line

In 3D, if a point lies on three concurrent spheres, one can obtain two different position informations using sphere centers and the three radii. Center points are assumed to be non-coplanar with each other and any three of the center points are not lying on a line. Additional information such as an another concurrent sphere which also contains the point, may narrow the possibilities down to one unique position. This technique is called *quadrilateration*.

3.3 Cayley - Menger Determinant

We use Cayley-Menger determinant [32] for collinearity checks between neighbor nodes:

Let v_A, v_B, v_C and v_D are stationary nodes in a wireless sensor network. $d(v_A, v_B)$ is the distance between node v_A and v_B.

v_A, v_B and v_C are collinear if and only if for any three nodes of v_A, v_B, v_C:

$$det \begin{bmatrix} 0 & d(v_A, v_B)^2 & d(v_A, v_C)^2 & 1 \\ d(v_A, v_B)^2 & 0 & d(v_B, v_C)^2 & 1 \\ d(v_A, v_C)^2 & d(v_B, v_C)^2 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} = 0$$

In our framework, Cayley-Menger determinant is also used by the mobile node to determine to which static node is belong to which floor.

$$det \begin{bmatrix} 0 & d(v_A, v_B)^2 & d(v_A, v_C)^2 & d(v_A, v_D)^2 & 1 \\ d(v_A, v_B)^2 & 0 & d(v_B, v_C)^2 & d(v_B, v_D)^2 & 1 \\ d(v_A, v_C)^2 & d(v_B, v_C)^2 & 0 & d(v_C, v_D)^2 & 1 \\ d(v_A, v_D)^2 & d(v_B, v_D)^2 & d(v_C, v_D)^2 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} = 0$$

Chapter 4

Mobile Assisted Range-based Indoor Localization (MARIL) Algorithm

In this chapter, MARIL algorithm is described. MARIL algorithm is a localization algorithm that solves the range-based localization problem for a connected WSN graph in a multi storey building with parallel floors. Floor plan of the building is assumed to be known apriori and all sensor nodes in the WSN are stationed on the floor level of each storey.

MARIL algorithm also gets a mobile sensor node's help yet it only interferes if the WSN has a graph structure that causes ambiguities and prevents application of trilateration or quadrilateration. Since it is costly in terms of time and energy, the mobile node is used only to fix the ambiguities of the graph structure that prevents localization to continue.

Beacon nodes serve as seeds to localization and a reference for the rest of the WSN graph. Localization continues until all of the sensor nodes are localized in the WSN graph or cannot continue anymore due to the ambiguities in the graph structure. This phase of the algorithm is called as *Passive Localization*.

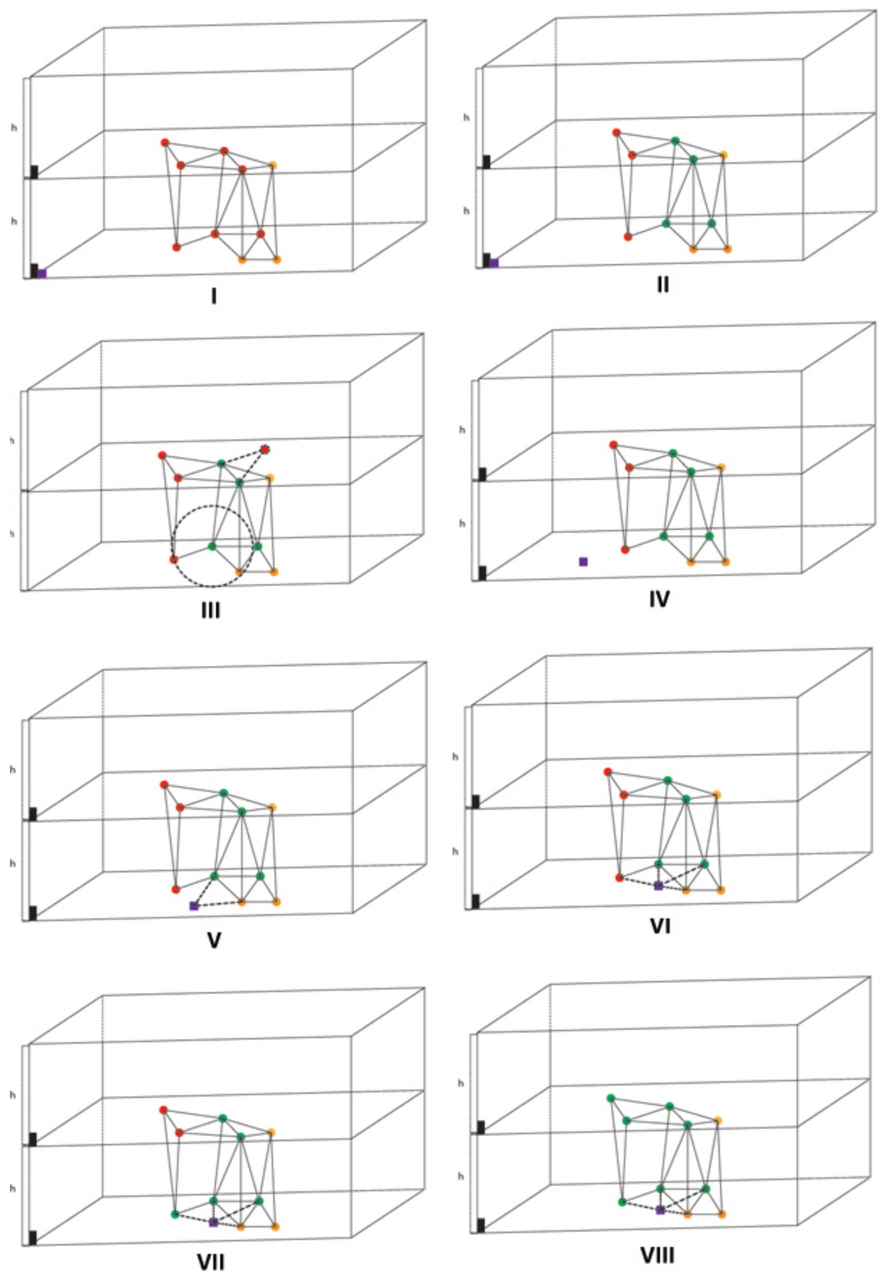

Figure 4.1: An example figure for MARIL algorithm. Red circles represent unlocalized nodes, green circles represent localized nodes, and yellow circles represent beacon nodes with apriori known positions. The purple square represents the mobile node. Lines between circles represent distance measurements.

If localization cannot continue and there are still unlocalized sensor nodes in the WSN graph, then the mobile node steps in, moves through the WSN graph according to the desired movement algorithm and localize an unlocalized sensor node or a group of unlocalized sensor nodes in order to help to continue localization until all of the sensor nodes in the WSN graph are localized. This phase of the algorithm is called as *Mobile Assisted Localization*. Passive Localization is discussed in Section 4.1 and Mobile Assisted Localization is discussed in Section 4.2.

An example scenario is illustrated in Figure 4.1. At the start, there are only beacon nodes and unlocalized nodes (I). Starting with the beacon nodes, localization continues until any ambiguity occurs (II). If any ambiguity occurs (III - the ambiguous situations are explained in Section 3.2.1), the mobile node steps in (IV). The mobile node moves until it founds measurements between three localized nodes to localize itself (V,VI). After it localized itself, it localizes an unlocalized node within it's range (VII). If the ambiguity is solved, localization continues without the mobile nodes help until all of the sensor nodes are localized (VIII).

4.1 Passive Localization

In this section, the Passive Localization phase of the MARIL algorithm is discussed. Aim of this phase is localizing every sensor node in the WSN graph in the building without getting help from the mobile node. Localization starts by localizing a common unlocalized neighbor node via taking the beacon nodes as seeds. After that, localization continues with known techniques like trilateration and quadrilateration as well as our proposed techniques for this algorithm like Node Floor Discovery and Grapple which are described in detail in the following subsections. Applications of these techniques depend on specific conditions and an order.

Node Floor Discovery (Section 4.1.1) and Grapple (Section 4.1.2) techniques are going to be discussed before the detailed description of the localization order of the Passive Localization phase.

