LOCALIZATION OF WIRELESS SENSOR NETWORKS

USING MULTIDIMENSIONAL SCALING

A Thesis

presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science

By

SRIDHAR TULABANDULA

Dr. Chengshan Xiao, Thesis Supervisor

DECEMBER 2007

The undersigned, appointed by the Dean of the Graduate School,

have examined the thesis entitled.

LOCALIZATION OF WIRELESS SENSOR NETWORKS USING MULTIDIMENSIONAL SCALING

Presented by Sridhar Tulabandula

A candidate for the degree of Master of Science

And hereby certify that in their opinion it is worthy of acceptance.

Dr. Chengshan Xiao

Associate Professor, Department of ECE, University of Missouri-Columbia.

Dr. Zhihai He

Assistant Professor, Department of ECE, University of Missouri-Columbia.

Dr. Sounak Chakraborty

Assistant Professor, Statistics Department, University of Missouri-Columbia.

Acknowledgements

I would like to profusely thank my advisor Dr. Chengshan Xiao for his valuable and constant support during the course of my work. I would also like to thank the committee members Dr. Henry He for his valuable suggestions and Dr. Sounak Chakraborty for spending his valuable time on my thesis report and seminar. I am thankful to Jun Tao who guided me on my successful thesis work. Thanks to my friends and colleagues who helped me out on miniature things.

I am grateful to my parents without their support, I would never be where I am.

Table of Contents

List of figures

LOCALIZATION OF WIRELESS SENSOR NETWORKS USING MULTIDIMENSIONAL SCALING

Sridhar Tulabandula

Dr. Chengshan Xiao, Thesis Supervisor

ABSTRACT

Wireless sensor networks found their way into a broad variety of applications including industrial automation, medical applications, highway monitoring, habitat monitoring, military applications, environmental applications, and at the bottom commercial applications like personal health diagnosis, automated grocery checkout, remote controlled heating and lighting, etc. Except commercial applications most of them require thousands of nodes deployed for sensing and controlling. Recent advances in internet, communications, information technologies, and sensor design made these applications possible and the design of cheap low power nodes using MEMS help envision all potential applications.

All these applications require knowing the locations of nodes, thus demanding a localization algorithm that is less complex in terms of computations, communication overhead and increasing the overall lifetime of the network with low life-cycle costs. In this thesis we give details of a simple mathematical technique, classical multidimensional scaling and how it solves the localization problem. It is simple for it does not have iterations or loops. This so called MDS-MAP algorithm is able to find the relative positions of nodes and with few anchor nodes available derives or maps the relative coordinates to absolute coordinates. When using a complex localization algorithm highly sophisticated nodes must be deployed and this increases the overall cost of deployment of the network. With a tradeoff between complexity and accuracy this less complex MDS-

MAP algorithm derives absolute positions of nodes with accuracy sufficient enough for most of the applications. In support we present simulation results of the MDS-MAP algorithm on three different topologies. Also discussed are the simulation setup of an ad hoc network and topology control of the network with varying network characteristics. All the simulations are carried out in MATLAB.

Chapter 1

Background on Ad Hoc Sensor Networks

1.1 Introduction

A wireless sensor node is a compact-size relatively inexpensive computational node capable of sensing or measuring the surrounding environmental conditions or other parameters, communicate with neighboring nodes to forward the sensed information, and also able to perform general computations on the data assembled from multiple sensors. Thus sensing (measuring), computing, and communication elements together form a wireless sensor node.

Wireless Sensor network (WSN) is an infrastructure comprising of hundreds or even thousands of such wireless sensor nodes (WNs) sprinkled or spread over a geographical area allowing an administrator to instrument, observe and react to events or phenomena in that particular environment. Typical Sensor networks communicate directly with the centralized processor whereas sensor networks with the following characteristics are called as ad hoc sensor networks – high density of sensor nodes and each one capable of mobile communication, has some sort of intelligence to process the sensed information, transmit the data. Thus an ad hoc WSN follows a sequence of operations demanding a less complex setup.

Before mentioning the requirements of a sensor network, going through few of many applications of wireless sensor networks helps us to better understand the requirements, and envision all other potential applications.

1.2 Motivating Applications

Security applications

- Self-organizing, fault tolerance, rapid access to data and computing characteristics of WSNs make them versatile for military applications which include tracking enemies, weapon targeting, monitoring inimical forces, battlefield surveillance, etc.
- Camera-enabled sensors used for surveillance in shopping malls and parking garages, detect the speed of vehicles on highways, etc.
- Monitor vehicle traffic flow on highways.
- To monitor and minimize fire hazards in chimneys, buildings and temperature of water pipes to prevent ice damage.
- Sensors employed with inventory prevent theft.
- On-ship or on-vehicle tampering detection.

Home/Building automation

- Control lighting, heating, and cooling systems from anywhere inside the home and customizable lighting schemes for reducing the consumption of energy and improves comfort.
- Control multiple home systems using a single remote increases the safety.
- Sensors together deployed with intelligence optimize the usage of electricity.

- Sensors deployed to sense unusual events like robbery, trigger fire alarm.

Industrial automation

- Continuous monitoring of hazardous, huge, costly equipment.
- Enhance employee safety by deploying smoke, CO detectors.
- Gather detailed data to help take preventive maintenance steps such as detecting poorly operating or damaged equipment, status changes of devices.
- Inventory management.
- Improve the performance and lifetime of products by integrating sensors into them.

Medical applications

- Body-worn medical sensors are emerging which are used to sense heartbeat, body temperature for elderly people, oxygen saturation.
- Track patients indoors and outdoors.
- Nanoscopic sensors used as a diagnostic tool in medicine.

Civil and Environmental applications

- In earthquake prone regions, sensors employed with building structures, bridges, dams, etc., self-diagnose the problems caused due to earthquakes and reports repairs to be done.
- Detecting/alerting to calamities like earthquakes, volcanoes, tornadoes, tsunamis, etc.
- Sensor nodes report climatic changes in difficult to reach locations.

- Microclimate sensors for habitat monitoring capable of sensing temperature, humidity, barometric pressure.
- Firebugs are sensors capable of sensing fires to report the fire behavior like its evolving direction in forests.

In brief wireless sensor networks can be employed if one wants to observe and to react to events or phenomena in any specified environment.

1.3 Requirements of a WSN

The above discussed applications demand the following requirements for WSNs.

One of the characteristics of an ad hoc sensor network is large sensor population. So the cost of individual node plays an important role in the overall cost of the sensor network deployment. Hence it follows that the cost of each node should be kept very low. The target is to build disposable sensor nodes that cost less than $1.

Aiming at less cost in building a sensor node implies limited functional capabilities the node is going to offer. Such disposable nodes need to fit into a very tight module say 1*1*1cm. The key hardware components of a sensor node include sensing unit, onboard storage unit, processing unit, communication unit, power unit and any application specific units. These size and hardware constraints require reliable packaging of sensor nodes.

Individual nodes in a WSN have limited processing speed, onboard storage, and communication bandwidth. The overall functionality of the networked nodes is only better by optimizing the battery life of individual elements so that the network operates for long periods of time. Low duty cycle operation of the sensing unit, less amount of data being transmitted by in-network data processing and multi-hop networking reducing long range transmission of data reduce the amount of power being consumed. The paper [10] discusses how to minimize the consumption of power by individual units of a node, other hardware constraints and presents a microelectromechanical systems (MEMS) device build cheaply and operate efficiently designed for environmental monitoring.

Sensor networks are deployed in harsh, hostile, or scattered environments challenging better management schemes.

Sensor network applications demand ad hoc networking techniques able to solve problems due to topology changes, node failures, and blocked nodes.

Figure 1.1: Sensor node for environmental monitoring

1.4 Location Problem

Why the location information is necessary

For each node in the network knowing its position is essential because -

1. Applications such as tracking endangered species, tracking wild fires, etc., demand the exact or at the least the approximate position of the sensors.
2. In security applications such as surveillance sensor networks, knowing the location enhances security.
3. Mobile sensor nodes are only controlled if the knowledge of their location is known.
4. Home automation, energy conservation, etc., result from location-based routing.
5. Locations are helpful in inventory management, habitat monitoring, etc.,
6. Location information is likely to create new applications.

Problem statement

In a network of thousands of nodes, given the geographical location of very few nodes (either by manually placing them or finding their location using GPS) determine the exact location of rest of the nodes whose position is unknown. The nodes whose geographical coordinates are known ahead are called as anchors.

The problem solution should be relatively simple because the solution algorithm will be executed at individual sensor nodes. The localization algorithm developed should be accurate enough using limited resources in terms of computation, memory, and communication overhead and for reasonable radio range, node density, and anchor-to-node ratio.

The localization algorithm should aim at the following design goals.

1. To meet energy requirements all the nodes should play equal role in the process of finding their location.
2. The algorithm should be developed for networks consisting of thousands of networks.
3. The algorithm should be robust enough to different types of network topologies, environments.
4. Algorithm should be tolerant to faults like node malfunctioning, power unavailability, jamming, and reachability impairments.
5. Localization algorithm should be accurate enough and should be able to find the locations all the nodes in the network.

1.5 Thesis outline

In chapter 2 literature survey of localization algorithms in wireless sensor networks is presented. In chapter 3 classical multi-dimensional scaling is presented with a mathematical description of how the positions are recovered in a relative map. In chapter 4 simulation setup is explained in terms of network parameters transmission range, k-connectivity and minimum node degree. It also presents a survey of algorithms that describe the importance of topology control by varying adjustable parameters like transmission range and how it increases the overall lifetime of the network. In chapter 5 the simulation results of MDS-MAP algorithm are presented on three different topologies, (a) Uniform random placement (b) Square grid placement, and (c) Hexagonal grid placement.

Chapter 2

Literature Review

All the motivating examples presented in chapter 1 require knowing the physical location of sensors. To solve this location problem a wide variety of location systems and techniques have been developed. This chapter presents the basic location sensing techniques and a survey of the localization schemes, though a good survey can be found in [7, 9, 11, and 12], a survey on design factors and protocols for sensor networks is presented in [8] and a survey on the research issues and challenges in the field are presented in [10].

2.1 Classification of localization algorithms

Relative versus absolute

Relative localization algorithms estimate relative position of the nodes i.e. the coordinate system is chosen by a group of nodes and is different from the original. It does not require any anchor nodes and in applications such as location aided routing relative positions are just sufficient than calculating the absolute positions.

Absolute localization algorithms on the other hand derive absolute positions of nodes making use of anchor nodes which broadcast their location information to the unknown nodes. Anchor nodes are those whose geographical locations are known prior to the localization process either by the use of GPS or through manual installation. The accuracy of the algorithm is greatly determined by the number of anchor nodes.

Centralized versus distributed

Centralized localization algorithms forward all the node measuring quantities to a central base station where the final computation or processing is carried out to derive either absolute or relative positions of the nodes.

On the other hand in distributed localization algorithms every node is responsible for performing computations to derive its position.

Range versus range free

Localization algorithms can be broadly classified into 2 categories namely range-based algorithms and range free algorithms.

In range–based algorithms fine grained information such as the distance between node pairs is exploited to compute the node locations. This distance information is obtained from,

1. Timing information, or the signal propagation time or time-of-flight of the communication signal is used to measure distance between the receiver and the reference point.
2. Time difference of arrival used to calculate the distance between two nodes.
3. Received signal strength information infers the distance between the receiver and the reference point from the fact that attenuation of the radio signal increases as the distance between the receiver and transmitter increases.

4. Direction of arrival methods use the angle at which signals are received at the reference point in some reference frame. Then position of the nodes can be calculated by triangulation technique.

On the other hand range free algorithms infer coarse grained information such as the proximity to a reference point to estimate the positions of nodes in the global network.

2.2 Location sensing techniques

Triangulation, multilateration, and proximity are the techniques used for location sensing. It uses the geometric properties of triangles to calculate node locations. Triangulation is classified into lateration, using distance measurements and angulation, using bearing angle information. In 2-dimension to calculate the node location using lateration distance information from 3 reference points is required and using angulation 2 angle measurements and 1 distance information is required.

Given measured and estimated distance values multilateration is used to maximum likelihood estimation of node positions by calculating minimum least square estimation of the error defined as the difference between measured and estimated values.

Proximity technique is used when there is no range information available. It reveals whether or not a node is in range or near to a reference point. Localization algorithms using this technique determine if a node is in proximity to a reference point by enabling the reference to transmit periodic beacon signals and whether the node is able to receive at least certain value of the beacon signals set as threshold. In a period t if it receives beacons greater than the set threshold then it is in proximity to that reference point.

2.3 Localization algorithms

2.3.1 LP and SDP based approaches

Convex position estimation: In [1], localization problem is addressed by solving connection-imposed convex constraint sets using linear programming.

Sensor node is modeled as covering a circle of radius equal to maximum range R. If two nodes a and b are in communication range then the distance between the two nodes is less than or equal to R is the radial constraint which can then be formulated as a linear matrix inequality using Schur's complements transformation. Stacking all such constraints builds the convex constraint set.

$$\left\| a - b \right\|_2 \leq R \xrightarrow{LMI} \begin{bmatrix} I_2 R & a - b \\ (a - b)^T & R \end{bmatrix} \geq 0 \qquad \rightarrow 2.1$$

Nodes equipped with laser transmitters are able to scan to check for its neighbors. When doing so if any node falls in the communication range, a cone can be imagined centered at the scanning node depending on the signal strength intensities. A cone is build for each neighbor detected and each cone contributes 3 linear matrix inequalities, 2 on the angle bound and one on the distance limit. These connectivity-induced convex constraint sets are then solved by linear programming techniques.

A similar approach to the one above is discussed in [3]. Along with the constraint specified in equation (2.1) if the exact distance $r_{a,b} \leq R$ is known one more constraint is set as,

$$\left\| a - b \right\|_2 \geq r_{a,b} \qquad \rightarrow 2.2$$

This constraint along with the constraint given in equation 2.1 impose tighter bound on the accuracy of position estimates. The constraint in equation (2) is not a convex constraint. SDP relaxation method is used to formulate it as a convex constraint.

2.3.2 APS

Ad hoc positioning system described in [2] is distributed, hop by hop positioning algorithm which utilizes the idea of distance vector routing and by deploying few GPS enhanced nodes which are otherwise called landmines determines absolute positions of all nodes in the network. Each node in the network will communicate with its immediate neighbors to exchange distance information to landmines and propagates through the entire network in a hop by hop manner. This algorithm's complexity is directly proportional to average node degree and the number of landmines. Three propagation methods are explained.

In DV-hop propagation method all the nodes in the network exchange distances to landmines in hops in a hop by hop manner so that every node has distances to all the landmines. After each landmark has distance to each other landmark correction is applied to the distance value. Each landmark calculates the correction value as distance in meters for one hop and floods the correction value through the network in a controlled manner. Controlled manner means if a node receives the correction value from the nearest landmine it will drop all subsequent ones. Now each node uses triangulation procedure to obtain its position estimation.

In DV-distance propagation method each node can calculate distance to its neighbors using the received signal strength and uses this value instead of distance in hops to find its position estimation. Euclidean propagation method uses true Euclidean distances to landmines and follows the same procedure as the DV-hop method.

2.3.3 AHLoS

Ad hoc localization system described in [4] is a distributed iterative algorithm. It has two phases. In the ranging phase nodes measure distances to their neighbors either using the received signal strength information or the time of arrival measurements. This paper gives details of finding distance information from both the RSSI and ToA measurements and finds out that the distance information obtained from ToA using RF and ultrasound is more accurate.

In the estimation phase each unknown node estimates the Euclidean distance to its neighbors based on the received beacon positions. Unknown node that receives the position of a beacon estimates its Euclidean distance to the beacon and then broadcasts its position to its neighbors as a beacon. This is an iterative process carried out until all or most of the unknown nodes estimate their positions.

Having both the measured and estimated distance values the algorithm uses iterative multilateration to find the positions of the unknowns. Iterative multilateration uses atomic multilateration. Necessary condition for an unknown to estimate its position using iterative multilateration is that the unknown should be within range of at least 3 beacons. The maximum likelihood estimator of the position is obtained by minimizing the error value i.e. the differences in the distances of measured and estimated values. The algorithm starts at an unknown which has at least 3 beacons in its range and finds the maximum likelihood estimation of its position using atomic multilateration, after which the unknown is treated as a beacon. This process continues till all the nodes are able to find 3 or more beacons in their range to find the ML estimate of their position.

