Anchor Free Positioning For Target Tracking in Wireless Sensor Networks

Project Report Submitted in partial fulfillment of the requirement
for the degree of

Master of Technology

in

Computer Science & Engineering

under the Supervision of

Dr. Shailendra Shukla

By

Gajendra Tyagi(132206)

Jaypee University of Information Technology
Waknaghat, Solan – 173234, Himachal Pradesh
May 2015

Certificate

This is to certify that project report entitled **"Anchor Free Positioning For Target Tracking in Wireless Sensor Networks"**, submitted by **Gajendra Tyagi** in partial fulfillment for the award of degree of Master of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision. This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Supervisor's Name:
Dr. Shailendra Shukla
Designation:
Assistant Professor(CSE)

Date:

Acknowledgement

First of all, I am very thankful to Almighty God for granting me the wisdom, health and strength for doing this work. Next, I am thankful to my supervisor **Dr. Shailendra Shukla** for providing me an opportunity to work under him. I am indebted to him for being a constant source of knowledge for me from taking of this thesis till now. He not only corrected me in the technical issues related to the localization of wireless sensor networks, but his painstakingly report correction as well as presentation helped me a lot to represent this research work in a novel way.

I owe a deep gratitude to **Dr. Hemraj Saini**, **Dr. Yashwant Singh**, **Mr. Suman Saha**, and **Mr. Ravindra Bhatt** for not only being the panel members of my thesis report evaluation, but also providing insightful comments at different times during this semester.

At last, I would like to say that all this would have been difficult to achieve without the support and patience of my parents. They have been a source of consistent motivation for this work. I love them for their perseverance and the moral character they provided to me.

Name of the student:
Gajendra Tyagi

Date:

CONTENTS

LIST OF FIGURES

ABBREVIATIONS

WSN	**W**ireless **S**ensor **N**etwork
GPS	**G**lobal **P**ositioning **S**ystem
BS	**B**ase **S**tation
TDMA	**T**ime **D**ivision **M**ultiple **A**ccess
TOA	**T**ime **O**f **A**rrival
TDOA	**T**ime **D**ifference **O**f **A**rrival
RSSI	**R**eceived **S**ignal **S**trength **I**ndicator
AOA	**A**ngle **O**f **A**rrival
MDS	**M**ulti **D**imensional **S**caling
APIT	**A**ppropriate **P**oint **I**n **T**riangle
SHARP	**S**imple **H**ybrid **A**bsolute **R**elative **P**ositioning
APS	**A**d-hoc **P**ositioning **S**ystem
MDL	**M**ulti **D**ulateration **L**ocalization
GMDL	**G**rouping **M**ulti **D**ulateration**L**ocalization
HBA	**H**ope **B**ased **A**pproach
LCB	**L**ocal **C**onnectivity **B**ased

Abstract

Wireless sensor networks are extremely being used in different environments to perform various monitoring tasks such as search, rescue, disaster relief, target tracking and a number of tasks in smart environments. In most sensor network applications, the information gathered by micro-sensors will be meaningless unless the location from where the information is obtained is known. This makes localization capabilities highly desirable in sensor networks. Till now, various localization algorithms have been proposed to fulfill the same objective. However, all of these algorithms are dependent on various factors to get the location information like GPS devices, a large number of anchor nodes, known network topology, communication cost etc. Accuracy is one more concern, which have to be look upon because it is directly dependent on the above mentioned factors. So, keeping all these things in mind, the main aim is to reduce the dependency on the anchor nodes without affecting the accuracy. Previous researches have shown the success of virtual coordinates in a wide variety of ad hoc and sensornet environments for the routing purpose. So this thesis report introduced the concept of virtual coordinates to make localization system anchor free, as well as proposed a new algorithm for the detection of boundary nodes of the network and the holes in the network having less percentage of false positivity.

Keywords: *Localization algorithms, Triangulation, Trilateration, Multidimensional Scaling, virtual coordinates, Holes.*

CHAPTER 1

INTRODUCTION

Wireless sensor network is a very vast and emerging arena in the field of computer science because of its wide applicability in the real world applications [2]. Scenarios like search, rescue, disaster relief, target tracking, telemedicine, weather forecasting, battlefield surveillance, battle damage assessment, home applications, detecting and monitoring car thefts and many more are handled effectively with the integration of wireless sensor network. In most of these sensor network applications, the information gathered by micro-sensors will be meaningless unless the location from where the information is obtained is known. Moreover, the location information should be accurate to get the quality information .This makes localization capabilities highly desirable in sensor networks. However, a lot of research work has been done to get the accurate position information of sensor nodes and the researchers have already dealt with various localization bottlenecks. Most of these algorithms are dependent on various factors to get the location information like GPS devices, a large number of anchor nodes, known network topology, communication cost etc. So, there is still a scope of improvement in these existing localization schemes in terms of making GPS device independent, reduction in number of anchor nodes, cost. Before moving on further, there are some definitions related to wireless sensor networks, applications and some key issues are discussed below to get better understanding of the need of localization.

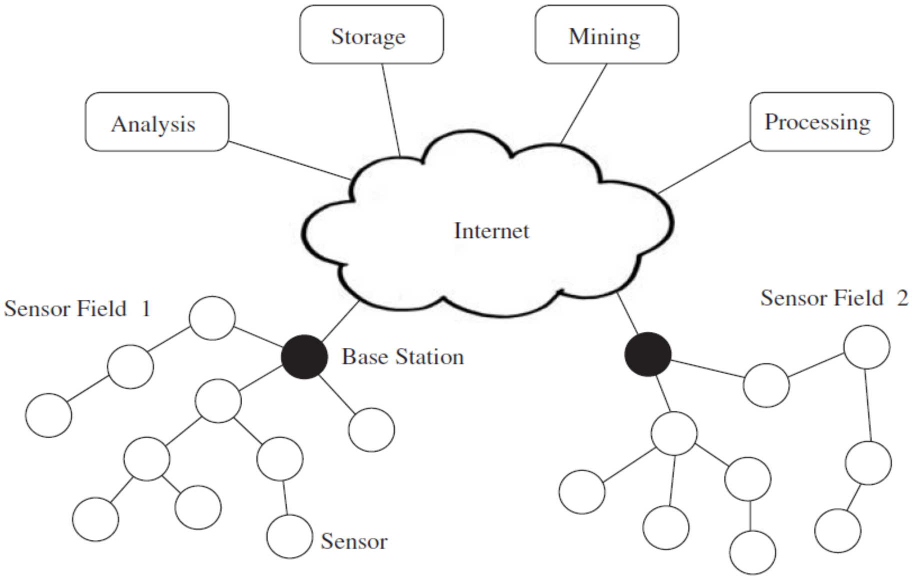


Figure 1.1: Wireless Sensor Networks

1.1 Definitions and Background

1.1.1 Sensing and Sensors

A technique that is used to gather information about any physical object, process or an event(i.e., changes in state such as a drop in temperature or pressure) is called sensing [1]. And the device which is used to perform such a sensing task is known as sensor. From a technical perspective, a sensor is a device that translates parameters or events in the physical world into signals that can be measured and analyzed.

1.1.2 Wireless Sensor Networks

While many sensors connect to controllers and processing stations directly (e.g., using local area networks), an increasing number of sensors communicate the collected data wirelessly to a centralized processing station. This is important since many network applications require hundreds or thousands of sensor nodes, often deployed in remote and inaccessible areas. Therefore, a wireless sensor has not only a sensing component, but also on-board processing, communication, and storage capabilities. With these enhancements, a sensor node is often not only responsible for data collection, but also for in-network analysis, correlation, and fusion of its own sensor data and data from other sensor nodes. When many sensors cooperatively monitor large physical environments, they form a wireless sensor network (WSN). Sensor nodes communicate not only with each other but also with a base station (BS) using their wireless radios, allowing them to disseminate their sensor data to remote processing, visualization, analysis, and storage systems [1]. For example, Figure 1.1 shows two sensor fields monitoring two different geographic regions and connecting to the Internet using their base stations.

1.2 Application areas

Sensor networks may consist of many different types of sensors such as seismic, low sampling rate magnetic, thermal, visual, infrared, acoustic and radar, which are able to monitor a wide variety of ambient conditions that include temperature, humidity, vehicular movement, lightning condition, pressure, soil makeup, noise levels, etc.

Sensor nodes can be used for continuous sensing, event detection, event ID, location sensing, and local control of actuators. The concept of micro-sensing and wireless connection of these nodes promise many new application areas. We categorize the applications into military, environment, health, home and other commercial areas. It is possible to expand this classification with more categories such as space exploration, chemical processing and disaster relief.

1.2.1 Military applications

Wireless device networks will be an integral a part of military command, control, communications, computing, intelligence, police work, and intelligence and targeting systems. Leaders and commanders will perpetually monitor the standing of friendly troops, the condition and therefore the availableness of the instrumentation and therefore the ammunition in a very tract by the utilization of device networks. crucial terrains, approach routes, methods and straits will be quickly coated with device networks and closely watched for the activities of the opposing forces. because the operations evolve and new operational plans square measure ready, new device networks will be deployed anytime for battle field police work. device networks will be incorporated into steerage systems of the intelligent ammunition. simply before or when attacks, device networks will be deployed within the topographic point to assemble the damage assessment information.

1.2.2 Environmental applications

Some environmental applications of sensor networks embody pursuit the movements of insects, birds, and little animals; watching environmental conditions that have an effect on farm animals and crops, irrigation, macro instruments for large-scale Earth watching and planetary exploration, chemical/biological detection, exactitude agriculture, biological, Earth, and environmental watching in marine, soil, and atmospherically contexts, forest re detection, earth science or geology analysis, flood detection.

1.2.3 Health applications

Wireless sensor networks has a wide application domain in the medical field. It is widely used in patient monitoring, drug administration, diagnostics in hospitals.

It is also used in, tele-monitoring of human physiological data, and tracking and monitoring doctors and patients inside a hospital.

1.2.4 Home applications

As technology advances, smart sensor nodes and actuators can be buried in appliances, such as vacuum cleaners, micro-wave ovens, refrigerators, and VCRs. These sensor nodes inside the domestic devices can interact with each other and with the external network via the Internet or Satellite. They allow end users to manage home devices locally and remotely more easily.

1.3 Key Issues in Wireless Sensor Networks

Some of the important issues in WSNs are stated below:

1. **Energy Efficiency:** Sensor nodes have limited battery capacity. This puts a constraint for other applications and on the lifetime of sensor node. Major sources of battery drainage include: (i) continuous sensing, (ii) transmission and reception modes of radio. Therefore, to increase the lifetime in unattended environments, efficient algorithms should be developed at each layer of WSN in concern with the less energy utilization. This includes techniques of data compression, data fusion (removal of data redundancy), rotation of cluster heads, and adaptive mechanisms for radio operations.
2. **Routing:** Topology of WSN changes too frequently; as new nodes are added or some nodes die due to meager resources. Therefore, to increase the connectivity, coverage, and remain updated of network topology, neighbour information should be disseminated timely. Furthermore, transmitting node should identify the best reliable shortest path to the sink node/base station. Therefore, routing serves as a bottleneck in overall efficiency of WSN.

3. **Time Synchronization:**Synchronizing time in sensor nodes serves as a basic prerequisite for various applications and protocols such as Time division multiple access (TDMA), Time difference of arrival (TDoA), Time of arrival (ToA) and so on. Basic property of WSNs, i.e., co-operation in communication, computation, sensing and actuation of different nodes solely depends on the time synchronization among nodes.
4. **Fault-Tolerance:** Reliability in WSNs is oftenly affected by various faults arising from environmental hazards, battery depletion, hardware malfunctioning and so on. Individual node failures should not affect the global performance of WSNs. This rate of failure may be high in harsh or hostile environments. In such cases, intended purpose of WSN is achieved by techniques such as load balancing, etc. Nodes should have the capability of self-testing, self-calibrating, self-recovering and so on.
5. **Localization:**For robust WSN, localization of nodes is one of the most important issue. Information sensed by a sensor node becomes useful only when its geographical location is tagged. Geographical routing is possible only after the localization, and other issues like spatial querying and load balancing can also be achieved.

For smooth functioning of WSNs each issue needs deep investigation. Some of these issues like synchronization, localization and data gathering needs much more attention. This is because these not only help in attaining the basic function of WSNs but also serve as prerequisite for other applications. In this thesis, we have concentrated on the localization issues in WSNs.

1.4 Motivation

Applications like telemedicine, weather forecasting, disaster monitoring, battlefield surveillance requires an exact location statistics of the sensors nodes to get the quality information. Suppose, in the case of battlefield surveillance, if the position of deployed sensor nodes for the surveillance purpose is wrongly estimated, the result of this mistake could be very cumbersome for the country benefit. Similarly in the case of telemedicine, the wrong estimate of deployed sensor nodes around a patient could give the wrong information about him to the doctor. However, this problem can be tackled by manual configuration of nodes in the sensor network. But again, it is also not feasible in the case of hundreds and thousands of deployable nodes. Another possible solution is to integrate GPS devices in each sensor node. It is also not possible because of the following reasons.