4.1.1 Node Floor Discovery

Aim of the Node Floor Discovery technique is to determine floor informations of the unlocalized nodes since the localization techniques we use depend on floor informations of the sensor nodes. If floor information of a sensor node is known, that node is called as *floor discovered node*.

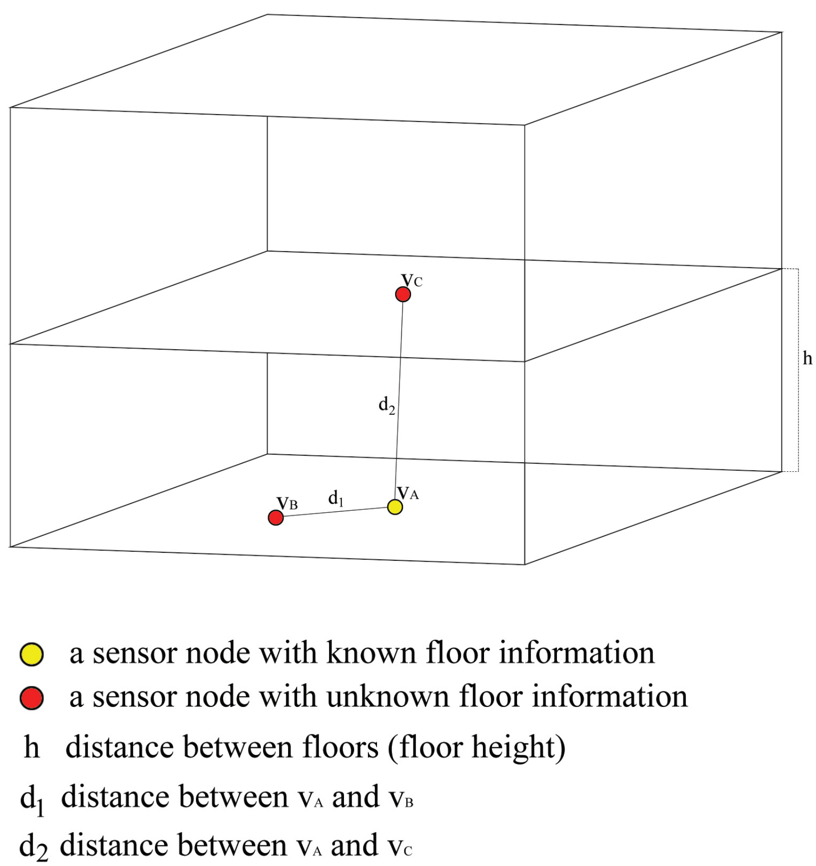

Figure 4.2: Node Floor Discovery - Assuming that $d_1 < h \leq d_2$ it can be said that v_A and v_B are on the same floor.

Node Floor Discovery uses distance information between two nodes. In our scenario, floors have apriori known heights as we know the floor plan of the building. Thus, we know that if the distance between two nodes is smaller than the floor height, than we can conclude that they are definitely on the same floor (Figure 4.2). Additionally, in Figure 4.2, it can be seen that v_C and v_A are not on the same floor, yet it cannot be concluded using MARIL since the distance between v_C and v_A is bigger than the floor height.

4.1.2 Grapple Localization Technique

For the conditions that causes flip ambiguity, one of the flip locations must be eliminated in order to provide the true relative position information. In the MARIL Algorithm scenario, there are two conditions which are crucial to solve in order to provide fully localization:

- Let $v_A, v_B, v_C \in G$ be three localized nodes and $F_x, F_y \in F$. Let $v_A, v_B \in F_x$ and $v_C \in F_y$. $v_D \in G$ be an unlocalized sensor node where $v_D \in \{Adj_A, Adj_B, Adj_C\}$. It is not possible to localize v_D via trilateration technique using v_A, v_B and v_C since that situation causes flip ambiguity. In order to solve it, Cross-Grapple technique is used (Section 4.1.2.1).
- Let $v_A, v_B, v_C \in G$ be three localized sensor nodes and $F_x, F_y \in F$. $v_A, v_B, v_C \in F_x$. $v_D \in G$ be an unlocalized sensor node where $v_D \in F_y$ is known and $v_D \in \{Adj_A, Adj_B, Adj_C\}$. It is not possible to localize v_D via trilateration technique using v_A, v_B and v_C since that situation causes flip ambiguity. In order to solve it, Vertical-Grapple technique is used (Section 4.1.2.2).

4.1.2.1 Cross-Grapple

If a node has distance measurements to three nodes with known positions on two adjacent floors, the localization using quadrilateration gives two possible solutions, as shown in Figure 4.3. A valid solution, in this case, should have a z-coordinate value consistent with the floor plan. Therefore, a solution with an inconsistent z-coordinate can easily be eliminated. This method is called as *Cross-Grapple*.

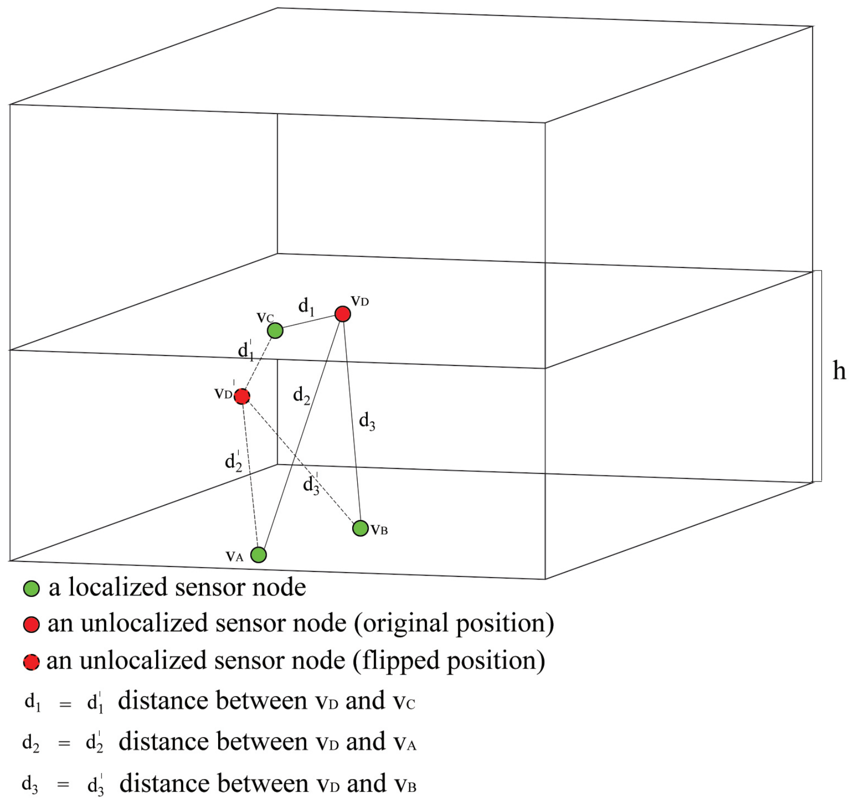

Figure 4.3: Cross-Grapple can eliminate flipped node position because v_C is neither on h = 80 nor h = 0

Although it is an exceptional situation, if both flipped and original positions are on an existing floor information due to different floor heights or seed node positions, Cross-Grapple cannot be applicable.

