LOCALIZATION FOR THREE DIMENSIONAL WIRELESS SENSOR NETWORKS

Dissertation submitted to the Jawaharlal Nehru University in partial fulfillment of the requirements for the award of the degree of

Master in Technology (Computer Science)

Submitted By

RAJESH KUMAR

Enrolment No. : 09/10/MT/16

School of Computer & Systems Sciences

Jawaharlal Nehru University

New Delhi - 110067

July' 2011

Dedicated to

TO MY PARENTS

ACKNOWLEDGEMENTS

I would like to express my greatest appreciation towards my dissertation supervisor Dr. R.C. Phoha for his continuous guidance, support and constant faith in me. He is always supportive and remarkably perceptive in advising in the right direction and correcting the vital details. He is a great mentor and I learned so much from him through this academic journey. He is the epitome of professionalism and perfectionism. His excellence and perfection in academics helped me in improving my skills and completing this excellent work. I could not have finished my dissertation in time without his untiring help in making every sentence concise and correct. Sadly, he passed away on 15/07/2011. I would like to thank Mr. Sushil Kumar for inspiring me to work in the exciting field of sensor network area.

I am very fortunate to have Prof. Sonajharia Minz, Dean of our school as my supervisor during MCA and M Tech course work who taught me the decision making lessons and helped me in improving my attitude towards people and life both. She has been excellent advisor cum guardian during my time in SC&SS.

My seminar evaluation committee members, Prof. K.K. Bhardwaj, Prof. C.P. Katti, Dr. Aditi Saran, and Dr. D.P. Vidyarthi provided invaluable feedback and advice on my topic finalization. During presentation they asked many question which later came as the crucial input in my dissertation work.

I am truly blessed to have come to Jawaharlal Nehru University, where I found intellectual professors and learned that the world is open for everyone to do miracles. Jawaharlal Nehru University's invigorating environment enabled my mind to broaden and my research to thrive.

I feel very fortunate for the opportunity to discuss potential applications of our sensor localization algorithms with Dr. D.K. Lobiyal; he provided inspiration and background for our algorithms to be practically useful.

I am indebted to my best friend Diksha Shukla for her encouragement. Without her continuous motivations, I would still be in my Dissertation dreams. I am also thankful to my friends Rakesh Ranjan, Shiv Prakash Tiwari, Sandeep Yadav (I.R.S.) for their unconditional love and academic support.

Finally, I give heartfelt thanks to my dearest parents, my sister and my brothers for their unyielding love and support. They provide a constant source of motivation in my life.

18/7/11

(Rajesh Kumar)

School of Computer & Systems Sciences
जवाहरलाल नेहरू विश्वविद्यालय
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067

DECLARATION

I hereby declare that the dissertation work entitled ***"LOCALIZATION FOR THREE DIMENSIONAL WIRELESS SENSOR NETWORKS"*** is an authentic record of my own work carried out during 2009-11 in partial fulfillment for the requirements for the degree of **Master of Technology (Computer Science & Technology)** and submitted to **School of Computer & Systems Sciences, Jawaharlal Nehru University**, New Delhi-110067, under the supervision of **Late Dr. R.C. Phoha.**

The matter contained in the dissertation has not been submitted for the award of any other degree or diploma anywhere.

18/7/11
Rajesh Kumar
M. Tech (2009-2011)
SC&SS, JNU
New Delhi-110067

Forwarded
12/12/11

प्रोफेसर कमेशु/Professor Karmeshu
डीन / Dean
स्कूल आफ कम्प्यूटर और पद्धति विज्ञान संस्थान
School of Com[illegible] and [illegible]tems Sciences
[illegible] विश्वविद्यालय
[illegible]iversity
[illegible]-110067

ABSTRACT

There are several essential issues (e.g., localization, deployment, and coverage) in wireless sensor networks. **Localization** is one of the most important subjects for wireless sensor networks since many applications such as environment monitoring, vehicle tracking and mapping depend on knowing the locations of sensor nodes. In addition, with location-based routing protocols, both routing and data forwarding are determined based on the geographic location. To solve the localization problem, either placing sensors manually or equipping each sensor with a GPS receiver. However, due to the large scale nature of sensor networks, those two methods become either inefficient or costly.

This dissertation describes many key concepts like characteristics, Applications, challenges, etc. related to the Wireless Sensor Network and specially describes problem of localization in detail. This work also discuses about the classification of localization methods i.e. range free and range based. It also discusses the localization scheme using the static anchor nodes and the mobile anchor nodes schemes.

This dissertation presents an efficient range-free localization mechanism for sensors in a 3-dimensional wireless sensor network based on the use of moving anchors. In the scheme, each anchor is equipped with a GPS receiver and broadcasts its location information as it flies through the sensing space. Each sensor node in the sensing area then estimates its own location by applying basic geometry principles to the location information it receives from the flying anchors.

The scheme eliminates the requirement for specific positioning hardware, and is independent of network densities and topologies. The performance of the proposed localization scheme is evaluated in a series of simulations performed using SINALGO software which support 3-D simulation and is compared to a reduced localization time, and a lower beacon overhead (Only 3 beacons are being used and sufficient in proposed scheme).

Finally, this paper is also comparing the proposed schemes with the earlier proposed schemes. The simulation results validate the gain in localization time, its accuracy, and the resulted overhead.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Introduction to a wireless sensor network

A WSN (Wireless Sensor Network) consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants[17][18].

Recent advances in wireless communications and electronics have enabled the development of low-cost, low-power and multi-functional sensors that are small in size and communicate in short distances. Cheap, smart sensors, networked through wireless links and deployed in large numbers, provide unprecedented opportunities for monitoring and controlling homes, cities, and the environment. In addition, networked sensors have a broad spectrum of applications in the defense area, generating new capabilities for reconnaissance and surveillance as well as other tactical applications.

There are several essential issues (e.g., localization, deployment, and coverage) in wireless sensor networks. *Localization* is one of the most important subjects for wireless sensor networks since many applications such as environment monitoring, vehicle tracking and mapping depend on knowing the locations of sensor nodes. In addition, with location-based routing protocols, both routing and data forwarding are determined based on the geographic location. To solve the localization problem, either placing sensors manually or equipping each sensor with a GPS receiver. However, due to the large scale nature of sensor networks, those two methods become either inefficient or costly, so researchers propose to use a variety of localization approaches for sensor network localization.

1.2 Architecture of a sensor in WSN [16]: A sensor node in a wireless sensor network usually consists of sub-systems illustrated in Fig1:

A computing subsystem : It consists of a microprocessor(microcontroller unit, MCU) which is responsible for the control of the sensors and execution of communication protocols. MCU's usually operate under various operating modes for power management purposes. But shuttling between these operating modes involves consumption of power, so the energy consumption levels of the various modes should be considered while looking at the battery lifetime of each node.

A communication subsystem: It consists of a short range radio which is used to communicate with neighboring nodes and the outside world. Radios can operate under the Transmit, Receive, Idle and Sleep modes. It is important to completely shut down the radio rather than put it in the idle mode when it is not transmitting or receiving because of the high power consumed in this mode.

A sensing subsystem: It consists of a group of sensors and actuators and links the node to the outside world. Energy consumption can be reduced by using low power components and saving power at the cost of performance which is not required.

A power supply subsystem: It consists of a battery which supplies power to the node. It should be seen that the amount of power drawn from a battery is checked because if high current is drawn from a battery for a long time, the battery will die even though it could have gone on for a longer time. Usually the rated current capacity of a battery being used for a sensor node is lesser than the minimum energy consumption required leading to the lower battery lifetimes. The lifetime of a battery can be increased by reducing the current drastically or even turning it off often.

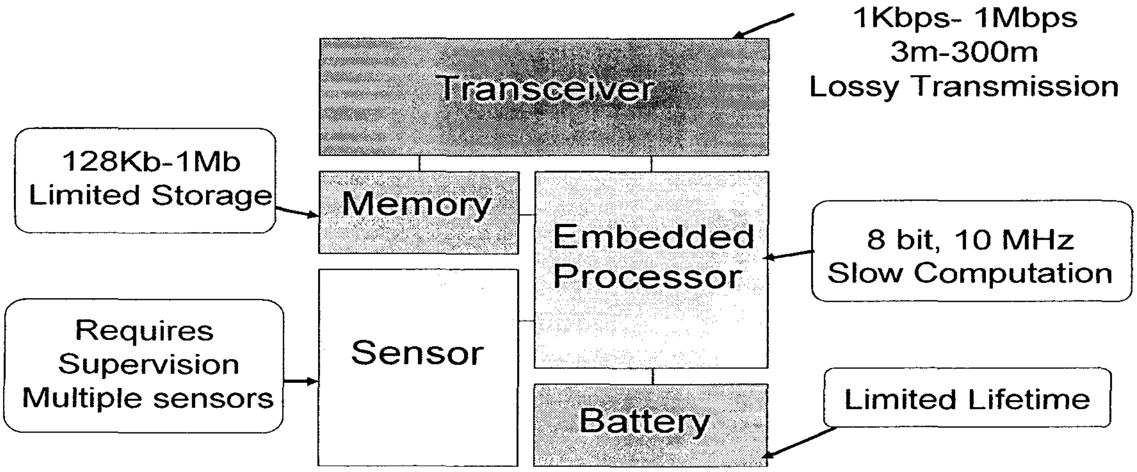


Fig 1: System architecture of a typical wireless sensor node [16]

[Sensor + Actuator + Microprocessor + Powering Unit + Communication Unit (RF Transceiver) + GPS (Optional)]

1.3 Characteristics of a Wireless Sensor Networks

Wireless Sensor Networks have some unique characteristics. Some of them are listed below **[21]:**

- Limited power they can harvest or store
- Ability to withstand harsh environmental conditions
- Ability to cope with node failures
- Mobility of nodes
- Dynamic network topology
- Communication failures
- Heterogeneity of nodes
- Large scale of deployment
- Unattended operation

1.4 Applications of sensor networks [22]:

There are various applications of WSN Specific applications include habitat monitoring, object tracking, fire detection, land slide detection and traffic monitoring. A WSN is deployed in a region where it is meant to collect data through its sensor nodes. Some of the important applications are described below in brief.

Environmental monitoring:

A number of WSNs have been deployed for environmental monitoring. Examples are Glacier, Earthquakes, Tornadoes, Tsunamis, Hurricanes monitoring etc.

-Greenhouse Monitoring:

WSN are also used to control the temperature and humidity levels inside commercial greenhouses. When the temperature and humidity drops below specific levels, the greenhouse manager must be notified via e-mail or cell phone text message, or host systems can trigger misting systems, open vents, turn on fans, or control a wide variety of system responses. Because some WSN are easy to install, they are also easy to move as the needs of the application change.

-Landslide Detection:

A landslide detection system makes use of a wireless sensor network to detect the slight movements of soil and changes in various parameters that may occur before or during a landslide. And through the data gathered it may be possible to know the occurrence of landslides long before it actually happens.

Industrial monitoring:

Numerous applications are in industrial monitoring exists for example pollution monitoring, physical conditions of mechanical parts, etc.

- Machine health monitoring:

WSN have been developed for machinery condition-based maintenance (CBM) as they offer significant cost savings and enable new functionalities. In wired systems, the installation of enough sensors is often limited by the cost of wiring, which is very costly like 20000-50000 Rs. per foot. Previously inaccessible locations, rotating machinery, hazardous or restricted areas, and mobile assets can now be reached with wireless sensors. Often, companies use manual techniques to calibrate, measure, and maintain equipment. This labor-intensive method not only increases the cost of maintenance but also makes the system prone to human errors. Especially in production industries like SAIL, NTPC, ONGC, BHEL etc, reduced manning levels make it imperative to install automated maintenance monitoring systems. WSN play an important role in providing this capability.