- In the presence of dense forests, mountains or other obstacles that block the line-of-sight from GPS satellites, GPS cannot be implemented.
- The power consumption of GPS will reduce the battery life of the sensor nodes and also reduce the effective lifetime of the entire network.
- In a network with large number of nodes, the production cost factor of GPS is an important issue.
- Sensor nodes are required to be small. But the size of GPS and its antenna increases the sensor node form factor.

  So an alternate solution of GPS is needed which should be cost effective, rapidly deployable and can operate in diverse environments.

A few applications indicating the importance of localization in WSNs is listed below:

- Sensors gather vital security related parameters such as radio communication, vigorous movements in an surveillance area, and report these to the back-end security system (a sink node). A prompt action by security personnel is possible only if location information is provided with the sensed information.
- On some occasions, some nodes may die due to the battery drainage or by physical forces. In such cases, new nodes to be injected or battery replacements can be achieved efficiently by adopting geographic routing rather than physical routing schemes. Geographic routing eases task of locating a faulty node as compared to physical routing.
- Location information is also used to divide the WSN into different clusters to facilitate collaborative processing and hierarchical routing. For each cluster, one node is chosen as cluster head which remains responsible for cluster inter connectivity and state maintenance.

1.5 Objective

Sensor nodes are low cost devices. Use of GPS to obtain location information will increase their cost. An alternative to the use of GPS is to obtain location information through localization algorithms. Use of localization algorithms mandate the deployment of a few location aware node. The remaining nodes are localized with the help of these location aware nodes.So the objective of this thesis includes:

- To reduce the dependency on location-aware nodes for localization.
- To develop a localization algorithm with no extra hardware cost.
- To reduce the error in localization, and localization time.

1.6 Organization of The Report

The report is organized into following chapters:

- **Chapter 1:** This chapter includes the basic introductory part of the thesis in which related definitions, applications and some important keys related to wireless sensor networks are mentioned.
- **Chapter 2:** All the necessary components and basics of a localization system are described in this chapter.
- **Chapter 3:** This chapter includes the detailed literature survey regarding the localization in wireless sensor network.
- **Chapter 4:** In this chapter, a new method for localization system which is based upon virtual coordinates is proposed and the necessary system model and assumptions are described in it.
- **Chapter 5:** Simulation and results along with the comparison with an existing approach made on the basis of false positivity are shown in this chapter.
- **Chapter 6:** Chapter 6 contains the future work on the proposed approach and concludes our work.

CHAPTER 2

BASICS OF LOCALIZATION SYSTEM

2.1 Localization System

The objective of localization is to find the physical coordinates of sensor nodes. These coordinates can be either global or relative. Localization is achieved with the help of a few location aware nodes usually referred as seeds/anchor nodes/beacon nodes. These anchor nodes are either manually programmed with their physical position or use the global positioning system (GPS) to determine their location.

There are three different stages in localization as shown in Figure - 2.1. They are: (i) distance/angle estimation between the nodes, (ii) position calculation of a single node, (iii) a localization algorithm - used for localization of whole network. Different techniques with varying accuracy and complexity exist at each stage. Localization error and localization time is the cumulative error and time respectively of each stage. These stages are explained in detail in subsequent sections.

2.1.1 Distance/Angle Estimation

This refers to the measurement of distance or angle between the transmitter and receiver node. Distance/Angle estimation is the pre-requisite for remaining two

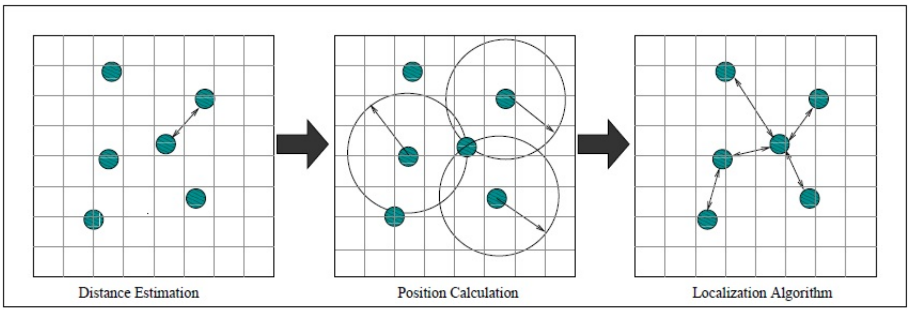


Figure 2.1: Three components of localization system.

phases of localization. Different techniques for distance/angle estimation include: time of arrival (ToA), time difference of arrival (TDoA), received signal strength indicator (RSSI), and angle of arrival (AoA).

2.1.1.1 Time of Arrival

The concept behind the time of arrival (ToA) method (also called time of flight method) is that the distance between the sender and receiver of a signal can be determined using the measured signal propagation time and the known signal velocity. For example, sound waves travel 343 m/s (in 20 C), that is, a sound signal takes approximately 30 ms to travel a distance of 10 m. In contrast, a radio signal travels at the speed of light (about 300 km/s), that is, the signal requires only about 30 ns to travel 10 m. The consequence is that radio based distance measurements require clocks with high resolution, adding to the cost and complexity of a sensor network. The one-way time of arrival method measures the one-way propagation time, that is, the difference between the sending time and the signal arrival time (figure 2.2(a), and requires highly accurate synchronization of the clocks of the sender and receiver. Therefore, the two-way time of arrival method is preferred, where the round-trip time of a signal is measured at the sender device(figure 2.2(b)). In summary, for one-way measurements, the distance

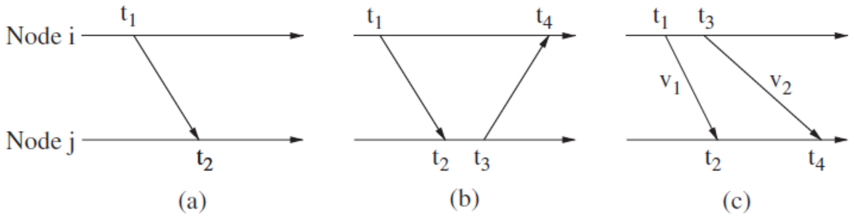


FIGURE 2.2: Comparison of different ranging schemes (one-way ToA, two-way ToA, and TDoA).

between two nodes i and j can be determined as:

$$dist_{i,j} = (t_2 - t_1) \times v \tag{2.1}$$

where t_1 and t_2 are the sending and receive times of the signal (measured at the sender and receiver, respectively) and v is the signal velocity. Similarly, for the two-way approach, the distance is calculated as:

$$dist_{i,j} = (t_4 - t_1) - (t_3 - t_2) \times v/2 \tag{2.2}$$

where t_3 and t_4 are the sending and receive times of the response signal. Note that with one way localization, the receiver node calculates its location, whereas in the two-way approach, the sender node calculates the receiver's location. Therefore a third message will be necessary in the two-way approach to inform the receiver of its location.

2.1.1.2 Time Difference of Arrival

The time difference of arrival (TDoA) approach uses two signals that travel with different velocities (Figure 2.2(c)). The receiver is then able to determine its location similar to the ToA approach. For example, the first signal could be a radio signal (issued at t_1 and received at t_2), followed by an acoustic signal (either immediately or after a fixed time interval $t_{wait} = t_3$-t_1). Therefore, the receiver can determine the distance as:

$$dist = (v_1 - v_2) \times (t_4 - t_2 - t_{wait}) \tag{2.3}$$

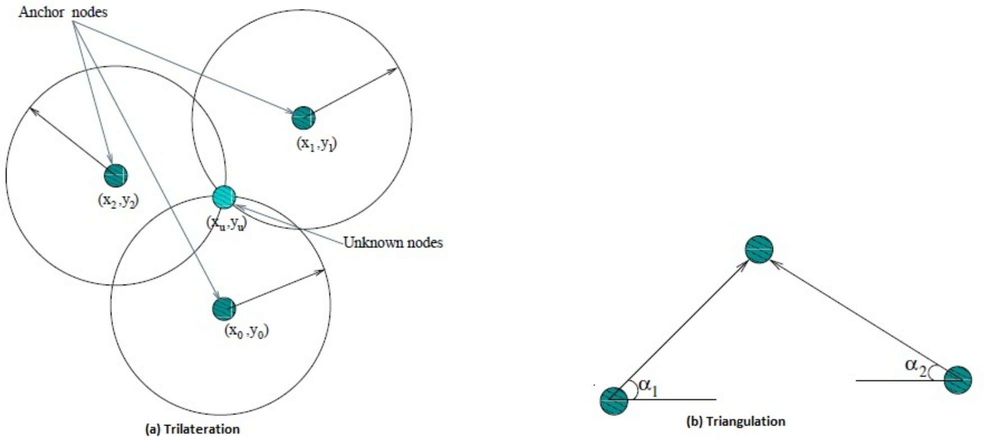
TDoA-based approaches do not require the clocks of the sender and receiver to be synchronized and can obtain very accurate measurements. The disadvantage of the TDoA approach is the need for additional hardware, for example, a microphone and speaker for the above example.

2.1.1.3 Angle of Arrival

Another technique used for localization is to determine the direction of signal propagation, typically using an array of antennas or microphones [1]. The angle of arrival (AoA) is then the angle between the propagation direction and some reference direction known as orientation For example, for acoustic measurements, several spatially separated microphones are used to receive a single signal and the differences in arrival time, amplitude, or phase are used to determine an estimate of the arrival angle, which in turn can be used to determine the position of a node. While the appropriate hardware can obtain accuracies within a few degrees, AoA measurement hardware can add significantly to the size and cost of sensor nodes.

2.1.1.4 Received Signal Strength

The concept behind the received signal strength (RSS) method is that a signal decays with the distance traveled.Acommonly found feature in wireless devices is a received signal strength indicator (RSSI), which can be used to measure the amplitude of the incoming radio signal [1]. Many wireless network card drivers readily export RSSI values, but their meaning may differ from vendor to vendor and there is no specified relationship between RSSI values and the signal's power levels. Typically, RSSI values are in the range of 0 . . . RSSI-Max, where common



(a) Trilateration

(b) Triangulation

Figure 2.3: Trilateration and Triangulation

values for RSSI-Max are 100, 128, and 256. In free space, the RSS degrades with the square of the distance from the sender. More specifically, the Friis transmission equation expresses the ratio of the received power Pr to the transmission power P_t as:

$$P_r/P_t = Gt \times Gr \times \lambda^2/((4 \times \pi)^2 \times R^2) \tag{2.4}$$

where G_t is the antenna gain of the transmitting antenna and G_r is the antenna gain of the receiving antenna. In practice, the actual attenuation depends on multipath propagation effects, reflections, noise, etc., therefore a more realistic model replaces R^2 in Equation (4) with R^n with n typically in the range of 3 and 5.

2.1.2 Position Calculation

Techniques used to estimate a node's location are trilateration, multilateration, and triangulation. Estimated distance and the position of anchor nodes is used to estimate a node's location.

2.1.2.1 Trilateration/Multilateration

Trilateration is a geometric technique used to determine the location of an unknown node with the help of three location aware nodes/anchor nodes. It uses distance between the anchor nodes and the unknown node. A pictorial view of this geometric technique for localizing an unknown node $(x_u;\ y_u)$ with anchor nodes $(x_i;\ y_i)$ is shown in Figure - 2.3. Distance measurements are never perfect. As a result it is difficult to get an accurate location. Distance measurement from more than three anchors is known as multilateration. This technique can be used to get a unique location.

We illustrate multilateration in a 2-dimensional space with known distances between anchor nodes and an unknown node as

$$d_1^2 = (x_1 - x_u)^2 + (y_1 - y_u)^2 \tag{2.5}$$

$$d_2^2 = (x_2 - x_u)^2 + (y_2 - y_u)^2 \tag{2.6}$$

$$.$$

$$.$$

$$.$$

$$d_n^2 = (x_n - x_u)^2 + (y_n - y_u)^2 \tag{2.7}$$

Subtracting equation (2.5) from (2.6) .. (2.7) gives

$$d_2^2 - d_1^2 = x_2^2 - x_1^2 - 2(x_2 - x_1)x_u + y_2^2 - y_1^2 - 2(y_2 - y_1)y_u \tag{2.8}$$

$$d_3^2 - d_1^2 = x_3^2 - x_1^2 - 2(x_3 - x_1)x_u + y_3^2 - y_1^2 - 2(y_3 - y_1)y_u \tag{2.9}$$

$$.$$

.

.

$$d_n^2 - d_1^2 = x_n^2 - x_1^2 - 2(x_n - x_1)x_u + y_n^2 - y_1^2 - 2(y_n - y_1)y_u \tag{2.10}$$

Rearranging, (2.8) .. (2.10) in matrix form, we obtain

$$\begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \\ . & \\ . & \\ .x_n - x_1 & y_n - y_1 \end{bmatrix} \begin{bmatrix} x_u \\ y_u \end{bmatrix} = (1/2) \begin{bmatrix} x_2^2 - y_2^2 - d_2^2 - (x_1^2 + y_1^2 - d_1^2) \\ x_3^2 - y_3^2 - d_3^2 - (x_1^2 + y_1^2 - d_1^2) \\ . \\ . \\ .x_n^2 - y_n^2 - d_n^2 - (x_1^2 + y_1^2 - d_1^2) \end{bmatrix}$$

Above matrix can be rewritten as

$$Au = b \tag{2.11}$$

Therefore, u can be derived as

$$u = (A^T A)^{-1} A^T b \tag{2.12}$$

2.1.2.2 Triangulation

Triangulation is a geometric technique that uses the trigonometry laws of sine and cosines on the angles of incoming signal α to estimate a unique location. A geometric computation of this is shown in Figure - 2.3(b).AoA measurement requires bulkier and expensive hardware such as multi-sectored antennae. This makes triangulation unsuitable for small sensor nodes.