4.1.2.2 Vertical-Grapple

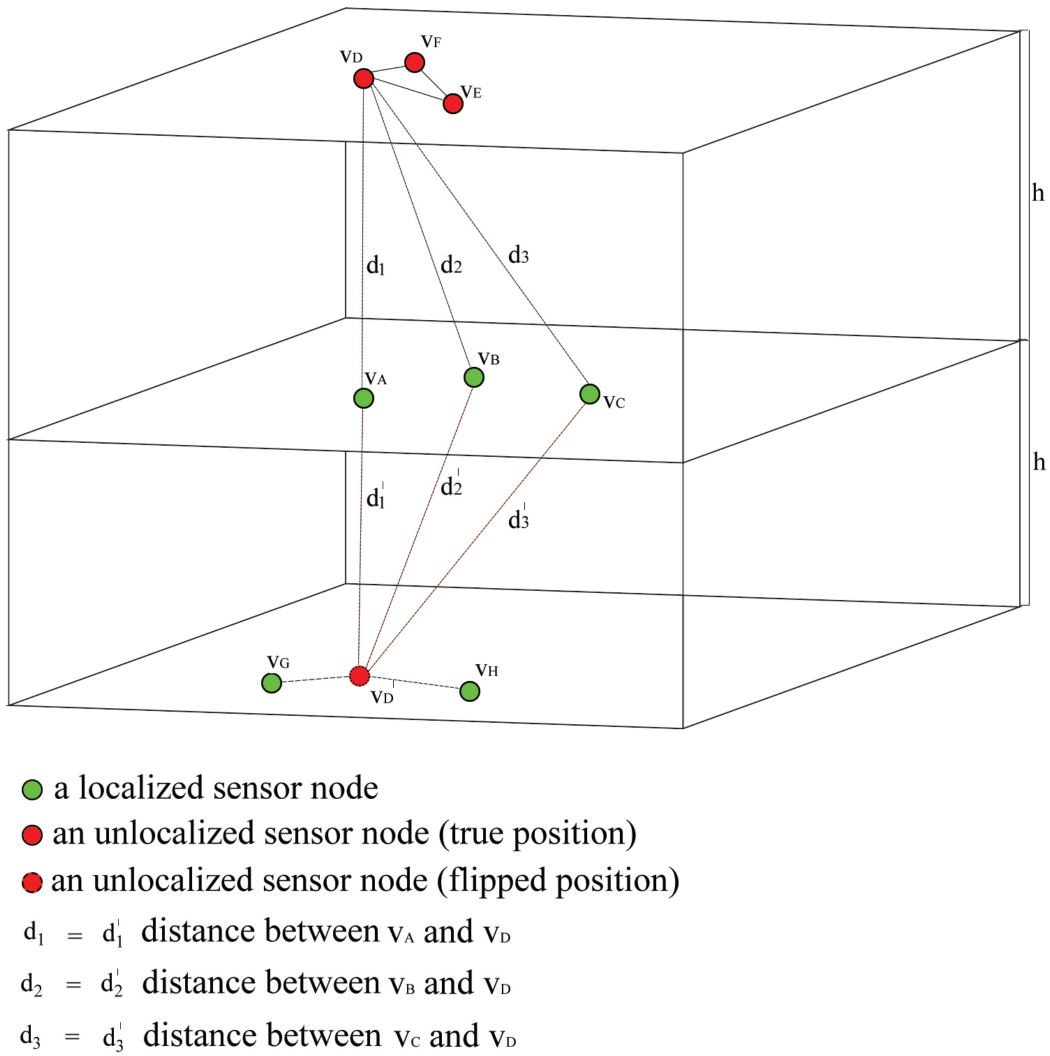

Figure 4.4: Vertical-Grapple checks the neighbor list of v_D and eliminate flipped position after seeing that v_G and v_H are not neighbors of v_D. It is assumed that both floors have same height value.

Vertical-Grapple technique uses three localized nodes which are on the same floor in order to localize common unlocalized neighbors on different floors.

In Figure 4.4, if a node v_D has distance measurements to three non-collinear nodes with known positions that are all on the same floor, but different than that of node v_D, then quadrilateration gives two possible solutions, both consistent with the floor plan. In this case, the neighbors of v_D with already assigned floors can be exploited to eliminate one of the two symmetrical solutions. This method is called as *Vertical-Grapple*. At worst case, both sets have the same elements, $\{v_A, v_B, v_C\}$, for these cases Vertical-Grapple is not applicable.

Vertical-Grapple technique is only necessary and only works when $range < 2 * h_{min}$ and h_{min} is the shortest floor height of the building. This technique is required only when quadrilateration or cross-grapple is not applicable and cannot reach higher floors to continue due to low range.

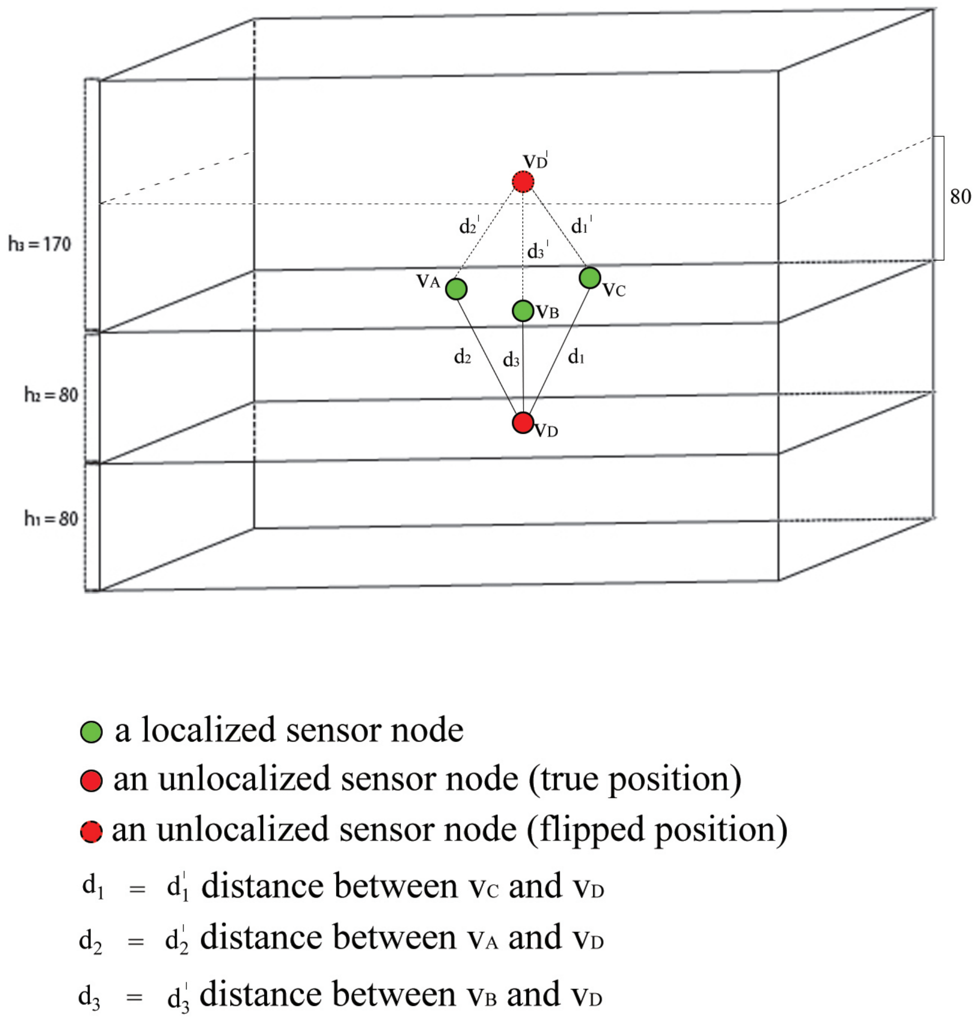

Figure 4.5: Flip position of v_D would not be on an existing floor, since the floor heights of the building are not the same.

If the floor heights of the building are not the same, one of the positions of v_D would not be on an existing floor. It can be seen on Figure 4.5, $v_D{}'$ is not on any floor of the building. Since we know the floor plan of the building which also gives us the height of every floor, $v_D{}'$ which is the flip position of v_D can be eliminated, since it is not neither on $h = 80$ nor $h = 330$.

4.1.3 Localization Order

In this subsection, localization order of the Passive Localization phase of the MARIL algorithm will be discussed. The order of the Passive Localization is given below:

- Cross Grapple starts localization with the beacon nodes until there are enough localized nodes to continue with quadrilateration.
- Quadrilateration continues the localization after there are enough localized sensor nodes to apply quadrilateration. The reason behind this is, quadrilateration does not depend on neither the floor height of the building nor the floor information of the sensor nodes.
- If quadrilateration cannot continue, trilateration steps in. Yet, trilateration only works on the sensor nodes which belong to the same floor. That's why, before trilateration, the floor information of eligible unlocalized nodes in the network is determined with the Node Floor Discovery technique. After that, trilateration continues to the localization.
- If both trilateration and quadrilateration cannot continue, the Cross Grapple or the Vertical Grapple techniques continues which are depending on both the floor information of the sensor nodes and the height information of the floors.