Water/Wastewater monitoring:

There are many opportunities for using WSN within the water/wastewater industries. Facilities not wired for power or data transmission can be monitored using industrial wireless I/O devices and sensors powered using solar panels or battery packs.

-Landfill ground well level monitoring and pump counter

Wireless sensor networks can be used to measure and monitor the water levels within all ground wells in the landfill site and monitor leach ate accumulation and removal. A wireless device and submersible pressure transmitter monitors the leach ate level. The sensor information is wirelessly transmitted to a central data logging system to store the level data, perform calculations, or notify personnel when a service vehicle is needed at a specific well.

-Water tower level monitoring

Maintaining the water levels in water towers (used to add water and create water pressure to small communities or neighborhoods during peak use times to ensure water pressure is available to all users) is important and requires constant monitoring and control. A wireless sensor network that includes submersible pressure sensors and float switches monitors the water levels in the tower and wirelessly transmits this data back to a control location. When tower water levels fall, pumps to move more water from the reservoir to the tower are turned on.

- Agriculture

Using wireless sensor networks within the agricultural industry is increasingly common. Gravity fed water systems can be monitored using pressure transmitters to monitor water tank levels, pumps can be controlled using wireless I/O devices, and water use can be measured and wirelessly transmitted back to a central control center for billing. Irrigation automation enables more efficient water use and reduces waste.

In fact, due to the pervasive nature of micro-sensors, sensor networks have the potential to revolutionize the very way we understand and construct complex physical system.

1.5 Challenges in wireless sensor networks

In spite of the diverse applications, sensor networks pose a number of unique technical challenges due to the following factors **[13][20]:**

Ad hoc deployment: Most sensor nodes are deployed in regions which have no infrastructure at all. A typical way of deployment in a forest would be tossing the sensor nodes from an aero plane. In such a situation, it is up to the nodes to identify its connectivity and distribution.

Unattended operation: In most cases, once deployed, sensor networks have no human intervention. Hence the nodes themselves are responsible for reconfiguration in case of any changes.

Energy: The sensor nodes are not connected to any energy source. There is only a finite source of energy, which must be optimally used for processing and communication. An interesting fact is that communication dominates processing in energy consumption. Thus, in order to make optimal use of energy, communication should be minimized as much as possible.

Dynamic changes: It is required that a sensor network system be adaptable to changing connectivity (for e.g., due to addition of more nodes, failure of nodes etc.) as well as changing environmental stimuli.

Thus, unlike traditional networks, where the focus is on maximizing channel throughput or minimizing node deployment, the major consideration in a sensor network is to extend the system lifetime as well as the system robustness.

1.6 Problem of Localization in wireless sensor networks : In sensor networks, nodes are deployed into an unplanned infrastructure i.e. ad-hoc manner where there is no a priori knowledge of location. ***The problem of localization is estimating the position or spatial coordinates of wireless sensor nodes.*** An immediate solution which comes to mind is GPS or the Global Positioning System.

However, there are some strong factors against the usage of GPS. For one, GPS can work only outdoors. Secondly, GPS receivers are expensive and not suitable in the construction of small cheap sensor nodes. A third factor is that it cannot work in the presence of any obstruction like dense foliage etc.

Thus, sensor nodes would need to have other means of establishing their positions and organizing themselves into a coordinate system without relying on an existing infrastructure.

1.7 Need for Localization

Self-localization capability is a highly desirable characteristic of wireless sensor networks. In environmental monitoring applications such as bush fire surveillance, water quality monitoring and precision agriculture, the measurement data are meaningless without knowing the location from where the data are obtained. Moreover, location estimation may enable countless of applications such as inventory management, intrusion detection, road traffic monitoring, health monitoring, reconnaissance and surveillance.

Example: – Raw data is of limited use, raw data combined with spatial information - very useful. In real-time monitoring of fire, Temperature combined with location is useful to find the source of the fire.

1.8 Classification of Localization methods [11]:

All algorithms can be classified in either range based or range-free methods.

1.8.1 Range-based localization method: The range based methods being the first one to appear. Their principle in localization is mainly depending on distance or angle between nodes to obtain unknown node's location. The first step is distance estimates and angle estimates. A number of approaches, such as time of arrival, time difference of arrival, angle of arrival, received signal strength, have been presented. The second step is location calculations. Trilateration, triangulation and maximum likelihood estimation are typical methods.

Before giving the details of different range methods, we will discuss how one can estimate inter-node distances.

Received Signal Strength Indicator (RSSI): The energy of the radio signal, viewed as an electromagnetic wave, decreases as it propagates in space. By knowing the original emitted power and comparing it to the received signal power, one can estimate the attenuation 'g' and deduce the distance via, for example, a free space path-loss model:

$$g = d^{-\alpha}$$

In this scheme the exponent α is around 2 in an open-space environment, but its value increases if the environment is more complex (walls, etc.) or less suitable for radio waves (metallic devices. Another issue is that there is no unique path from the transmitter to the receiver. Any reverberations of the signal will influence the received strength, so it has to be measured at the appropriate moment. Some consider the first peak, whereas others prefer an average of the first periods.

Time based methods (ToA, TDoA): These methods record the time-of-arrival (ToA) or time-difference-of-arrival (TDoA). The propagation time can be directly translated into distance, based on the known signal propagation speed. These methods can be applied to many different signals, such as RF, acoustic, infrared and ultrasound. TDoA methods are impressively accurate under line-of-sight conditions. But this line-of-sight condition is difficult to meet in some environments. Furthermore, the speed of sound in air varies with air temperature and humidity, which introduce inaccuracy into distance estimation. Acoustic signals also show multi-path propagation effects that may impact the accuracy of signal detection.

Angle-of-Arrival (AoA): AoA estimates the angle at which signals are received n use simple geometric relationships to calculate node positions. Generally, AoA techniques provide more accurate localization result than RSSI based techniques but the cost of hardware of very high in AoA.

Note: Range-based localization methods have the advantage of fine resolution. However, extra hardware and additional energy consumption restricted the application of range-based methods.

1.8.2 Range-free localization method: Range-free localization methods use the information of topology and connectivity for location estimation. In range free techniques, the position of sensor node is identified on the basis information transmitted by *nearby anchor nodes or neighboring nodes,* based on hop or on triangulation basis. The various range free techniques are APIT, DV-Hop localization etc.

APIT: The APIT idea is to divide the environment into triangles, given by beaconing nodes. An individual node's presence or absence in each of those triangles will allow reducing the possible location area. This goes until all the possible sets are exhausted, or the desired accuracy reached.

The APIT algorithm is then ran at every node:

- Receive locations from n anchors.
- For each possible triangle, test if inside or not.
- If yes, add it to the InsideSet.
- Break if accuracy reached.
- Estimate position as CenterOfGravity (∩Ti belongs to InsideSet).

For testing if the node is inside or not a triangle according to the Point-in-Triangle (PIT) test, it needs to move. To cope with situations where nodes are static and unable to move, an Approximate PIT test is defined according to: If no neighbor of M is further from/closer to all three anchors A, B and C simultaneously, M assumes that it is inside triangle ABC. Otherwise, it is outside.

This is of course subject to errors, especially if the node is close to one of the network's edges, or if the neighbors have an irregular placement. The error has never exceeded 15 % (on their particular scenarios).

An important aspect of this solution is that APIT uses indeed signal strength, but not as an approximate for a distance. It just assumes that signal strength decreases monotonically with the distance (usually valid). Thus it is used to compare distance, and APIT is still a range-free algorithm.

Multi-hop: Multi-hop methods are mainly range-free, but can also use estimation of the distances. Their purpose is to compute a connectivity graph, and then trying to make it fit the known positions as good as possible.

Multi Dimensional Scaling: In a large sensor network, Multi Dimensional Scaling (MDS) only uses connectivity information, i.e. which nodes are within communication range of which others. The process has three steps:

- Rough estimation of the distance between each possible pair of nodes.
- MDS to derive locations fitting the estimated distances.
- Optimization by taking the known positions into account.

The system is modeled by a connectivity graph, the edges having the value 1 (if the distances are known, the values are used instead). This gives a symmetric matrix, which is run in a classical all-pairs shortest-path algorithm.

The resulting distance matrix is used in classical MDS, and gives a relative map locating each node.

Linear transformations (scaling, rotations, reflections, translations) are used to fit the anchors' estimated positions to the correct ones, and perhaps all other known positions, if any. There are many types of MDS techniques: metric/non-metric, classical/replicated, weighted, and deterministic/probabilistic. Classical MDS, where the proximities are used as being distances, seems to be the best choice in this issue .The Euclidian distance has then to be as close as possible to the proximities (least squares).

N-Hop Multilateration Multi-hop multilateration technique is aiming to give to give nodes that are several hops away from beacons the possibility to collaborate in finding better position estimates. By allowing this type of collaboration, the ratio of beacons to nodes can be decreased.

Range-free methods have some advanced characteristics, such as low cost, small communication traffic, no extra hardware and flexible localization precision. Because of these special characteristics of range-free methods, they were been regard as a promising solution for the localization problem in WSN.

1.9 Localization with Static and Mobile Sensor Nodes [8]:

There are mainly two strategies which followed by most of the researchers as give below:
Enable a few static sensor nodes in the network with GPS equipped devices. These nodes will help in locating their neighbors depending on the placement strategy. The rest of sensor nodes will collectively get localized with the help of their respective neighboring nodes. *Drawback of this scheme is if there is an error while computing the nodes location, this error gets rippled in computations related to next tiers of neighbors and so on.*
Take a few GPS enabled mobile sensor nodes to move within the network and help in locating the other sensor nodes. *This scheme was proved to be better than any existing range free localization scheme for three dimensional wireless sensor networks.*

1.10 Criteria for good localization algorithms:

Self-organized: Because of the lack of localization infrastructure in wireless sensor network, the self-organization of localization algorithm is necessary.
Robustness: Localization algorithm is immune to node failure and distance estimation error.
Efficient energy: Message exchange is an indicator of energy consumption. How much communication overhead is required?
Distributed calculation: Does localization algorithm only uses localized information?
Scalability: Is a localization algorithm scalable to the number of nodes in network?

1.11 Organization of dissertation: The dissertation is organized in chapters which contain introduction, related work, proposed work, simulation results, conclusion and future work.

CHAPTER 2: RELATED WORK

Localization in three dimensional wireless sensor networks is still a very good area of research work. Only a few solutions are proposed to address the problem using both range based and range free techniques. I will discuss some of them based on the *range free* technique.

- Novel Centroid Localization Algorithm for Three-Dimensional Wireless Sensor Networks [7].

- Sensor Position Determination with Flying Anchors in Three-Dimensional Wireless Sensor Networks which works on the principle "A perpendicular line passing through the center of a sphere's circular cross section also passes through the center of that sphere"[1].

- Localization scheme for 3-Dimensional wireless sensor networks using GPS enabled mobile sensor nodes which work on the principle "If any point is at the surface of sphere then it will satisfy the sphere equation"[8].

2.1 Novel Centroid Localization Algorithm for Three-Dimensional Wireless Sensor Networks:

This proposed localization algorithm will not use 2D centroid theorem, but present the centroid theorem of coordinate-tetrahedron in the volume-coordinate system, which acts as a key component of our estimation approach. Using centroid theorem of coordinate-tetrahedron, the proposed positioning algorithm can be used in 3D WSNs and also can improve location accuracy than the conventional centroid localization algorithm. This work will be focusing on the second and third of the listed above [7].

2.2 Sensor Position Determination with Flying Anchors in Three-Dimensional Wireless Sensor Networks: This paper presents a range-free position determination (localization) mechanism for sensors in a three-dimensional wireless sensor network based on the use of flying anchors.