2.1.3 Localization algorithm

Localization algorithm is the last and most important stage of localization system. It utilizes the information collected in previous two stages. It defines how this information can be transformed to localize sensor nodes cooperatively. Cooperative localization refers to the collaboration between sensor nodes to find their locations. Mostly, accuracy of this stage is effected by the ranging method, deployment environment, and the relative geometry of unknown nodes to the anchor nodes.

Broadly, localization algorithms in WSNs can be divided into two categories: (i) centralized, and (ii) distributed. Centralized localization requires the migration of internode ranging and connectivity data to a sufficiently powerful central base station and then the migration of resulting locations back to respective nodes [3]. Centralization allows an algorithm to undertake much more complex mathematics than is possible in a distributed setting. Whereas in distributed localization, all the relevant computations are done on the sensor nodes themselves and the nodes communicate with each other to get their positions in a network.

On the basis of ranging method used, localization algorithms for WSNs can be broadly categorized into two types: (i) range based, and (ii) range free. Range based localization algorithms use the range (distance or angle) information from the beacon node to estimate the location [4]. Several ranging techniques exist to estimate an unknown node distance to three or more beacon nodes. Based on the range information, location of a node is determined. Some of the range based localization algorithm includes: Received signal strength indicator (RSSI), Angle of arrival (AoA), Time of arrival (ToA), Time difference of arrival (TDoA).

Range-free localization algorithms use connectivity information between unknown node and landmarks. A landmark can obtain its location information using GPS or through an artificially deployed information. Some of the range-free localization algorithm includes: Centroid, Appropriate point in triangle (APIT), and DV-HOP. In centroid the number of beacon signals received from the pre-positioned beacon nodes is counted and localization is achieved by obtaining the centroid of received beacon generators. DV-HOP uses the location of beacon nodes, hop counts from beacons, and the average distance per hop for localization. A relatively higher ratio of beacons to unknown nodes, and longer range beacons are required in APIT. They are also more susceptible to erroneous reading of RSSI.

Range-based algorithms achieve higher localization accuracy, at the expense of hardware cost and power consumption. Range-free algorithms have lower hardware cost and are more efficient in localization. A brief review of different localization algorithms proposed in the literature for wireless sensor networks is presented in the next chapter.

CHAPTER 3

RELATED LITERATURE SURVEY

3.1 Localization algorithms

Localization algorithm is the last and the most important part of a localization system. Broadly, there are two categories of localization algorithms in WSNs: (i) centralized, and (ii) distributed. In centralized localization system, all the data collected by nodes is migrated to a sufficiently powerful central base station which then respond back to the nodes with their resulting locations [3]. In distributed localization, all the sensor nodes takes equal participation in the location computation, and communicate this location with each other. If we consider the ranging method used, there are again two classifications:(i) range based, and (ii) range free. Details of these classes are already described in the previous chapter.

This paragraph summarizes the state of art related to localization algorithms. Shang et al. [5] proposed a centralized range based algorithm which utilizes the MDS(multidimensional scaling) for the localization purpose. Disadvantage with this approach is the use of MDS which is slow, even if we have sufficient number of anchor nodes. Huang et al. [6] introduced a range free technique which utilizes the triangle structures to divide the whole region. Shortcoming of this method is its requirement of high number of beacon nodes. Savvides et al. [7] came up with a collaborative multilateration approach that consists of a set of mechanisms that enables nodes found several hops away from location aware beacon nodes collaborate with each other to estimate their locations with high accuracy. A lot of message passing and computational cost is its major drawback. Priyantha et al. [8] proposed AFL(anchor free localization) algorithm. In this algorithm, nodes start from a random initial coordinate assignment and converge to a consistent solution using only local node interactions. This algorithm is susceptible to local minima. Cheng et al. [9] integrated the two techniques multidimensional scaling and proximity based mapping. But it doesn't work when there are limited number of anchor nodes. Ahmed et al. [10] proposed a Simple Hybrid Absolute Relative Positioning (SHARP) which uses multidimensional scaling (MDS) and Ad-hoc Positioning System (APS) for localization. Again the use of multidimensional scaling reduces its efficiency. Kannak et al. [11] proposed an approach based on simulated anneling. when the node density is low, it is possible that a node is flipped and still maintains the correct neighborhood. In this situation, the proposed algorithm fails to identify the flipped node. Lee et al.[12] introduced the multi duolateration technique in which four anchor nodes are required to initiate the localization process. Additional hardware requirement for jumper setting is its main drawback.Detailed description of all localization algorithms is given below.

3.1.1 Title: Localization from mere connectivity [5]

Authors: Y Shang, W Ruml, Y Zhang
Published in: MobiHoc
Year: 2003
Overview

It is a Centralised range based algorithm. 3 main steps of this algorithm are:

1. First in the scheme, the shortest path between all the possible pairs of nodes within that region is computed using the dijkstra's or floyd's algorithm for

shortest path. Then the distance matrix is constructed for MDS using the shortest path distances.

2. In the next step, distance matrix is being processed using classical MDS, to retrieve the rest 2 (or 3) largest Eigen values and Eigen vectors so as to construct a 2D or 3D map which provide the location for each node. However these locations can be accurate with respect to each other and the map will be rotated and flipped relative to the actual node position.
3. On the basis of the position of the appropriate anchor nodes (3 or more for 2-D, 4 or more for 3-D), transformation from the relative map to the absolute map on the foundation of actual position of anchors which consists of scaling, rotation and reaction. The aim is to reduce the sum of squares of the errors between the actual position and the transformed one.

Strength: There is no need of anchor nodes to initiate the process. Relative map can be build even without anchor nodes and next with three or more anchor nodes, Then a transformation is done from relative map to absolute coordinates.

Weakness: An initial global information of the network is mandatory to start with this algorithm and centralized computation is done which is not a good point for a very large number of sensor nodes. Performance of classical MDS is not good when large no. of anchor nodes are available, means if there already exists a large number of anchor nodes, than in comparison to other algorithms which computes the location on the basis of anchor nodes, MDS-MAP is quite slower.

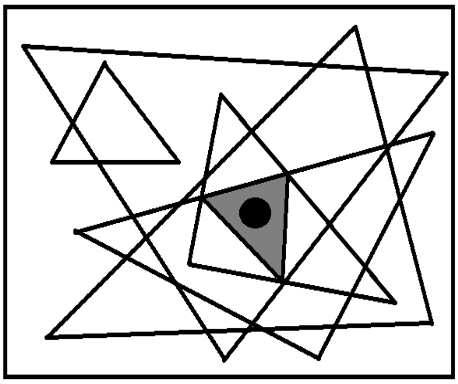


Figure 3.1: Area-based APIT Algorithm Overview.

3.1.2 Title:Range-Free Localization Schemes for Large Scale Sensor Networks [6]

Authors:T He, C Huang, BM Blum, JA Stankovic
Published in: Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking (MobiCom'03)
Year:2003

Overview

This paper proposed an area based range free localization scheme, named as APIT. There is a requirement of heterogeneous network of sensor nodes containing few nodes as anchor nodes (nodes with known location). As shown in figure 5, APIT employs a novel area based approach by dividing the region into triangular sections between beaconing nodes. With the use position of anchor nodes, the diameter of projected area in which the node reside can be reduced and an accurate location estimation can be done.

There are four steps in the APIT algorithm: (1) Beacon exchange, (2) PIT Testing, (3) APIT aggregation and (4) COG calculation. Each node perform these steps in a distributed manner.

PIT Testing: To test a node that if it is inside or outside a particular triangle, PIT testing is done over the node. Two types of PIT testing are described here. First one is perfect PIT Test, in which movement of node is considered. However in approximate PIT test, information of neighbors is considered to make the decision. PIT test theory of both the types is given below.

Perfect P.I.T Test Theory: If it is possible that a point next to M is further/-closer to points A, B, and C at the same time, then M is exterior of $\triangle$ ABC. Else, M is inside $\triangle$ ABC.

Approximate P.I.T Test: If there is no neighbor of M which is further from or closer to all the anchors A, B and C concurrently, Then M accepts that it is inside $\triangle$ ABC. Else, accepts that it is outside the triangle.

PIT Aggregation: Results are aggregated in APIT using a grid SCAN algorithm (Figure 3.2). Maximum area is represented by a grid array, in which it is assumed that a node will reside. If the node position comes to inside a particular region using the APIT test, then the value corresponding to that grid region is incremented otherwise decremented. Then the maximum overlapping area (e.g. the grid area with value 2 in Figure 4), is calculated using the resulting information. Then the center of gravity is calculated for position approximation.

Strength: The strength of APIT lies in its simplicity and ease of implementation.Cost involved in computation (message passing) is not very much as compared to MDS (n^3 time complexity for n number of nodes).

Weakness: APIT requires a high ratio of beacons to nodes and longer range beacons to get a good position estimate. For low beacon density this scheme will

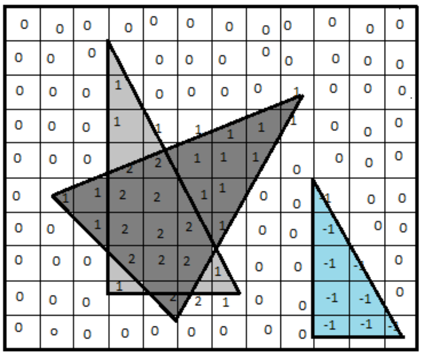


Figure 3.2: SCAN Approach

not give accurate results.Also approximate PIT test is not very accurate always. It could give wrong results also which is mentioned in the paper.

3.1.3 Title: The Bits and Flops of the N-hop Multilateration Primitive For Node Localization Problems [7]

Authors:A Savvides, H Park, MB Srivastava

Published in: WSNA Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications.

Year: 2002

Overview Author proposed a collaborative multilateration method in which a number of mechanisms are used to find the location of nodes which are several hops away from known location beacon nodes that is also with high accuracy. There are three phases in this approach.

Forming Collaborative sub trees: A computation sub tree constitutes a configuration of unknowns and beacons for which the solution of the position estimates of the unknown can be uniquely determined. The requirement of one-hop multilateration for an unknown node is that it is within the range of at least three beacons. If the beacons are not directly connected to the nodes but lies inside a

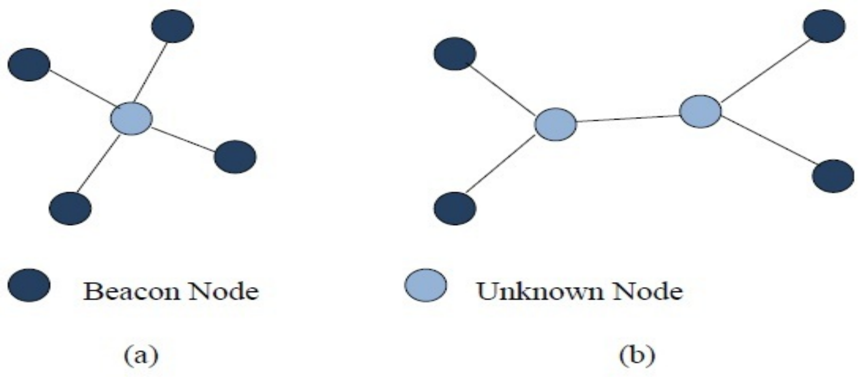


FIGURE 3.3: (a) One-hop Multilateration (b) Two-hop Multilaretation.

two hop radius from the unknown location node, then it is represented by a two hop multilateration.

Obtaining initial estimates: This phase is explained by the help of Figure 3.4. In Figure 3.3 A and B are beacons where C is the unknown node. If a is distance between A and C, then a will bound the x coordinate of C to the left and to the right of the x coordinate of A, x_A–a and x_A+a. Similarly beacon B which is two hops away from C, bounds the coordinate of C within x_B-(b+c) and x_B+(b+c). by knowing the information, C can determine that its x coordinate bounds with respect to beacons A and B are x_B+(b+c) and x_A-a. The similar method is applied on the y coordinates. Bounds on x and y coordinates are combined by, and a final bounding box of the region is obtained.

Position refinement: Third step is the position refinement step. Kalman fletr implementation is used to refine the initial node positions. Now as most unknown nodes are not directly connected to beacons, they use the initial estimates of their neighbors as the reference points for estimating their locations. As soon as an unknown node computes a new estimate, it broadcasts this estimate to its neighbors, and the neighbor use it to update their own position estimates.