- If at least one node is localized after the quadrilateration, localization continue with the quadrilateration again. This cycle continues until there are not any eligible node for the Passive Localization.

If localization cannot continue with the Passive Localization, the mobile node steps in and the Mobile Assisted Localization phase starts.

4.2 Mobile Assisted Localization

In this section, the Mobile Assisted Localization phase is described with the movement strategy we present. Whenever the Passive Localization of the algorithm is stopped, if there are still unlocalized nodes in the WSN graph, that means WSN graph has an ambiguous situation and the Passive Localization cannot continue anymore. Aim of this technique is solving the ambiguous situation by interfering to the WSN graph by a mobile node and continue localization. The mobile node always starts from the bottom-most floor which has at least one unlocalized node. Also, the mobile node keeps track of the floor number it is currently on. When the mobile node start it's movement, it has the position information of any localized sensor node, floor information of any floor discovered sensor node, and neighborhood information of every sensor node in the WSN, in addition to the floor plan of the building and the positions of the beacon nodes.

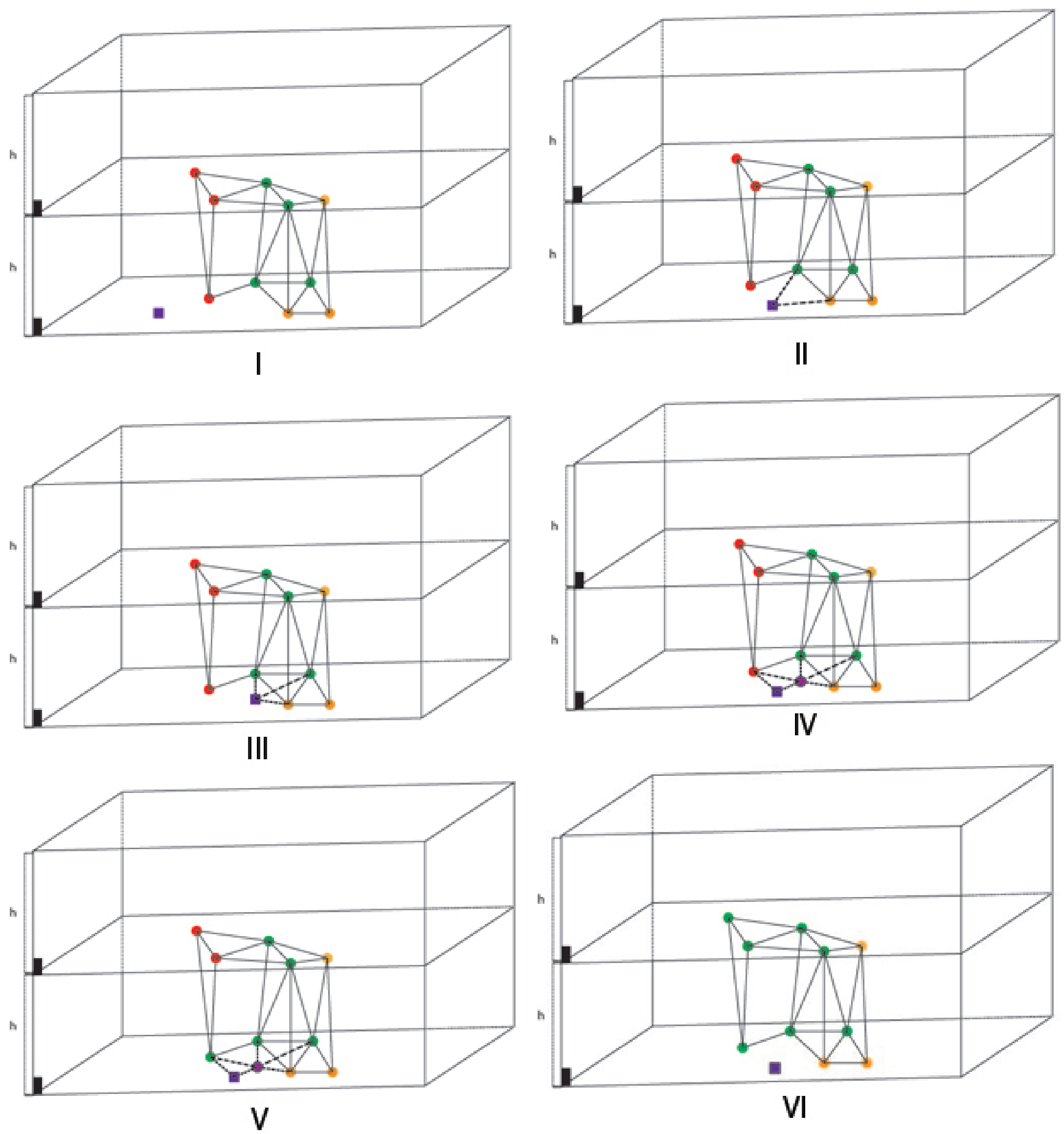

Figure 4.6: An example figure for the Mobile Assisted Localization. Red circles represent unlocalized nodes, green circles represent localized nodes, yellow circles represent beacon nodes with apriori known positions. Purple square represents the mobile node and purple circle represents a dropped virtual node. Lines between circles represent distance measurements.

In Figure 4.6, an example Mobile Assisted Localization scenario is illustrated. If the Passive Localization cannot continue to the localization because of an ambiguous situation, the mobile node steps in (I). First it needs to localize itself relative to the stationary sensor nodes. In order to do that, at least three measurements between the localized sensor nodes with the mobile node are required. Until it finds a required condition, the mobile node moves according to desired movement strategy (II). After the mobile node successfully localizes itself, it looks for any undetected sensor nodes within it's range (III). In order to localize an unlocalized node, at least three measurements are required. If the extra measurement added by the mobile node is still not enough to localize an unlocalized node, the mobile node saves the coordinate and measurement information of the current position of itself and moves to take an additional measurement information in order to localize the corresponded unlocalized node. The saved measurement and position data, acts like an actual sensor node and called as *a virtual node* (IV). After the unlocalized node is localized with the help of the mobile node (V), the Passive Localization continues, if the ambiguity is solved and the virtual node information is removed from the mobile nodes memory (VI). If the ambiguity still persists, the mobile node continues until the ambiguity is solved.

All measurement points to the unlocalized node, either virtual nodes or a localized sensor node, must be in a non-collinear fashion. In order to check this, the mobile node uses Cayley-Menger determinant with the three points that is using for localization. If the determinant value is zero, that means the points are collinear (Section 3.3). In that case, the mobile node discards the current virtual node and drops an another one to a different point.

The localization technique used for the Mobile Assisted Localization depends on the position of the unlocalized sensor node.

- If an unlocalized node is in the same floor with the mobile node, trilateration technique is used.
- If an unlocalized node is in a different floor from the mobile node and the distance between the mobile node and the unlocalized sensor node is smaller then $2 * h_{min}$ where h_{min} is the shortest floor height in the building, the vertical-grapple technique is used. The reason behind this is explained under Section 4.1.2 in this chapter.

If the unlocalized sensor node is not a floor discovered node, in order to determine the floor information of that node, the mobile node drops three virtual nodes and uses Cayley-Menger determinant using the unlocalized node and the virtual nodes. If the result is zero, that means the mobile node and the unlocalized node is in the same floor (Section 3.3).

4.2.1 Movement Strategy

Although it works independently, MARIL algorithm requires a movement strategy in order to determine the mobile nodes behavior efficiently. The mobile node must interfere to the graph, only when it is absolutely necessary and with minimum moving action since it is costly in terms of energy and time.

The mobile node continues it's movement until it finds an eligible spot to localize itself and interfere to the graph. In order to prevent the possibility of getting lost, the mobile node always stays in the range of localized nodes while moving around to travel within the building. Only exception to this is when the mobile node changes floors. In case no localized nodes are present, however, the mobile node follows a random movement pattern on the floor until a localized node is found.

The main objective of the mobile node is to reach to an unlocalized node in order to pinpoint its location. Until all the nodes are localized, it is guaranteed that there exists at least one unlocalized node with a localized neighbor in the network. This is a direct outcome of the assumption that the underlying network graph is connected. By moving towards the localized node, the mobile node gets into the range of an unlocalized neighbor, and making its own distance measurements, resolves the ambiguity associated with the unlocalized nodes.