This proposes a solution for localization which works on basic principle that, *"a perpendicular line passing through the center of a sphere's circular cross section also passes through the center of that sphere".*

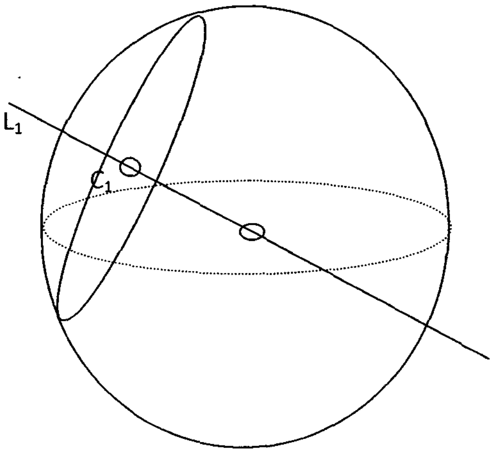


Fig. 2: A perpendicular line passing through center of the cross section of sphere also passes through the center of the sphere

The algorithm for the above proposed solution can be give as below:

1. *Select 4 beacon points using the flying anchor concept.*
2. *Assume 2 circular cross sections C_1, C_2 as shown in the Fig 4.*
3. *Find the centre of the circle C_1, and C_2.*
4. *Now find the lines L_1 and L_2 Passing through the centre of the circle C_1, and C_2 respectively and perpendicular to the circular cross sections too as shown in the Fig 5.*
5. *Find the intersection point of the L_1 and L_2 which will be the centre of the sphere and the desired location of the sensor node.*

1. ***How to select the 4 beacon points using flying anchors ?***

To establish two circular cross sections which is required as per stated above in geometric corollary the localization mechanism must identify a minimum of four endpoints on the sphere, i.e. four locations at which the flying anchors pass through the surface of the imaginary sphere. As it travels through the sensing space, *each flying anchor periodically broadcasts beacon messages detailing its ID, position, and timestamp. Meanwhile, each sensor node maintains a set of beacon points and a Visitor List.* The beacon point is regarded as an endpoint on the communication sphere of the sensor node, and the Visitor List maintains the IDs and respective lifetimes of the flying anchors currently passing through this sphere. When a sensor node receives a beacon message from a flying anchor, it first checks whether the flying anchor is already recorded in its Visitor List.

If it is missing, a beacon point, i.e., the current position of the flying anchor, is logged, and the ID and corresponding lifetime of the flying anchor are added to the Visitor List. If the flying anchor is already present in the Visitor List, the beacon message is ignored and the lifetime of the anchor is simply extended. When the lifetime of the flying anchor expires, its final beacon message is recorded as a beacon point and the corresponding entry for the anchor is deleted from the Visitor List [1].

Fig. 3 illustrates an example of the beacon point selection procedure. In this example, a flying anchor A moves from (x, y, z) to (x', y', z') broadcasting beacon messages at an interval of t, where $t=T_{i+1}$ *-* $T_{i,}$ *and i=0, 1, 2, 3.... 7. The beacon messages at T2 is considered as a beacon point* $(A, (x_2, y_2, z_2))$ *by sensor node S so Node S add an entry* $(A, (x_2, y_2, z_2), T2 + L)$ *to its visitor list, where 'L' is the predefined lifetime of the flying anchor and has a value larger than the beacon interval 't' (L=at, a >1). The lifetime of Node A is increased by L when it arrives at T3, T4, T5, and T6, respectively. Once node A moves out of the communication sphere of Node S and its lifetime expire, its beacon message at T6 is logged as a beacon point and Node S deletes the entry for Node A from its Visitor List.*

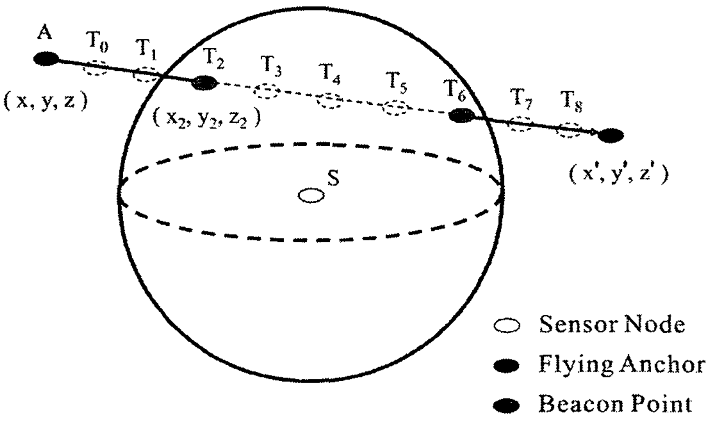


***Fig 3: Beacon Point Selection on the surface of sphere* [1]**

2. ***How to select 2 circles of out of 4 beacon points?*** The Fig 4 illustrates the selection of 2 circles using the 3 beacon points. B$_1$, B$_2$, B$_4$ representing the circular cross section having the centre C$_1$ and B$_1$, B$_3$, B$_4$ representing the second circular cross section having centre C$_2$. The lines passing through the centre C$_1$ and C$_2$ will cross at S which is the desired point i.e. the location of the Sensor Node.

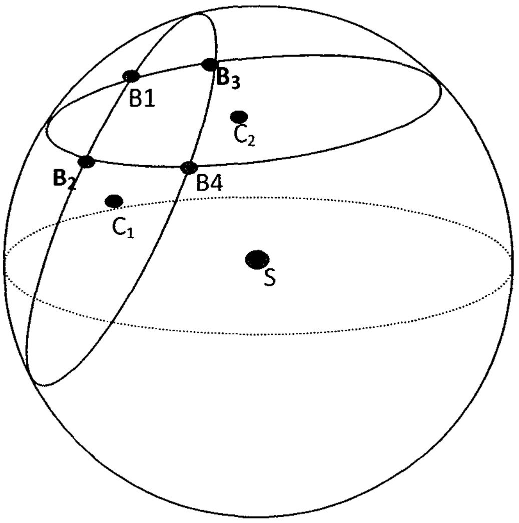


Fig 4: Selection of circular cross section

3. ***How to find the centre of the circles i.e. C_1, and C_2 ?***

Fig5 illustrates the calculation of the centre of the sphere, Based on the geometric corollary that the perpendicular bisector of any chord of a circle passes through the center of that circle, the intersection point of the perpendicular bisectors of two chords of the same circle is located at the center of that circle.

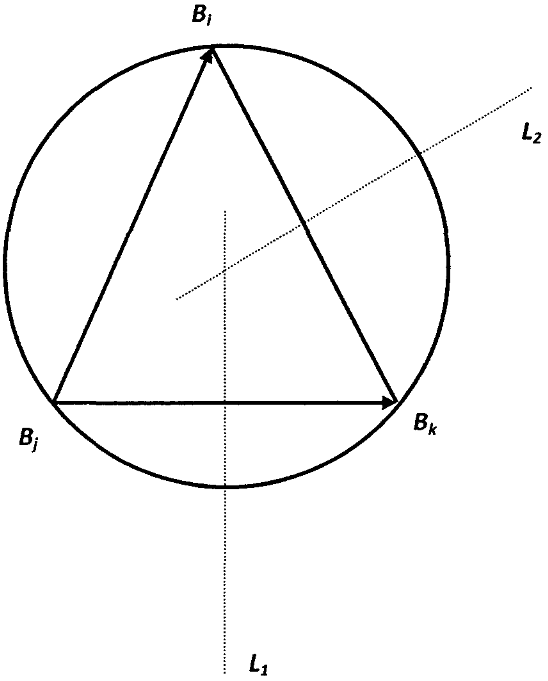


Fig 5: Centre of the Circular cross section

4. ***Calculation of centre C_1, C_2, L_1, L_2 and the final location of the sensor node i.e. the centre of the sphere:***

As shown in Fig. 6, the perpendicular line L_1 passes through both the center of the circular cross-section C_1 and the center of the sphere S. Similarly, L_2 passes through both C_2 and S. The intersection point of L_1 and L_2 is, therefore, S. In the localization mechanism, the center of the sphere is taken to indicate the *sensor node's position*, and the radius of the sphere corresponds to the maximum range over which the sensor node can detect the location information broadcast by the flying anchor.

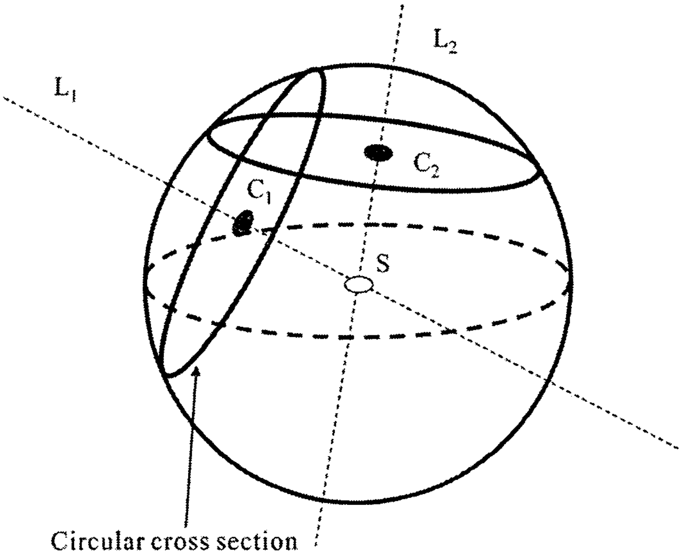


Fig 6: intersection of L1 and L2 shows the centre of the sphere

Once four beacon points have been identified, two circular cross sections are constructed and the perpendicular lines passing through the centers of the circular cross sections determined. In Fig. 4, it is assumed that a set of beacon points (B_i, B_j, B_k, B_l) has been selected and there corresponding locations are (x_i,y_i,z_i), (x_j,y_j,z_j), (x_k,y_k,z_k), (x_l,y_l,z_l). Arbitrarily B_i,B_j,B_k and B_j, B_k, B_l are chosen to make 2 circular cross sections S and S'respectively. As shown, the two circular cross sections intersect at B_j and B_k. Determining the perpendicular lines passing through centers of the cross section requires the normal vectors and centers of the two circular cross sections to be known. The normal vector of S(N) is obtained by the cross prodcut of vector B_iB_j and,B_jB_k similarly the normal vector of S'(N') is given by the cross product of the vectors B_jB_k and B_kB_l.

$$\bar{N} = vector\ BiBj * vector\ BjBk\ .\ (N_x,N_y,N_z) \qquad (1)$$

$$\bar{N}\ '= vector\ BjBk * vector\ BkBl =(N'_x,N'_y,N'_z) \qquad (2)$$

Based on the geometric corollary that the perpendicular bisector of any chord of a circle passes through the center of that circle, the intersection point of the perpendicular bisectors of two chords of the same circle is located at the center of that circle. Consider the circular cross-section S, for example. The normal vector of the perpendicular bisector of the chord vector B_iB_j i.e. (L_{ij}) can be generated by the cross product of $\vec{N}$ and vector B_jB_k gives the normal vector of the perpendicular bisector L_{jk}, i.e.