2.3.4 GPS less technique

Range free or course-grained connectivity-metric method for localization in outdoor environments is described in [5]. This approach assumes spherical radio propagation and same transmission range for all the nodes. Nodes with known positions form overlapping regions of coverage in the network denoted as R_i. These reference nodes transmit periodic beacon signals. A node listens or receives beacon signals from various reference points for a duration t. Each reference point transmits S samples of its position in this time period t periodically such that a node is able to have enough connectivity metric value for a reference node i. Connectivity metric is given as

$$CM_i = \frac{\text{(Number of beacon signals received for the duration t Transmitted by refernce point i)}}{\text{(Number of packets sent or transmitted by refernce point i for this period t)}} \times 100$$

Based on the connectivity metric value a node determines the set of reference points to which it should be localized to get its position estimation. This is done by verifying if the connectivity metric value for a reference point i is above a threshold value. Then the node estimates its position as the centroid of the overlap regions of the new reference set.

2.3.5 APIT

In [6] the author describes a novel area-based range-free localization algorithm APIT. This approach assumes homogenous sensor network and anchors equipped with high power transmitters. In this area-based approach the entire area is isolated into triangles with vertices as anchors as many as all the combinations of anchor nodes. The next step is to narrow down the presence of a unknown node in a particular region. This procedure is called point-in-triangulation test. An unknown node forms a set of anchor nodes from which it is able to receive beacon signals. For every combination of forming a triangle it tests whether it falls within the triangle. After required accuracy is achieved it determines its position as the center of gravity of intersection of all the triangles it determined.

2.3.6 GPS free approach

The approach presented in [13] is an infrastructure free, self positioning and distributed. It does not employ anchor nodes or does not use any GPS enhanced nodes and is only designed for relative position estimation. It builds a coordinate system by making use of the distance measured between neighbors using angle of arrival technique. The algorithm begins with each node building its own local coordinate system choosing its own frame of reference with it at the origin and 1-hop and 2-hop neighbors as the elements. After each node is ready with its local coordinate system, an arbitrary nodes coordinate system is chosen say i as the reference and all other nodes in the network rotate their local coordinate system to the direction of system i. Then all the 2-hop neighbors of the system i merges its local coordinate system into the reference coordinate system and this is continued still all the nodes merge into the reference system.

This algorithm can’t be used in applications where absolute position of nodes is required.

2.3.7 MDS based approach

Multi-dimensional scaling (MDS) based approach presented in [14] is a centralized algorithm which can be used for both relative position estimation and absolute position estimation of nodes. Provided with just the connectivity information absolute position of the nodes can be determined with limited accuracy and with the distance information available position estimation with better accuracy is obtained. With no anchors available the algorithm settles with relative positions and with only 3 anchors available, absolute positions are calculated in 2-dimension. In this thesis, MDS is tested on various topologies and thus the simulation results are presented in chapter 5.

Chapter 3

Classical Multi-dimensional Scaling

Multidimensional scaling (MDS) is a data analysis technique used to model similarity or dissimilarity data on a set of objects as distances between points in a geometric plane. The main idea in performing MDS is to present graphical representation of data, more understandable than the data itself.

Though multidimensional scaling has its origin in psychophysics, its existence in a variety of related geometric models explains the different purposes it is used for. MDS is used as a model for similarity judgments, sociologists use it for structural hypothesis testing and in the present day it is mostly used as a data exploration technique. In the process of solving the localization problem we explore multidimensional scaling as a data exploration technique.

3.1 Classification of Multidimensional scaling

Many different types of MDS exist. This classification done is based on either geometry or dimension used to map the data, or the number of similarity matrices used in the scaling, or the statistical error or the stress function being optimized, or the mapping function. It's often that the proximities recorded in the experiments are transformed in some optimal way. Proximities in this context are called pseudo distances. MDS models can also be classified as metric and non-metric MDS based on how proximities and the pseudo distances are related.

Metric MDS

If the relation between the proximities and the pseudo distances is continuous then they fall under metric MDS. The first model for MDS is Classical metric multidimensional scaling or simply classical MDS. It is the simplest for it gives analytical solution requiring no iterations to compute the coordinates. Classical MDS is discussed in detail later in this chapter.

Other metric MDS models are absolute MDS which means that the proximities and the pseudo distances are the same, ratio MDS means the ratio of proximities and pseudo distances is an integer, interval MDS where the pseudo distances equals the sum of a constant and ratio of the proximities, i.e., if p_{ij} are the proximities among two sets of objects and d_{ij} are the pseudo distances then

$d_{ij} = a + bp_{ij}$

If the data are related to algebraic operations then they are metric MDS.

Non-metric MDS

In non-metric MDS models pseudo distances are related to the proximities as,

If $p_{ij} < p_{mn}$, then $d_{ij} < d_{mn}$

3.2 Classical MDS

Classical scaling can be performed only if the similarity or dissimilarity matrix is distances and computes the coordinates that explain the dissimilarity matrix. Let $S_{n\times 2}$ be the similarity matrix, where each row represents the coordinates of that point i along 2 dimensions. We are considered with calculating the distances between all the n points.

The Euclidean distance between any two coordinate points say $\mathbf{x}(x_1, x_2, \ldots\ldots, x_n)$ and $\mathbf{y}(y_1, y_2, \ldots.., y_n)$ is given by norm of the difference vector $\mathbf{x}$-$\mathbf{y}$ represented by $\|\mathbf{x}\text{-}\mathbf{y}\|$.

$$\|\mathbf{x}\text{-}\mathbf{y}\| = \sqrt{(\mathbf{x}-\mathbf{y})'(\mathbf{x}-\mathbf{y})}$$

$$\Rightarrow \|\mathbf{x}\text{-}\mathbf{y}\| = [\,(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots\ldots\ldots + (x_n - y_n)^2\,]^{1/2} \qquad \rightarrow 3.1$$

Let the matrix of squared Euclidean distances be denoted by $\Delta^{(2)}(S)$ or simply $\Delta^{(2)}$ and Δ_{ij} represent the distance between coordinates i and j. Then using equation 3.1

$$\Delta^{(2)}(S) = \Delta^{(2)} = \begin{bmatrix} 0 & \Delta_{12}^2 & \Delta_{13}^2 & \cdots & \Delta_{1n}^2 \\ \Delta_{21}^2 & 0 & \Delta_{23}^2 & \cdots & \Delta_{2n}^2 \\ \vdots & & \ddots & & \vdots \\ \Delta_{21}^2 & \Delta_{12}^2 & \Delta_{n3}^2 & \cdots & 0 \end{bmatrix}, \qquad \rightarrow 3.2$$

Where, $\Delta_{ij}^2(S) = \Delta_{ij}^2 = \sum_{a=1}^{2} (s_{ia} - s_{ja})^2$

$$\Rightarrow \Delta_{ij}^2 = \sum_{a=1}^{2} (s_{ia}^2 + s_{ja}^2 - 2s_{ia}s_{ja}) \qquad \rightarrow 3.3$$

Substituting 3.3 in the individual elements of the matrix in equation 3.2, we get

$$\Delta^{(2)} = \sum_{a=1}^{2} \begin{bmatrix} s_{1a}^2 & s_{1a}^2 & \cdots & s_{1a}^2 \\ s_{2a}^2 & s_{2a}^2 & \cdots & s_{2a}^2 \\ \vdots & & \ddots & \vdots \\ s_{na}^2 & s_{na}^2 & \cdots & s_{na}^2 \end{bmatrix} + \begin{bmatrix} s_{1a}^2 & s_{2a}^2 & \cdots & s_{na}^2 \\ s_{1a}^2 & s_{2a}^2 & \cdots & s_{na}^2 \\ \vdots & & \ddots & \vdots \\ s_{1a}^2 & s_{2a}^2 & \cdots & s_{na}^2 \end{bmatrix}$$

$$+ 2 \sum_{a=1}^{2} \begin{bmatrix} s_{1a}s_{1a} & s_{1a}s_{2a} & \cdots & s_{1a}s_{na} \\ s_{2a}s_{1a} & s_{2a}s_{2a} & \cdots & s_{2a}s_{na} \\ \vdots & & \ddots & \vdots \\ s_{na}s_{1a} & s_{na}s_{2a} & \cdots & s_{na}s_{na} \end{bmatrix}$$

Where, diagonal elements s_{ij}^2 for i = j are 0.

$$\Rightarrow \Delta^{(2)} = c1' + 1c' - 2SS', \qquad \rightarrow 3.4$$

Where, 1 is an n×1 vector of ones, and the matrix say B=SS′ is called the scalar product matrix, and c is a vector consisting the diagonal elements of the scalar product matrix, i.e., $c = \sum_{a=1}^{2} s_{ia}^2$.

3.2.1 Eigen decomposition of a matrix

Every square matrix can be decomposed into product of several matrices. Eigen decomposition is one which can be performed on only symmetric ones. Consider a square matrix A of size n×n. Matrix A can now be decomposed into

$$A = Q\Lambda Q' \text{ or } AQ = Q\Lambda \qquad \rightarrow 3.5$$

Where Q is orthonormal and Λ is a diagonal matrix. A matrix is orthonormal if QQ′ = I which means $Q' = Q^{-1}$. Equation 3.5 can also be written as a system of Eigen equations as

$Aq_i = \lambda_i q_i$, where $q_i \neq 0$ and i=1, 2,……, n

The values in the diagonal of Λ are the Eigen values of A and the column vectors of Q are the Eigen vectors of A.

3.2.2 Recovery of coordinates

Given the matrix of similarities S between pair of objects the first step is to calculate the matrix of squared distances $\Delta^{(2)}(S)$. The second step is to arrive at the scalar product matrix $B = SS'$ from $\Delta^{(2)}(S)$, which can be done as follows.

Rewriting equation 3.4, $\Delta^{(2)} = c1' + 1c' - 2SS'$ and multiplying both sides by centering matrix $T = I - n^{-1}11'$ where I is the identity matrix and 1 is a vector of ones.

$T\Delta^{(2)}T = T(c1' + 1c' - 2SS')T = Tc1'T + T1c'T - T(2B)T.$

Centering a vector of ones yields a vector of zeros yielding the first two terms in the above equation to zeros. Thus,

$T\Delta^{(2)}T = -T(2B)T$ and since B here is column centered it has no effect. Multiplying both sides by -1/2 gives

$-\frac{1}{2}T\Delta^{(2)}T = B.$

Since B is symmetric it can be decomposed as given in equation 3.5.

$B = Q\Lambda Q' = (Q\Lambda^{1/2})\,(Q\Lambda^{1/2})' = SS'$

$\Rightarrow S = Q\Lambda^{1/2}$

In calculating S the negative Eigen values and its Eigen vectors are ignored. And the recovered matrix S is rotated and has a different coordinate system than the original.

Thus in our localization problem solving, classical multidimensional scaling yields relative location estimation of the nodes. And availability of at least 3 anchors (2-dimension) or 4 anchors (3-dimension) the relative map can be transformed to an absolute map.

Chapter 4

System-level Simulation setup of an Ad Hoc network

This chapter presents the characteristics of a multi-hop wireless network namely minimum node degree, k-connectivity, and change in the topology with transmission power or transmission range. This helps how to set the parameters for system-level simulation of multi-hop wireless networks. We start with discussing the basic graph theory used to represent a wireless network, define the parameter terms and then present simulation results presented in [15].

4.1 Graph Theory

A graph is a pair G = (V, E) where the elements of V are vertices are points and the elements of E are its edges or lines. E is a set of two element subsets of V. Wireless multi-hop networks are represented as a graph with V as the set of nodes and E as the set of wireless multi-hop communication links between node pairs.

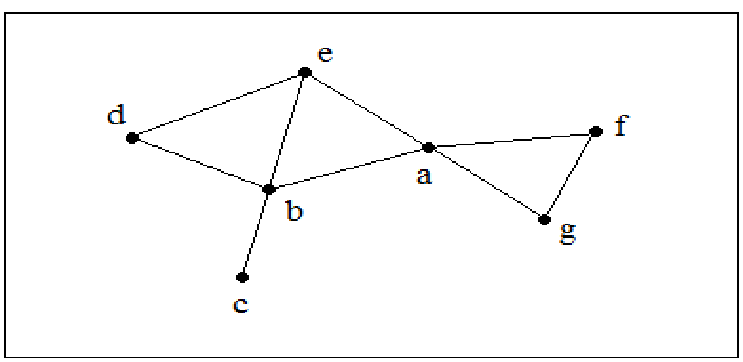


Figure 4.1: Topology of an ad hoc network

In the above network V = {a, b, c, d, e, f, g} and E = {(a, b), (a, e), (a, f), (a, g), (b, c), (b, d), (b, e), (f, g)}.

Also we assume that the communicational links are bidirectional which means communication link relations on node pairs are symmetric, i.e., in the above set E a pair (a, b) also means (b, a). In terms of graph theory we only consider undirected graphs.

4.2 Network characteristics

The fundamental characteristics of a multi-hop wireless network are minimum node degree, k-connectivity and topology. Topology can be well controlled by dynamically varying the transmission power or transmission range at the nodes. Simulation results in this chapter depict how the transmission range varies with given node density ($\frac{n}{A}$) for almost k-connected network. Almost a k-connectivity network implies probabilities to build k-connected networks. With this relation one can achieve

- A fully connected network,
- An ad hoc network with no isolated node,
- A network where each node a certain number of neighbors.

given number of nodes per unit area, and the minimum transmission range.

4.2.1 Topology

Topology of a network is the set of wireless communication links between node pairs utilizing a routing mechanism. Desired topology can be achieved by adjusting controllable parameters like transmit power, antenna direction. Most of the wireless networks operate on batteries and less battery life of a node results in node failure which leads to sub networks considerably reducing the capacity, increasing end to end packet delays. Hence controlling the topology of the network by adjusting the transmission power dynamically at each node plays an important role in the wireless network lifetime. One method given in [16] is to use energy efficient distributed algorithm where each node decides its transmission power maintaining connectivity and the resulting topology increases overall network lifetime. This cone-based topology control algorithm is simple and efficient, not dependant on the deployment region, does not require the location information of the nodes, chooses minimum power paths between node pairs having more than one wireless link, makes fewer assumptions regarding the propagation channel, and each node requires only local information for determining its transmission power. This algorithm is carried out in two phases.

Neighbor discovering phase: In this phase each node beacons with increasing power, initially p to maximum power P in steps ($p < P$) to build its neighbor set. Each time a node increases its power from p to P in steps it determines new neighbors and adds them to its neighbor set. This is done based on a node u broadcasting a message and the nodes that receive u's message send back an acknowledgement to u. A node stops building its neighbor set if the transmission power reaches the maximum value P or if it determines that there is at least one neighbor for any cone with angle α. For $\alpha <= \frac{2\pi}{3}$ the algorithm yeilded good connectivity and minimum power paths.

Edge removal phase: In this phase the node degree is reduced without effecting the minimum power paths and the connectivity by removing redundant edges in each nodes neighbor set. This phase is utilized to reduce interference and incresing throughput.

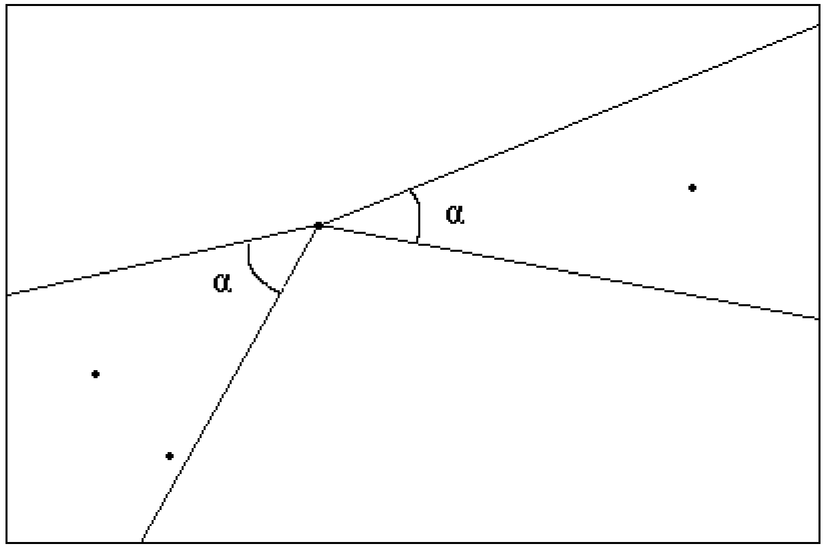


Figure 4.2: Neighbor set determination

4.2.2 Minimum Node degree

Let the graph G = (V, E) is nonempty and represents n nodes and the communication links in a wireless multi-hop network. The degree of a node v, denoted as $d_G(v)$ is the set of neighbors or its number of links. In terms of graph theory $d_G(v)$ is the number of edges $|E_G(v)|$ at v. If a node u is isolated then $d_G(u) = 0$.