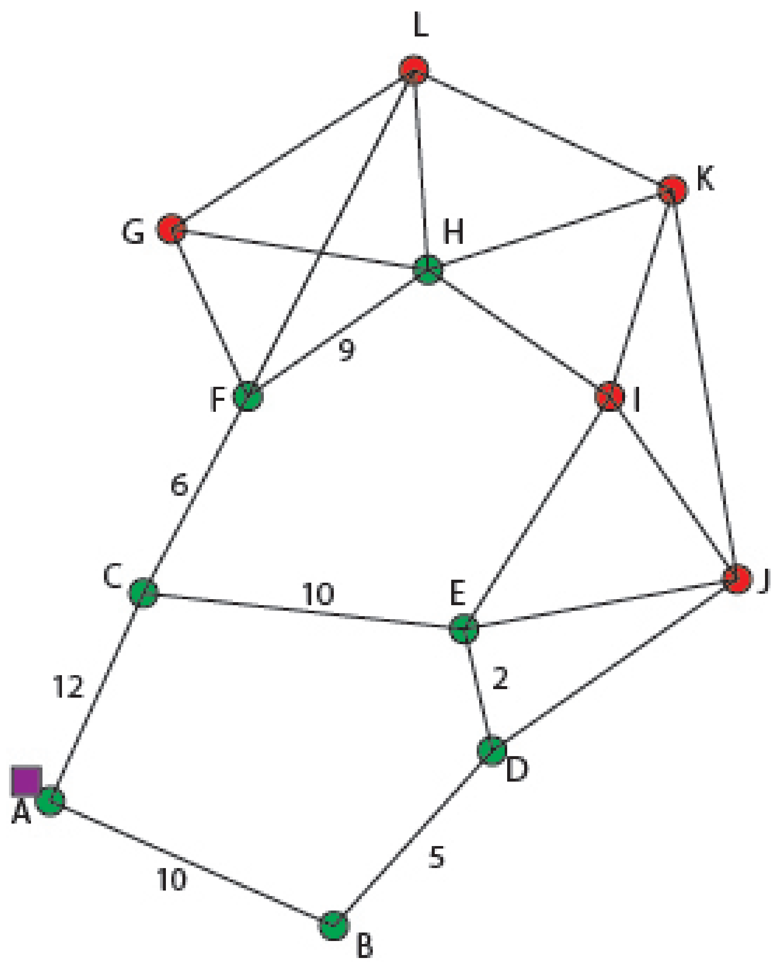

Figure 4.7: Nodes G, I, J, K, L are unlocalized, and the rest of the nodes are localized. Purple square represents the mobile node. Based on the given figure, localizing nodes G and L each has a gain of two, and I, J, and K each has a gain of five. The mobile node first localizes the closest node with the highest gain value.

Examining the network graph with respect to the localized and the unlocalized nodes, the mobile node computes a gain value for every unlocalized node. The gain of an unlocalized node is an estimate of its contribution towards a complete localization upon its selection as the next node to be localized by the mobile node. Therefore, the gain of an unlocalized node is defined simply as the number of additional nodes that can be localized in the next Passive Localization phase as soon as the mobile decides to localize it. The travel plan is generated to visit the closest nodes with the highest gain values. The Mobile Assisted Localization phase uses the *impact point* measure to select a candidate node to visit, which is the ratio of the node's gain to the shortest distance of that node from the mobile node. If we examine Figure 4.7, nodes G, I, J, K, and L are all unlocalized while the rest of the nodes are localized. The mobile node (purple square) calculates the gain of localizing each unlocalized node. For example, localizing node G will enable the localization of node L using trilateration, therefore the gain of node G is two. Localizing node J on the other hand, enables the localization of nodes G, I, K, and L, therefore has a gain value of five. In order to localize node G, the mobile node has to travel to the closest localized neighbor, node F. Therefore, the travel cost is 18. Using the gain and travel cost, impact point of node G is calculated as $\frac{2}{18}$. In the example figure, node J has the highest impact point ($\frac{5}{15}$), therefore, the mobile node generates the travel plan to localize node J first.

Chapter 5

Simulation Results

In this section, we present the simulation results of our MARIL algorithm. In order to carry out the simulations, we developed an in-house simulator written in Java. It simulates a five storey building with each floor having an area of 1000x1000 units, and a height of 80 units (Figure 5.1). As our main focus is on the localization quality, we assumed that each floor is free of obstacles, and none are placed in the building. We also assume in the simulations that the floor plan of the building is known, and the mobile node changes it's floor using apriori known entrance and exit points (stairs or elevators, etc.). In all the experiments, the mobile node enters the building from the first floor, and keeps track of the floor number it is currently on. Each experiment is conducted 100 times, and average values are reported.

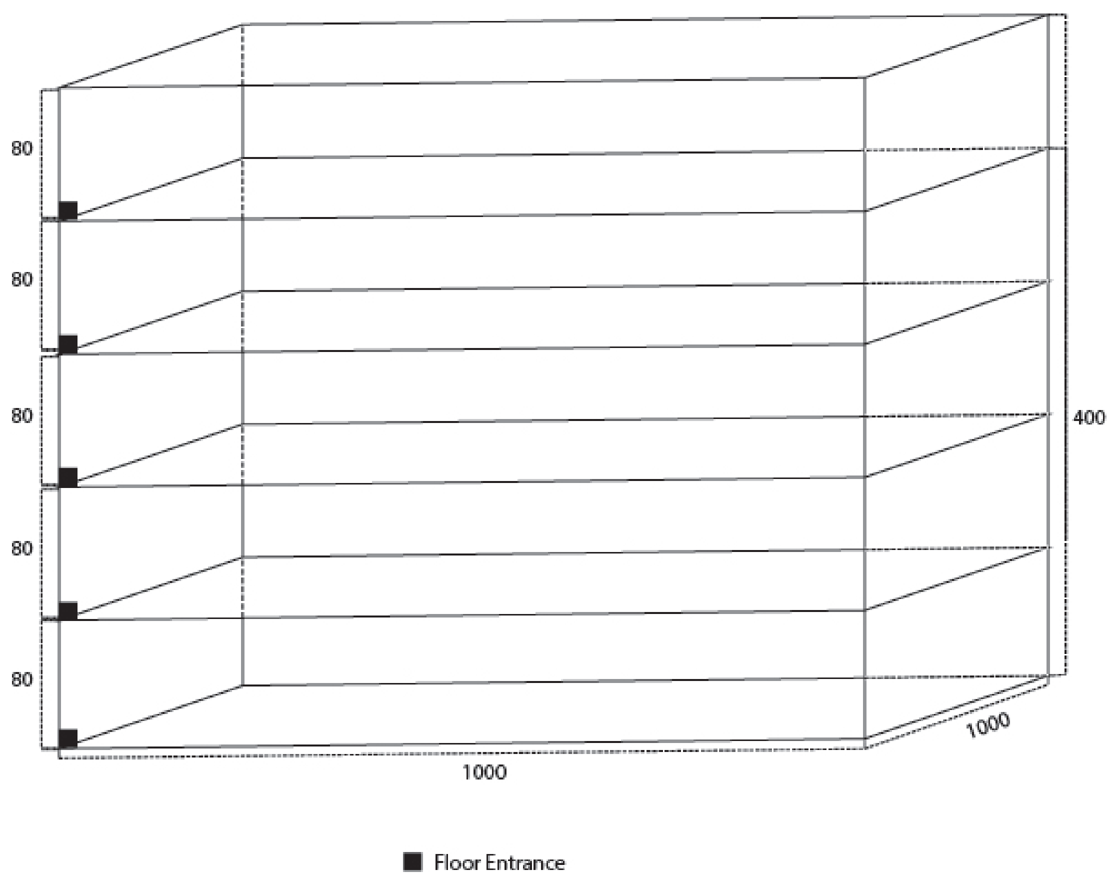

Figure 5.1: Representation of the simulated building.