$$vector\ L_{ij} = \vec{N} * vector\ BiBj\ = (l,m,n) \qquad (3)$$

$$vector\ L_{jk} = \vec{N} * vector\ BjBk = (p,q,r) \qquad (4)$$

Similarly for the cross section S', vector vector L'_{ij} and vector L'_{jk} are given by

$$vector\ L'_{ij} = \vec{N} * vector\ BjBk\ = (l,m,n) \qquad (5)$$

$$vector\ L'_{jk} = \vec{N} * vector\ BkBl = (p,q,r) \qquad (6)$$

Suppose that the equation of the straight lines L_{ij} , $L_{jk,}$ $L'_{ij,}$ and L'_{jk} can be expressed by the following:

$$L_{ij}: \frac{x_{ij} - a_{ij}}{l} = \frac{y_{ij} - b_{ij}}{m} = \frac{z_{ij} - c_{ij}}{n} = t_{ij} \qquad (7)$$

$$L_{jk}: \frac{x_{jk} - a_{jk}}{p} = \frac{x_{jk} - b_{jk}}{q} = \frac{x_{jk} - c_{jk}}{r} = t_{jk} \qquad (8)$$

$$L'_{ij} = \frac{x_{jk} - a_{jk}}{l'} = \frac{x_{jk} - b_{jk}}{m'} = \frac{x_{jk} - c_{jk}}{n'} = t'_{ij} \qquad (9)$$

$$L'_{jk} = \frac{x_{kl} - a_{kl}}{p'} = \frac{x_{kl} - b_{kl}}{q'} = \frac{x_{kl} - c_{kl}}{r'} = t'_{jk} \qquad (10)$$

Based on the corollary that states that the perpendicular bisector of a chord contains the center of the circle, the intersection point of L_{jk} and L'_{ij} (L'_{ij} and L'_{jk}) is located at the center of the circle. Therefore the centre of the circle (C) will be

$$X_c = l * t_{ij} + \left(\frac{x_i + x_j}{2}\right),$$

$$Y_c = m * t_{ij} + \left(\frac{y_i + y_j}{2}\right),$$

$$Z_c = n * t_{ij} + \left(\frac{y_i + y_j}{2}\right)$$

$$\textit{Where } t_{ij} = \frac{p*\left(\frac{y_i+y_j}{2}-\frac{y_j+y_k}{2}\right)-q*\left(\frac{x_i+x_j}{2}-\frac{x_j+x_k}{2}\right)}{q*l-p*m} \qquad (11)$$

Similarly the center of C' will be given as

$$\mathrm{X_c}' = l'*t'_{kl} + \left(\frac{x_j+x_k}{2}\right),$$

$$\mathrm{Y_c}' = m'*t'_{kl} + \left(\frac{y_j+y_k}{2}\right),$$

$$\mathrm{Z_c}' = n'*t'_{kl} + \left(\frac{y_j+y_k}{2}\right)$$

$$t'_{ij} = \frac{p'*\left(\frac{y_j+y_k}{2}-\frac{y_k+y_l}{2}\right)-q'*\left(\frac{x_j+x_k}{2}-\frac{x_k+x_l}{2}\right)}{q'*l'-p'*m'} \qquad (12)$$

The equation of the perpendicular line L and L' are given by

$$\mathrm{L} = \frac{x-x_c}{N_x} = \frac{y-y_c}{N_y} = \frac{z-z_c}{N_z} = t \qquad (13)$$

$$\mathrm{L}' = \frac{x-x'_c}{N'_x} = \frac{y-y'_c}{N'_y} = \frac{z-z'_c}{N'_z} = t' \qquad (14)$$

The center of the sphere can then be calculated by solving the intersection point of L and L'. *The estimated location of the sensor node is, therefore, shown to be*

$$X_s = N_x*T+X_c,$$

$$Y_s = N_y*T+Y_c,$$

$$Z_s = N_z*T+Z_c$$

$$\text{Where T} = \frac{\left(N'_x*(Y_c-Y_c')-N'y*(X_c-X_c')\right)}{N'_y*N_x-N'_x*N_y} \qquad (15)$$

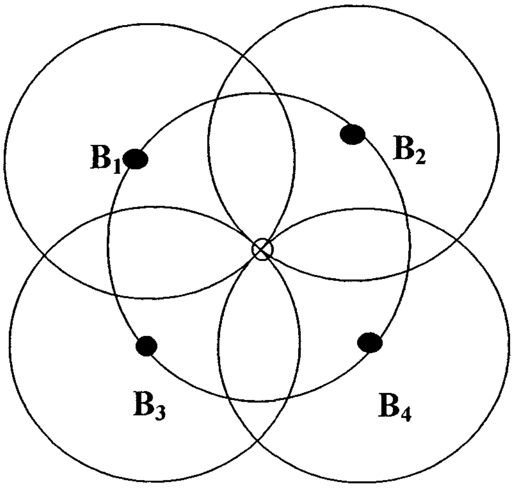
Another paper proposes the different way of doing the same using the other basic principle of geometry i.e. *"If any point is at the surface of sphere then it will satisfy the sphere equation"* described below in detail.

2.3 Localization scheme for 3-Dimensional wireless sensor networks using GPS enabled mobile sensor nodes which works on the principle *"If any point is at the surface of sphere then it will satisfy the sphere equation" and one important assumption is also there that is "all the mobile sensor nodes deployed in a field have same the radio communication range"* [8].

The algorithm for the above proposed solution can be give as below:

1. *Select 4 beacon points using the flying anchor concept.*
2. *Substitue the 4 beacon points in equation of sphere to get three different equations.*
3. *Solve those equations to get the desired centre of sphere i.e. required location of the sensor node.*

Calculation of the location of the node i.e. the centre of the sphere according to the above algorithm: Let the first beacon message received by static sensor node is broadcasted by the mobile anchor nodes from position $B_1(x_1,y_1,z_1)$ and that of second, third and fourth are from B_2 (x_2,y_2,z_2), $B_3(x_3,y_3,z_3)$, B_4 (x_4,y_4,z_4).



● Anchor Node (Beacon) ○ Node to be localized

Fig 7: Four beacon points on the surface of the spherical range of the static sensor

As we can see that all the beacons are lying on the surface of sphere so using the geometric corollary *"If any point is at the surface of sphere then it will satisfy the sphere equation"* we can write as follows:

$$(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2 = r^2 \qquad (1)$$

$$(x-x_2)^2 + (y-y_2)^2 + (z-z_2)^2 = r^2 \qquad (2)$$

$$(x-x_3)^2 + (y-y_3)^2 + (z-z_3)^2 = r^2 \qquad (3)$$

$$(x-x_4)^2 + (y-y_4)^2 + (z-z_4)^2 = r^2 \qquad (4)$$

As we assumed that communication range of all the mobile sensors are equal. Substituting the values of 'r' from equation (2) to equation (1) and equation (3), and from equation (3) to equation (4), we get

$$(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2 = (x-x_2)^2 + (y-y_2)^2 + (z-z_2)^2 \qquad (5)$$

$$(x-x_2)^2 + (y-y_2)^2 + (z-z_2)^2 = (x-x_3)^2 + (y-y_3)^2 + (z-z_3)^2 \qquad (6)$$

$$(x-x_3)^2 + (y-y_3)^2 + (z-z_3)^2 = (x-x_4)^2 + (y-y_4)^2 + (z-z_4)^2 \qquad (7)$$

On solving (5), (6), (7) we get

$$X = \begin{bmatrix} \left(\left(x_4^2 - x_3^2\right) + \left(y_4^2 - y_3^2\right) + \left(z_4^2 - z_3^2\right)\right)/2 & y_1 - y_2 & z_1 - z_2 \\ \left(\left(x_4^2 - x_1^2\right) + \left(y_4^2 - y_1^2\right) + \left(z_4^2 - z_1^2\right)\right)/2 & y_2 - y_3 & z_2 - z_3 \\ \left(\left(x_2^2 - x_1^2\right) + \left(y_2^2 - y_1^2\right) + \left(z_2^2 - z_1^2\right)\right)/2 & y_3 - y_4 & z_3 - z_4 \end{bmatrix} / \Delta \qquad (8)$$

$$Y = \begin{bmatrix} \left(\left(x_4^2 - x_3^2\right) + \left(y_4^2 - y_3^2\right) + \left(z_4^2 - z_3^2\right)\right)/2 & y_1 - y_2 & z_1 - z_2 \\ \left(\left(x_4^2 - x_1^2\right) + \left(y_4^2 - y_1^2\right) + \left(z_4^2 - z_1^2\right)\right)/2 & y_2 - y_3 & z_2 - z_3 \\ \left(\left(x_2^2 - x_1^2\right) + \left(y_2^2 - y_1^2\right) + \left(z_2^2 - z_1^2\right)\right)/2 & y_3 - y_4 & z_3 - z_4 \end{bmatrix} / \Delta \qquad (9)$$

$$Z = \begin{bmatrix} \left(\left(x_4^2 - x_3^2\right) + \left(y_4^2 - y_3^2\right) + \left(z_4^2 - z_3^2\right)\right)/2 & y_1 - y_2 & z_1 - z_2 \\ \left(\left(x_4^2 - x_1^2\right) + \left(y_4^2 - y_1^2\right) + \left(z_4^2 - z_1^2\right)\right)/2 & y_2 - y_3 & z_2 - z_3 \\ \left(\left(x_2^2 - x_1^2\right) + \left(y_2^2 - y_1^2\right) + \left(z_2^2 - z_1^2\right)\right)/2 & y_3 - y_4 & z_3 - z_4 \end{bmatrix} / \Delta \qquad (10)$$

$$\Delta = \begin{bmatrix} x_1 - x_2 & y_1 - y_2 & z_1 - z_2 \\ x_2 - x_3 & y_2 - y_3 & z_2 - z_3 \\ x_3 - x_4 & y_3 - y_4 & z_3 - z_4 \end{bmatrix} \qquad (11)$$

If the mobile beacons received are in one plane that is parallel to the basic axes x, y, and z, then the computed value of $\Delta = 0$. Thus, we need to remove the last entry of mobile sensor node's location and look for another value. In other cases the algorithm will execute as normal and no special consideration is needed. The obtained values of x, y, and z specifies the location of static sensor node L. Also, once a node has calculated its location, the beacon messages will be automatically removed as their lifetime expires.

2.4 Analysis of the related works:

There are only few papers addressing the range free localization in three dimensional WSNs. And as the approach presented in [1] has been proved by the authors to be the best scheme, we analyze and compare our approach with their approach to check the better of the two.

Number of Computations

The numbers of computations required in previously proposed approaches were quiet high. In [1] number of computation required for computing a sensor's location is 153 (i.e. 88 multiplications and 65 additions). However, in our proposed approach the number of computations is 98 (i.e. 54 for multiplying and 44 addition). Thus, we have reduced the number of computation required by 33% which is an additive gain in case of sensor networks.

Space Complexity While comparing our approach with the chord selection approach [1], it is found that our approach takes lesser space than chord selection approach. As in [1] it takes 34 variables to compute the location of a sensor node. However, in our approach it takes 28 variables. Thus, it also saves memory resource to a little extent.

Chord Selection Criteria vs. Points in a plane In [1] Ou and Ssu have considered the chord selection criteria to avoid beacons. This criterion selects those chords that are built with these beacons having angle greater than 10 degrees between them. Otherwise, it will lead to the location of the center of the sphere above or below the actual center. This problem will also be there for larger angle between them. But, in our approach occurrences of all coplanar beacons only lead to non determination of the center. Only one non-coplanar beacon will be sufficient to determine the center effectively. Hence, there is a low probability of discarding any position information obtained through beacon messages which results in saving of time, computation and communication overheads.

As we can see that [8] has been proved to be the best out of three best scheme I have mentioned here. *In this dissertation I have tried to develop a new scheme which has been proved to be better than the best of all these.* In the next chapter I will be discussing the problem definition, environment, assumptions, and proposed work.

CHAPTER 3: PROPOSED WORK

3.1 Problem Definition:

Problem of Localization in 2D: Consider the case when we have deployed a sensor network consist of N sensors at locations S = {S_1, S_2... S_N}. Let Si_x refer to the x-coordinate of the location of sensor 'i' and let Si_y refer to the y-coordinate. Out of these entire sensor nodes are aware of their own positions; these nodes are known as *anchors or beacons*. All the other nodes localize themselves with the help of location references received from the anchors.