The minimum node degree of G is given by

$$D_{min}(G) = \min \{ d_G(v) \mid v \varepsilon V \}.$$

4.2.3 k-connectivity

A graph is said to be connected if for every pair of vertices there exists a path between them in G. Otherwise the graph is said to be disconnected.

If a wireless multi-hop network is connected then a node can communicate with each other node over one or many communication hops.

And a graph is said to be k-connected if every two nodes are connected via k different multi-hop paths. Figure 4.1 shows a 1-connected network of 7 nodes.

Equivalently, a graph is k-connected if removal of (k-1) nodes still leaves the graph connected or it's the minimum number of nodes whose removal or failure will disconnect the graph.

4.3 Probability of minimum node degree vs. node density, transmission range

The following assumptions are made on the network.

1. Random uniform distribution of nodes over a large area A.
2. Usage of omni-directional antennas at the nodes and the transmission range r_0 of a node follows

   $P(r) \alpha r^{-\gamma} P_0$,

   Where γ is the path loss exponent and P(r) is the received power at a distance r.
3. All the communication links are bidirectional.

   Given an ad hoc network with n nodes and homogeneous node density $\rho = \frac{n}{A}$ nodes per unit area, probability of the event that no node in the network is isolated is given by

   $$P(d_{min} > 0) = (1 - e^{-\rho\pi r_0^2})^n \quad \rightarrow 4.1$$

   And the probability that the network has a minimum node degree $d_{min} \geq n_0$ is given as

   $$P (d_{min} \geq n_0) = \left(1 - \sum_{N=0}^{n_0-1} \frac{(\rho\pi r_0^2)^2}{N\,!}\right)^n \quad \rightarrow 4.2$$

4.4 Simulation results

A uniform random generator is used to generate the n coordinates and distribute them uniformly over an area of 10^6 unit^2. A radio range is set and minimum node degree of the resulting network is calculated for each set of n and r_0 values.

The simulation results support the analytical results given in equations (4.1) and (4.2)

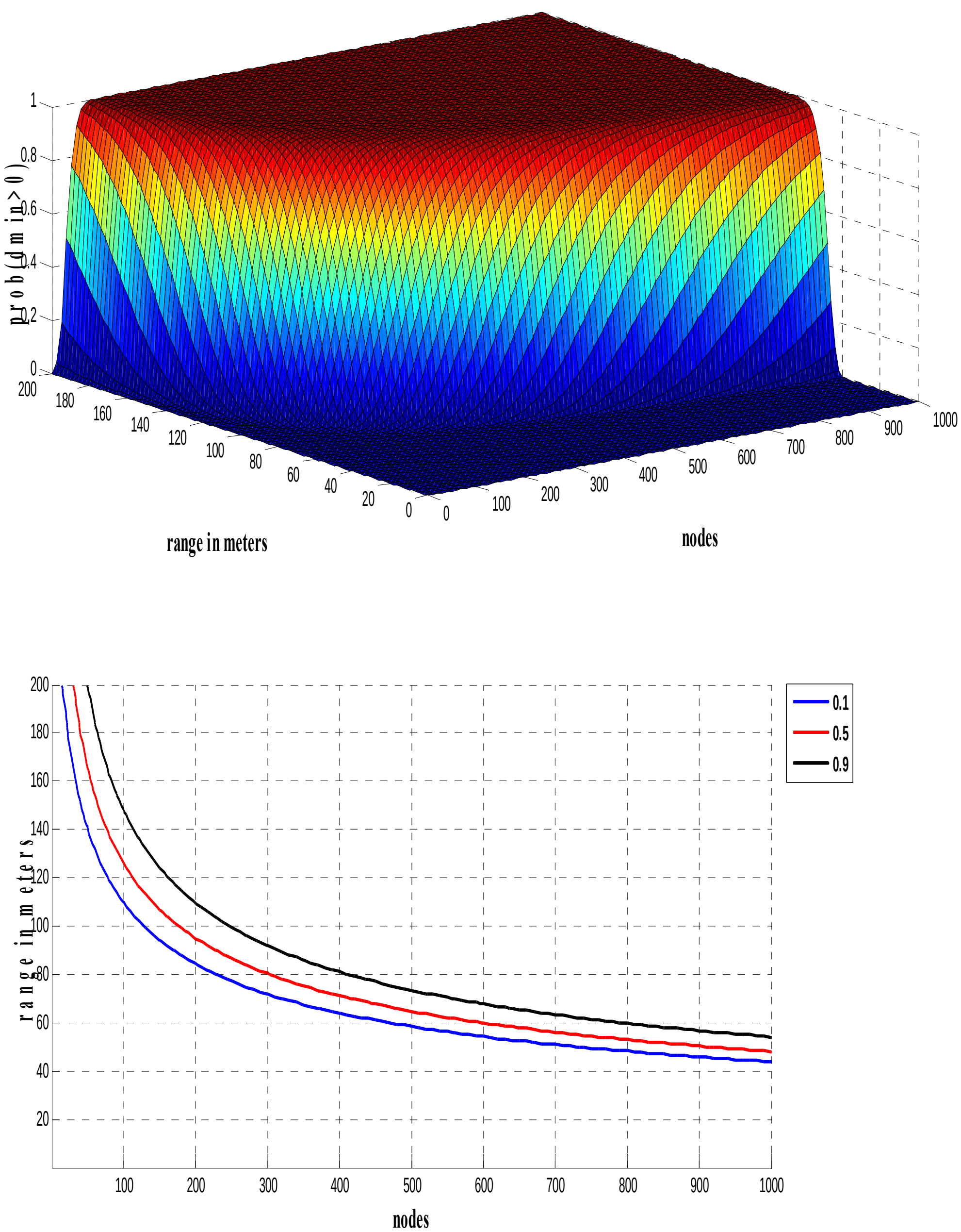


Figure 4.3: Probability that no node is isolated (System area = 10^6 unit^2)

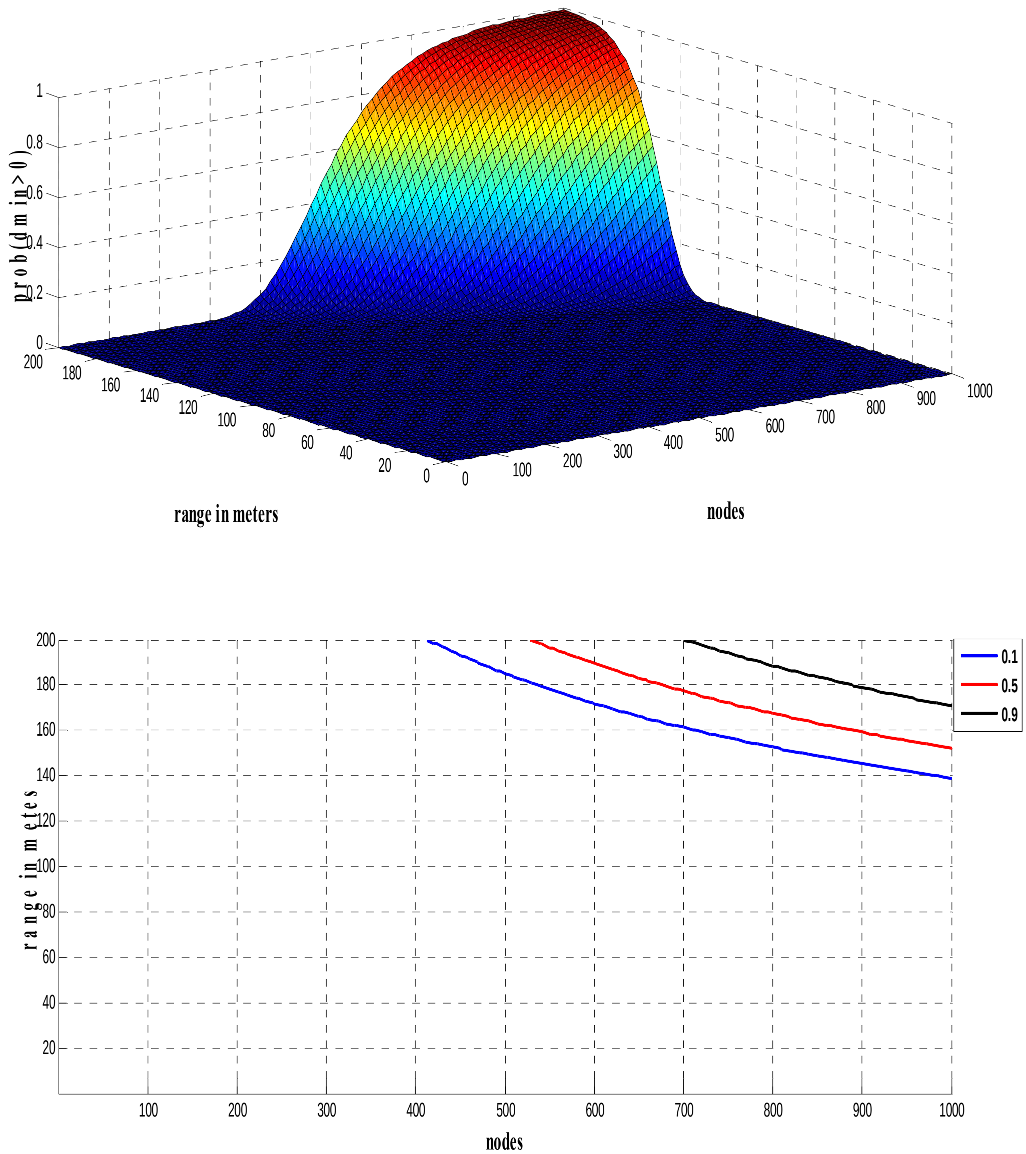


Figure 4.4: Probability that no node is isolated (System area = 10^7 unit^2)

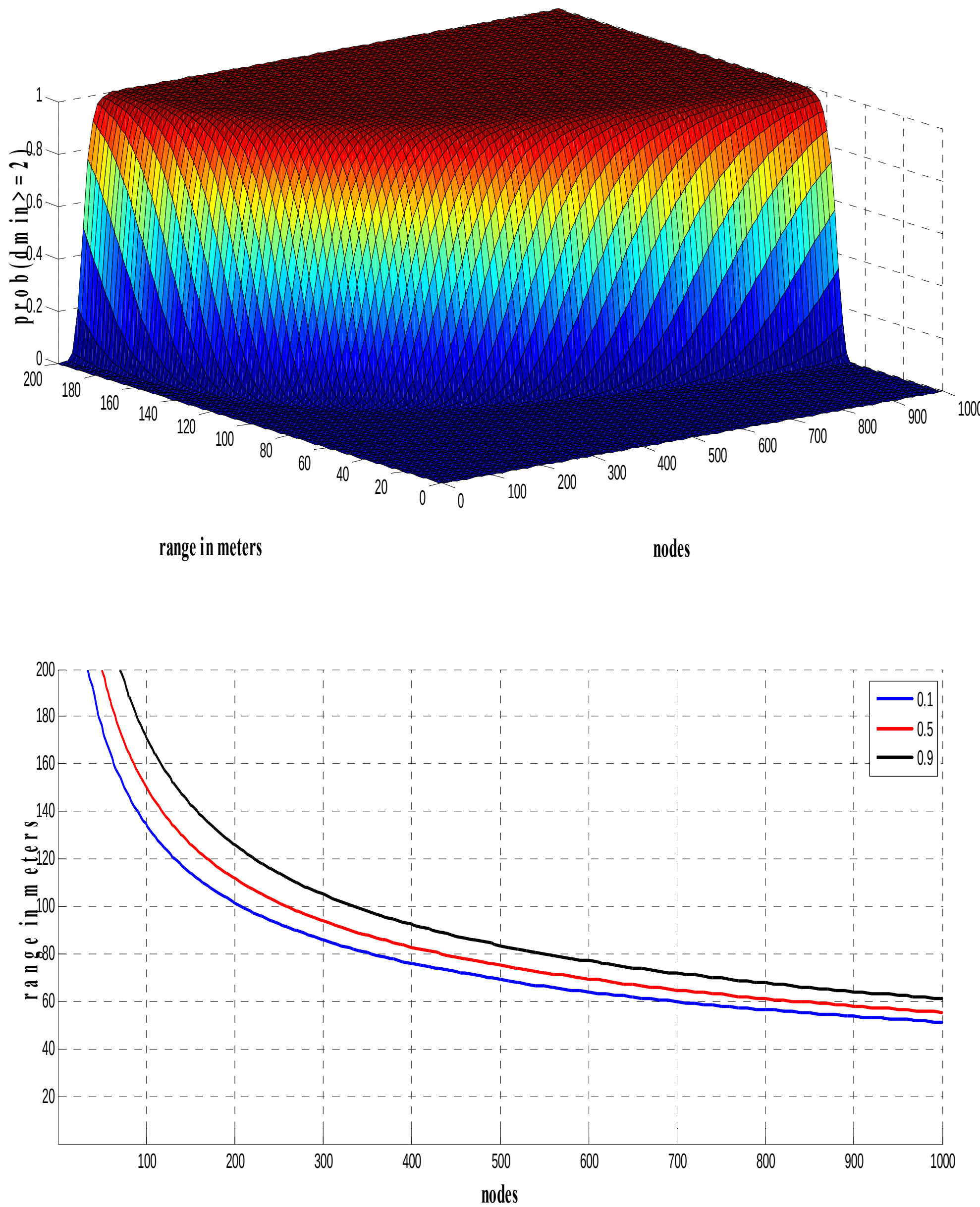


Figure 4.5: Probability that each node has at the least two neighbors (System area = 10^6 unit^2)

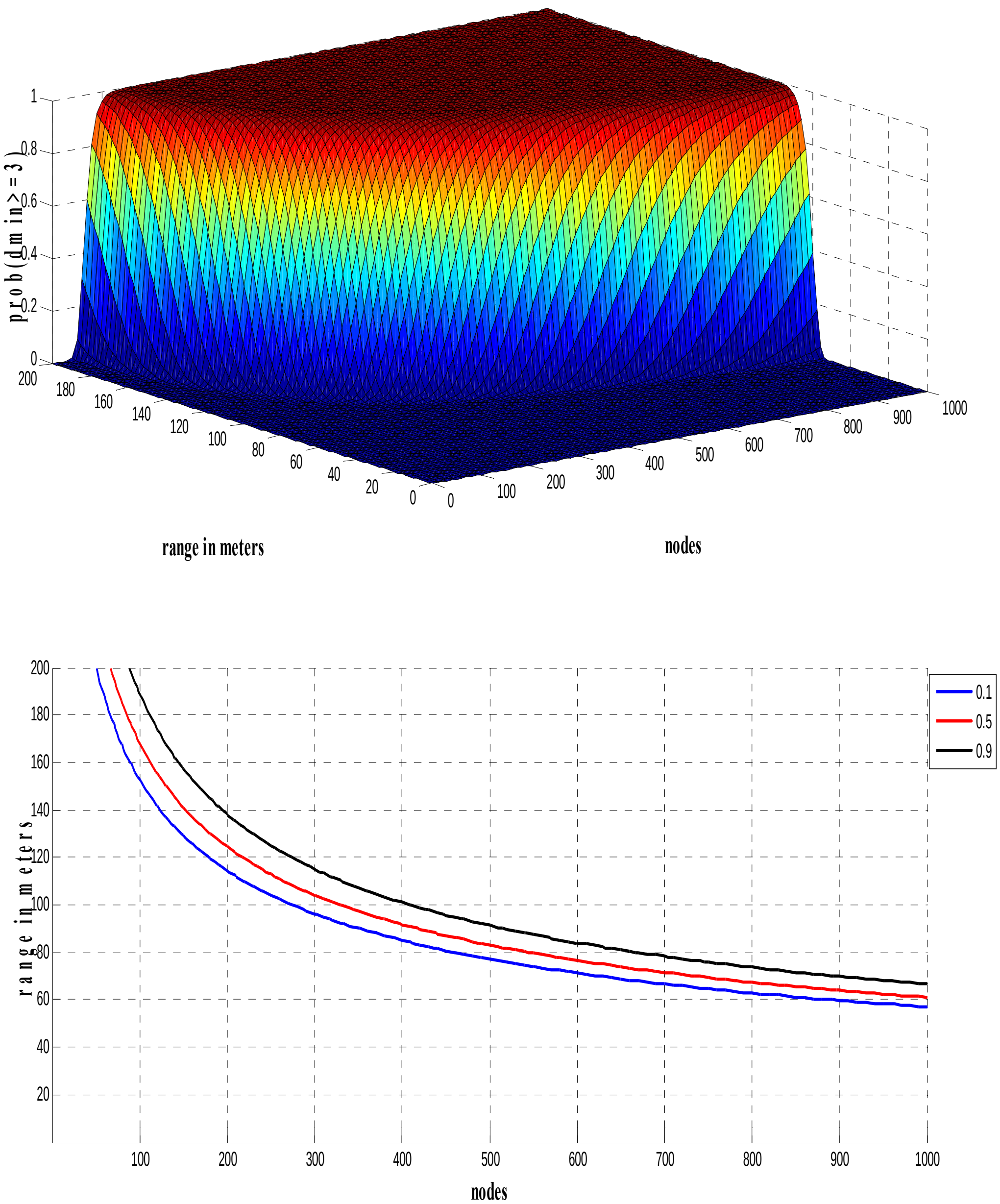


Figure 4.6: Probability that each node has at the least three neighbors (System area = 10^6 unit^2)