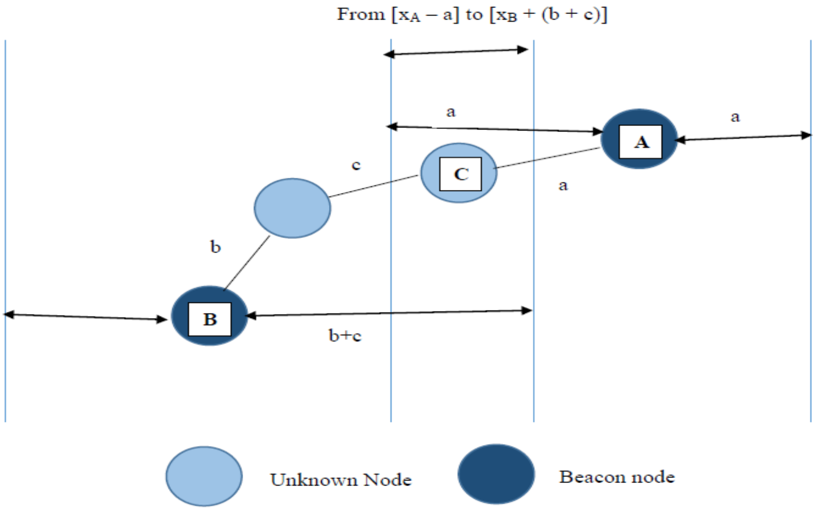


FIGURE 3.4: X coordinates bounds for C using initial estimates

Strength: Enjoys all the advantages of distributed approaches and it accurately estimate the locations of nodes that are not directly connected to anchor nodes.

Weakness: A lot of message passing and computational cost is involved.A lot of anchor nodes are required to compute the position of unknown nodes.

3.1.4 Title: Anchor-free distributed localization in sensor networks [8]

Authors:NB Priyantha, H Balakrishnan, E Demaine
Published in: Proceedings of the 1st international conference on Embedded networked sensor systems.
Year: 2003

Overview The authors propose an Anchor Free Localization (AFL) algorithm where nodes start from a random initial coordinate assignment and converge to a consistent solution using only local node interactions. The algorithm proceeds in two phases and it assumes the nodes as point masses connected with strings and use force-directed relaxation methods to converge to a minimum-energy configuration. The first phase is a heuristic that produces a graph embedding which looks similar to the original embedding. The authors assume that each node has a unique identifier and the identifier of node i is denoted by ID_i and the hop-count between nodes i ad j is the number of nodes $h_{i,j}$ along the shortest path between i and j. The algorithm first elects the five reference nodes in which four nodes n_1, n_2, n_3 and n_4 are selected such that they are on the periphery of the graph and the pair (n_1, n_2) is roughly perpendicular to the pair (n_3, n_4). The node n_5 is elected such that it is in the middle of the graph. At first the node with smallest ID is selected. Next the reference node n_1 is selected to maximize $h_{1,2}$. After that n_3 is selected to minimize — $h_{1,3}$ – $h_{2,3}$ — and the tie-breaking rule is to pick the node that minimizes $h_{1,3} + h_{2,3}$. In the next stage n_4 is selected to minimize — $h_{1,4}$ – $h_{2,4}$ — and the ties are broken by picking the node that maximizes $h_{3,4}$. Next n_5 is selected which minimizes — $h_{1,5}$ – $h_{2,5}$ — and from contender nodes pick the node that minimizes — $h_{3,5}$ – $h_{4,5}$ —. So node n_5 is the center of the graph and node n_1, n_2, n_3, n_4 becomes the periphery of the graph. Now for all nodes ni the heuristics uses the hop-counts $h_{1,i}$, $h_{2,i}$, $h_{3,i}$, $h_{4,i}$, and $h_{5,i}$ from the chosen reference nodes to approximate the polar coordinates (ρ_i, Θ_i) where

$$\rho_i = h_{5,i} * R \tag{3.1}$$

$$\Theta_i = tan^{-1}[(h_{1,i}–h_{2,i})/(h_{3,i}–h_{4,i})] \tag{3.2}$$

In the optimization, the magnitude of F_i for each node n_i is zero and the global energy of the system E is also zero and the algorithm converges.

Strength: Extensive simulations show that the proposed algorithm outperforms incremental algorithm by both being able to converge to correct positions and by being significantly more robust to errors in local distance estimate.

Weakness: limitation of this approach is that the algorithm is susceptible to local minima.

3.1.5 Title:Localization in sensor networks with limited number of anchors and clustered placement [9]

Authors: King-Yip Cheng, King-Shan Lui and Vincent Tam
Published in: Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE
Year: 2007

Overview There are two core localization technique are involved in this approach named as multidimensional scaling and proximity based map. Initially some primary anchor nodes are denoted from the set of anchor nodes. Then some anchor nodes are denoted as secondary anchors which localize themselves using multidimensional scaling. Remaining nodes, which are neither primary nodes nor secondary nodes are denoted as the normal nodes. Localization of these normal nodes is done in the second phase using the proximity distance mapping. An invitation packet containg the unique ID, a counter which is initialized to zero and factor k_s which is used to control the number of secondary nodes is sent to one of its neighbors. Bernoulli trial having success rate of p is performed by those normal nodes which receives the invitation packet. If the result is true, then it changes itself to secondary node, and increment the counter by 1. Same process is repeated until it reaches the maximum value ks. Same process is then repeated by primary anchor nodes. Invitation packets are now sent by primary anchor nodes which contains ID and coordinates to all of its neighbors. It also contains a counter to the hop that the packet has travelled which is named as proximity. Secondary anchor nodes also repeats the same process but the left the coordinate field blank. Unique ID and proximity values will then be stored by each node which they have received in the packets. Is the packet is a repeated packet, means if it has received before, then the proximity value is checked again, and if it is larger, then the packet will be dropped. Else its value is updated and the packet will be forwarded to the neighbors. Thus, only the shortest path is reflected by the proximity value. When an anchor node knows about the distance to all other anchors, proximities are sent by it to other anchors, so that others can also do the same thing. In this way, all the nodes will know about their proximities information between every pair of anchors. Now classical MDS can be applied by secondary nodes to know their location.

So after the first phase, MDS provides the proximities of primary and secondary anchors and a proximity distance mapping is calculated. These position estimates and the proximities are then distributed to the normal nodes so then they can also find out their location by processing the proximity vector.

Strength: Computation cost is minimum in this scheme. The complexity of Classical MDS is O(m^3)where m is the number of nodes. . For PDM, complexity is O(n^3)where n is number of anchors. As it is a composition of MDS and PDM. So it has a complexity of O(mn^3)where mn is the total number of primary and secondary anchors.

Weakness: It is not a good approach with limited number of anchors.

3.1.6 Title: SHARP: A New Approach to Relative Localization in Wireless Sensor Networks [10]

Authors: AA Ahmed, H Shi, Y Shang

Published in: "Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference"

Year: 2005

Overview This approach also uses the multidimensional scaling with the combination of Ad-hoc positioning system. That's why it is named as Hybrid Absolute Relative Positioning (SHARP). There are three major steps in this approach. A set of reference nodes is selected randomly or from the outer boundary of the network in the first step. Nodes selected in the first step are relatively localized using the multidimensional scaling in the second step. Before applying MDS, a shortest path is computed between each pair of reference nodes. Now we have some set nodes which knows about their coordinates after first and second step. Third step is the actual localization step i.e. APS, in which remaining nodes in the network localize themselves using the anchor nodes. Shortest path distance is used by each node and finally multilateration is used to estimate the position.

3.1.7 Title: Simulated annealing based wireless sensor network localization [11]

Authors:AA Kannan, G Mao, B Vucetic
Published in: Journal of Computers, Vol 1, No 2 (2006)
Year: 2006

Overview : In this approach, the concept of simulated annealing is utilized to localize the sensor nodes. Suppose, there is a sensor network of n total number of nodes out of which m nodes are anchor nodes, then n-m nodes will be the nodes with unknown location. There are two steps in the algorithm. In the first step, simulated annealing is used to determine the location of sensor nodes using the distance constraints. Let us define the set Ni as a set containing all one hop neighbors of node i. The localization problem can be formulated as:

$$Min \sum_{i=M+1ton} \sum_{j\epsilon N_i} (d_{ij}^{\bigwedge} - d_{ij})^2 \tag{3.3}$$

In equation (1), d_ij is the measured distance between node i and its neighbor j; $d_{ij}^{\wedge} = \sqrt{(x_i^{\wedge} - x_j^{\wedge})^2 + (y_i^{\wedge} - y_j^{\wedge})^2}$ is the estimated distance; $(x_i^{\wedge}, y_i^{\wedge})$ and $(x_j^{\wedge}, y_j^{\wedge})$ are the estimated coordinates of node i and its one hop neighbor j respectively and the cost function $CF = \sum_{i=M+1ton} \sum_{j\epsilon N_i} (d_{ij}^{\wedge} - d_{ij})^2$. Then according to Simulated Annealing coordinate estimate $(x_i^{\wedge}, y_i^{\wedge})$ of any chosen node i is given a small displacement in a random direction and the new value of the cost function is calculated for the new location estimate. If$\Delta(CF) \leq 0, (\Delta(CF) = CF_{new} - CF_{old})$ then the perturbation is accepted and the new location estimate is used as the starting point of the next step.Otherwise the probability that the displacement is accepted is $P(\Delta(CF)) = exp(-\Delta(CF)/T)$. Here P is a monotonically increasing function of T and T is a control parameter.

In the next step of the algorithm, elimination of errors caused by flip ambiguity is done. The reason of flip ambiguity is that when a node's neighbors are placed in positions such that they are approximately on the same line, this node can be reflected across the line of best fit produced by its neighbors with essentially no change in the cost function. Based on this observation the authors define a complement set comp(N_i) of the set Ni as a set containing all nodes which are not neighbors of node i. If R is the transmission range of the sensor node and the estimated coordinate of node $j\epsilon comp(N_i)$ is such that $d_ij^{\wedge} < R$, then the node j has been placed in the wrong neighborhood of node i, resulting in both nodes i and j having each other as wrong neighbors. So the minimum error due to the flip is $d_ij^{\wedge} - R$ and the new localization problem can be formulated as in equation 3.4.

$$Min \sum_{i=M+1ton} (\sum_{j\epsilon N_i} (d_{ij}^{\wedge} - d_{ij})^2 + \sum (d_{ij}^{\wedge} - R)^2) \tag{3.4}$$

The paper presented a novel simulated annealing based localization algorithm which mitigates the flip ambiguity problem. By simulations the authors the authors show that the proposed algorithm gives better accuracy than the semi-definite programming localization. They show that the proposed algorithm does not propagate error in localization. The proposed flip ambiguity mitigation method

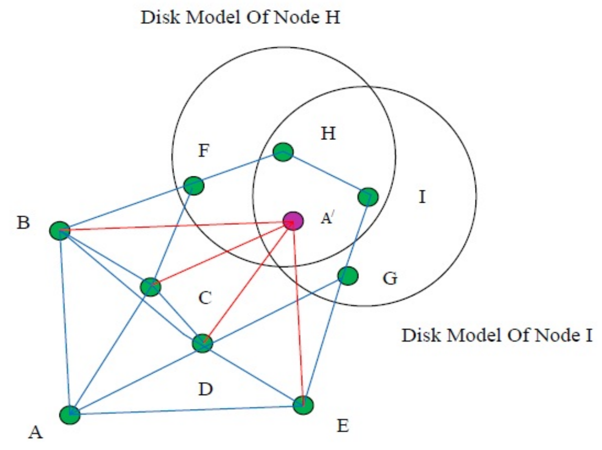


FIGURE 3.5: Illustration of flip ambiguity

is based on neighborhood information of nodes and it works well in a sensor network with medium to high node density. However when the node density is low, it is possible that a node is flipped and still maintains the correct neighborhood. In this situation, the proposed algorithm fails to identify the flipped node. **Strength:** They show that the proposed algorithm does not propagate error in localization. The proposed flip ambiguity mitigation method is based on neighborhood information of nodes and it works well in a sensor network with medium to high node density.

Weakness: when the node density is low, it is possible that a node is flipped and still maintains the correct neighborhood. In this situation, the proposed algorithm fails to identify the flipped node.

3.1.8 Title: Grouping multi-duolateration localization using partial space information for indoor wireless sensor networks [12]

Authors:H Lee, S Lee, Y Kim, H Chong

Published in: Consumer Electronics, IEEE Transactions

Year: 2010

Overview : This paper proposed a localization algorithm termed multiduolateration localization (MDL) and grouping multiduolateration localization (GMDL) for indoors by employing jumper setting of nodes. Their algorithm operate in two stages: First, edge nodes are localized using internal division and then the remaining surface nodes, are localized using localized edge nodes. It uses four beacon nodes placed at the corners of field. Localization accuracy of MDL and GMDL depends on the localization of edge nodes. It results in more error propagation as one wrongly localized edge node affects location estimation of all those surface nodes which use it as a reference node.

Weakness: Additional hardware is required for jumper setting, and initially four anchor nodes are needed to begin the process of localization.