We conduct experiments to identify the limitations of quadrilateration as a localization algorithm, and test how MARIL helps improve the overall quality of indoor localization. Therefore, in our first experiment, we compare the overall effect of quadrilateration, trilateration, grapple, and the mobile node on the localization percentages, i.e., the percentage of the nodes localized. Figure 5.2 presents how the localization percentage changes with increasing radio range for 150 nodes. The y-axis is the localization percentage of each algorithm, while the x-axis represents three different radio ranges. For each range, the contributions of the algorithms to the localization percentage are depicted separately. We can see from the figure that quadrilateration is less effective for wireless ranges close to 100, and gets more effective as the wireless range increases. For wireless range 100, quadrilateration localizes only 32% of the nodes in the first run of the Passive Localization phase until the mobile interferes. With the help of the mobile, quadrilateration can find beacon nodes to localize, and increases the percentage to 50% at the end of the localization. For range 100, localization percentages of the rest of the algorithms can be listed as 16% by cross-grapple, 20% by trilateration, 3% by vertical-grapple, 9% by the mobile node, and 2% for the beacon nodes. The success of quadrilateration increases for larger ranges, as nodes can communicate to nodes more than two floors away when the ranges are larger than 160, and this increases the chances for quadrilateration to find non-coplanar localized nodes.

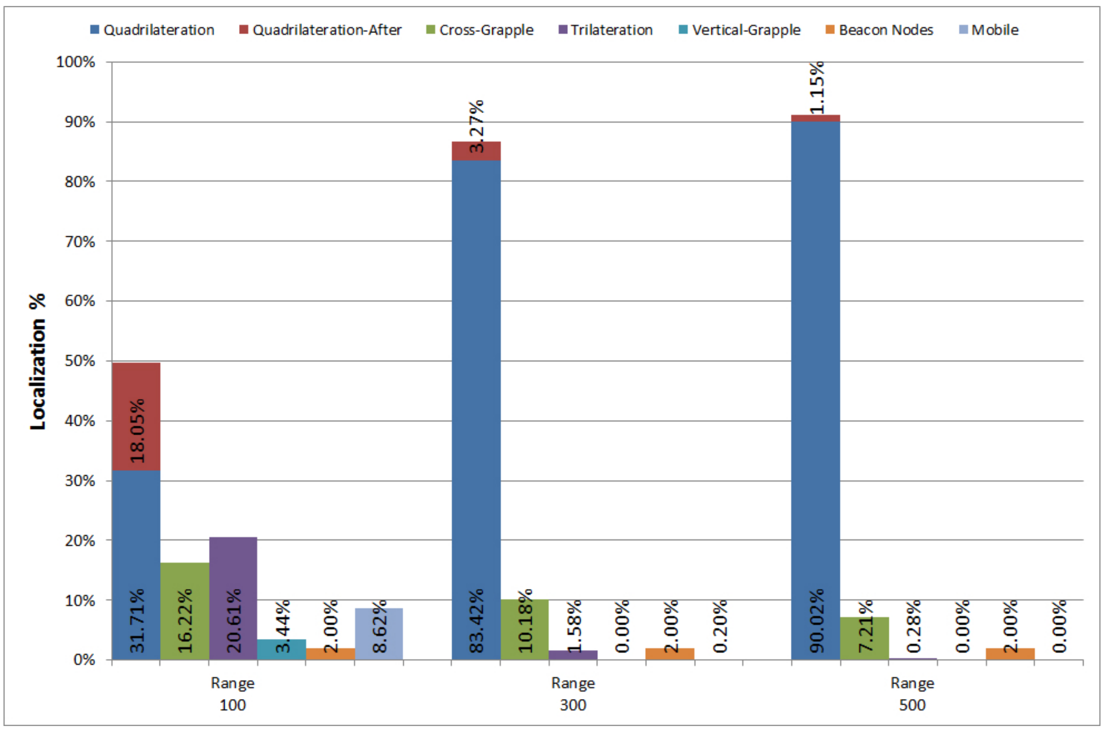

Figure 5.2: Change in localization percentage for each algorithm for various wireless range values.

We also observe from Figure 5.2 that the mobile node's intervention is minimal for low ranges, and practically zero for larger ranges. As the movement of the mobile node is costly both in terms of time and energy spent, minimizing its movement is an important outcome of our algorithm. We can conclude from Figure 5.2 that for low radio ranges, quadrilateration have problems to work to completion, therefore, the Passive Localization and the Mobile Assisted Localization phases with the supplementary algorithms like grapple are required to carry on localization. For higher radio ranges, the interference of the mobile node is not required. However, using a higher wireless range means consumption of more energy for each node, therefore, it is not cost effective. This result also suggests that there is a trade-off in terms of using a mobile node, or a higher wireless range.

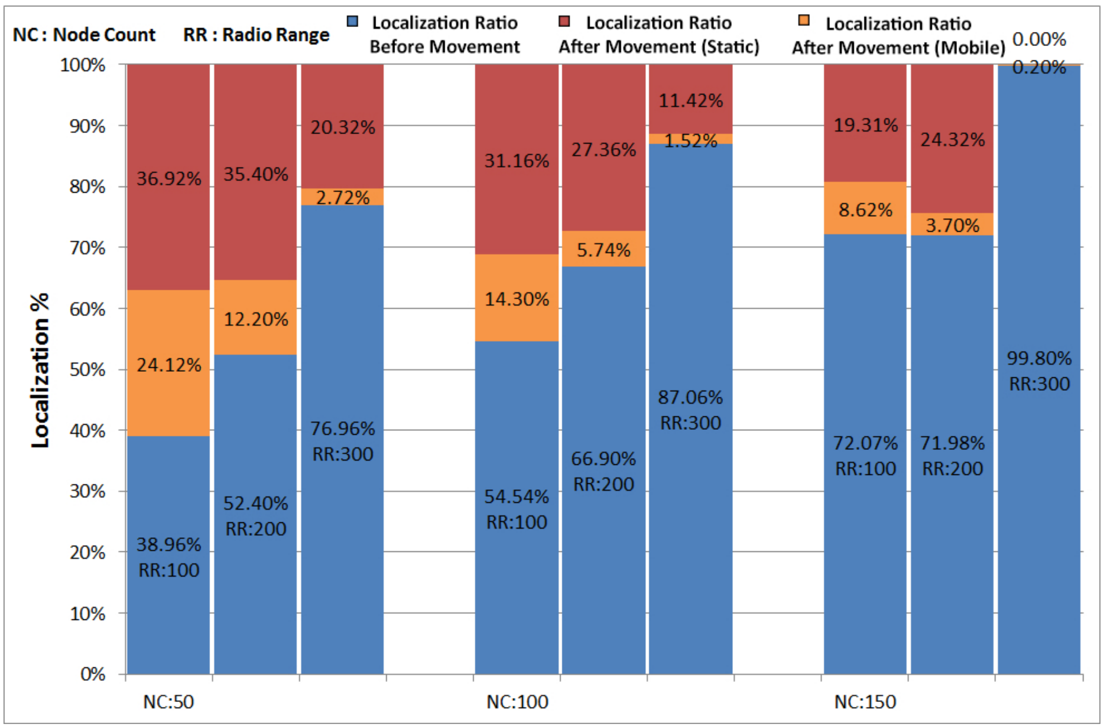

Figure 5.3: Change in localization percentage for each algorithm for various wireless range values.

In MARIL algorithm, when the Passive Localization is unable to localize any more nodes, the mobile node interferes so that the Passive Localization algorithms will have a chance to resume when there are now more nodes with known positions than in the previous run. In order to observe the percentage of the nodes that are localized before and after the first interference of the mobile, we conduct an experiment as shown in Figure 5.3. In this figure, we see the percentage of nodes localized before (in blue), and after the first interference of the mobile (in red) for node ranges 100, 200, 300, and for number of nodes varying from 50 to 150. From the figure, we observe that only a small percentage of the nodes are localized by the Mobile Assisted Localization phase (orange), and even this small contribution lets the passive algorithms to carry on, and localize more of the previously unlocalized nodes.