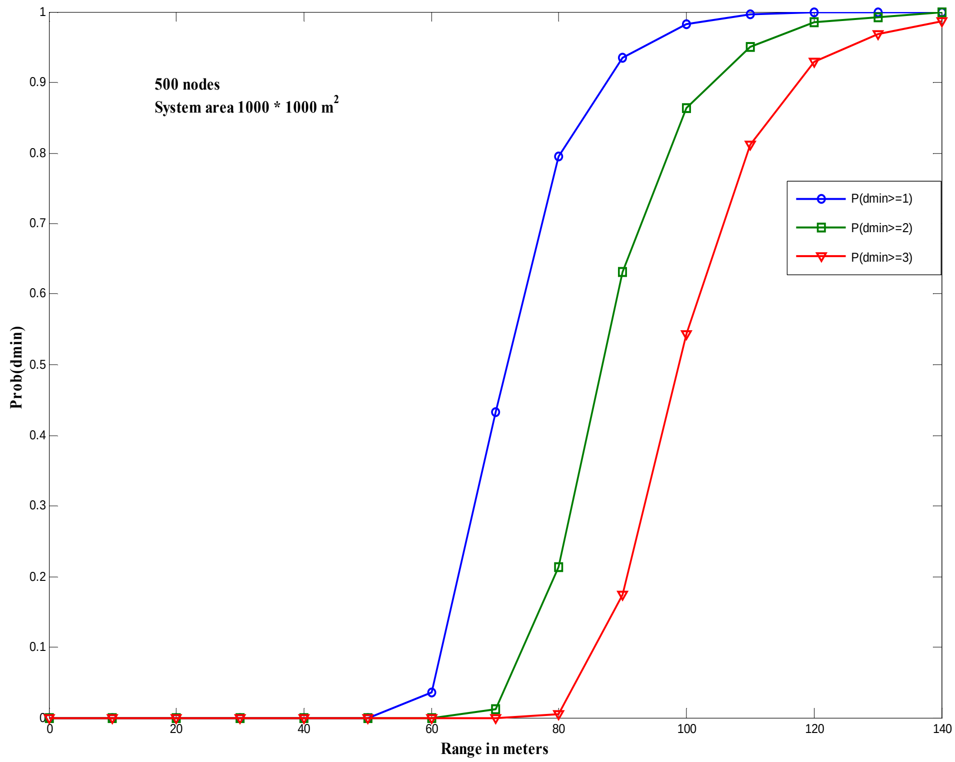


Figure 4.7: Probability of $d_{min} \geq n_0$

Chapter 5

MDS-MAP

All the localization algorithms in literature start with, the localization problem can be solved by installing GPS at each sensor node but since wireless ad hoc networks are densely populated using GPS at each node increases the overall cost. A single GPS costs around $100 and with incremental uplink and downlink costs the idea is of no interest.

MDS-Map algorithm is based on Multi-Dimensional Scaling which is a data analysis technique discussed in chapter 3. Some applications require to know the absolute geographic positions of nodes and some applications just need the relative position of the nodes. With MDS-MAP both relative map and absolute map are possible.

Relative map

Relative map would give us a new arrangement of the underlying network such that the distances between the nodes in this new arrangement maintain the desired distance relationships. Relative map does not provide us with the absolute coordinates of the nodes but provides with useful information in cases when not enough anchor nodes are available to find the absolute coordinates. Relative map is rotated and flipped version of the original network.

Absolute map

Nodes which know their position ahead are called anchor nodes. These nodes know their position either through GPS or through manual installation. With only 3(2-dimensional), 4(3-dimensional) such anchor nodes available relative map can be easily transformed to absolute map.

5.1 MDS-MAP procedure

It consists of 3 steps.

1. In the first step the shortest distances between every pair of nodes is calculated using either Dijkstra's or Floyd's all pairs shortest path algorithm. This is the distance matrix that serves as input to the Multi-dimensional scaling in step 2.
2. Classical multi-dimensional scaling is applied to the distance matrix. As stated in chapter 3 MDS gives relative map of the true node positions.
3. In this step we transform the relative map into absolute map give sufficient number of anchor nodes.

Simulations are carried out for two different cases. First case is when only the connectivity or proximity information is available. Each node by some local communication channel knows the nodes that fall into its range or that it can exchange data with single hop link. We approximate the distance value as 1 if a node falls into the range of another or else zero. Using this edge matrix of ones and zeros we calculate shortest path distances between every node pairs using Dijkstra's algorithm. Second case is when the edge matrix input to the shortest path algorithm consists of distance between nodes with limited accuracy.

5.2 Simulation results

The simulation results are carried out on 3 different topologies. Nodes randomly generated and uniformly distributed over a square area, n^2 nodes placed in nr by nr square grid with placement errors, and nodes placed on hexagonal grid with placement errors. With placement error e_p, $r \times e_p \times N(0, 1)$ values are added to node's positions where r is the unit length and N (0, 1) is a zero mean unit variance normal distribution. With distance information available we model true distance d blurred as $d \times (1 + N(0, e_r))$ where e_r is the range error and $N(0, e_r)$ is a zero mean e_r variance normal distribution.

In the case of uniform random placement 200 random nodes with range equal to 3 meters are uniformly placed on a 20×20 meter^2 area. Since there is no constraint on the placement of anchors, 4 anchors are chosen randomly. With only connectivity or proximity information available, that is distance vector to MDS technique is build by taking 1 if any two nodes are in communication range and a 0 if not this algorithm derives absolute position of nodes shown in figure 5.3. The intermediate relative map is shown in figure 5.2. With the knowledge of distances between nodes measured by some method this algorithm derives the absolute map of the network as shown in fig 5.5 and the intermediate relative map are shown in figure 5.4. We modeled known distances between nodes as true distances blurred with 5% range error. Figure 5.6 shows the performance comparison of MDS-MAP when proximity information and distance information are available. Different connectivity levels correspond to varying ranges of nodes. The same set of experiments is conducted for square grid and hexagonal grid deployments.

5.2.1 Uniform random placement

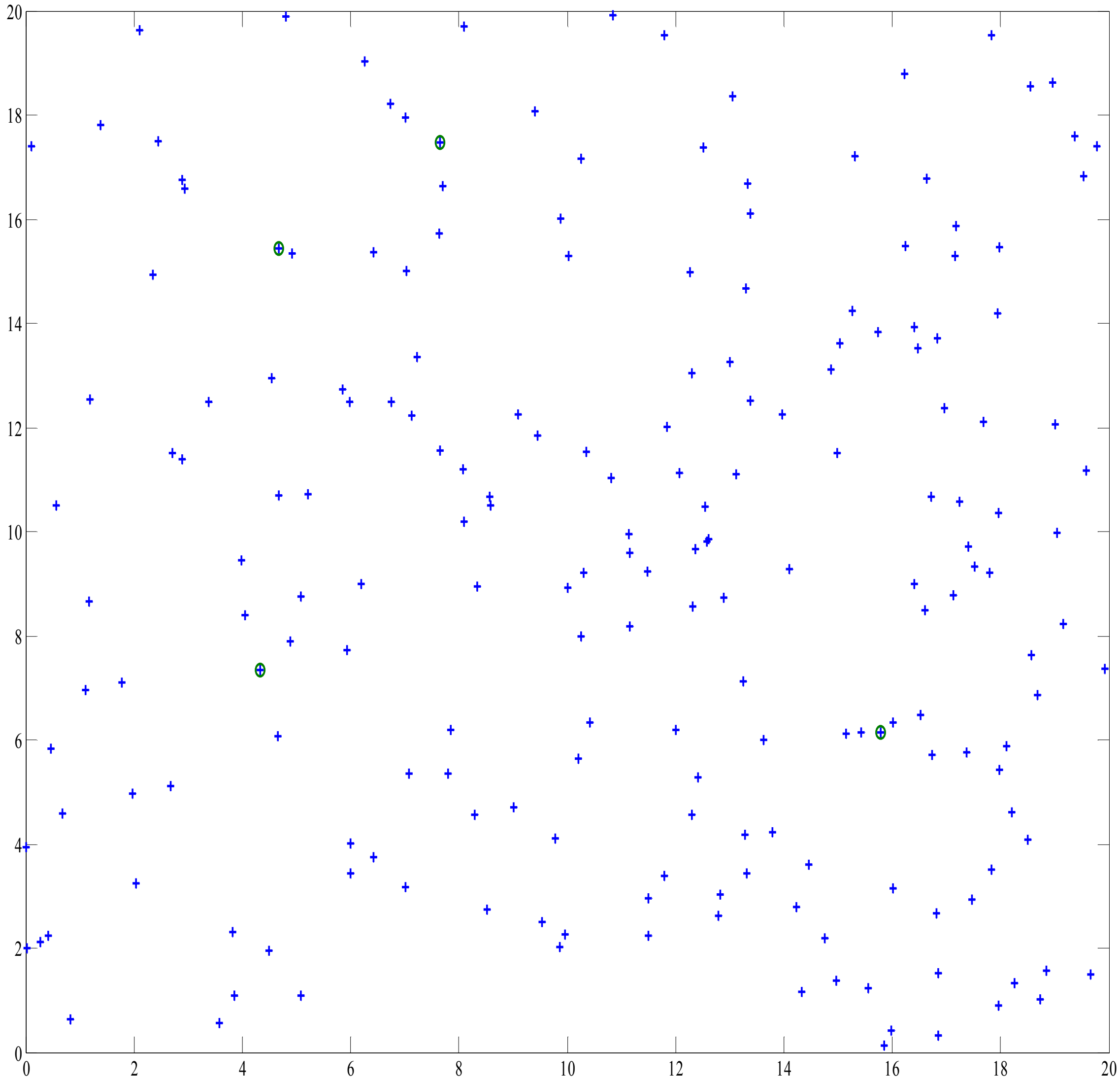


Figure 5.1: 200 nodes randomly generated to cover 20*20 meter^2 square area. Range of each node is 3 meters which resulted in an average connectivity of 12.89. Above graph also shows 4 anchor nodes which are selected randomly.

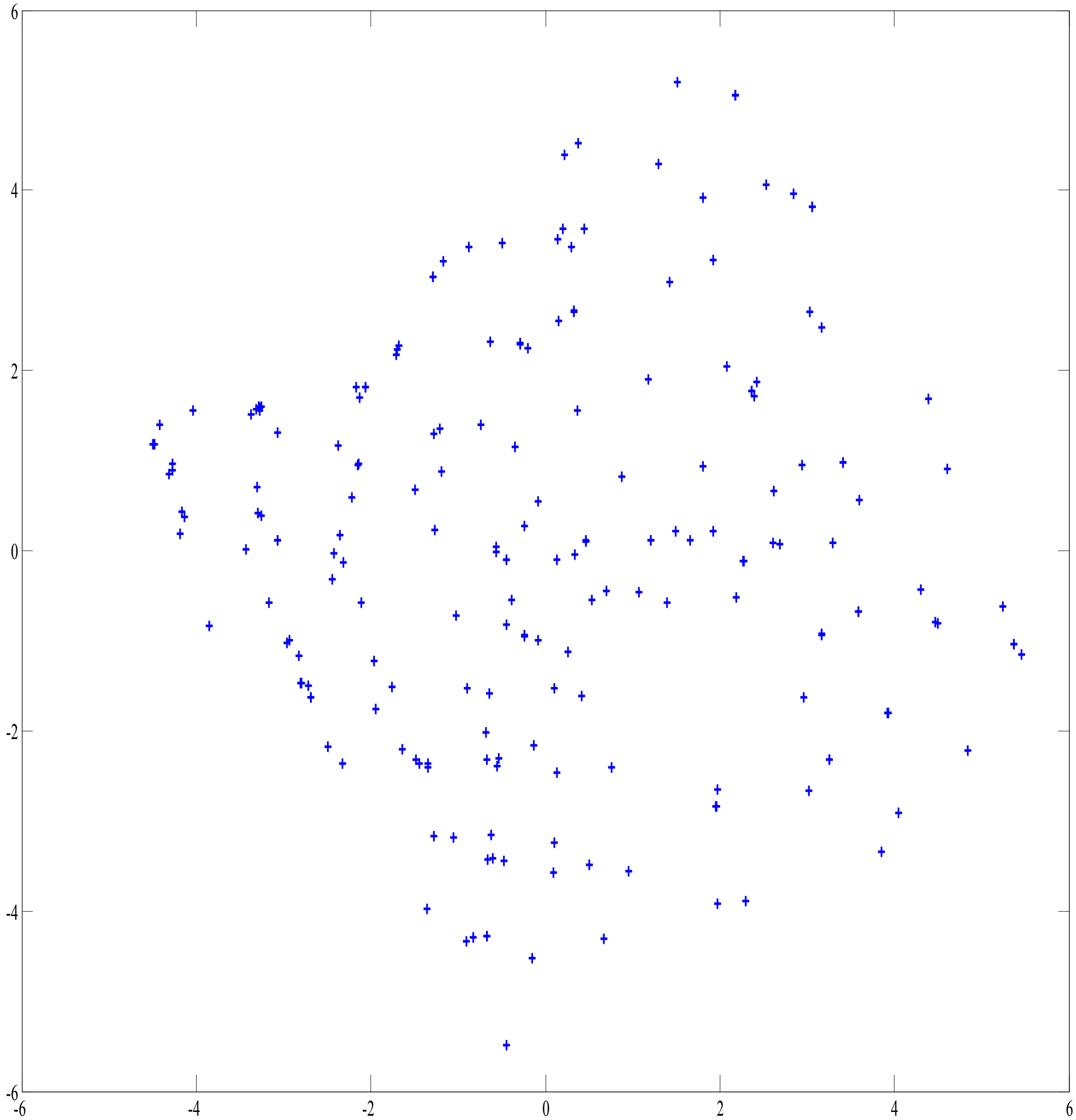


Figure 5.2: Multi-Dimensional Scaling (MDS) of the network using proximity information only i.e., distance vector input for MDS is built by taking 1 if connectivity between any 2 nodes exists or else 0.