3.2 Holes and Boundary detection algorithms

Khan et al.[13] introduced an algorithm to find the boundary of holes in the network. However there are number of algorithm found in literature for hole and boundary detection. Khan et al. [14], contains the major portion of the related literature. S. babaie and S.S. Pirahesh [15] lacks behind because of the assumption of known location information with the use of GPS which increases the overhead of power consumption. Fekete et al. [16] assumes the uniform node distribution and applied some statistical functions. Saukh et al [17], taken an assumption of particular patterns, which may not always appear. In Fanke and Klain [18], there is a large communication overhead because it requires a set of seed nodes for the initialization of boundary detection. In Wang et al. [19], node degree must be larger than 11. Also there is a need of synchronization between the nodes. Kroller et al. [20] searches for the flower structure which requires node degree at least from 20 to 30 in the random deployment of nodes. It also requires 8 hop neighbors of every node. Bi et al. [21], there is a large communication overhead of message passing between the nodes. Khan et al. [13], [22] is one of the best approach found till date which works only on the basis of local connectivity but there are a large number of iterations involved in path construction phase. Also the nodes in the last set defined in the paper includes its 1 hop neighbors also, which increases false positivity. It doesnt work well when node degree is less than 7. Detailed description of hop based approach for holes and boundary detection is given below.

3.2.1 Title: Hop-based approach for holes and boundary detection in wireless sensor networks [13]

Authors:I.M. Khan, N. Jabeur, S. Zeadally

Published in: IET Wireless Sensor Systems

Year: 2012

Overview : In this paper, author have proposed a hop based approach for the boundary detection. The whole approach is divided in three phases. First phase is the information collection phase in which, each node collects the information about their 1-hop neighbors, 2-hop neighbors and so on based on the number of boundaries to be detected. second phase is the path construction phase, in which each node n will find the communication links to the nodes present in their x-hop neighbors. Third phase is the path checking phase in which the intersection of two sets that is first set and last set is checked. if the intersection of the sets contains nothing, means if it is a null, then the node n is dented as a boundary node, else it is not a boundary node. The figure 3.6 illustrates the overall approach.

Weakness: The above explained approach to find the boundary nodes seems to be a very good approach theoretically. But when it is really implemented, we found that it is working only for a very dense network and with node degree above

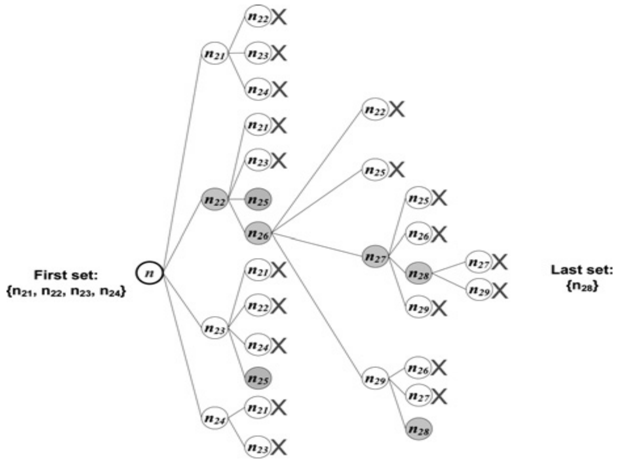


FIGURE 3.6: Illustration of hop based approach

5. Result are not very good for the less number of nodes and the node degree less than 5. Also, it is finding the exact boundary when the nodes are deployed in a grid form, but for a practical situation, it is almost impossible for the nodes to be in a grid form.

3.3 Virtual coordinates

There are a number of virtual coordinate assignment algorithms exist in the literature. Here we are elaborating two very important algorithm, which are the basis of all other proposed algorithms. Rao et al.[23] in year 2003 introduced an approach which is based on the spring relaxation procedure, which firstly assigns the coordinates to perimeter nodes, then all other interior nodes. Leong et al. [24] in year 2007 proposed Gspring (modified spring relaxation procedure) approach to provide the virtual coordinates. It also deals with dead ends in the network. Detailed description of these two methods is given below.

3.3.1 Title: Geographic Routing without Location Information [23]

Authors: Ananth Rao, Sylvia Ratnasamy, Christos Papadimitriou, Scott Shenker†, Ion Stoica

Published in: Mobicom

Year: 2003

Overview : Rao et al. had earlier proposed the NoGeo family of coordinate assignment algorithms for ad hoc wireless networks [13].In their algorithm, two nodes are designated as beacon nodes to initialize the detection of perimeter nodes from a heuristic based on their hop count from the beacons. After the detection of perimeter nodes, O(p^2) messages are exchanged, where p is the number of perimeter nodes, and an error minimization algorithm is used by the perimeter to determine the coordinates.Finally, an imaginary circle is used for the projection of perimeter nodes and nodes determine their virtual coordinates using a relaxation algorithm that works by averaging the coordinates of neighboring nodes.

3.3.2 Title: Greedy Virtual Coordinates for Geographic Routing [24]

Authors: Ben Leong, Barbara Liskov and Robert Morris

Published in: Network Protocols, 2007. ICNP 2007. IEEE International Conference

Year: 2007

Overview : In this paper, author proposed an approach for generating virtual coordinates with the help of which, new usable coordinates can be produced quickly and hence can improve the routing performance of existing geographic routing algorithms. It starts from a set initial coordinates which are derived from a set selected boundary nodes. Possible dead ends are detected by greedy embedding spring coordinates and then it uses the modified spring relaxation algorithm to incrementally adjust the virtual coordinates and to increase the convexity of voids in the virtual routing topology. By doing so, the probability of ending up of packets at dead ends get reduced during greedy forwarding. This approach is much better than the Nogeo approach, and one of the best approach to derive the virtual coordinates. In fact it achieves around 10 to 15 % better routing stretch than actual physical coordinates.

3.4 Summary

This chapter discussed about the previous approaches of localization with their merits and demerits. All these approaches somehow dependent upon a number of anchor nodes. In 2nd section of this chapter we have given a short description of various holes and boundary detection algorithm and have provided a detailed description of hop based approach, as we have compared our algorithm with this approach. 3rd section of this chapter provided the details of two pioneer work related to virtual coordinates which is a part of our proposed approach for the localization of nodes.

CHAPTER 4

LOCALIZATION USING VIRTUAL COORDINATES

4.1 Introduction to Virtual Coordinates and graph centrality

The Concept of virtual coordinates is basically introduced for the situations where information about location is not available at the sensor nodes by using GPS(Global positioning system) or some other techniques. These scenarios may be some indoor location, dense forest, underwater, underground mines etc. Till now, various algorithms have been proposed to provide the virtual coordinates to nodes for the routing purpose. Two pioneer approaches to provide virtual coordinates are Nogeo and Gspring which are already described in the previous chapter.

In this report, a new tentative approach to provide the virtual coordinates to the anchor nodes is presented. This technique is based on the concept of centrality in the social networks. Basically, four types of centrality has been described in the literature for the social network analysis. These are closeness centrality, graph centrality, stress centrality, betweenness centrality [25].

A wireless sensor network is conveniently described as a graph G=(V,E), where the set V of vertices represents all the sensor nodes and the set E of edges represents links between these nodes.A path is defined from s ϵ V to t ϵ V as an alternating sequence of vertices and edges, beginning with s and ending with t, such that each edge connects its preceding with its succeeding vertex. The length of a path is the sum of the weights of its edges. We use d_G(s, t) to denote the distance between vertices s and t, i.e. the minimum length of any path connecting s and t in G. By definition, d_G(s, s) = 0 for every s ϵ V , and d_G(s, t) = d_G(t, s) for s; t ϵ V .

Several measures capture variations on the notion of a vertex's importance in a graph. Let $\sigma_{st} = \sigma_{ts}$ denote the number of shortest paths from sϵV to tϵV , where $\sigma_{ss} = 1$ by convention. Let $\sigma_{st}(v)$ denote the number of shortest paths from sϵt that some vϵV lies on. The following are standard measures of centrality:

$$C_c(v) = 1/\sum_{t\epsilon V} d_G(v,t) \quad \text{(closeness centrality)} \tag{4.1}$$

$$C_G(v) = 1/max_{t\epsilon V} d_G(v,t) \quad \text{(graph centrality)} \tag{4.2}$$

$$C_S(v) = \sum_{s\neq v\neq t\epsilon V} \sigma_{st}(v) \quad \text{(stress centrality)} \tag{4.3}$$

$$C_B(v) = \sum_{s\neq v\neq t\epsilon V} (\sigma_{st}(v)/\sigma_{st}) \quad \text{(betweenness centrality)} \tag{4.4}$$

High centrality scores thus indicate that a vertex can reach others on relatively short paths, or that a vertex lies on considerable fractions of shortest paths connecting others. In the proposed approach, graph centrality is being utilised to provide the virtual coordinates to the nodes, which is described in detail in the next section.

4.2 Problem Statement and Contribution

4.2.1 Definitions

- *Definition 1:* **Central Node (ς):** A node $\varsigma \varepsilon V$ is said to be a central node if its graph centrality is maximum. Graph centrality criteria is already defined in the previous section.
- *Definition 2:* **Perimeter Node (ρ):** A node $\rho \varepsilon V$ is said to be a perimeter node if it lies on the imaginary circle formed after the first step of the algorithm.
- *Definition 3:* **Non-Perimeter Node (η):** A node $\eta \varepsilon V$ is said to be a non perimeter node if it lies inside the network boundary.
- *Definition 4:* **Anchor node (α):** A node $\alpha \varepsilon V$ is said to be a anchor node if it is assigned with the virtual coordinate using the virtual coordinate algorithm.
- *Definition 5:* **Target Node (τ):** A node $\tau \varepsilon V$ is said be a target node if its location(coordinates) needed to be traced using localization technique as the final step.
- *Definition 6:* **Network Boundary:** Network boundary is the imaginary circle formed by the perimeter nodes after the first phase of algorithm.

4.2.2 Problem Formulation

Problem can be formulated as follows: Our first problem is to provide the virtual coordinates to the set of anchor nodes $\alpha \varepsilon V$, so that these can be used to trace the target nodes τ in the situations where no location information is available like dense forest and underground conditions. Our second problem is to trace the target nodes τ with the help of anchor nodes α using some localization technique like trilateration.

Mathematically, the problem can be formulated as follows: given a multihop network, represented by a graph G = (V, E), and a set of anchor nodes $\alpha \varepsilon V$, their positions X_α, Y_α for all α from V, we want to find the position X_τ, Y_τ for all unknown target nodes τ.

4.2.3 Assumptions and System Model

In this work, it is assumed that the sensor nodes are homogeneously spread into a region L such that it form a connected graph G(V, E), where V represents the set of nodes and E represents the link connecting these nodes.The neighbour set of a node v($v \varepsilon V$) is N(v). Each sensor node has some communication range R knows the distance to its direct one hope neighbours using some ranging technique lke RSSI(16), TOA or TDOA.It is also assumed that the sensor does not know anything about their location. The idea behind our algorithm is to provide some initial virtual coordinates to anchor nodes(α) using the graph centrality based virtual coordinate technique.Then using RSSI,(Received signal strength indicator), distance between target nodes(mobile nodes present in the network) and atleast three anchor nodes can be calculated. In the third phase, using trilateration technique location of target node(τ) can be determined.

4.2.4 Proposed Approach

We have divided our proposed approach for tracing the targets in a wireless sensor network in 3 phases. In first phase we are providing virtual coordinates to the set of anchor nodes(α).Second phase is the distance estimation stage, in which distance between anchor nodes(α) and target nodes(τ) is calculated using RSSI.

In third phase we are tracing the position of target nodes using some mathematical technique like trilateration.

Phase 1: Assignment of virtual coordinates to anchor nodes

Suppose a Wireless sensor network is represented by a Graph G(V, E) containing n number of nodes. Initial task of the algorithm is to assign some virtual coordinates to anchor nodes(α). To do this, all the nodes are divided into two categories, Perimeter nodes(ρ) and Non-Perimeter nodes(η). Perimeter nodes have to be mapped to an imaginary circle to form a network boundary. This task is done by the central node(ς). So the first task of this approach is to elect the central node which is done on the basis of graph centrality $C_G(v)$.

1. ***Election of Central Node:*** Each node in the network will broadcast a hello message which is piggybacked with a hop counter. This is how nodes will know about the hop count distance to each node in the network. Then, each node will calculate it's graph centrality by comparing their maximum hop count distance using the formula given below.

   $$C_G(v) = 1/max_{t \epsilon V} d_G(v,t) \quad \text{(graph centrality)} \tag{4.5}$$

   The node with the maximum graph centrality will be elected as the central node and assigned with coordinates (0,0).

2. ***Detection and assignment of Virtual coordinates to Perimeter Nodes:*** In the first step, all the nodes know the hop count distance to each node. So a distance vector D is build at each node. Central node will select it's maximum hop count(max_d) distance node and assign it virtual coordinates(max_d,0). Again select next maximum hop count node from the center and assign the virtual coordinates to it using cosine rule as described below.

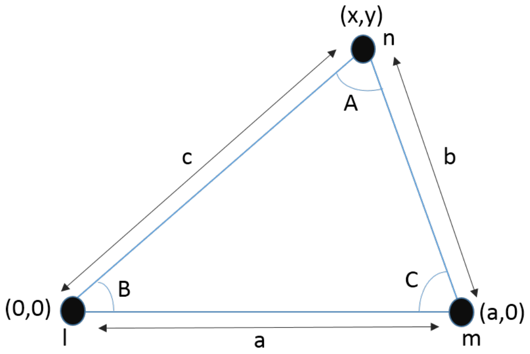


FIGURE 4.1: A trianlge showing 3 nodes with their known distances.