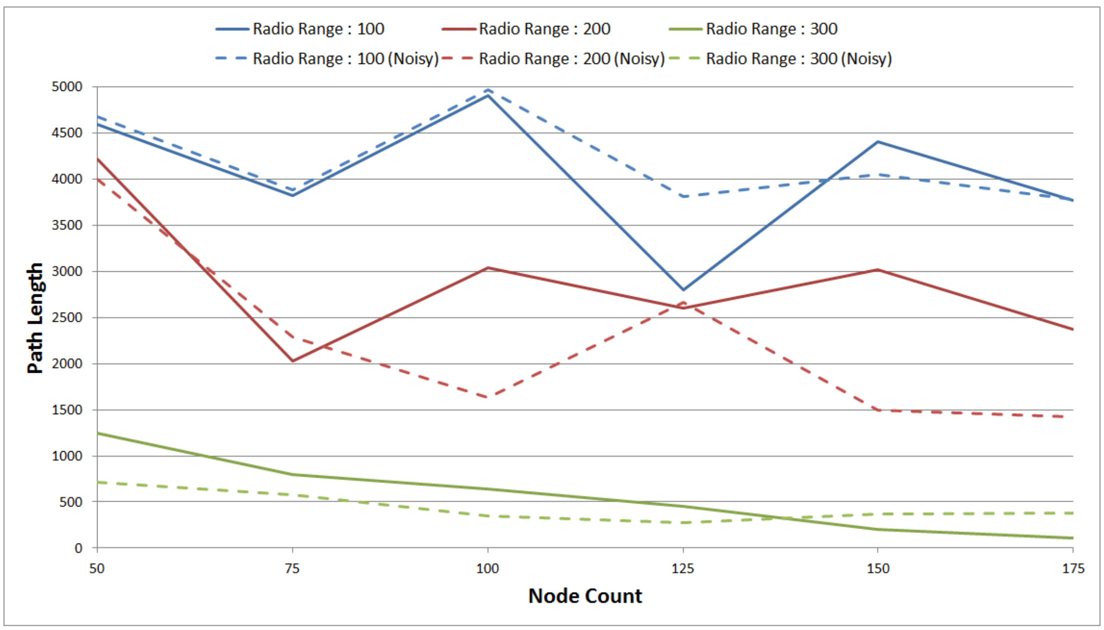

Figure 5.4: Effect of node count and radio range over the path length for noisy and noiseless scenarios.

In Figure 5.4, we depict the total path length traveled by the mobile node during localization, for various radio ranges, and node counts. We present the results for both noisy and noiseless scenarios. In the noisy scenario, we assume an error of ±3% in distance measurements, and ±8% in movement errors. As the range increases, the Passive Localization performs most of the work, and the total path length traveled by the mobile decreases. Even in small ranges, distances traveled by the mobile are small compared to the size of the whole floor. For range 100, the path length varies between 4000 and 5000, while if the mobile node swept the whole floor it would cost at least 5000 units distance per floor. Therefore, we can conclude that the total path length is reduced by a factor of 5 for range 100, and a factor of at most 93 for range 300.

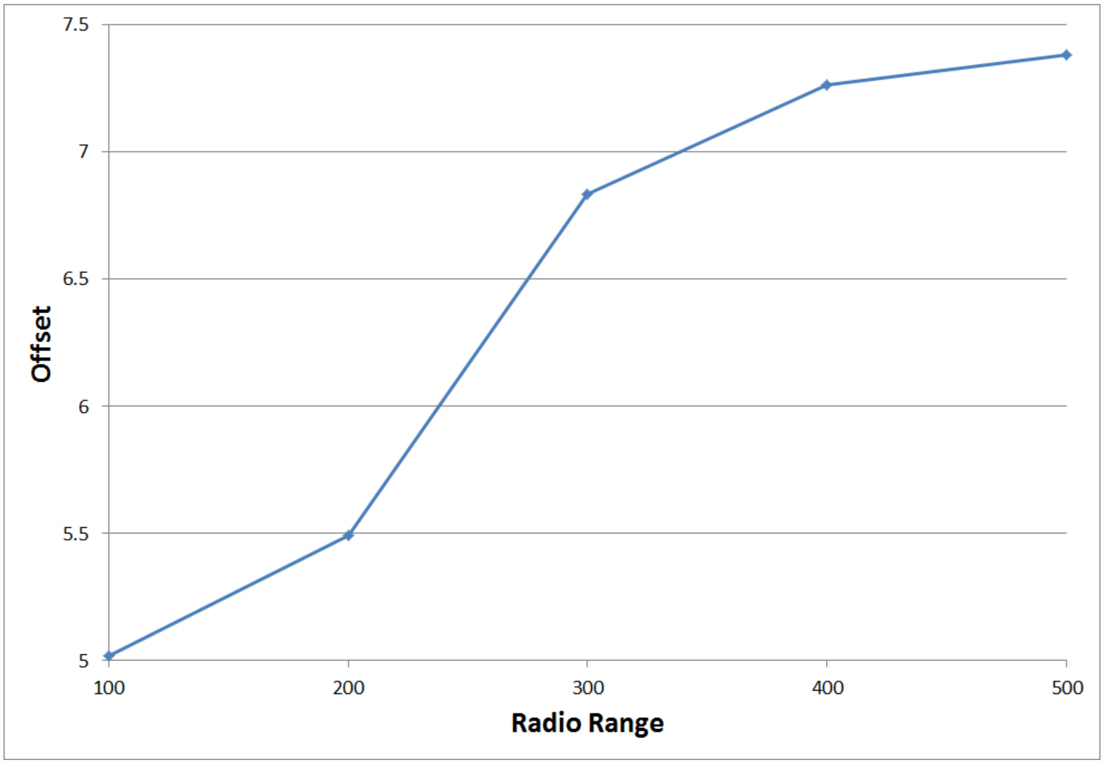

Figure 5.5: Change of localization offset for various radio ranges.

In Figure 5.5, we see the effect of increasing the range on the localization error, depicted as offset, with ±3% noise on distance measurements, and ±8% movement error on distance traveled by the mobile node. Localization offset is the average of the distances between the actual and the computed positions of the nodes. From the figure we can conclude that increasing wireless range improves the localization, and reduces the mobile node usage, however, at the cost of less accuracy on the positions of the nodes.

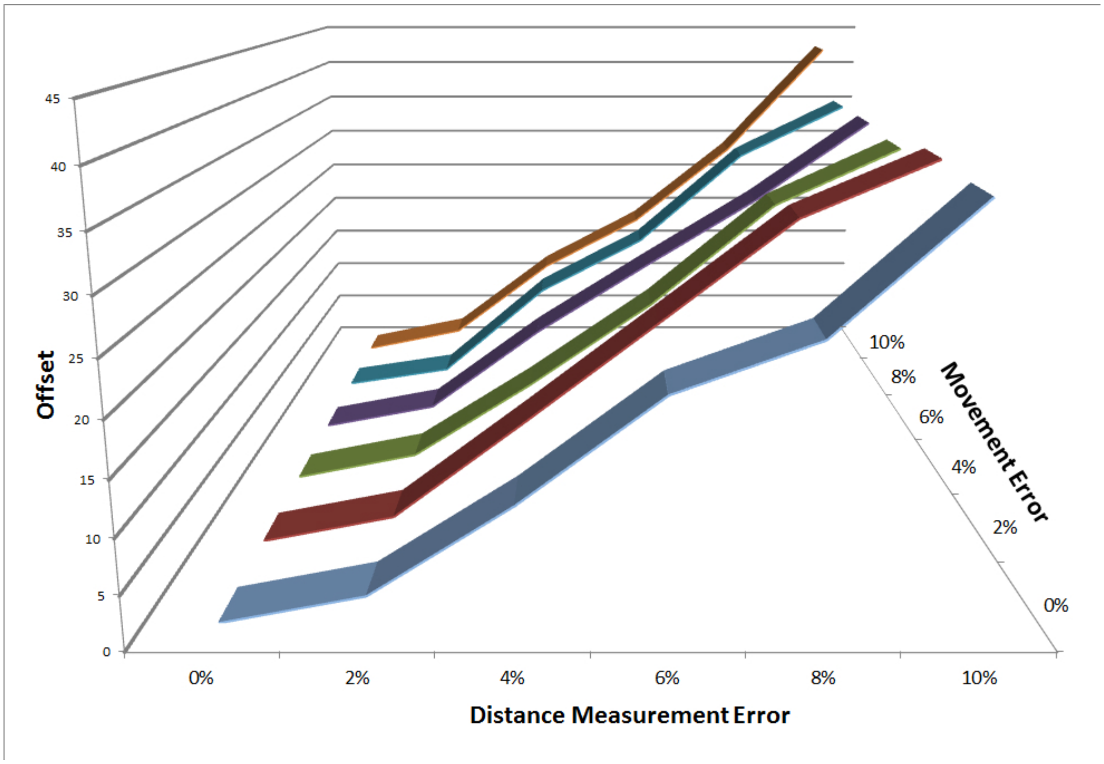

Figure 5.6: Change of localization offset for various radio ranges.