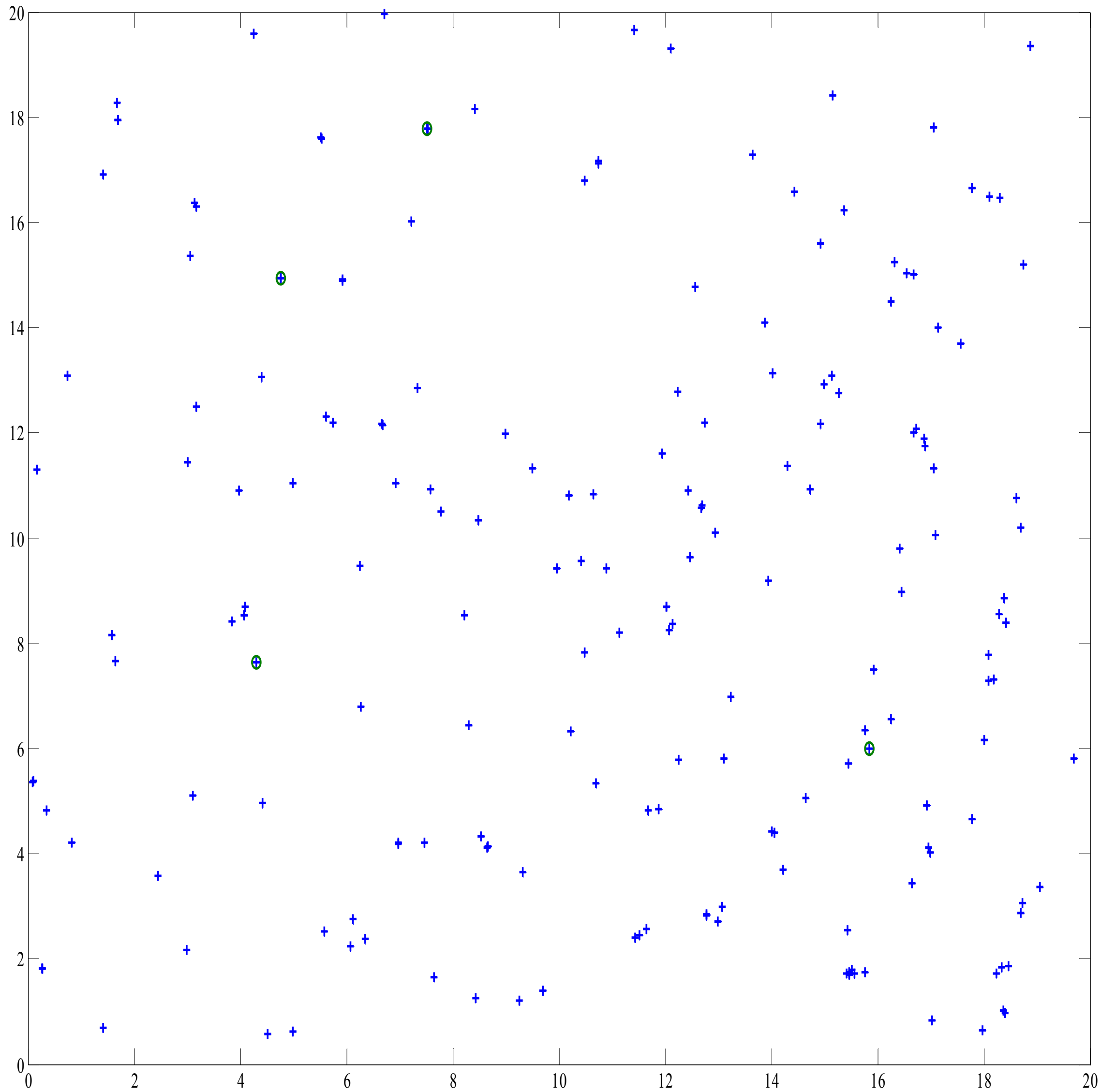


Figure 5.3: Final position estimation of the network using proximity information by translation, reflection, orthogonal rotation, and scaling (MDSMAP) of the result obtained through MDS using the 4 anchor nodes. The average position estimation error obtained is 0.8584 meters which is 28.6138% of range.

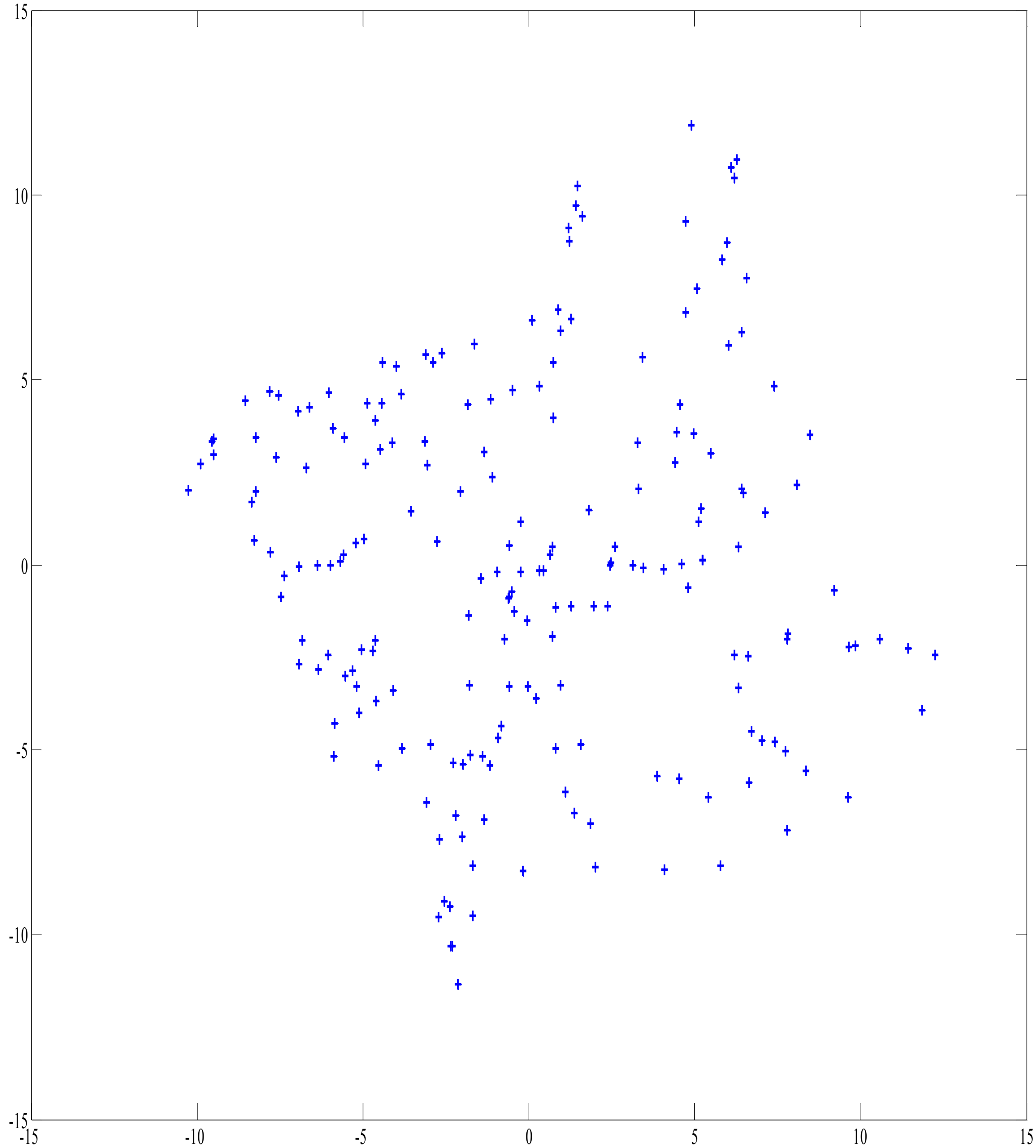


Figure 5.4: Multi-Dimensional Scaling (MDS) of the network using available distance information (distance between nodes with 5% range error) as the distance vector input.

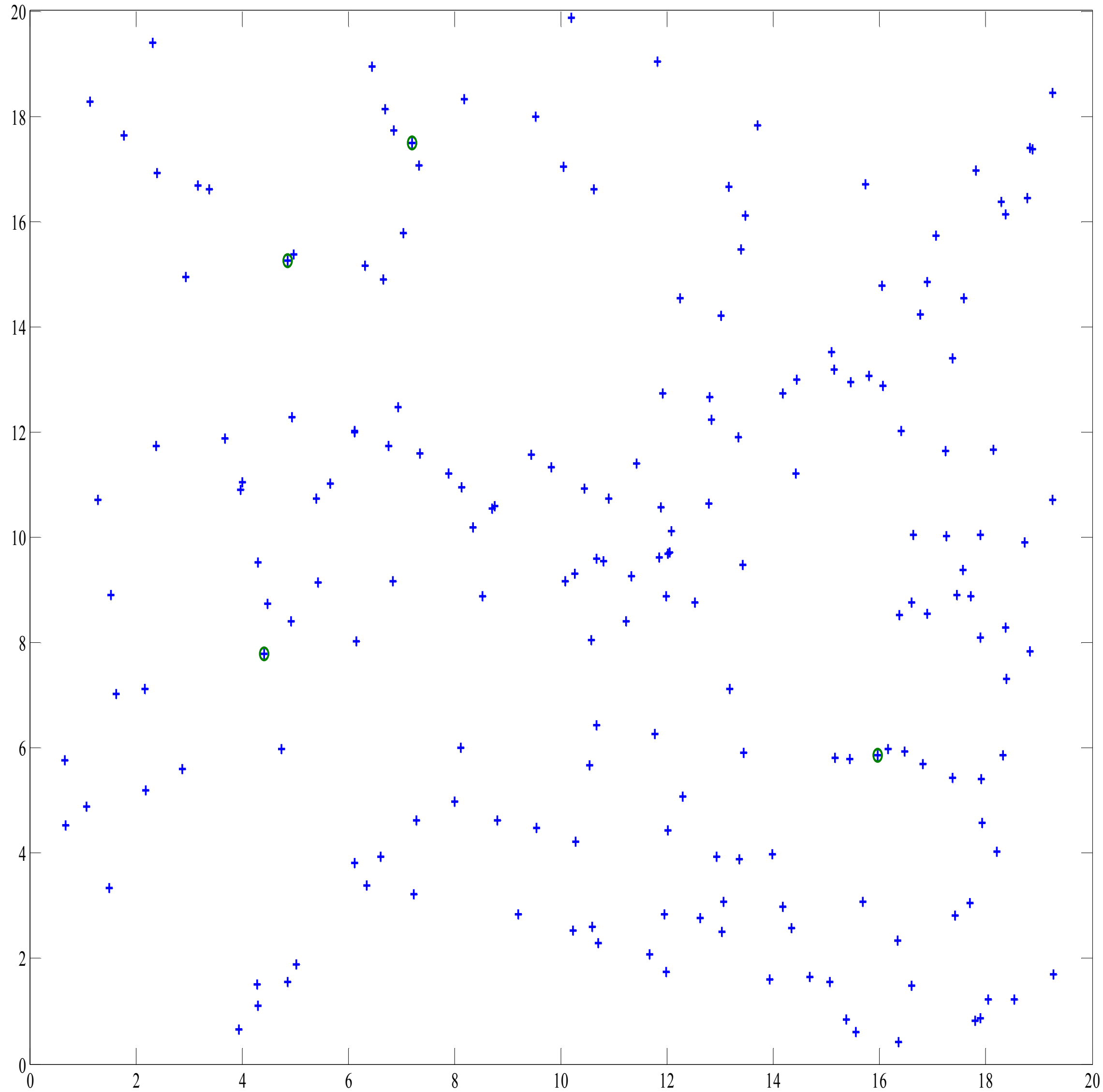
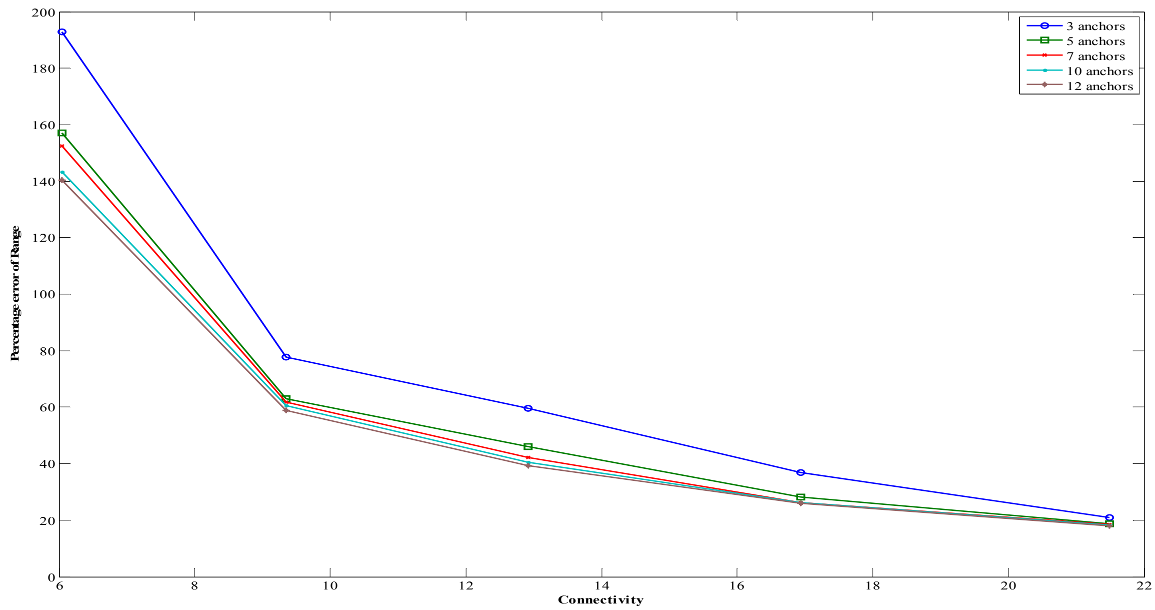
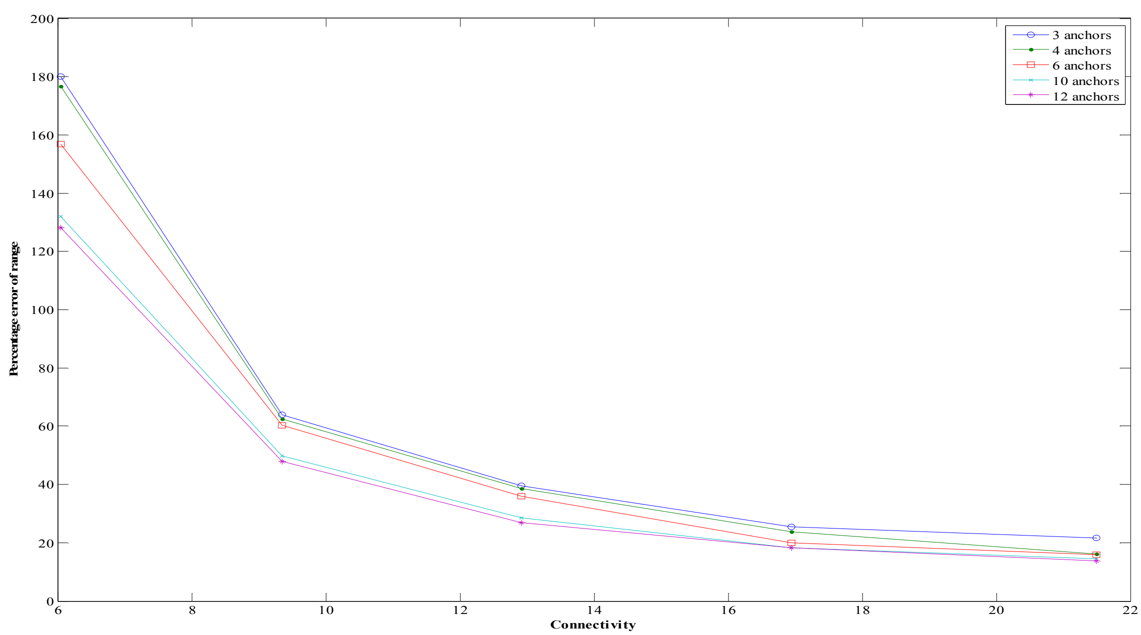


Fig 5.5: Final position estimation of the network using distance measured between connected neighbors (with 5% range error), by translation, reflection, orthogonal rotation, and scaling (MDS-MAP) of the result obtained through MDS using the 4 anchor nodes. The average position estimation error obtained is 0.5934 meters which is 19.7811% of the range.