So, mathematically the localization problem can be formulated as follows: *Given a multi-hop network, represented by a graph G = (V, E), and a set of beacon nodes B, their positions {X_b, Y_b} for all b ε B, we want to find the position {X_u, Y_u} for all unknown nodes u ε U.* **[2]**

Problem of Localization in 3D:

WSNs are physically impossible to be deployed into the area of absolute plane in the context of real-world applications. For applications like Deployment of WSNs for *surveillance of terrains, study of underwater ecosystem, space monitoring & exploration etc;* accurate location information in 3D-Wireless Sensor Networks is crucial. So, a good localization schemes for accurate localization of sensors in three dimensional spaces is a crucial area of research. The above well defined 2D problem can be extended to 3D as follows: *Mathematically it will be extended as Given a multi hop network, represented by a graph G = (V, E), and a set of beacon nodes B, their positions {X_b, Y_b, Z_b} for all b ε B and their communication range is 'r', we want to find the position {X_u, Y_u, Z_u} for all unknown nodes u ε U.*

3.2 System Environment and Assumptions [1]: System environment looks like shown in the Fig 8. Environment consists of a large number of sensor nodes and a smaller number of flying anchors (moving anchors). The sensor nodes are randomly deployed in the sensing space and are designed to execute a specific monitoring task. The anchors fly through the sensing space transmitting their current locations such that each sensor node can then estimate its position.

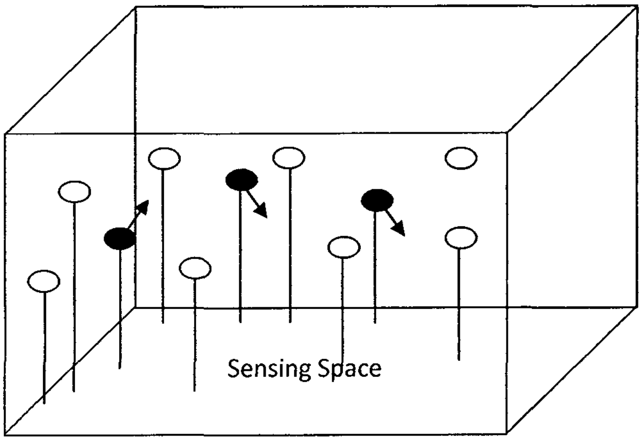


Fig 8: Example of System Environment

● *Moving Anchors*

○ *Sensor Nodes*

3.3 Assumptions [8]:

Static Nodes: Static nodes are the wireless sensor nodes which are deployed in ad-hoc manner and have to be localized. All static sensor nodes have identical sensing ability, computational ability, and the ability to communicate. We also assume that, the initial battery powers of the nodes are identical at deployment.

Anchor Nodes: Anchor nodes are the wireless sensor nodes which is GPS enabled and know its location. These nodes are very few in number and are homogeneous in nature. But, anchors have sufficient energy to both fly and broadcast during the localization process. The communication range of mobile sensor nodes are assumed not to change drastically during the entire localization algorithm runtime and also not to change significantly within the reception of four beacon messages by a particular static node.

Communication Range: All nodes have identical communication range '***r***'. Communication is Omni-directional and is broadcast in nature. The connectivity region of each node can be represented by a sphere of radius '***r***', having the sensor node at its center.

Hearing: Mobile sensor nodes will continuously broadcast their location. As soon as any static sensor node comes within its communication range, it will receive the broadcast message.

Deployment: All the sensor nodes are deployed randomly over the surveillance volume.

3.4 Objective:

Objective of this work is to improve existing localization scheme that localizes the randomly deployed sensor nodes in three dimensional spaces with low computation and communication overhead using range free technique based on the moving anchor concept.

3.5 Proposed Localization Scheme: The proposed scheme in this dissertation follows the same concept of moving anchors (flying anchors) in which few GPS enabled mobile sensor nodes to move within the network and help in locating the other sensor nodes. This scheme was proved to be better than any existing **range free** localization scheme for three dimensional wireless sensor networks.

The proposed method of calculating the sensor node position in this dissertation uses following geometric principles.

1. A perpendicular line passing through the center of a sphere's circular cross section also passes through the center of that sphere. As shown in Fig. 2, the perpendicular line L_1 passes through both the center of the circular cross-section C_1 and the center of the sphere S.
2. If any point is at the surface of sphere then it will satisfy the equation of sphere. For example standard equation of sphere having its center at (x_S, y_S, z_S) and radius 'R' can be expressed as $(x-x_S)^2 + (y-y_S)^2 + (z-z_S)^2=R^2$. If there is a point ($x_1$, y_1, z_1) lies on the sphere then it will satisfy the equation of the sphere. Thus,

$$(x_1 - x_S)^2 + (y_1 - y_S)^2 + (z_1 - z_S)^2 = R^2$$

3. Area of a triangle is the half of magnitude of cross product of side vectors. If ABC is a triangle then area of ΔABC is,

$$\Delta ABC = \frac{1}{2}\left|\overline{BA} \times \overline{BC}\right|$$

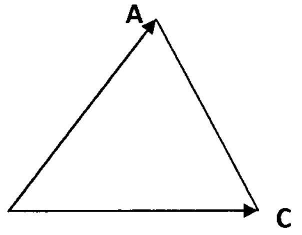


Fig.10: Area of triangle

4. If two vectors are inclined at an angle then sum of these vectors is represented by the diagonal of the parallelogram formed by these two vectors. If $\overline{P}$ & $\overline{Q}$ are two vectors inclined at some angle then ,

$$\overline{P}+\overline{Q}=\overline{R}$$

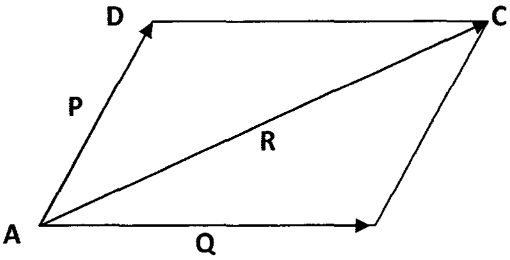


Fig.11: Sum of vectors

3.6 The algorithm for newely proposed approach is as follows:

1.Select only 3 beacon (rather than 4 as in previous proposed methods) points using the flying anchor concept.

2. Substitue the 3 beacon points in equation of sphere to get 2 different equations say eqaution 1 and equations 2.

3.Now assume one circular cross sections say C_1 using all the three chosen beacons.

4.Find the centre of the circle C_1.

5.Now find the line L_1 Passing through the centre of the circle C_1 and perpendicular to the circular cross sections too; as shown in the Fig 2.

6.Now solving the equation of line L_1, equation 1 and equation 2, we get the required location of the sensor.

3.7 Computation of sensor node position i.e. centre of the sphere:-

3.7.1 Selection of beacon points using the flying anchor concept:

Each flying anchor periodically broadcasts beacon messages detailing its ID, position, and timestamp. Meanwhile, each sensor node maintains a set of beacon points and a Visitor List. The beacon point is regarded as an endpoint on the communication sphere of the sensor node, and the

Visitor List maintains the IDs and respective lifetimes of the flying anchors currently passing through this sphere. When a sensor node receives a beacon message from a flying anchor, it first checks whether the flying anchor is already recorded in its Visitor List. If it is missing, a beacon point, i.e., the current position of the flying anchor, is logged, and the ID and corresponding lifetime of the flying anchor are added to the Visitor List. If the flying anchor is already present in the Visitor List, the beacon message is ignored and the lifetime of the anchor is simply extended. When the lifetime of the flying anchor expires, its final beacon message is recorded as a beacon point and the corresponding entry for the anchor is deleted from the Visitor List [1].

Fig. 3 illustrates an example of the beacon point selection procedure. In this example, a flying anchor A moves from (x, y, z) to (x', y', z') broadcasting beacon messages at an interval of t, where $t=T_{i+1} - T_{i,}$ and i=0, 1, 2, 3.... 7. The beacon messages at T2 is considered as a beacon point (A, (x_2, y_2, z_2)) by sensor node S so Node S add an entry (A, (x_2, y_2, z_2), T2 + L) to its visitor list, where 'L' is the predefined lifetime of the flying anchor and has a value larger than the beacon interval 't' (L=at, a >1). The lifetime of Node A is increased by L when it arrives at T3, T4, T5, and T6, respectively. Once node 'A' moves out of the communication sphere of Node S and its lifetime expire, its beacon message at T6 is logged as a beacon point and Node S deletes the entry for Node A from its Visitor List. Similarly we proceed and select 3 beacons say B_1, B_2 and B_3.

3.7.2 Getting 2 equations using the eqation of the sphere and the 3 beacon points:

Consider first beacon message received by the static sensor node is broadcasted from the position $B_1(x_1, y_{1,} z_1)$ and that of the second and third are from B_2 (x_2, y_2, z_2) and B_3 (x_3, y_3, z_3) respectively.

Let S (x_s, $y_{s,}$ z_s) be the sensor node whose location is to be determined and 'R' be the communication range of all sensor nodes. Now the communication range of sensor node L can be represented by-

$$(x-x_S)^2+(y-y_S)^2+(z-z_S)^2=R^2 \qquad (1)$$

As we know that beacons are selected on the surface of the spherical range on the sensor node, therefore the beacons B1(x1, y1, z1), B2 (x2, y2, z2) and B3 (x3, y3, z3) will satisfy the equation of the sphere given in (1) and it will give (2), (3), and (4) as below:

$$(x_1 - x_S)^2 + (y_1 - y_S)^2 + (z_1 - z_S)^2 = R^2 \qquad (2)$$

$$(x_2 - x_S)^2 + (y_2 - y_S)^2 + (z_2 - z_S)^2 = R^2 \qquad (3)$$

$$(x_3 - x_S)^2 + (y_3 - y_S)^2 + (z_3 - z_S)^2 = R^2 \qquad (4)$$

In our assumption we have assumed that the range 'R' of each static sensor will be identical, solving (2), (3), (4) we get (5), (6) as below:

$$(x_1 - x_S)^2 + (y_1 - y_S)^2 + (z_1 - z_S)^2 = (x_2 - x_S)^2 + (y_2 - y_S)^2 + (z_2 - z_S)^2 \qquad \mathbf{(5)}$$

$$(x_1 - x_S)^2 + (y_1 - y_S)^2 + (z_1 - z_S)^2 = (x_3 - x_S)^2 + (y_3 - y_S)^2 + (z_3 - z_S)^2 \qquad \mathbf{(6)}$$

3.7.3 One circular cross sections say C$_1$ using all the three chosen beacons: Fig 12 illustrate this part very well. There are 3 beacons B1, B2, B3 and a circular cross section drawn using the beacons points B$_1$, B$_2$, B$_3$ and a line L$_1$ passing throughh both, centre of the circular cross section and the centre of the sphere S.