In Figure 5.6, change of offset for various values of movement and measurement errors are presented. We can observe from the figure that offset is more sensitive to the distance measurement error among radios, and less sensitive to the movement error. This can be attributed to the fact that, mobile node uses distance measurements to nodes with known positions to localize itself, therefore movement errors are not accumulated throughout the travel of the mobile node.

Chapter 6

Conclusion

In this thesis, we propose a novel mobile assisted range-based indoor localization algorithm specifically designed for multi-storey buildings. We analyze the weaknesses of quadrilateration algorithm in multi-storey indoor environments, and propose methods to overcome these weaknesses. A mobile node is used, when necessary, to aid localization, and the algorithms developed aim to minimize the interference and the travel time of the mobile node.

BIBLIOGRAPHY

[1] Akcan, H., Evrendilek, C. (2012). 'GPS-free directional localization via dual wireless radios'. *Comput. Commun.*, 35, pp. 1151–1163.

[2] Akcan, H., Kriakov, V., Brönnimann, H. and Delis., A. (2006) 'GPS-Free node localization in mobile wireless sensor networks.'*In Proceedings of the 5th ACM International Workshop on Data Engineering for Wireless and Mobile Access, Chicago, Illinois, USA (MobiDE'06)*, pp 35–42.

[3] Akcan, H., Kriakov, V., Bronnimann, H. and Delis, A. (2010). 'Managing cohort movement of mobile sensors via GPS-free and compassfree node localization'. *Journal of Parallel and Distributed Computing* 70(7), pp. 743–757.

[4] Capkun, S., Hamdi, M. and Hubaux., J.P. (2002) 'GPS-free Positioning in Mobile Ad Hoc Networks.'*Cluster Computing*, 5(2), pp. 157–167.

[5] Iyengar, R. and Sikdar., B. (2003) 'Scalable and distributed GPS free positioning for sensor networks.'*In Proceedings of the IEEE International Conference on Communications (ICC'03), Anchorage, AK, USA*, pp. 338–342.

[6] Moore, D., Leonard, J., Rus, D. and Teller., S. (2004). 'Robust distributed network localization with noisy range measurements.'*In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys'04), Baltimore, MD, USA* pp. 50–61.

[7] Park,J., Demaine,E.D. and Teller,S. (2008). 'Moving-baseline localization'. *Proceedings of Information Processing in Sensor Networks(IPSN)*, pp. 15–26.

[8] Priyantha, N. B., Balakrishnan, H., Demaine, E. and Teller., S. (2003) ‘Anchor-free distributed localization in sensor networks. ’*In Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys’03), Los Angeles, CA, USA*, pp. 340–341.

[9] Priyantha, N. B., Chakraborty, A. and Balakrishnan, H. (2000) ‘The cricket location-support system. ’*In Proceedings of the Sixth International Conference on Mobile Computing and Networking (MOBICOM’00), Boston, MA, USA,* pp. 32–43.

[10] Boukerche, A., Oliveira, H. A. B. F., Nakamura, E. F. and Loureiro. A. A. F. (2008) ‘Vehicular ad hoc networks: A new challenge for localization-based systems. ’*Computer Communications, 31(12)* pp. 2838–2849.

[11] Hightower, J. and Borriello, G. (2001). ‘Location systems for ubiquitous computing’. *IEEE Computer*, 34(8), pp. 57–66.

[12] Yemini, Y. (1979). ‘Some theoretical aspects of position-location problems’. *20th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 1–8.

[13] Saxe, J. B. (1979) ‘Embeddability of weighted graphs in k-space is strongly np-hard.’*In 17th Allerton Conf. Commun. Control Comput.*, pp 480–489.

[14] Eren, T., Goldenberg, O., Whiteley, W., Yang, Y., Morse, A., Anderson, B. and Belhumeur, P. (2004) ‘Rigidity, computation, and randomization in network localization.’*In INFOCOM volume 4*, pp 2673–2684

[15] Evrendilek, C. and Akcan, H. (2011). ‘On the complexity of trilateration with noisy range measurements’ *Communications Letters, IEEE* 15(10), pp. 1097–1099.

[16] Harmer, D., Russell, M., Frazer, E., Bauge, T., Ingram, S., Schmidt, N., Kull, B., Yarovoy, A., Nezirovic, A., Xia, L., Dizdarevic, V. and Witrisal, K. (2008). ‘Europcom: Emergency ultrawideband radio for positioning and communications ’. *In IEEE International Conference on Ultra-Wideband (ICUWB’08), volume 3* pp 85–88.

[17] Bulusu, N., Heidemann, J. and Estrin, D. (2000) 'Gps-less low cost outdoor localization for very small devices.'. *IEEE Personal Communications Magazine,7(5)*, pp 28–34.

[18] Liu, J., Zhang, Y., and Zhao, F. (2006) 'Robust distributed node localization with error management.'*In Proceedings of the 7th ACM Interational Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc?06)*, pp 250–261.

[19] Savarese C., Rabaey, J. M. and Langendoen, K. (2002) 'Robust positioning algorithms for distributed ad-hoc wireless sensor networks.'*In USENIX* , pp 317–327.

[20] Porretta, M., Nepa, P., Manara G. and Giannetti, F. (2008) 'Location, location, location.'*IEEE Vehicular Technology Magazine*, pp 20–29.

[21] Gezici, S., Tian, T., Giannakis, G., Kobayashi, H., Molisch, A., Poor, H. and Sahinoglu, Z. (2005) 'Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks.'*Signal Processing Magazine, IEEE*, 22(4), pp 70–84.

[22] Connelly, R. (2005). 'Generic global rigidity'. *Discrete & Computational Geometry* 33(4), pp. 549–563.

[23] Aspnes, J., Eren, T., Goldenberg, D.K., Morse, A.S., Whiteley W., Yang Y.R., Anderson, B.D.O., and Belhumeur, P.N. (2006). 'A theory of network localization'. *IEEE Trans Mobile Computing*, 5(12), pp. 1663–1678.

[24] Olson,E., Leonard,J.J. and Teller,S. (2006). 'Robust range-only beacon localization'. *IEEE J. Oceanic Eng.*, 31(4), pp. 949–958.

[25] Priyantha, N.B., Balakrishnan, H., Demaine, E.D., Teller, S. (2005). 'Mobile-Assisted localization in wireless sensor networks'. *IEEE INFOCOM*, pp. 172–183.

[26] Djugash, J., Singh, S, Kantor, G. and Zhang, W. (2006). 'Range-Only SLAM for robots operating cooperatively with sensor networks'. *IEEE International Conference on Robotics and Automation*, pp. 2078–2084.

[27] Dantu, K., Rahimi,M. Shah, H., Babel,S., Dhariwal,A. and Sukhatme,G.S. (2005). 'Robomote: enabling mobility in sensor networks'. *Proc. Inter. Symp. Information Processing in Sensor Networks (IPSN '05)*, pp. 404–409.

[28] Hu, l. and Evans,D.(2004). 'Localization for mobile sensor networks'. *MOBICOM*, pp 45–57.

[29] Choset,H., Konukseven,I. and Rizzi,A. (1997). 'Sensor based planing: A control law for generating the generalized voronoi graph'. *IEEE International Conference in Advanced Robotics*, pp. 333–338.

[30] Laman, G. (2002). 'On graphs and rigidity of plane skeletal structures'. *Journal of Engineering Mathematics, vol. 4*, pp. 331–340.

[31] Goldenburg, D.K., Krishnamurthy, W., Maness, W.C., Yang,Y., Young,A., Morse A.S., Savvides, A. and Anderson, B.D.O. (2005). 'Network localization in partially localizable networks'. *IEEE INFOCOM*, 1, pp. 313–326.

[32] Blumenthal L. (1953). Theory and Applications of Distance geometry. Clarendon Press, Oxford.