(a) Using proximity information



(b) Using Distance measured between connected neighbors (5% range error)

Figure 5.6: Average position error of MDS-MAP

5.2.2 Square grid placement

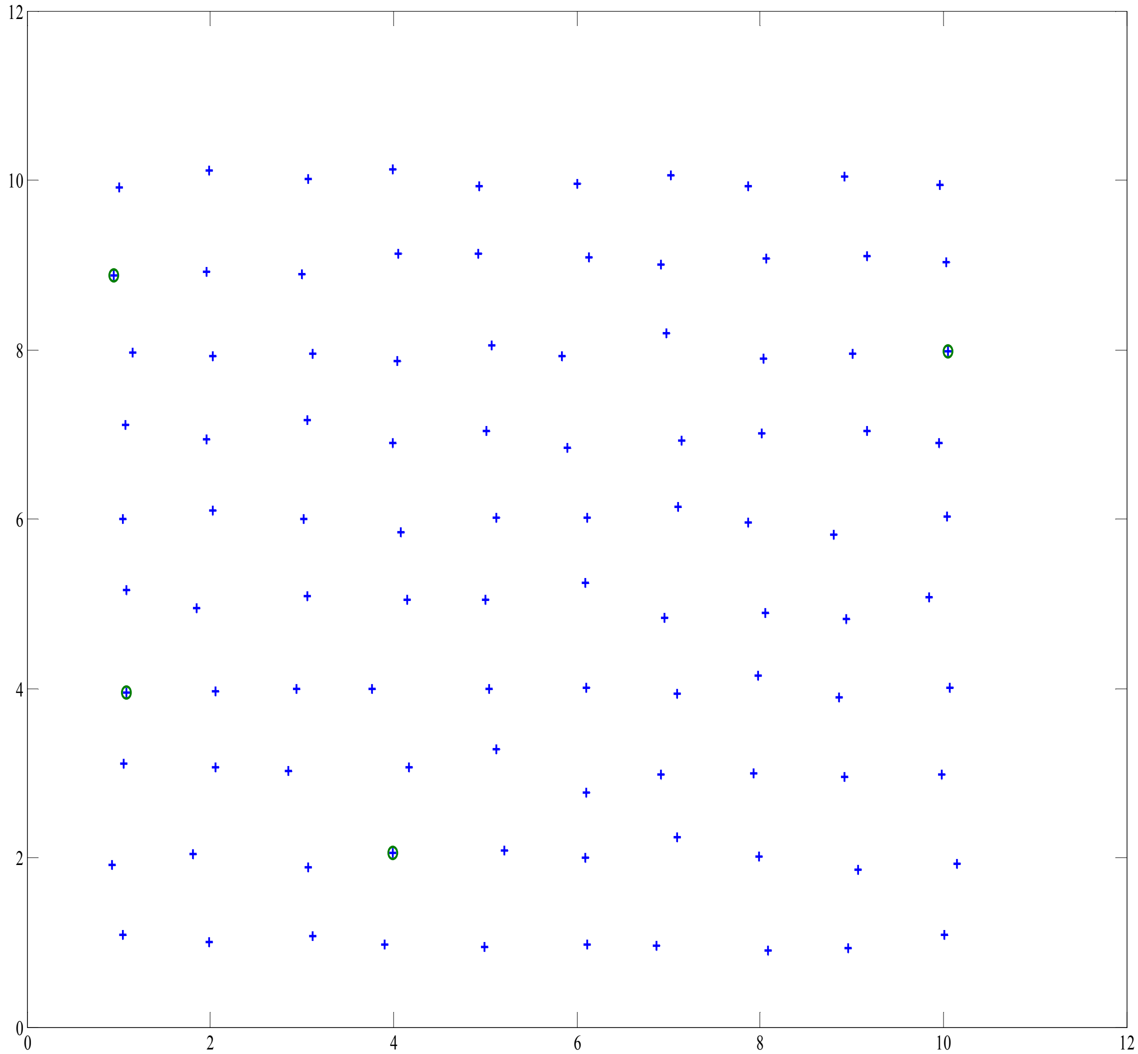


Figure 5.7: 100 nodes placed on 10×10 square grid with 10% placement error. Range of each node is 1.4 meters which resulted in an average connectivity of 5.08. Above graph also shows 4 anchor nodes which are selected randomly.

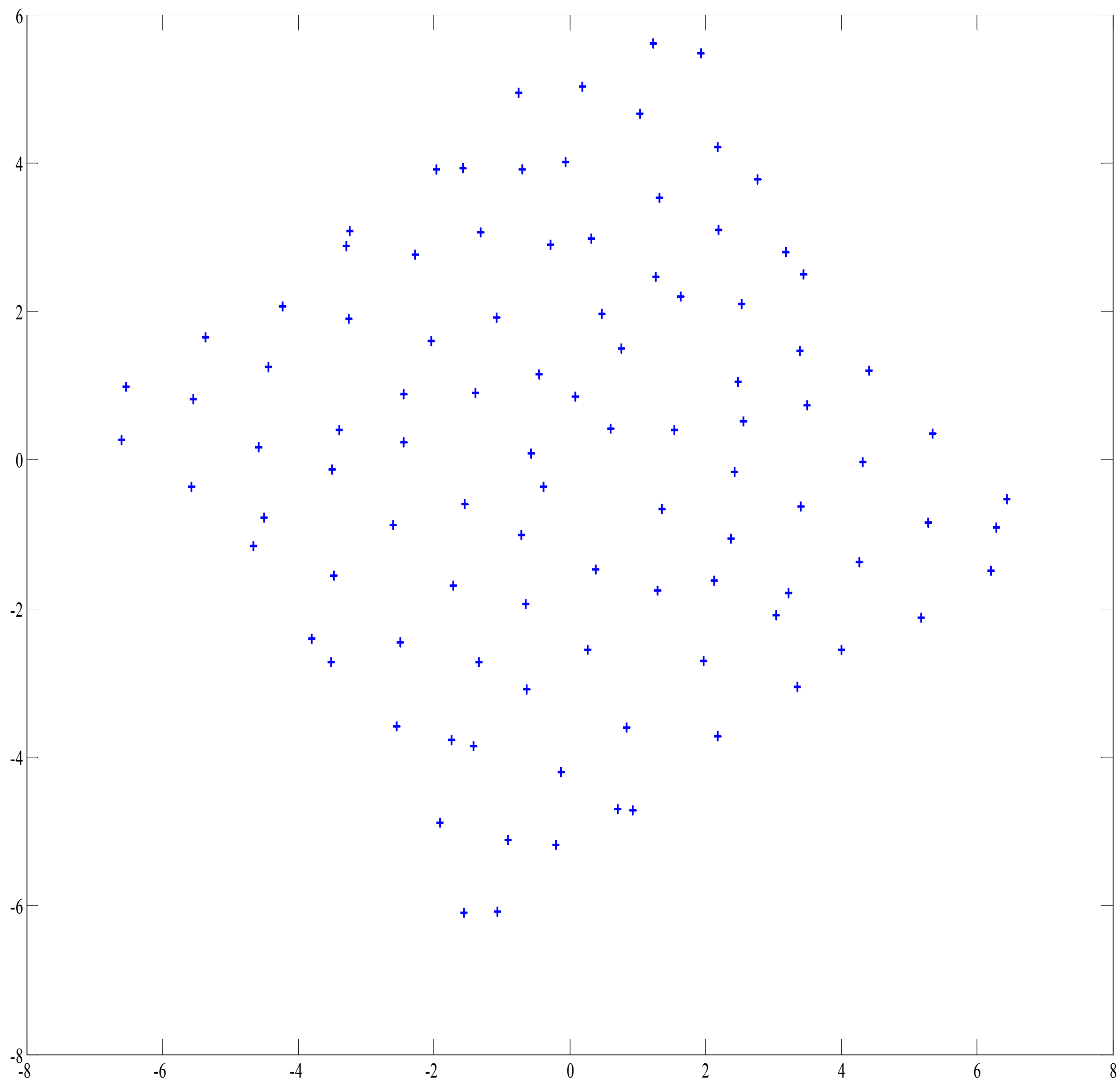


Figure 5.8: Multi-Dimensional Scaling (MDS) of the network using proximity information only i.e., distance vector input for MDS is built by taking 1 if connectivity between any 2 nodes exists or else 0.

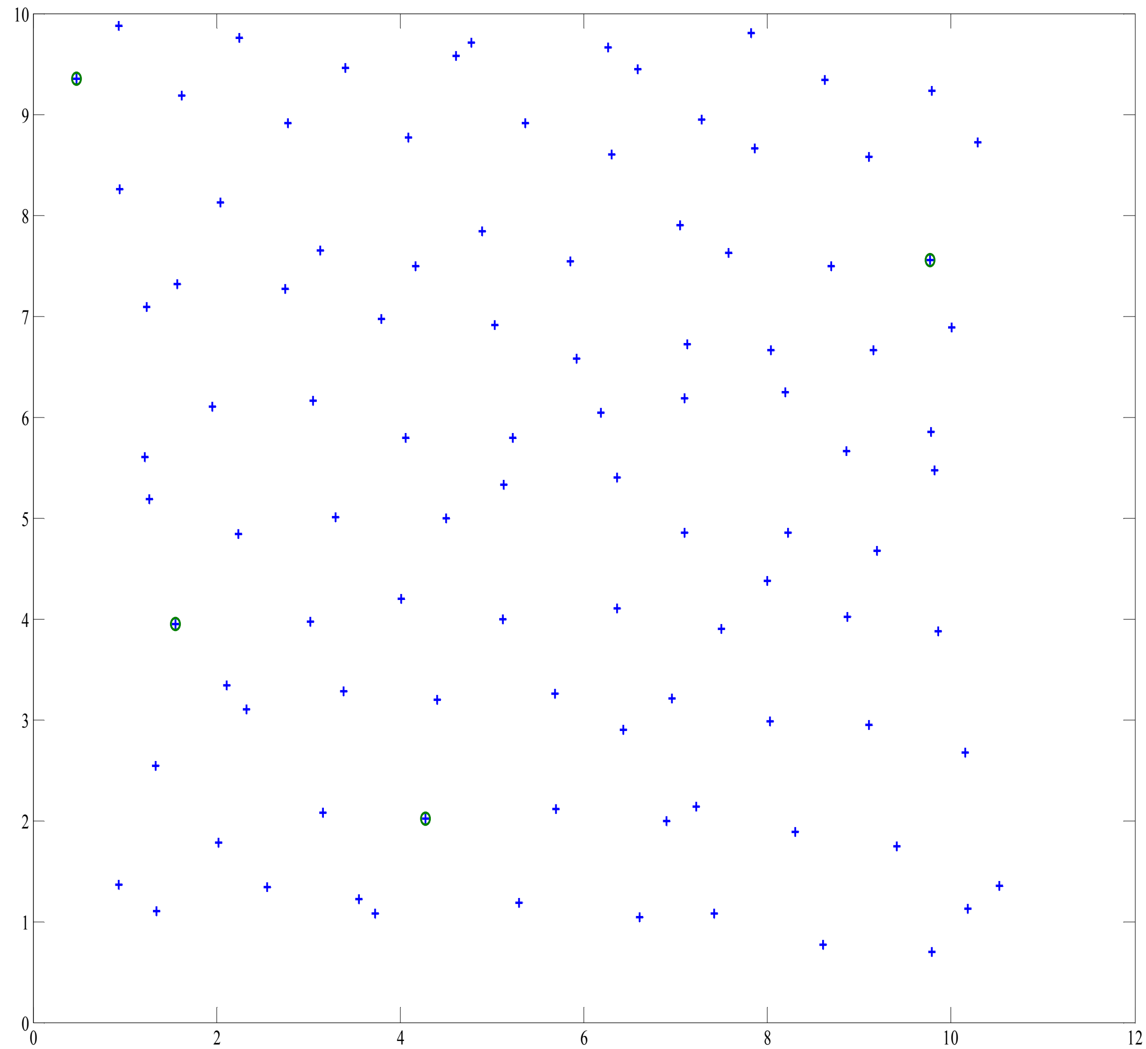


Figure 5.9: Final position estimation of the network using proximity information by translation, reflection, orthogonal rotation, and scaling (MDSMAP) of the result obtained through MDS using the 4 anchor nodes. The average position estimation error obtained is 0.3596 meters which is 25.6847% of range.

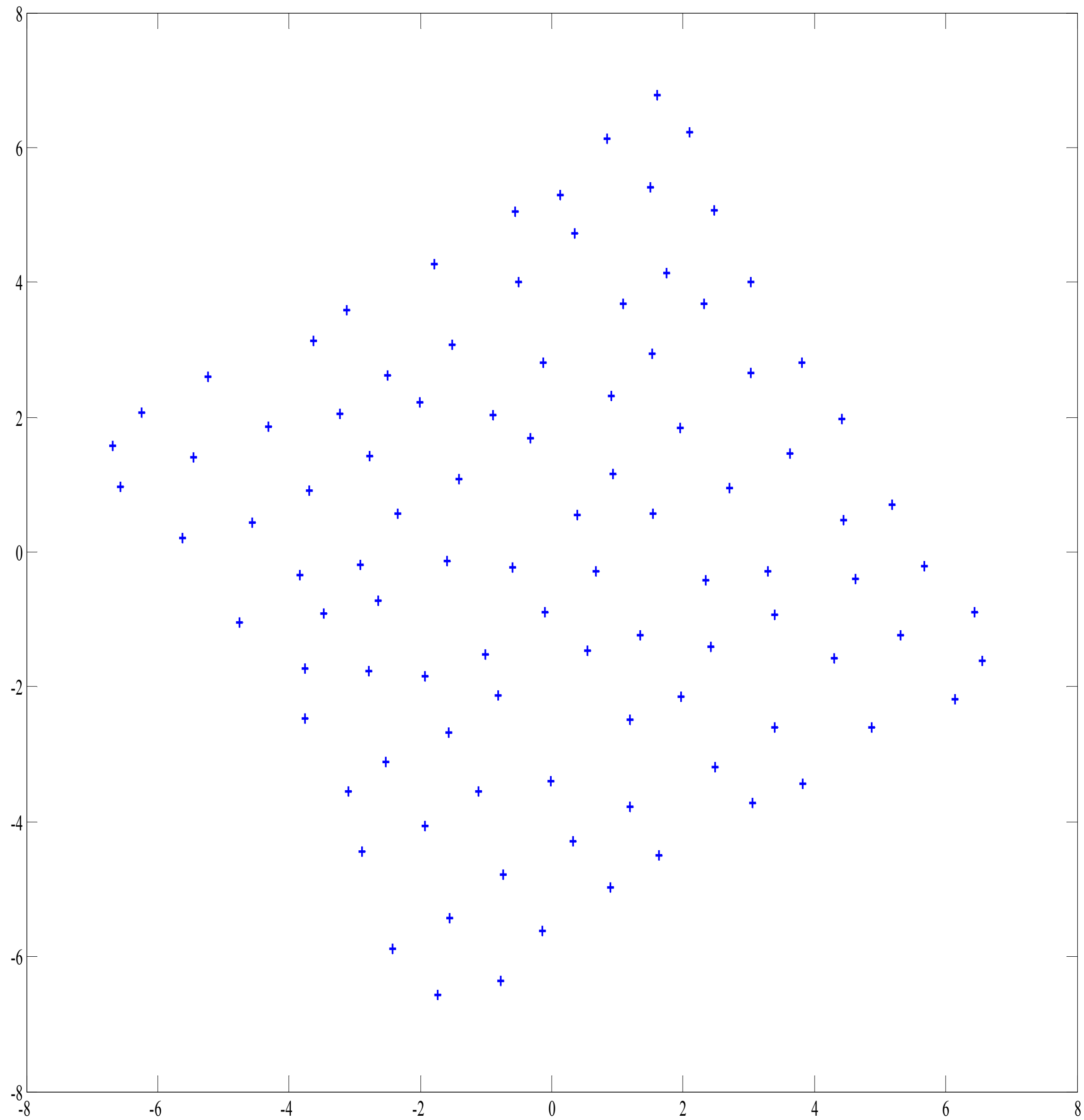


Figure 5.10: Multi-Dimensional Scaling (MDS) of the network using available distance information (distance between nodes with 5% range error) as the distance vector input.