Accordng to cosine rule, we have

$$a^2 = b^2 + c^2 - 2bc\cos(A) \tag{4.6}$$

$$b^2 = a^2 + c^2 - 2ac\cos(B) \tag{4.7}$$

$$c^2 = a^2 + b^2 - 2ab\cos(C) \tag{4.8}$$

These equations can be converted to find the angles as follows:

$$A = \cos^{-1}(b^2 + c^2 - a^2/2bc) \tag{4.9}$$

$$B = \cos^{-1}(a^2 + c^2 - b^2/2ac) \tag{4.10}$$

$$C = \cos^{-1}(a^2 + b^2 - c^2/2ab) \tag{4.11}$$

here it is needed to calculate the angle made at the central node, which is $\angle B$ calculated by equation 4.10. Now as we know the angle made at the central

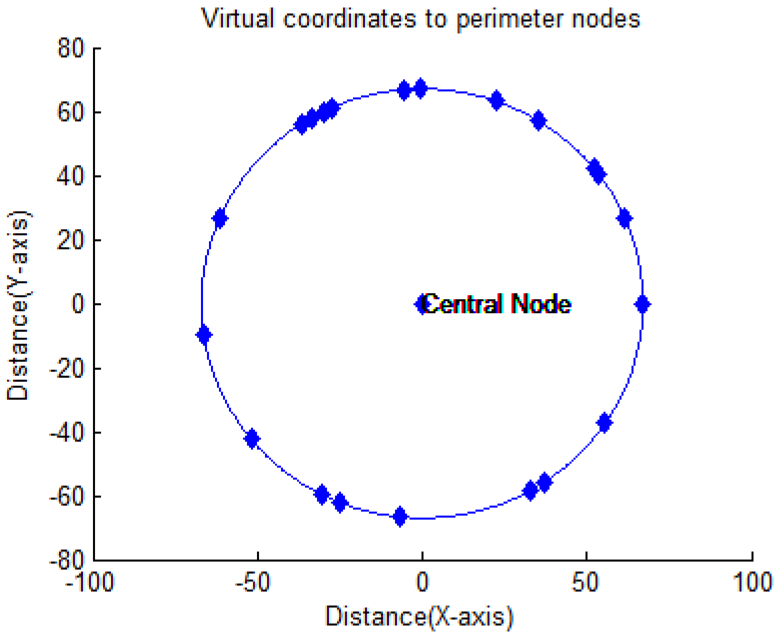


FIGURE 4.2: Circle representing the perimeter nodes.

node, we can assign coordinates to node n(x,y) as $(a\cos(B), a\sin(B))$. In the similar way, we assign coordinates to remaining k-2 perimeter nodes by assuming central node(ς) and first perimeter node as the reference nodes shown in figure 4.2.

This is how, we assign coordinates to k perimeter nodes by using the central node. Now the next task is to assign coordinates to remaining n-k non-perimeter nodes, which is explained in the next section.

3. ***Detection of Perimeter Nodes and holes in the network using local connectivity:*** However, we have already designed an approach to find out the perimeter nodes using the central node, but as the boundary and hole detection is another major problem in the wireless sensor network and which also serves as a base to many application like target tracking, so here we are proposing another method which is not based on the availability of central node.

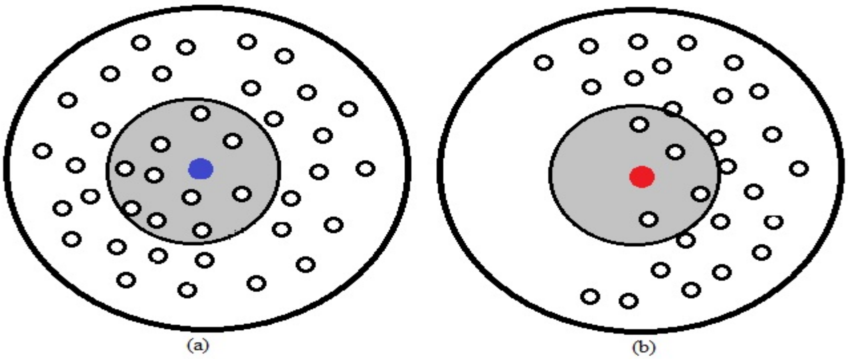


FIGURE 4.3: (a) Blue node is an interior node as it is having sufficient number of nodes in its range (b) Red node is a boundary node because of lack of nodes in its range.

The key notion of our algorithm is to find out the number of neighboring nodes that covers a particular node N. If the number of neighboring nodes is sufficiently high, it means, the coverage of that region is good, and it cannot be a hole. On the other hand, if a node N doesn't have sufficient number of neighboring nodes, it means that node is lies somewhere around the region, where coverage is less and hence can be declared as a boundary node as shown in figure 4.3.

We have divided our algorithm in two phases.

Phase 1: Computation of threshold Value

Let us consider a graph G (V, E), representing a wireless sensor networks. As we know that the sensor nodes have the capability of communication and computation, so all the nodes (V) will send a message M1$< Node_id, R_F_A, Seq_No. >$ to their one hope neighbors requesting them to send back an acknowledge message M2 $< Node\_id, Ref\_M1, Seq\_No. >$ containing their unique ids. Here R_F_A means request for acknowledgement and Seq_No. is used to identify duplicate messages. The nodes which will be active at that time, will respond with an acknowledgement message M2. Each node will then count the received acknowledgement messages and in this way, they will be able to identify the number of their active one hop neighbor nodes.

Each node will then construct a message M3$< node_id, Neighbor_count >$ containing the node id and their one hop neighbor count and broadcast it into the whole network, so that each node will know about the neighbor counts of all the nodes. In this way, we can say that the network now has the information about the one hop neighbor counts of each node. A vector will be constructed at each node which contains the node_id and neighbor_count of all the active nodes in the network. Now each node at their own can compute a threshold value (TH_{value}) by using the following formula.

$$TH_{value} = \frac{\sum_{i=1to|V|} Neighbor\_Count(i)}{|V|} \tag{4.12}$$

here, $|V|$= total number of active nodes in the network

Phase 2: Testing of the nodes

In this phase, each node can identify their status on the basis of threshold value in a distributed manner. Target Nodes can simply run an algorithm to compare their one hop neighbor_count value with the threshold. If it is less than threshold value, than the node will declare itself as a boundary node. Otherwise it will be an interior node.

4. ***Assignment of Virtual coordinates to Non-Perimeter Nodes:*** In the previous section, we find out the coordinates of k perimeter nodes.In this section, we describe a iterative method by which all n-k perimeter nodes can determine their coordinates using relaxation procedure.The analogy, borrowed from the theory of graph embeddings, is that each link – each neighbor relation– is represented by a force that pulls the neighbors together.If we hold its neighbors fixed, then a node's equilibrium position (the one where the sum of the forces is zero) is where the its x-coordinate is the average of its neighbors' x-coordinates (and, again, the same for y-coordinate)[16]. We

use these facts to motivate an iterative procedure where each non-perimeter node periodically updates its virtual coordinates as follows:

$$x_i = \sum_{n \epsilon neighbour\_set(i)} x_n / size\_of(neighbor\_set(i)) \tag{4.13}$$

$$y_i = \sum_{n \epsilon neighbour\_set(i)} y_n / size\_of(neighbor\_set(i)) \tag{4.14}$$

Phase 2: Distance estimation between anchor nodes (α) **and Target nodes**(τ) We provided the virtual coordinates to all the anchor nodes present in the network. Now, in this phase, we calculate the distance between the anchor nodes (α) and target nodes(τ). In chapter 2, we have already seen that, their are many techniques to calculate distances mentioned in [28-31].In a WSN, every node has a radio, if we can calculate the distance using this radio model, then no extra hardware is required on nodes for distance estimation. So here, we are using RSSI method to calculate distances, which is based on the fact that the radio signal strength attenuates from one node to another.In theory, the energy of a radio signal diminishes with the square of the distance from the signal's source. As a result, a node listening to a radio transmission should be able to use the strength of the received signal to calculate its distance from the transmitter. RSSI is often coupled with surrounding noise, and even may vary completely for different surroundings. But a careful propagation analysis and proper statistical techniques may be used to eliminate these problems to a maximum.

The principle of RSSI ranging describes the relationship between transmitted power and received power of wireless signals and the distance among nodes [30] [31]. This relationship is shown in the Equation.4.15 :

$$P_r = P_t x (1/d)^n \tag{4.15}$$

where P_r received power of wireless signals, P_t the transmitted power of wireless signal, d the distance between the sending nodes and receiving nodes, n the transmission factor whose value depends on the propagation environment.

Taking 10 times logarithm on both sides, we obtain the following relation: In RSSI calculations, RSSI is given, which is derived from Pr in decibel meters (dBm). It is hence clear from above equations that RSSI α log Pr and also P $\alpha 1^n/d$. So, we can calculate RSSI as follows:

$$RSSI = -10n\log(d) + C \tag{4.16}$$

RSSI has a linear relationship with logarithm of distance, where C is some constant, we can represent Eq. 2 as:

$$RSSI = -m\log(d) + C \tag{4.17}$$

Such that n=m/10.

Phase 3: Location estimation of Target Nodes(τ) Now, we have distances between anchor nodes(α) and the target nodes (τ) and also the coordinates of all the anchor nodes. So in this phase, we need to calculate the location (coordinates) of target nodes (τ). Their are some mathematical techniques which have been already described in chapter 2 . Trilateration is one of the technique which can be used for the purpose. We have already discussed trilateration in chapter 2.

4.2.5 Algorithms

Here we are explaining the algorithm used for election of central node, perimeter nodes and non perimeter nodes.

Algorithm 1 Election of Central Node(ς)

```
Require: Nodes know their one hop neighbour
Ensure: Central node($\varsigma$)
1: For each nodes
     Broadcast a message $< node\_id, hop\_counter >$ in the network, where node_id is id of each node from where the message is initialized and hop_counter is the counter of each hop.
2: At each node, build a distance vector D containing distances to all nodes.
3: For each node
     Select max_hop_count from their distance vector D.
4: For each node
     Calculate Graph centrality $C_G$ and broadcast it into the network.
5: Return Maximum $C_G$ as the central node ($\varsigma$) and assign it coordinates (0,0).
```

Algorithm 2 Centrality based Detection and virtual coordinate assignment to k perimeter nodes

```
Require: Central node and Distance vector D build in Algorithm 1
Ensure: Virtual coordinates to perimeter nodes($\rho$)
1: Select max_hop_count from the Distance Vector D of central node($\varsigma$).
2: Assign it coordinates (max_hop_count,0).
3: repeat 4 to 5 k-1 times
4: Select next maximum hop_count from D of $\varsigma$ and find it's coordinates(x,y) as follows:
5: Calculate internal angle made at $\varsigma$ using cosine rule i.e. equation 4.10.
6: Assign $x = max\_hop\_count * \cos(B)$.
7: Assign $y = max\_hop\_count * \sin(B)$.
8: Return (x,y) as coordinates of this perimeter node($\rho$).
```

Algorithm 3 Virtual coordinates for n-k Non-perimeter nodes(η).

```
Require: Nodes know their one hop neighbor and know the perimeter nodes.
Ensure: Virtual coordinates to non-perimeter nodes($\eta$).
1: Initially assign equal coordinates $(X_\eta, Y_\eta)$ that exist within the network boundary to all non-perimeter nodes.
2: Repeat 3 to 4 some s times to reach in equilibrium state.
3: For all n-k non-perimeter nodes ($\eta$):
     Build 1-hop neighbor set.
4: Update coordinates $(X_\eta, Y_\eta)$ using equation 4.13 and 4.14.
5: Return stabilized network with known coordinates of $\rho$ and $\eta$.
```

Algorithm 4 Local Connectivity based Boundary detection algorithm

```
Require: Nodes know about the local connectivity
Ensure: Perimeter nodes and Interior nodes
All active nodes send message M1$< Node\_id, R\_F\_A, Seq\_No. >$ to their one hop neighbors to make request for the acknowledgement. Where $Node\_id$ is the id of sender node, $R\_F\_A$ indicates request for acknowledgement and $Seq\_No.$ is the message number.
Neighboring Nodes will send the acknowledgement message M2$< Node\_id, Ref\_M1 >$ to their senders. Where $Node\_id$ is the id of receiver node and $Ref\_M1$ indicates that the message is an acknowledgement of M1.
Each node will count the acknowledgement messages (M2) received to know the number of one hop neighbors.
A message M3$< node\_id, Neighbor\_count >$ is broadcasted into the network by each node. Where $Neighbor\_count$ indicates the number of one hop neighbors.

A Vector $\nu$ is Constructed at each node containing $\nu < Node\_ids, Neighbor\_count >$.
A $TH_{value}$ is computed by using the formula 1.
Now, for any node $N_0$
if $Neighbor\_count < TH_{value}$ then
    Node $N_0$ will declare itself as a Boundary node($\rho$).
else
    Node $N_0$ will declare itself as an Interior node.
end if
```

CHAPTER 5

SIMULATION AND RESULTS

This chapter covers the results and simulation analysis of our proposed approach. To compare the results of our algorithm, we also implemented one of the existing algorithm named as hop based approach for boundary detection in wireless sensor networks. Reason behind implementing only this approach is it's resemblance with our approach as both the approaches are using the hop connectivity to find the boundary nodes. For the implementation purpose, we have used matlab. In all the simulations, Range of nodes is assumed to be uniform. The reason of this assumption is that, if we consider non uniform ranges, then the graph will not be undirected. For the accurate analysis of our algorithm, we conducted the simulation in different scenarios and environments described in the following the following subsections. In all the simulations, we have used red color for the boundary nodes and blue color for the interior nodes.