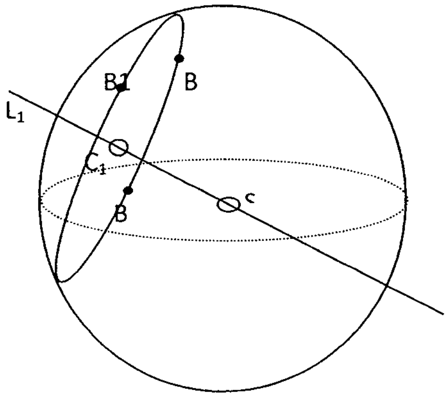


Fig. 12: Circular cross section using three beacons and line L1 passing through C$_1$ and S

3.7.4 How to find the centre of the circle ?

I have proposed two different methods to server this purpose and named these *Vector method* and *perpendicular bisector method.*

1. **Vector method:** We can see in th Fig. 13 that $\overline{OC} = \overline{OP} + \overline{PC}$ and that can be expressed as below:

$$\overline{OC} = \frac{\overline{OB_1} + \overline{OB_2}}{2} + \overline{PC} = x_C\hat{i} + y_C\hat{j} + z_C\hat{k}$$

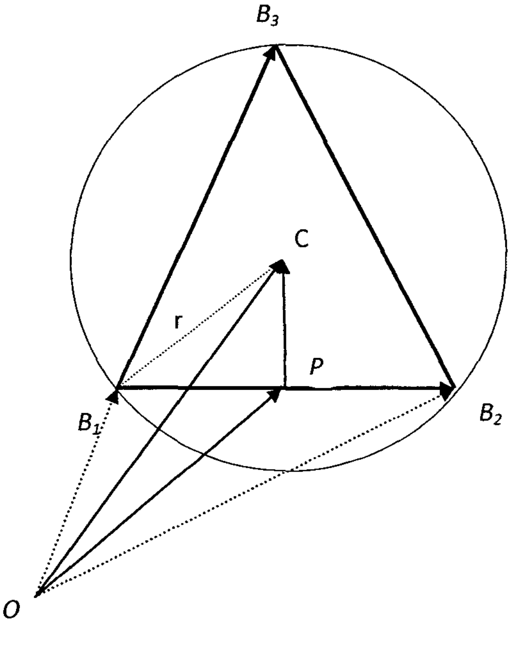


Fig. 13: Centre using the vector method

Let C (x_c,y_c,z_c) is the centre of the circular cross section and radius is 'R' and the area of the traingle is 'a'. Now PC can be given as $PC = \sqrt{r^2 - (a/2)^2}$

where 'a' can be given as:

$$a = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

*and '**r**' can be given as below:*

$$r = \frac{abc}{4\Delta}$$

where a, b, c can be calculated as below:

$$a = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

$$b = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2 + (z_3 - z_2)^2}$$

$$c = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2 + (z_1 - z_3)^2}$$

and Δ Can be calculated as

$$\Delta = \frac{1}{2}\left|\overline{B_1B_2} \times \overline{B_1B_3}\right|$$

$$\Delta = \frac{1}{2}\left(\sqrt{\Delta_x{}^2 + \Delta_y{}^2 + \Delta_z{}^2}\right)$$

Where

$$\Delta_x = (y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1)$$

$$\Delta_y = (z_3 - z_1)(x_2 - x_1) - (x_3 - x_1)(z_2 - z_1)$$

$$\Delta_z = (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)$$

The unit vector in the direction of PC can be given as below:

$$\hat{PC} = \frac{\vec{N} \times \overline{B_1B_2}}{\left|\vec{N} \times \overline{B_1B_2}\right|} = d_1\hat{i} + d_2\hat{j} + d_3\hat{k}$$

Where $\overline{N} = \overrightarrow{B_1B_2} \times \overrightarrow{B_1B_3} = (N_X, N_Y, N_Z)$ and

$$N_X = (y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1)$$

$$N_y = (z_3 - z_1)(x_2 - x_1) - (x_3 - x_1)(z_2 - z_1)$$

$$N_Z = (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)$$

Further,

$$d_1 = \frac{N_Y(z_2 - z_1) - N_Z(y_2 - y_1)}{m}$$

$$d_2 = \frac{N_Z(x_2 - x_1) - N_X(z_2 - z_1)}{m}$$

$$d_3 = \frac{N_X(y_2 - y_1) - N_Y(x_2 - x_1)}{m}$$

$$m = \sqrt{(N_Y(z_2 - z_1) - N_Z(y_2 - y_1))^2 + (N_Z(x_2 - x_1) - N_X(z_2 - z_1))^2 + (N_X(y_2 - y_1) - N_Y(x_2 - x_1))^2}$$

Now using,

$$\overline{OC} = \frac{\overline{OB_1} + \overline{OB_2}}{2} + \overline{PC} = x_C\hat{i} + y_C\hat{j} + z_C\hat{k}$$

We get the centre of the circular cross section in the form as given below:

$$x_C = \frac{x_1 + x_2}{2} + d_1 * PC$$

$$y_C = \frac{y_1 + y_2}{2} + d_2 * PC$$

$$z_C = \frac{z_1 + z_2}{2} + d_3 * PC$$

2. Perpendicular bisector method:

Consider the circular cross-section , the normal vector to the perpendicular bisector of the chord vector **B$_1$B$_3$** i.e. (L$_1$) can be generated by the cross product of $\vec{N}$ and vector **B$_1$B$_3$** and similarily L$_2$ can be generated by the cross product of $\vec{N}$ and **B$_3$B$_2$**, i.e.

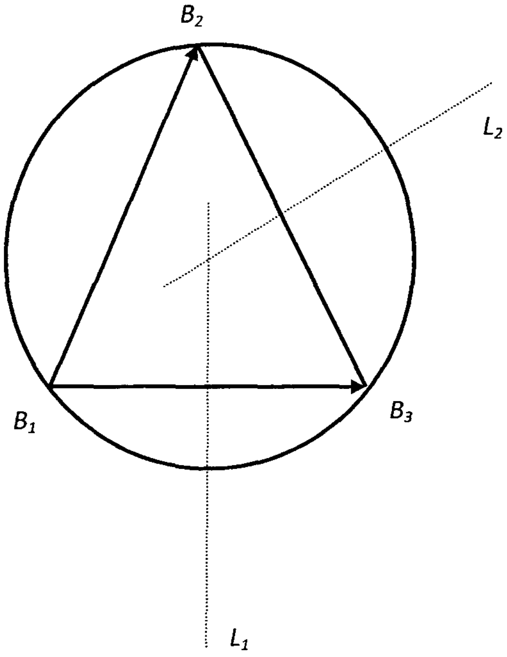


Fig. 14: Centre using the perpendicular bisector method

$$\text{vector } L_1 = \vec{N} * \text{vector } B_1B_3 = (l,m,n) \tag{7}$$

$$\text{vector } L_2 = \vec{N} * \text{vector } B_3B_2 = (p,q,r) \tag{8}$$

Suppose that the equation of the straight lines L_1 and L_2 can be expressed by the following:

$$L_1: \frac{x-a_1}{l} = \frac{y-b_1}{m} = \frac{z-c_1}{n} = t_1 \tag{9}$$

$$L_2: \frac{x-a_2}{p} = \frac{y-b_2}{q} = \frac{z-c_2}{r} = t_2 \tag{10}$$

Where, (a_1, b_1, c_1) and (a_2, b_2, c_2) are the mid points of B_1B_3 and B_3B_2 respectively.Based on the corollary that states that the perpendicular bisector of a chord contains the center of the circle, the intersection point of L_1 and L_2 is at the center of the circle. Therefore the centre of the circle (C) will be ,

$$x_C = l*t_1 + \frac{(x_1 + x_3)}{2}$$

$$y_C = m*t_1 + \frac{(y_1 + y_3)}{2}$$

$$z_C = n*t_1 + \frac{(z_1 + z_3)}{2}$$

& $$t_1 = \frac{p*\left(\frac{(y_1 + y_3)}{2} - \frac{y_3 + y_2}{2}\right) - q\left(\frac{(x_1 + x_3)}{2} - \frac{x_3 - x_2}{2}\right)}{q*l - p*m} \qquad (11)$$

3.7.5 Finding the line L₁ passing through the centre of the circle C₁ and perpendicular to the circular cross section:

Equation of the line L_1passing through the centre of the circle $C(x_c, y_c, z_c)$and perpendicular to the plane of circle i.e. in the direction of $\vec{N}$,

$$\frac{x - x_C}{N_X} = \frac{y - y_C}{N_Y} = \frac{z - z_C}{N_Z} = t \qquad (12)$$

3.7.6 Solving the equation of line L₁, equation 1 and equation 2, to get the required location of the sensor:

The line passing through centre of circular cross section of the sphere and perpendicular to the plane of cross section will also pass through the centre of the sphere." Thus the centre of sphere (x_S, y_S, z_S) will lie on the line (6) and we get the equation as follows:

$$\frac{x_S - x_C}{N_X} = \frac{y_S - y_C}{N_Y} = \frac{z_S - z_C}{N_Z} = t \qquad (13)$$

On solving equation (5), (6) and equation (13), we will get the centre of the sphere S (x_s, y_s, z_s) i.e. the required location of the sensor node,

$$x_S = t * N_X + x_C$$

$$y_S = t * N_Y + y_C$$

$$z_S = t * N_Z + z_C$$

&

$$t = \frac{(x_1^2 + y_1^2 + z_1^2) - (x_2^2 + y_2^2 + z_2^2) - 2(x_1 - x_2)x_C - 2(y_1 - y_2)y_C - 2(z_1 - z_2)z_C}{2(x_1 - x_2)N_X + 2(y_1 - y_2)N_Y + 2(z_1 - z_2)N_Z} \quad (14)$$

3.8 Analysis of the proposed and previous methods:

Many of the papers published on 3D localization using Range free localization method and **[1]** and **[8]** has been proved better than others in Accuracy, Number of computations, and Space complexity. I will analyse and compare proposed scheme in this work with the above two best approaches.

3.8.1 Computational Complexity: As we know that WSNs have limited energy so reducing even little computations any how is an additive gain. Therefore I have compared/analysed my proposed method with two previously proposed best approaches to show the gain over the best existing methods to present the additive gain.

There are three approaches I am comparing, [1] Chord Selection Approach, [8] Point will lie on the surface of sphere concept and my proposed approach which contains two methods of finding the centre of the circular cross section viz. vector method and perpendicular bisector method.

Chord Selection Approach [1]: Using the chord selection approach the number of multiplications are eighty eight, number of additions are sixty five and the number of square roots to be calculated are zero.

Point will lie on the surface of sphere concept [8]: Using the point method number of multiplications required is fifty four, number of additions to be done is forty four and the number of square roots to be calculated is five.

Proposed Approach:

Vector Method: There are two approaches in the proposed method; the first is the vector method which requires sixty eight numbers of multiplications, forty six number of additions and number of square roots to be calculated are five.

Perpendicular bisector method (Best of All the above): The first is the vector method which requires forty nine numbers of multiplications, forty five number of additions and number of square roots to be calculated are zero.

From the above statistics it has been calculated and observed that the proposed method has an additive gain 5% over [8] and 39% over [1] using the Perpendicular bisector method for finding the centre of the circle.

3.8.2 Space Complexity: As we know that WSNs have limitations of memory because we are focusing on the WSNs made up of cheap sensors which will obviously having the limitations of memory space, so reducing the space complexity is also an additive gain.

The comparison among the methods given bellow:

1. Chord Selection Approach [1]: 34
2. Point will lie on the surface of sphere concept [8]: 28
3. My proposed approach:

(i) Vector Method: 28

(ii) Perpendicular bisector method (Best of All the above):24

From the statistics it is obvious that we have additive gain of approx 14% using the second proposed approach.

3.8.3 Gain in terms Number of Beacons: We know that *"in any sensor node most of the energy is consumed in computation and in message transmission and reception"* therefore reducing the number of beacons being used to localize reduces the transmission and reception of the beacon messages. ***Number of beacons used by the previous and current proposed schemes to localize a single node is as below:***

1. *Chord Selection Approach [1]:* ***4***
2. *Point will lie on the surface of sphere concept [8]:* ***4***
3. *My proposed approach:*
    - *(i) Vector Method:* ***3***
    - *(ii) Perpendicular bisector method* ***(Best of All the above):3***

The proposed scheme is using only 3 beacon points rather than four as proposed in [1] and [8]. So proposed scheme reduces the number of beacons being used up to 25% which reduces the messages transmission/reception, so the energy consumption is being reduced by 25% which is and additive gain over [8] and more percentage of gain over [1] because [1] is discarding some of the beacons.

Note: from the above analysis it is obvious that the proposed scheme is having additive gain over the previous proposed approached in each metric.