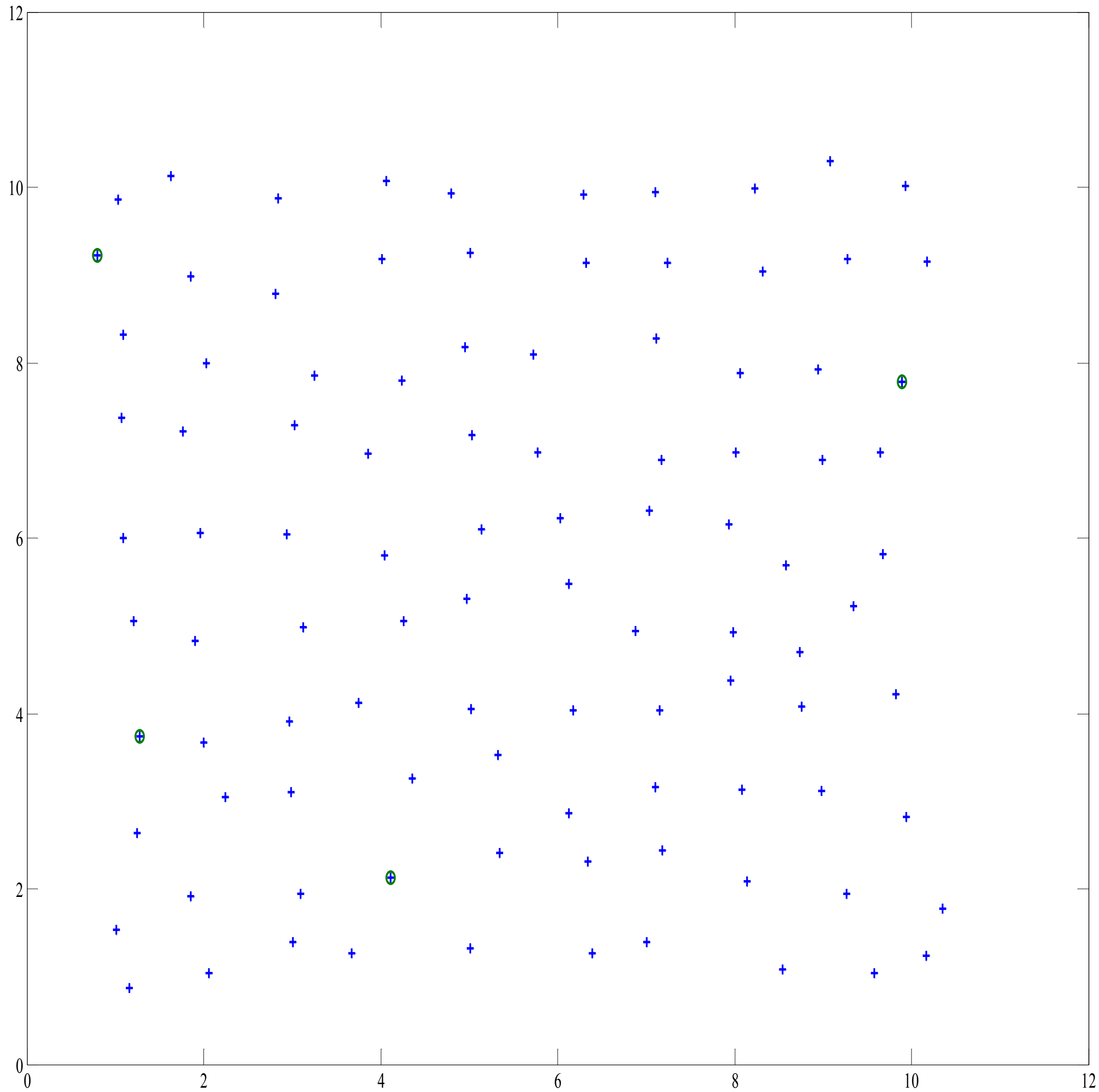
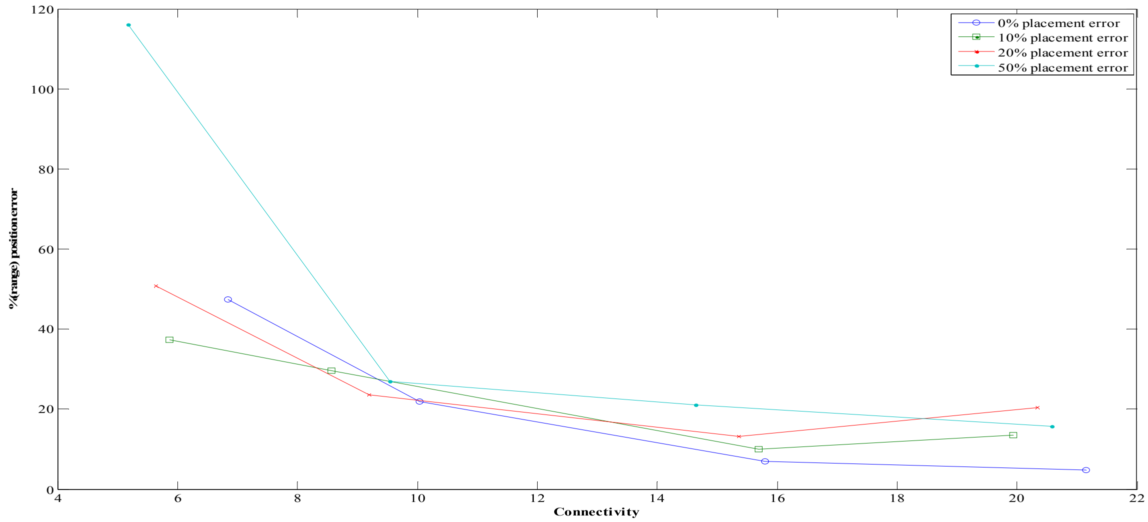
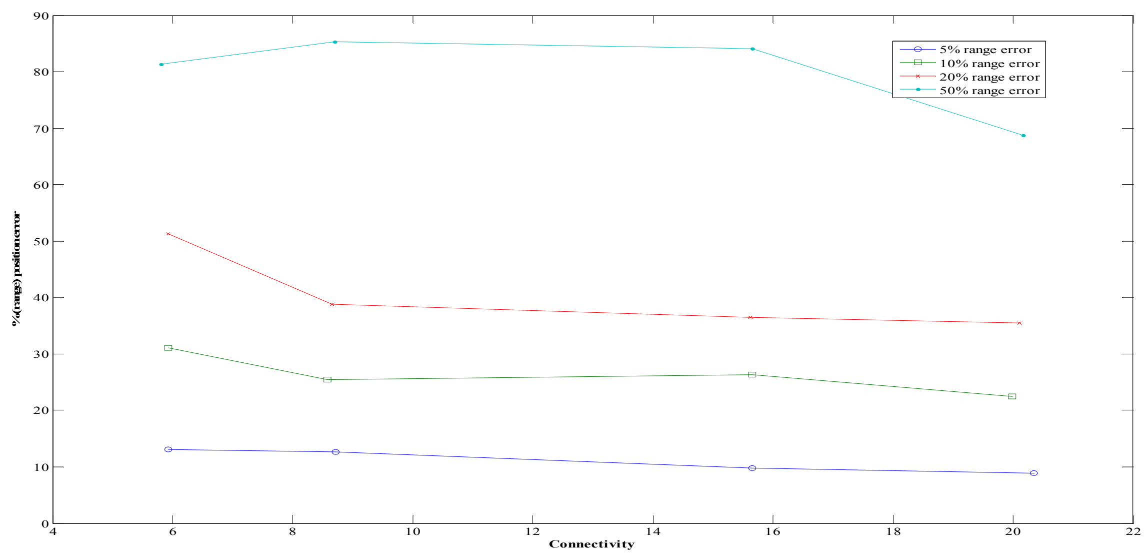


Figure 5.11: Final position estimation of the network using distance measured between connected neighbors (with 5% range error) by translation, reflection, orthogonal rotation, and scaling (MDS-MAP) of the result obtained through MDS using the 4 anchor nodes. The average position estimation error obtained is 0.2125 meters which is 15.1781% of the range.



(a) Using proximity information



(b) Using distance measured between neighbors

Figure 5.12: Average position error of MDSMAP on square grid (with 10% placement error) networks of 100 nodes. 4 different connectivity values correspond to ranges 1.5, 2.0, 2.5, 3.0 meters. Number of anchors chosen = 4.

5.2.3 Hexagonal grid placement

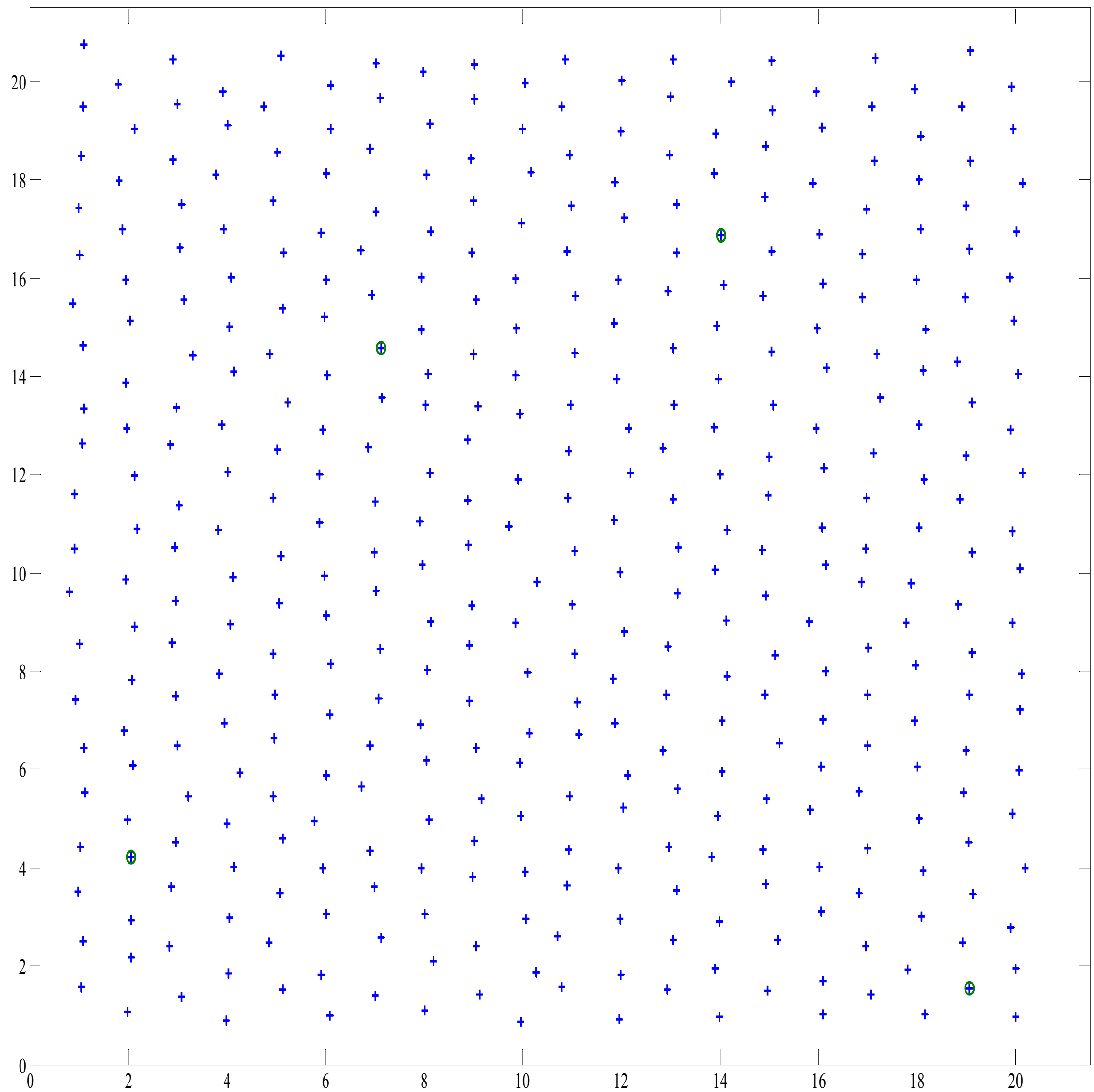


Figure 5.13: 400 nodes placed on 20×20 hexagonal grid with 10% placement error. Range of each node is 1.4 meters which resulted in an average connectivity of 5.4950. Above graph also shows 4 anchor nodes which are selected randomly.

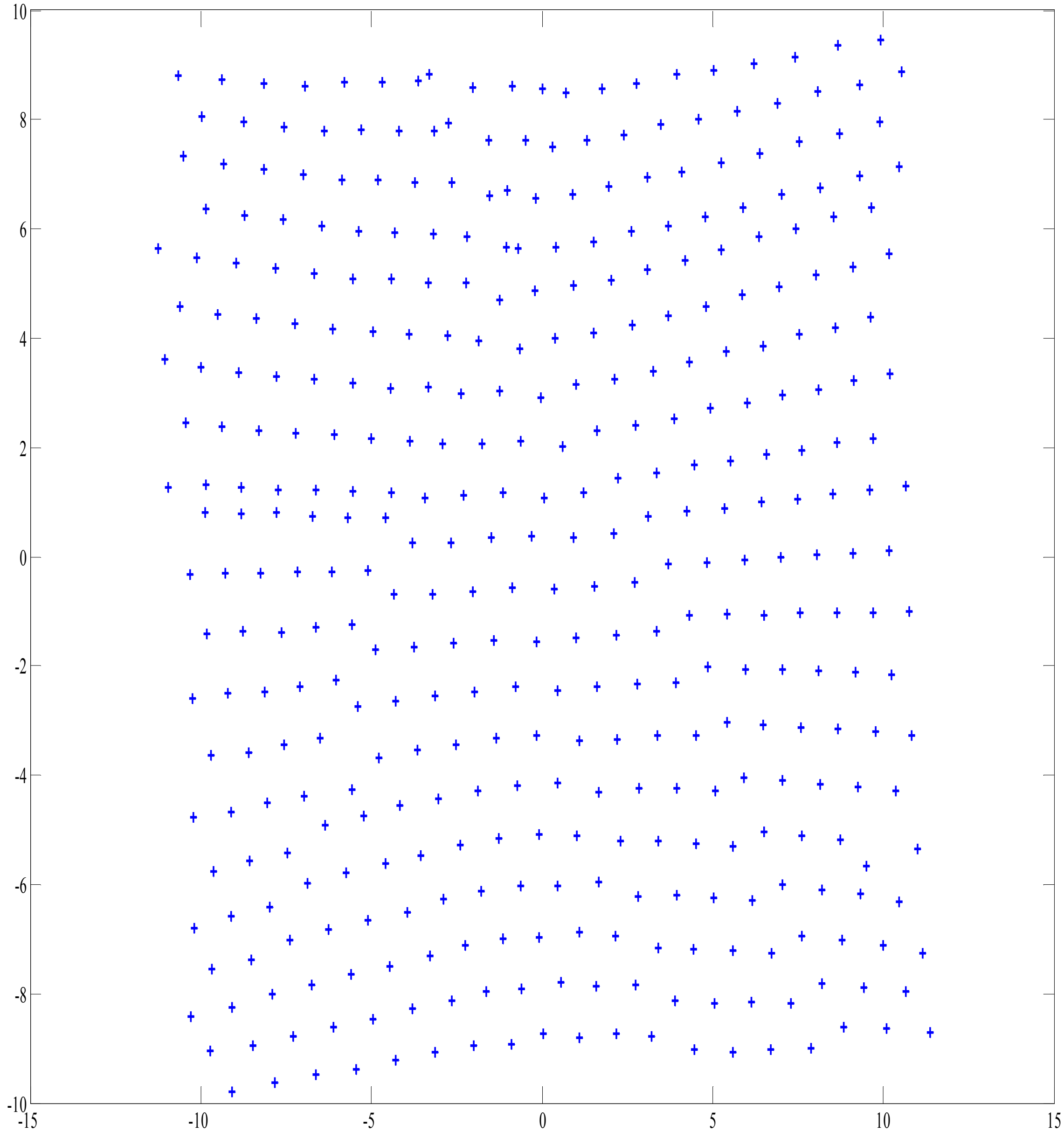


Figure 5.14: Multi-Dimensional Scaling (MDS) of the network using proximity information only i.e., distance vector input for MDS is built by taking 1 if connectivity between any 2 nodes exists or else 0.