The following section and figures describes all the results and simulations done over different number of nodes.

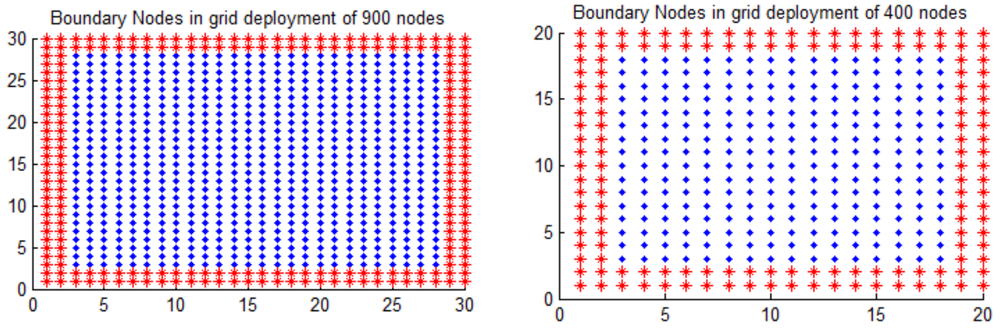


Figure 5.1: Double layer boundary detection in 900 and 400 nodes using hop-based boundary detection.

5.1 Results of existing hop based boundary detection algorithm

This section contains the results of the existing algorithm(Hop based approach for boundary detection in wireless sensor networks). As it is an iterative algorithm, we used 2-hop neighbors list for the implementation purpose. Firstly we took the scenario of grid deployment of nodes. The number of nodes which we have taken are 400 and 900 deployed in a 20×20 and 30×30 area respectively with the radio range 1.5 which sufficiently makes the node degree very high as per the requirement of the algorithm. Figure 5.1 clearly showing the double boundary layer in both cases as we have used 2-hop neighbor list for the path construction phase. We can also find the 3_rd boundary layer by using 3-hop neighbor list, but the computation cost will increase.

For the second case, we deployed the nodes randomly which is the actual scenario in wireless sensor networks. The number of nodes taken are 100 and 300 which are deployed in 1000×1000 area with radio range 200. Figure 5.2 showing the results of this algorithm for the random deployment which are not very good. A large number of nodes identifies himself as boundary nodes in the case of 100 nodes.

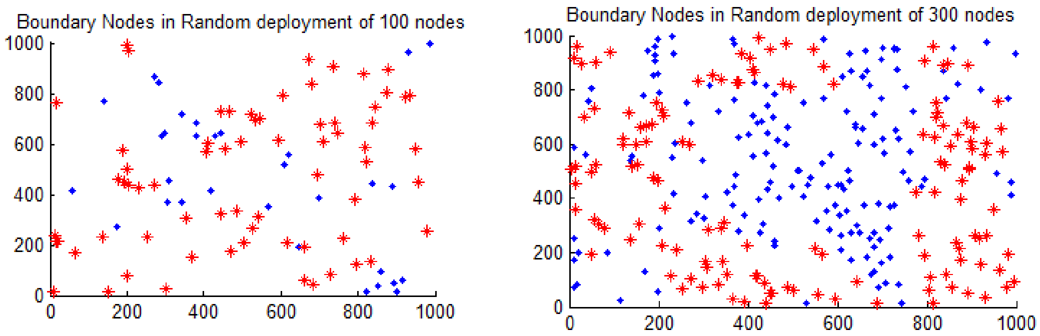
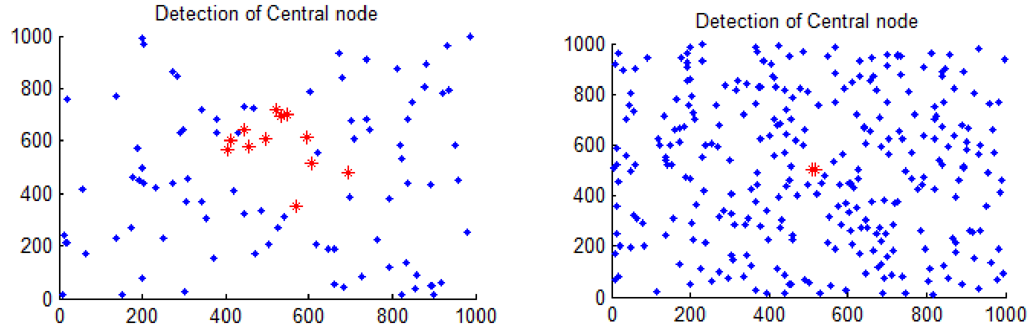


FIGURE 5.2: Boundary detection in 100 and 300 nodes using hop-based boundary detection.

While this is not so in the case of 300 because number of nodes got increased. All this is because of disjoint of two sets(first set and last set) defined in the algorithm.

5.2 Results of Centrality based detection and virtual coordinate assignment to perimeter nodes

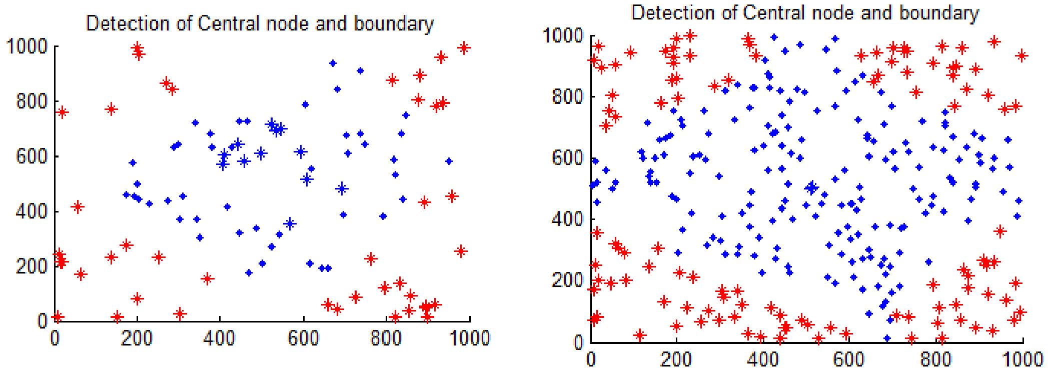
We implemented the first phase of our approach i.e. detection of perimeter nodes and assignment of virtual coordinates to these nodes using graph centrality in matlab. Again we took the same network of 100 and 300 nodes deployed in 1000 × 1000 with radio range 200. Figure 5.3 showing the detection of central node in both the cases. Figure 5.4 showing the detection of perimeter nodes. As we can see, it is showing better results than the existing algorithm in the case of random deployment. Virtual coordinate assignment is already shown in the figure 4.2 in which perimeter nodes are virtually mapped to a circle with center representing the central node.



Detection of Central node

Detection of Central node

FIGURE 5.3: Detection of central nodes in 100 and 300 number of nodes using the proposed centrality based algorithm.



Detection of Central node and boundary

Detection of Central node and boundary

FIGURE 5.4: Detection of perimeter nodes in 100 and 300 number of nodes using the proposed centrality based algorithm.

5.3 Results of local connectivity based boundary and hole detection algorithm

5.3.1 Boundary detection in grid and random deployment of nodes

In this section, we have shown the grid and random deployment of nodes. Firstly, we simulated our algorithm on 900 nodes deployed in grid manner in an area of 30×30 meters which also contains 4 holes. Figure 5.5 shows the detected boundary

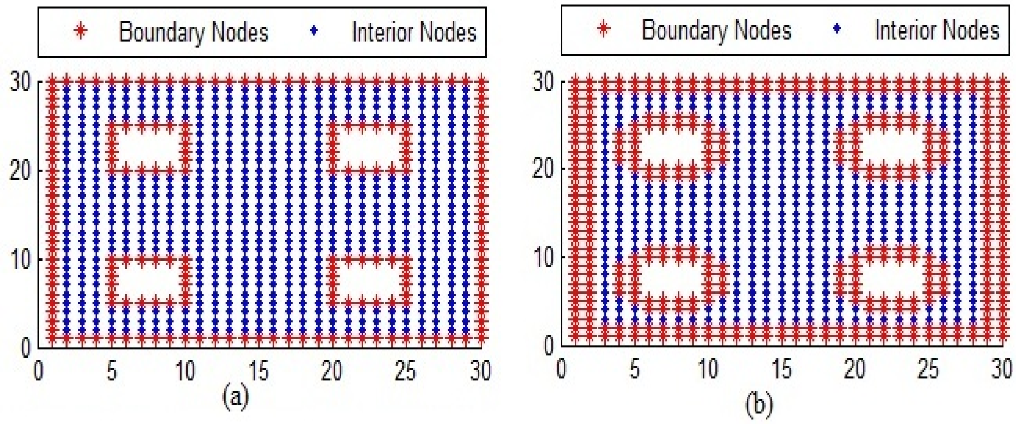


FIGURE 5.5: Boundary detection in grid deployment of 900 nodes when radio range is (a)1.5 meter, (b)3 meter.

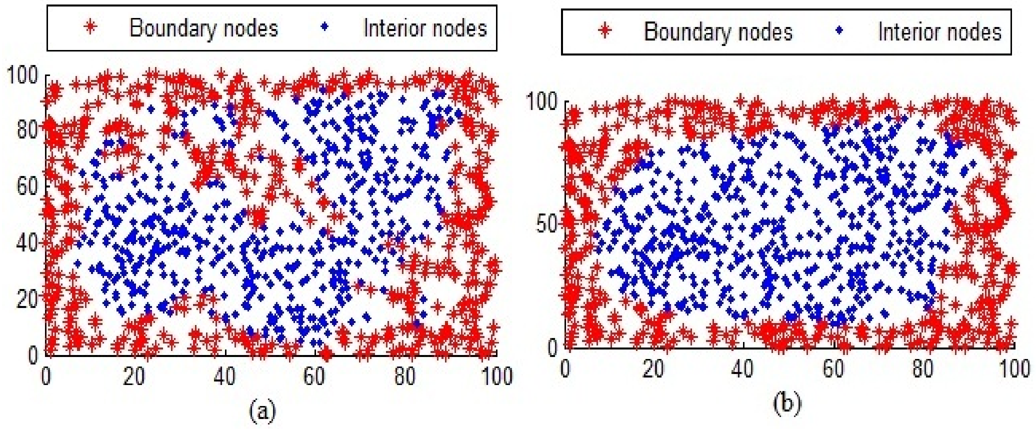


FIGURE 5.6: Boundary detection in random deployment of 1000 nodes when radio range is (a)15 meter, (b) 25 meter.

nodes marked with red color when radio range of nodes is (a) 1.5 meter and (b) 3 meter. Secondly we simulated our algorithm on 1000 nodes deployed in an area of 50×50 meters. Figure 5.6 shows the detection of boundary nodes when radio range is (a) 15 meter, (b)20 meter, (c)25 meters. We can see that the false detection of boundary nodes is less in 25 meter radio range.

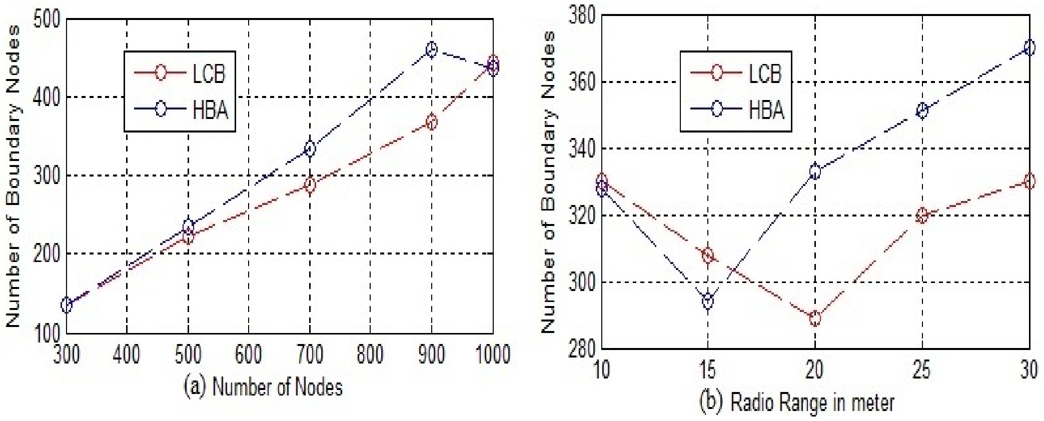


FIGURE 5.7: Comparison of Number of Boundary nodes between LCB and HBA. (a) With different number of nodes, (b) With different Radio range.

5.3.2 Impact of changing the number of nodes and radio range on boundary detection

This subsection includes a comparison analysis of our algorithm with hop based approach for holes and boundary detection [22].[13] is an extended work of [22], in which author made slight changes in their last set construction. They included the one hop neighbors of the last set nodes in path checking phase which is the major reason of its performance reduction.