CHAPTER 4: SIMULATION AND RESULTS

I have simulated the proposed scheme using Simulator for Networking Algorithm (SINALGO) which provides simulation framework for 2D as well as 3 dimensional networking algorithms as 3 dimensional environments are not supported by the wireless extensions of the Network Simulator -2.

4.1 Simulation Environment: The simulations are performed on a region having volume of 1000*1000*1000 $m^3$and deployed 3000 static sensor nodes. The environment is shown in the Fig 15. And deployed GPS enabled mobile sensor nodes of 1%, 2%, 3%, 4%, and 5% of total number of deployed static sensor nodes.

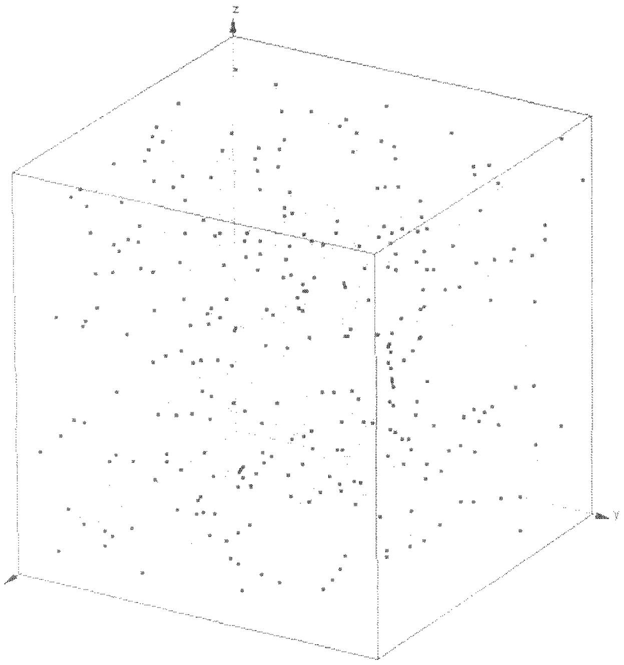


Fig.15: simulation environment

The mobile sensor nodes are moved according to random way point method and random direction walk. In random way point technique, the mobile nodes are deployed randomly. They randomly choose any destination and move in that direction with constant speed. Once they reach the destination they repeat the whole process. The random direction walk is similar to random way point walk. The only difference is the choice of the target. Instead of picking a random point from the deployment field, the random direction chooses a direction in which the node should walk, and how long the node should walk in this direction. If the node hits the boundary of the deployment area, it is reflected just as a billiard ball.

4.2 Simulation Settings: The other parameter settings for the simulation are summarized below:

Synchronous Mode:	True
Interference Model:	No
Distribution Model:	Random
Connectivity Model:	UDG
Reliability Model:	Reliable Delivery

General Configuration:

Command line arguments:

Java VM arguments: -agentlib:jdwp=transport=dt_socket,suspend=y,address=localhost:3910 -Dfile.encoding=Cp1252

Classpath: F:\ASSIGNMENTS\DISSERTATION DRAFT\NETWORK_SIMULATOR_
SOURCE\workspace\localizationin3d\binaries\bin;F:\ASSIGNMENTS\DISSERTATION
DRAFT\NETWORK_SIMULATOR_SOURCE\workspace\localizationin3d\binaries\
jdom.jar

Configuration settings

Simulation Area

dimensions = 3
dimX = 1050
dimY = 1050
dimZ = 1000

Simulation

asynchronousMode = false
mobility = true
interference = false
interferenceIsAdditive = true

```
   canReceiveWhileSending = true
   canReceiveMultiplePacketsInParallel = true
   edgeType = sinalgo.nodes.edges.Edge
   exitOnTerminationInGUI = false
   initializeConnectionsOnStartup = false
   refreshRate = 1
   generateNAckMessages = false
   handleEmptyEventQueue = true
   javaCmd = java
   javaVMmaxMem = 500
Random number generators
   useSameSeedAsInPreviousRun = false
   useFixedSeed = true
   fixedSeed = 77654767
 Logging
   logFileName = logfile.txt
   outputToConsole = true
   logToTimeDirectory = true
   logConfiguration = true
   eagerFlush = false
GUI
   extendedControl = false
   drawArrows = false
   zoomStep = 1.2
   wheelZoomStep = 1.05
   minZoomFactor = 0.05
   draw3DGraphNodesInProperOrder = true
   usePerspectiveView = true
   perspectiveViewDistance = 40
 Background map in 2D
```

```
  useMap = false
  map =
Models
  DefaultMessageTransmissionModel = ConstantTime
  DefaultConnectivityModel = sample3: AntennaConnection
  DefaultDistributionModel = Random
  DefaultInterferenceModel = NoInterference
  DefaultMobilityModel = NoMobility
  DefaultReliabilityModel = LossyDelivery
  DefaultNodeImplementation = sample3:RMobileNode
  showModelsOfAllProjects = false
Node storage, position transformation
  guiPositionTransformation2D = sinalgo.gui.transformation.Transformation2D
  guiPositionTransformation3D = sinalgo.gui.transformation.Transformation3D
  nodeCollection2D = sinalgo.runtime.nodeCollection.Geometric2DNodeCollection
  nodeCollection3D = sinalgo.runtime.nodeCollection.Geometric3DNodeCollection
Export Settings
  epsToPdfCommand = epstopdf %s
  epsDrawDeploymentAreaBoundingBox = true
  epsDrawBackgroundWhite = true
Animation Settings
  showMessageAnimations = false
  messageAnimationEnvelopeWidth = 30.0
  messageAnimationEnvelopeHeight = 20.0
  messageAnimationEnvelopeColor = r=255,g=255,b=0
Diverse Settings
  showOptimizationHints = false
  drawEdges = true
  drawNodes = true
  shownEventQueueSize = 10
```

```
   outputTextFieldHeight = 200
   arrowLength = 8
   arrowWidth = 1
   defaultRoundNumber = 1
Custom settings
antenna/inviteintervall/distribution = Uniform
 antenna/inviteintervall/max = 12
 antenna/inviteintervall/min = 6
 antenna/refreshrate = 2
 antenna/size = 15
 edge/activecolor = 0xdd7700
 edge/passivecolor = 0x8888cc
 geometricnodecollection/rmax = 100
 lossydelivery/droprate = 0.1
 messagetransmission/constanttime = 1
 mobilenode/size = 4
 node/defaultsize = 10
 randomdirection/movetime/distribution = Uniform
 randomdirection/movetime/max = 20
 randomdirection/movetime/min = 5
 randomdirection/nodespeed/distribution = Gaussian
 randomdirection/nodespeed/mean = 2
 randomdirection/nodespeed/variance = 6
 randomdirection/waitingtime/distribution = Poisson
 randomdirection/waitingtime/lambda = 10
 udg/rmax = 100
Seed for Random Number Generators
 Fixed seed: 77654767
End of settings
```

4.3 Performance Metric: Three metrics were established to evaluate the performance of the localization mechanisms by [1]:

(1) Average location error: The average distance between the estimated location (Xe_i, Ye_i, Ze_i) and the actual location (X_i, Y_i, Z_i) of all the sensor nodes, i.e.

$$\text{Average location error} = \frac{\sum\sqrt{(Xe_i - X_i)^2 + (Ye_i - Y_i)^2 + (Ze_i - Z_i)^2}}{Number\_of\_SensorNodes}$$

(2) Average localization time: The average time required for all the sensor nodes to compute their locations, i.e.

$$\text{Average localization time} = \frac{\sum Localization\_Time}{Number\_of\_SensorNodes}$$

(3) Beacon Overhead: The average number of beacon messages broadcast by the flying anchors during the total localization time, i.e.

$$\text{Beacon overhead} = \frac{Number\_Of\_Beacons\_Messages}{Number\_Of\_Flying\_Anchors}$$

In any sensor node most of the energy is consumed in computation and in message transmission and reception and the above two metrics defines energy efficiency.

4.4 Results: Simulations are performed using SINALGO and on the basis of some metrics following results were observed:

4.4.1 Time of Localization Vs Number of Nodes: Simulations have been performed on different data sets varying in numbers of static nodes for proposed method and method discussed in [8]. The graph shown in the Fig 16 best illustrate the comparison.

Let $T_{[8]}$ is the total time taken in localization by the scheme proposed in [8] and $T_{[8]}$ can further be divided into t_1 (i.e. Time taken to select 4 beacons), t_2 (i.e. Time taken in calculation). Initially t_1 will be negligible with respect to the t_2 because the static nodes will be enough dense and gain will be 5%, gradually it will be difficult to select the beacons so t_1 will increase and when t_1 will increase t_2 will be negligible with respect to t_1 so maximum gain here will be 25%.

$T_{[8]} = t_1 + t_2$ (A)

Now the time taken in localization using the proposed scheme can be given as below in the form of (A)

$\mathbf{T_{proposed} = .75 * t_1 + .95 * t_2}$

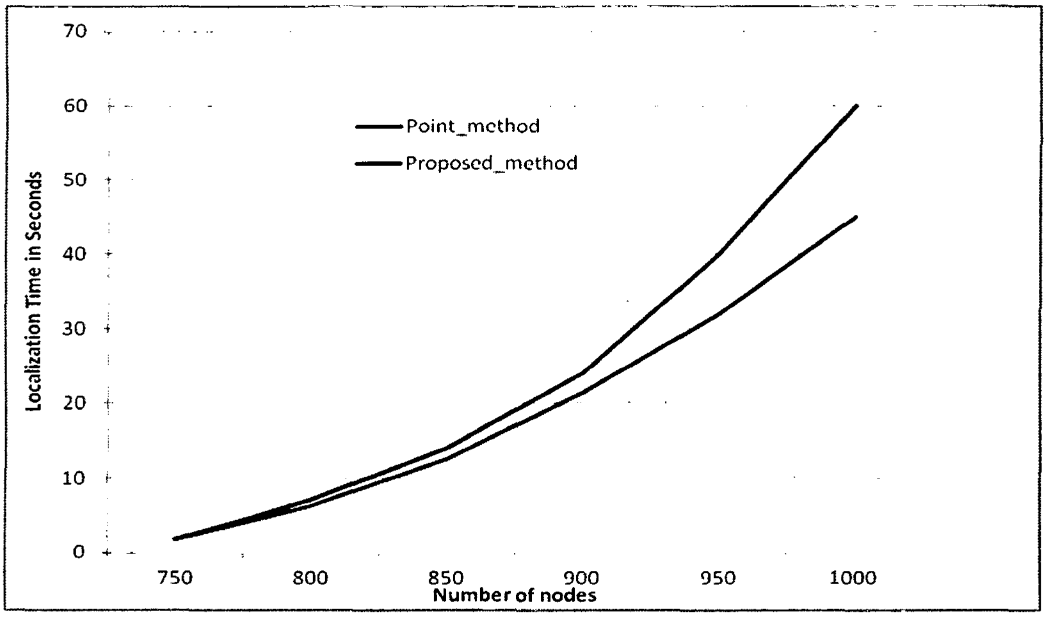


Fig 16: Comparison of Time of Localization vs. No. of nodes localized

From the Fig 16 it is obvious that the localization time taken in proposed localization scheme is less than the scheme proposed in [8].