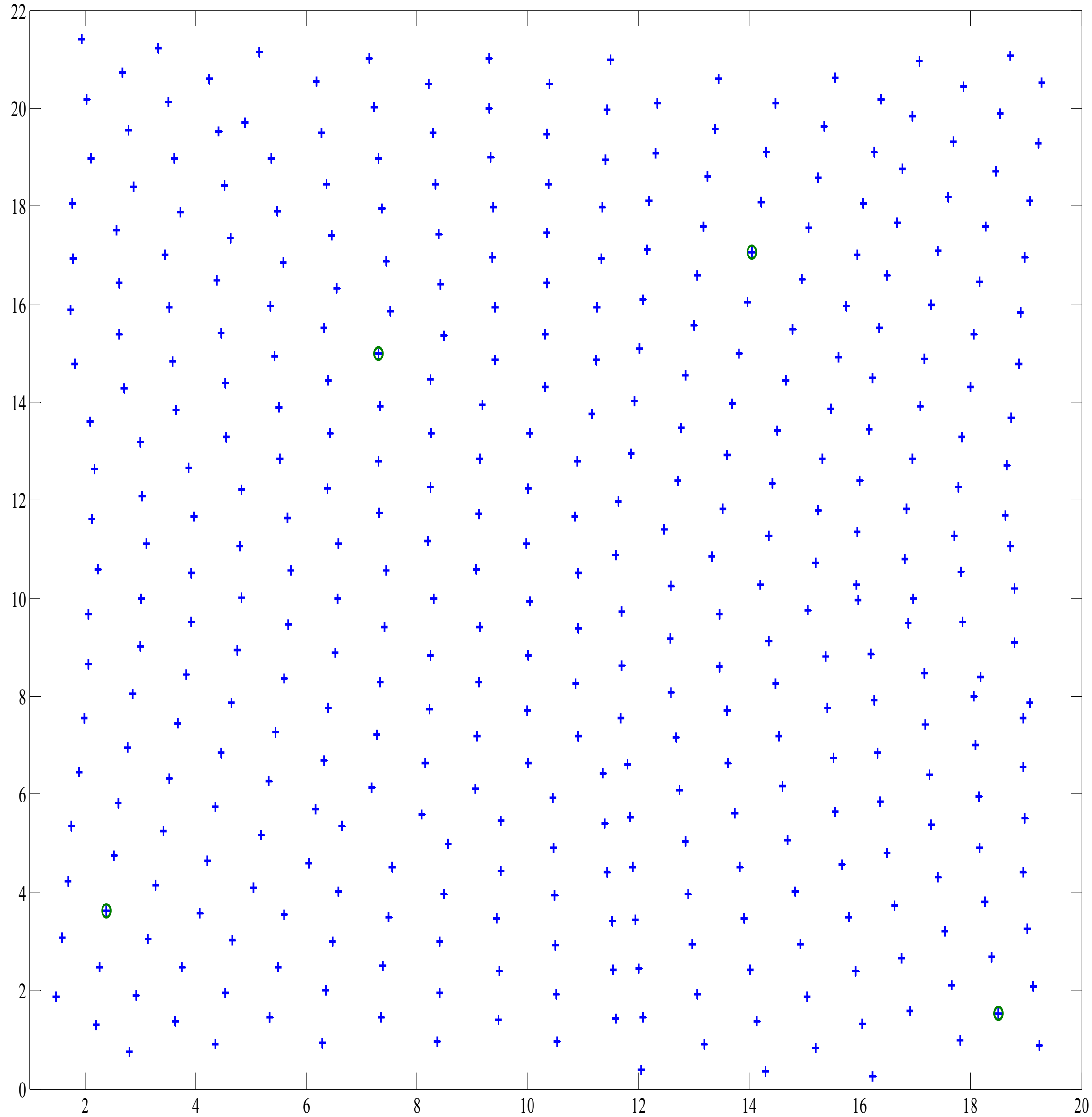


Figure 5.15: Final position estimation of the network using proximity information by translation, reflection, orthogonal rotation, and scaling (MDSMAP) of the result obtained through MDS using the 4 anchor nodes. The average position estimation error obtained is 0.6171 meters which is 44.0808% of range.

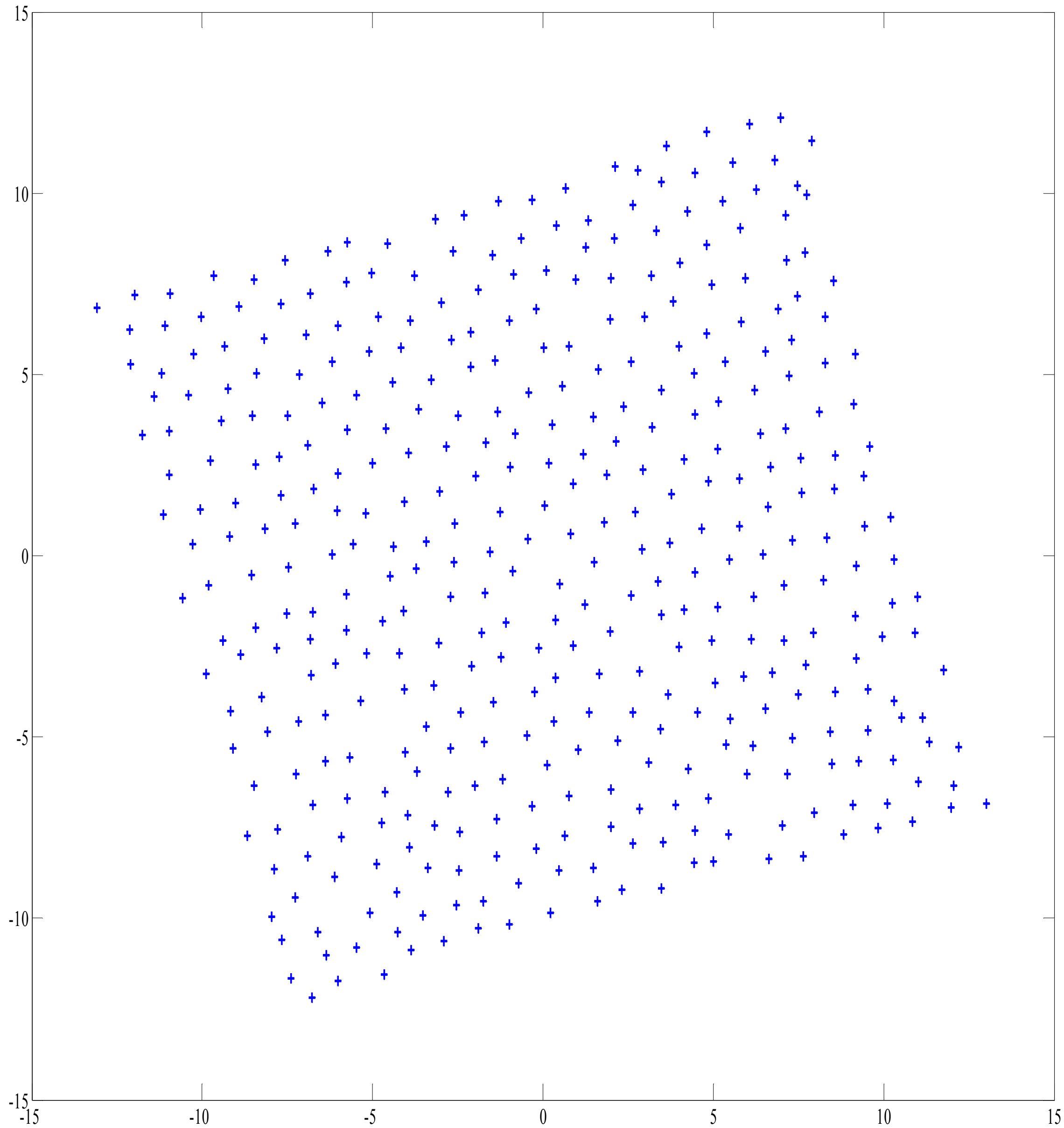


Figure 5.16: Multi-Dimensional Scaling (MDS) of the network using available distance information (distance between nodes with 5% range error) as the distance vector input.

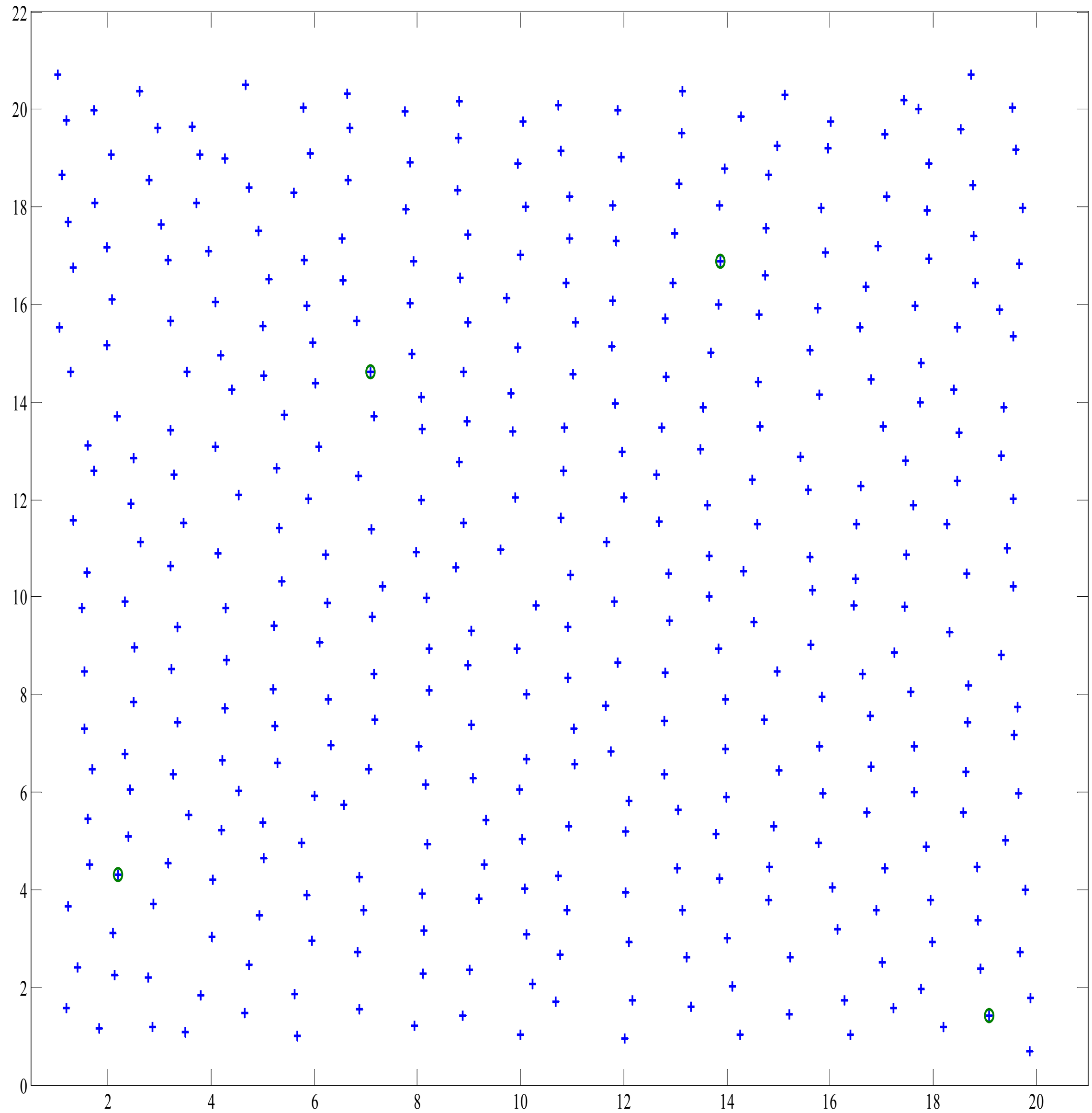


Figure 5.17: Final position estimation of the network using distance measured between connected neighbors (with 5% range error) by translation, reflection, orthogonal rotation, and scaling (MDSMAP) of the result obtained through MDS using the 4 anchor nodes. The average position estimation error obtained is 0.2514 meters which is 17.9582% of the range.

Chapter 6

Conclusions and Future work

In this thesis we presented a simple approach for solving the localization problem in wireless sensor networks. This simple mathematical approach is able to derive the locations of nodes with accuracy equal to 20% of R, where R is the range of each node. All the algorithms in the literature either settle with an absolute map or relative map of the network. But MDS-MAP algorithm is able to derive both relative and absolute map of the network. If there is no anchor available the algorithm settles with relative map and is good enough with just 3 anchor nodes (2-dimensional) to translate the relative map to an absolute map. Also of interest is there are no rules where to place the anchor nodes within the network. This is quiet helpful in applications of sensor networks deployed in harsh environment to position anchor nodes in difficult to reach positions. This algorithm is of great interest in applications like location aided routing where absolute positions are not necessary and also low budget applications that can not afford highly sophisticated devices for anchors.

Whatever the localization algorithm is capable of, wireless sensor networks are often a big failure. This is only due to node failures which in fact causes due to limited power at each node. Node failures result in several secluded regions within the network. Idea to solve this problem is in-network processing, that is, each node play equal role in determining its position. But, further a better approach is needed to solve this problem.

This should be designing a localization algorithm addressing efficient power usage at individual node elements. First the algorithm must assign power values to each node to attain certain connectivity. Secondly it should solve the localization problem. This will at least decrease the number of node failures, thus increasing the overall lifetime of the network. And in mobile sensor networks dynamic assignment of power levels to the nodes should be of great interest.

REFERENCES

[1] L. Doherty, K. S. J. Pister and L. El Ghaoui, “Convex position estimation in wireless sensor networks,” *IEEE INFOCOM*, vol. 3, pp. 1655-1663, April 2001.

[2] D. Niculescu and B. Nath, “Ad-hoc positioning system,” *IEEE GLOBECOM,* vol. 5, pp. 2926-2931, Nov. 2001.

[3] P. Biswas and Yinyu Ye, “Semi-definite programming for ad hoc wireless sensor networks,” *IEEE Information processing in sensor networks,* pp. 46-54, April 2004.

[4] Andreas Savvides, Chih-Chieh Han and Mani B. Srivastava, “Dynamic fine-grained localization in ad-hoc networks of sensors,” in *Proceedings of international conference on mobile computing and networking,* pp. 166-179, 2001.

[5] N. Bulusu, J. Heidemann and D. Estrin, “GPS-less low-cost outdoor localization for very small devices,” *IEEE Personal communications,* vol. 7, pp. 28-34, Oct. 2000.

[6] Tian He, Chengdu Huang, Brian M. Blum, John A. Stankovic and Tarek Abdelzaher, “Range-free localization schemes for large scale sensor networks,” in *Proceedings of international conference on mobile computing and networking,* pp. 81-95, 2003.

[7] Jeffrey Hightower and Gaetano Borriella, “Location Systems for Ubiquitous Computing,” *IEEE computer*, vol. 34, pp. 57-66, Aug 2001.

[8] Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam and Erdal Cayirci, “A survey on sensor networks,” *IEEE Communications magazine*, vol. 40, pp. 102-114, Aug. 2002.

[9] Jeffrey Hightower and Gaetano Borriello, “A survey and taxonomy of location systems for ubiquitous computing,” *Technical report UW-CSE,* Aug. 2001.

[10] C. David , Deborah Estrin and Mani Srivastava, “Overview of sensor networks,” *IEEE Computer,* vol. 37, pp. 41-49, Aug. 2004.

[11] Dragos Niculescu, “Positioning in ad hoc sensor network,” *IEEE Network,* vol. 18, pp. 24-29, Aug. 2004.

[12] Guolin Sun, Jie Chen, Wei Gua, and K.J. Ray Liu, “Signal processing techniques in network aided positioning: a survey of state-of-the-art positioning design,” *IEEE Signal processing magazine,* vol. 22, pp. 12-23, July 2005.

[13] S. Capkun, M. Hamdi, J. Hubaux, “GPS-free positioning in mobile ad hoc networks,” in *Proceedings of Hawaii International conference on System Sciences,* Jan. 2001.

[14] Yi Shang, Wheeler Ruml, Ying Zhang, and Markus P. J. Fromherz, “Localization from mere connectivity,” in *Proceedings of ACM International symposium on mobile ad hoc networking and computing*, pp. 201-212, June 2003.

[15] Christian Bettstetter, “On the minimum node degree and connectivity of a wireless multi-hop network,” in *Proceedings of ACM International symposium on mobile ad hoc networking and computing,* pp. 80-91, 2002.

[16] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi-Min Wang, “Distributed topology control for power efficient operation in multi-hop wireless ad hoc networks,” *IEEE INFOCOM* , vol. 3, pp. 22-26, April 2001.