For our convenience, we named our algorithm as Local connectivity based algorithm (LCB), and existing hop based approach[13] as HBA. Figure 5.7 shows two different scenarios of comparison, firstly we compared our algorithm on the basis of change in number of nodes , secondly we analyzed the effect of change in radio range. Plots in figure 5.7 (a) and (b) clearly shows the difference between our algorithm and the existing algorithm.

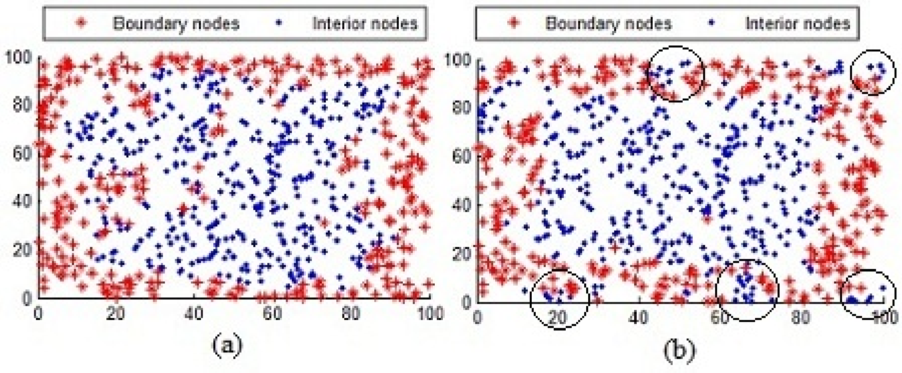


Figure 5.8: Boundary nodes detected in (a)LCB, (b) HBA. Encircled nodes are undetected Boundary nodes

5.3.3 A comparison analysis based on false positivity

In this subsection, a comparison analysis is done based on accurate boundary detection and false positivity of boundary nodes between LCB and HBA. False positivity may be defined as the identification of a node as a boundary node, but if in fact it is an interior node. Figure 5.8(a) clearly shows the accurate boundary detection in LCB and the undetected boundary nodes in HBA are encircled in figure 5.8(b). Plot in figure 6 shows the difference between the two algorithms based on percentage of false detection. Firstly we can see that, percentage of false positivity decreases with increase in radio range, in both approaches. Secondly, we can say that our approach has better false positivity than HBA on different radio ranges as illustrated in figure 5.9. However, at radio range 15, false positivity of our approach is more than HBA. This is because of some topological changes in the network. But after than, it gradually decreases. We found 5.88, 4.66, 4.1 % of false positivity at radio range 20, 25 and 30 respectively. While in case of HBA, we found 6.44, 5.87, 5.0 % at radio range 20, 25 and 30 respectively. However more accurate results and simulation analysis deserves a further study and we are still working on it.

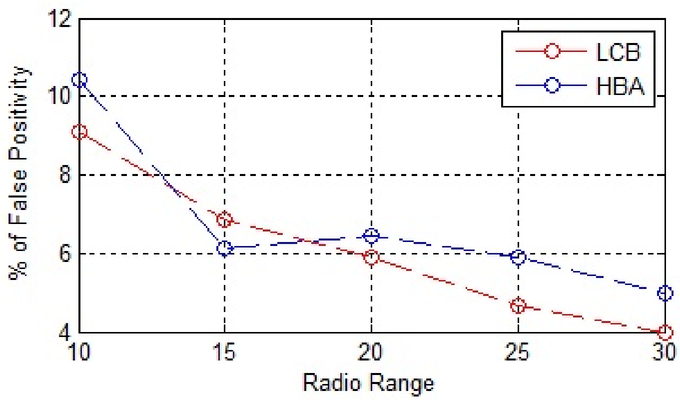


FIGURE 5.9: Comparison between LCA and HBA on the basis of false positivity.

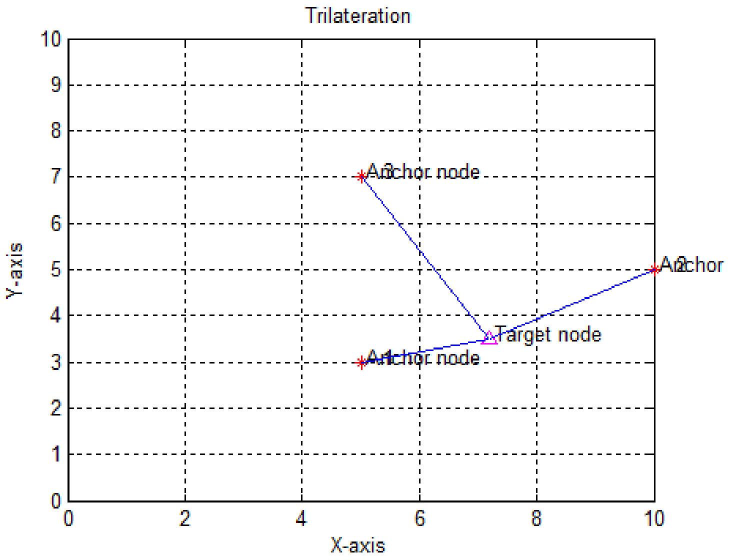


FIGURE 5.10: Target node Localized using 3 anchor nodes.

Using algorithm 3 that is virtual coordinates to non perimeter nodes, we have successfully provided the virtual coordinates to all the nodes in the network. Now we have all the nodes with their coordinates, so any target can be easily traced using trilateration technique. To apply trilateration method, we need to broadcast the RSSI values, so as to compute the distance between nodes. This has been implemented in the cooja simulator. and then we have build a prototype of trilateration shown in figure 5.10.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Localization in wireless sensor networks have received increasing attention over the last one decade. It not only provides the geographical position of a sensor node but also fills the pre-requisite for geographic routing, spatial querying, and data dissemination. With the continuous research in localization of sensor networks, a number of effective algorithms have been proposed, but the stability has not yet reached. This is because of the meager resources (storage, battery, processor) and the harsh deployment environments. Currently, none of the localization techniques is able to full-fill all these constraints. Most existing localization algorithms for static WSNs were designed to work with at least three anchor nodes except in those cases where directional antenna is used. Usage of antenna not only increases the cost, but also the size of node as well as complexity of the algorithm. As the number of anchor nodes required in a network increases, overall cost of the network also increases. In addition, energy drainage of the network increases, but the localization time of the whole network decreases. Further, anchor nodes installed with GPS do not work well everywhere. Therefore, at present we are in the need of a novel technology that will solve the following problems: (*i*) reduce the dependency on anchor nodes, (*ii*) localize sensor nodes in areas where GPS do not work well, (*iii*) minimize the localization error. In this work, we have proposed localization technique based on virtual coordinates provided to the anchor nodes.

We completed the first part of localization system i.e. to provide virtual coordinates to the initial anchor nodes. We have developed a distributed algorithm which is based on the concepts of centrality in social networks . We have developed one more algorithm to detect perimeter nodes using the local connectivity in the wireless sensor networks. Finally we made an attempt to compare the results of our algorithm with the Hop based approach for boundary nodes detection . Results of this comparison are given in the result section of chapter 5. Phase 2 that is related to the computation of RSSI values is implemented in contiki's Cooja simulator that is specially desinged for low power embedded devices like sensor motes.

In this work, we have proposed a distributed approach for holes and boundary detection which uses only the one hop neighbor information. Moreover, there is no assumption regarding the distance computation so Computation overhead is also reduced in our approach. During the simulation of grid deployment of nodes, we found that LCA is able to find the boundary nodes perfectly whereas in HBA, all boundary nodes gets detected except some corner nodes shown in figure 5. We also found LCA better than HBA on the basis of false positivity however we are also planning to optimize our work by using ego centrality which purely works on local information. As it is not implemented in on the real sensor nodes, we can not say that the computation of RSSI values is accurate, because there are lot of environmental factors involved in its computation. So we have also planned to implement our virtual coordinate approach on real sensor motes using contiki platform.

BIBLIOGRAPHY

[1] W. Dargie and C. Poellabauer, *Fundamentals of wireless sensor networks: theory and practice:* John Wiley and Sons, 2010.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, pp. 393-422, 2002.

[3] A. Pal, "Localization algorithms in wireless sensor networks: Current approaches and future challenges," *Network Protocols and Algorithms*, vol. 2, pp. 45-73, 2010.

[4] G. Han, H. Xu, T. Q. Duong, J. Jiang, and T. Hara, "Localization algorithms of wireless sensor networks: a survey," *Telecommunication Systems*, vol. 52, pp. 2419-2436, 2013.

[5] Y. Shang, W. Ruml, Y. Zhang, and M. P. Fromherz, "Localization from mere connectivity," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking and computing*, 2003, pp. 201-212.

[6] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 81-95.

[7] A. Savvides, H. Park, and M. B. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proceedings of the*

1st ACM international workshop on Wireless sensor networks and applications, 2002, pp. 112-121.

[8] N. B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller, "Anchor-free distributed localization in sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003, pp. 340-341.

[9] K.-Y. Cheng, K.-S. Lui, and V. Tam, "Localization in sensor networks with limited number of anchors and clustered placement," in *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*, 2007, pp. 4425-4429.

[10] A. A. Ahmed, H. Shi, and Y. Shang, "Sharp: A new approach to relative localization in wireless sensor networks," in *Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on*, 2005, pp. 892-898.

[11] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based wireless sensor network localization," *Journal of Computers*, vol. 1, pp. 15-22, 2006.

[12] H. Lee, S. Lee, Y. Kim, and H. Chong, "Grouping multi-duolateration localization using partial space information for indoor wireless sensor networks," *Consumer Electronics, IEEE Transactions on*, vol. 55, pp. 1950-1958, 2009.

[13] I. M. Khan, N. Jabeur, and S. Zeadally, "Hop-based approach for holes and boundary detection in wireless sensor networks," *Wireless Sensor Systems, IET*, vol. 2, pp. 328-337, 2012.

[14] I. Khan, H. Mokhtar, and M. Merabti, "A survey of boundary detection algorithms for sensor networks," in *Proceedings of the 9th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, 2008.

[15] S. Babaie and S. S. Pirahesh, "Hole detection for increasing coverage in wireless sensor network using triangular structure," arXiv preprint arXiv:1203.3772, 2012.

[16] S. P. Fekete, A. Kröller, D. Pfisterer, S. Fischer, and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *Algorithmic Aspects of Wireless Sensor Networks, ed: Springer*, 2004, pp. 123-136.

[17] O. Saukh, R. Sauter, M. Gauger, and P. J. Marrón, "On boundary recognition without location information in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 6, p. 20, 2010.

[18] S. Funke and C. Klein, "Hole detection or: how much geometry hides in connectivity?," in *Proceedings of the twenty-second annual symposium on Computational geometry*, 2006, pp. 377-385.

[19] Y. Wang, J. Gao, and J. S. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proceedings of the 12th annual international conference on Mobile computing and networking*, 2006, pp. 122-133.

[20] A. Kröller, S. P. Fekete, D. Pfisterer, and S. Fischer, "Deterministic boundary recognition and topology extraction for large sensor networks," *in Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, 2006, pp. 1000-1009.

[21] K. Bi, K. Tu, N. Gu, W. L. Dong, and X. Liu, "Topological hole detection in sensor networks with cooperative neighbors," *in Systems and Networks Communications, 2006. ICSNC'06. International Conference on*, 2006, pp. 31-31.

[22] I. Khan, H. Mokhtar, and M. Merabti, "A new self-detection scheme for sensor network boundary recognition," in *Local Computer Networks, 2009.LCN 2009. IEEE 34th Conference on*, 2009, pp. 241-244.

[23] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in *Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 96-108.

[24] B. Leong, B. Liskov, and R. Morris, "Greedy virtual coordinates for geographic routing," in *Network Protocols, 2007. ICNP 2007. IEEE International Conference on*, 2007, pp. 71-80.

[25] U. Brandes, "A faster algorithm for betweenness centrality*," *Journal of Mathematical Sociology*, vol. 25, pp. 163-177, 2001.

[26] N. Linial, L. Lovasz, and A. Wigderson, "Rubber bands, convex embeddings and graph connectivity," *Combinatorica*, vol. 8, pp. 91-102, 1988.

[27] I. Stojmenovic, *Handbook of sensor networks: algorithms and architectures* vol. 49: John Wiley and Sons, 2005.

[28] P. Kumar, L. Reddy, and S. Varma, "Distance measurement and error estimation scheme for RSSI based localization in Wireless Sensor Networks," in *Wireless Communication and Sensor Networks (WCSN), 2009 Fifth IEEE Conference on*, 2009, pp. 1-4.

[29] M. Srbinovska, C. Gavrovski, and V. Dimcev, "Localization estimation system using measurement of RSSI based on Zigbee standard," in *Conference Proceedings of the 17th International Scientific and Applied Science Conference (Electronics 2008)*, 2008.

[30] A. Faheem, R. Virrankoski, and M. Elmusrati, "Improving RSSI based distance estimation for 802.15. 4 wireless sensor networks," in *Wireless Information Technology and Systems (ICWITS), 2010 IEEE International Conference on*, 2010, pp. 1-4.

[31] A. T. Parameswaran, M. I. Husain, and S. Upadhyaya, "Is rssi a reliable parameter in sensor localization algorithms: An experimental study," in *Field Failure Data Analysis Workshop (F2DA09)*, 2009, p. 5.