4.4.2 Average Localization Time Vs % Number of moving anchors:

Simulations has been performed for all the three methods and following result have been observed

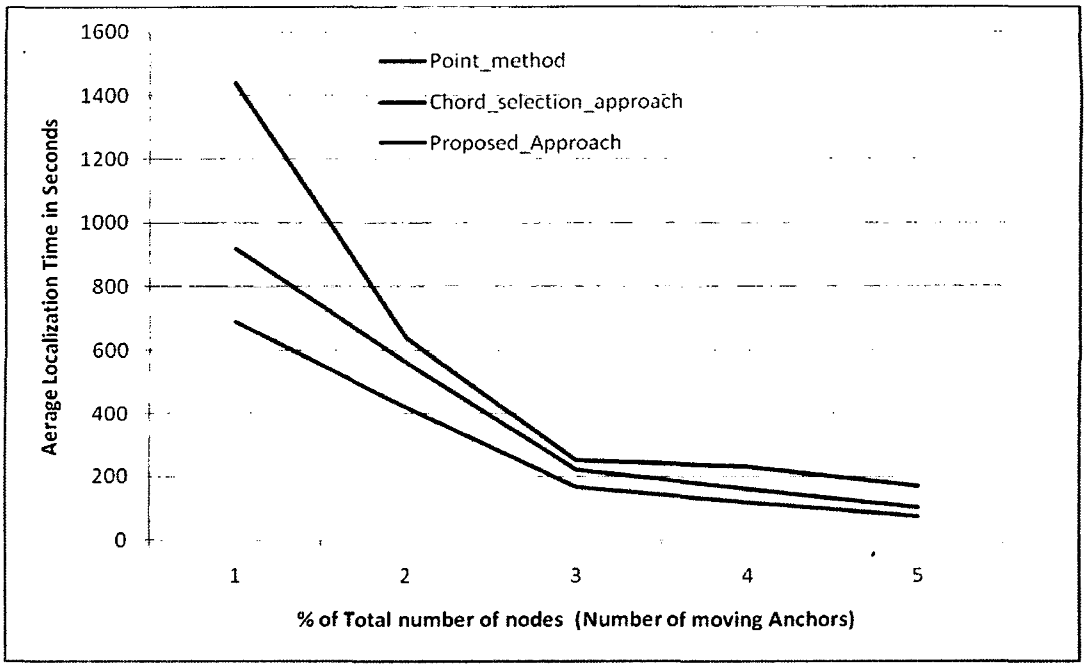


Fig 17 : Average Localization Time Vs % Number of moving anchors

4.4.3 Number of Moving Anchors Vs Beacon Overhead: In the performance metrics Beacon Overhead is given as Average number of beacon messages broadcast by the flying anchors during the total localization time, i.e.

$$\text{Beacon overhead}=\frac{Number\_Of\_Beacons\_Messages}{Number\_Of\_Flying\_Anchors}$$

As we can see that beacon overhead is directly proportional to Number of beacon messages, and in proposed approach number of beacon messages is reduced significantly that is 25%. Furthermore no beacon message is being discarded in any case. Following results (shown in graph below) has been observed in comparing all the three approaches.

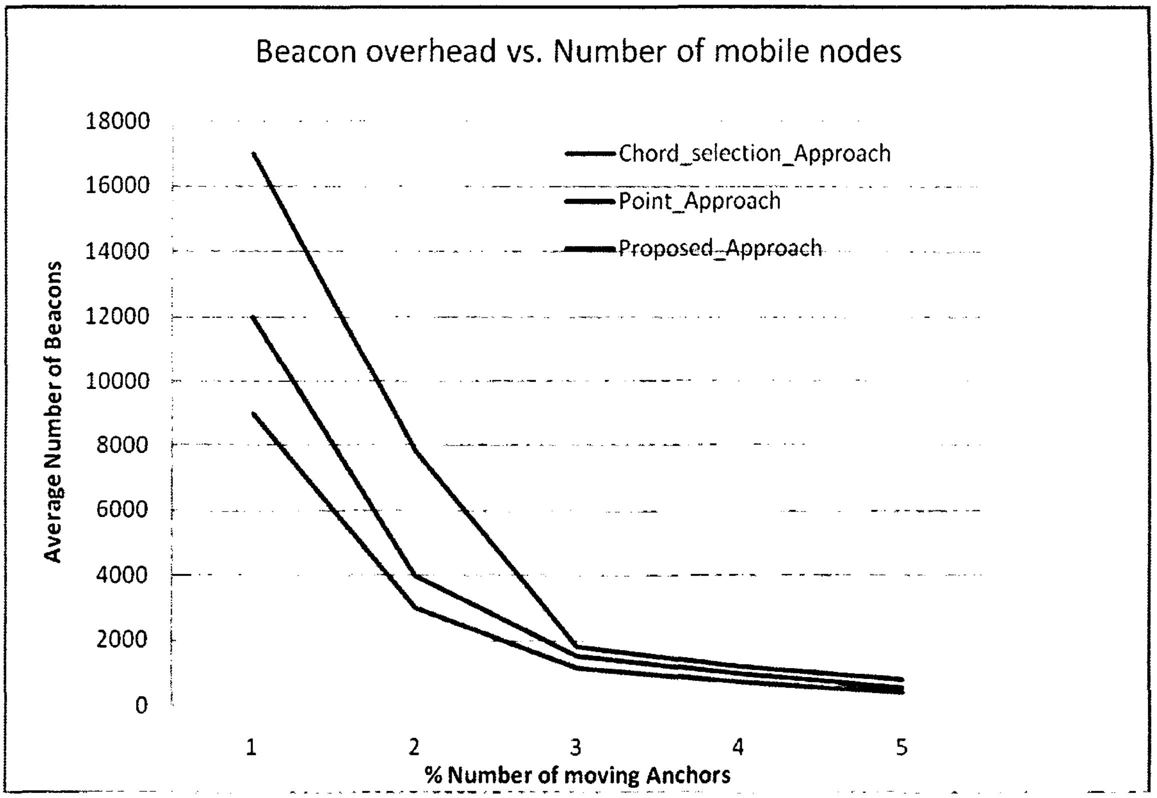


Fig 18 : Beacon overhead vs. Number of mobile nodes

CHAPTER 5: CONCLUSION & FUTURE WORK

This work proposes and implements a range-free localization scheme for 3-dimensional networks. This work is truly based on the flying (Moving) anchor concept. The proposed mechanism utilizes the location information transmitted from flying anchors to determine the positions of the sensor nodes. Based on standard geometric corollaries, each sensor node computes its position locally.

The proposed scheme performs independent of the influence of network densities and topologies. The localization mechanism is accurate, distributed, scalable, and power efficient. A series of simulations have been performed using the SINALGO network simulator to evaluate the performance of the mechanism and to compare its results against those provided by [1] and [8]. The simulation results have shown this position determination mechanism outperforms both proposed methods in [1] and [8] in terms of a reduced localization time, a lower beacon overhead and increased energy efficiency.

Already localized nodes can participate in process of localization, so new algorithms can be developed using proposed method for improving efficiency.

REFERENCES:

[1] Chia-Ho Ou and Kuo-Feng Ssu. (2008) *"Sensor position determination with flying anchor in three dimensional wireless sensor networks."* In IEEE Transactions on Mobile Computing, pages 1084--1097, September 2008.

[2] Amitangshu Pal, *"Localization Algorithms in Wireless Sensor Networks: Current Approaches and Future Challenges",* Department of Electrical and Computer Engineering, The University of North Carolina at Charlotte, 9201 University City Blvd, Charlotte, North Carolina 28223-0001, apal@uncc.edu

[3] Brian M. Blum, John A. Stankovic, Tarek Abdelzaher, Tian He and Chengdu Huang (2005), *"Range-free localization and its impact on large scale sensor networks"* In ACM Transactions on Embedded Computing Systems, volume 4, pages 877 -- 906, November 2005.

[4] Wei Yawen, Yu Zhen, Guan Yong (2007), *"Location Verification Algorithms for Wireless Sensor Networks"* Iowa State University,Ames, Iowa 50011, weiyawen, yuzhen, yguan@iastate.edu

[5] Jiang Jin, Yi Wang, Chen Tian, Wenyu Liu and Yijun Mo. (2007) *"Localization and synchronization for 3d underwater acoustic sensor networks."* In Springer, Ubiquitous Intelligence and Computing 2007, pages 622-631, Hong Kong, China, July 2007.

[6] Paschos G. S., Vagenas E. D., Kotsopoulos S. A., *"Real-time Localization for Wireless Sensor Networks with multiple beacon transmissions",* Wireless Laboratory, University of Patras, Greece, e-mail: gpasxos@ee.upatras.gr, svagenas@ ee.upatras.gr, kotsopoulos@ee.upatras.gr

[7] Marcelo Martins, Hing Cheung So, Hongyang Chen, Pei Huang and Kaoru Sezaki(2008), *"Novel centroid localization algorithm for three-dimensional wireless sensor networks."* In IEEE Transactions on Wireless Communication, Networking and Mobile Computing, pages 1-4, October 2008.

[8] Vibha Yadav, Manas Kumar Mishra, A.K. Singh and M. M. Gore (2009), *"localization scheme for three dimensional wireless sensor networks using GPS enabled mobile sensor nodes"*

Department of Computer Science & Engineering, MNNIT, Allahabad, INDIA in International Journal of Next-Generation Networks (IJNGN),Vol.1, No.1, December 2009

[9] Wei Cheng, Amin Y. Teymorian, Liran Ma, Xiuzhen Cheng (2008), *"Underwater Localization in Sparse 3D Acoustic Sensor Networks"*, Department of Computer Science, The George Washington University, Washington, DC, USA. Xicheng Lu, Zexin Lu ,School of Computer Science, National University of Defense Technology,Changsha, Hunan, PRC, Email: {xclu,lzx}@nudt.edu.cn

[10] Zhen Hu, Zhengxun Song, Hongzuo, *"Localization in Wireless Sensor Networks Using a Mobile Anchor Node",* School of Electronics and Information Engineering, Changchun University of Science and Technology. Dongbing Gu, Department of Computing and Electronic Systems, University of Essex, Colchester, United Kingdom.

[11] CHRAIBI YOUSSEF, (Sweden 2005) *"Localization in Wireless Sensor Networks",* Master's Degree Project, Stockholm, Sweden 2005, IR-RT-EX-0523

[12] Bachrach Jonathan and Taylor Christopher, *"Localization in Sensor Networks",* Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 {taylorc, bachrach}@mit.edu

[13] Bharathidasan Archana, Anand Vijay, Ponduru Sai, *"Sensor Networks: An Overview",* Department of Computer Science, University of California, Davis, CA 95616

[14] Sahni Sartaj and Xu Xiaochun, (September 7, 2004), *"Algorithms For Wireless Sensor Networks",* Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611

[15] Srinivasan, A. and J. Wu, 2007. *"A Survey on Secure Localization In Wireless Sensor Networks. In: Wireless And Mobile Communications",* Furht, B. (Ed.). CRC Press, Boca Raton, London

[16] Dhekne Prabhakar, August 24, 2006, *"Wireless Sensor Network"*, Bhabha Atomic ResearchCentre , Talk at SASTRA [www.sastra.edu/core/guestlect/1dhekne240806.ppt]

[17] I.F. Akyildiz, W. Su*, Y. Sankarasubramaniam, E. Cayirci,(20 December 2001), *"Wireless sensor networks: A survey"*, Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA,

[18] Kay Romer and Friedemann Mattern, (December 2004), *"The Design Space of Wireless Sensor Networks"*, Institute for Pervasive Computing, ETH Zurich. {roemer|mattern}@inf.ethz.ch

[19] Thomas Haenselmann, (2006-04-05), *"Sensornetworks, GFDL Wireless Sensor Network* ", textbook, http://pi4.informatik.uni-mannheim.de/~haensel/sn_book, retrieved 2006-08-29 .

[20] N. Bulusu, D. Estrin, L. Girod and J. Heidemann, (July 2001) *"Scalable Coordination for wireless sensor networks: Self-Configuring Localization Systems,"* In Proceedings of the Sixth International Symposium on Communication Theory and Applications (ISCTA 2001), Ambleside, Lake District,UK.

[21] http://en.wikipedia.org/wiki/WSN#Characteristics

[22] http://en.wikipedia.org/wiki/Wireless_sensor_network

[23] http://www.howstuffworks.com/gps1.htm

[24] http://www.isi.edu/nsnam/ns/