Localization in Underwater sensor Networks

Priyansh Kumar Dubey
Department of Computer Science and Engineering
National Institute of Technology Karnataka
Surathkal, Mangalore, India
priyansh.dby25@gmail.com

***Abstract*—This project proposes a new localization technique where whenever a new node is introduced in an underwater sensor network, we will be able to derive its coordinate only by examining the coordinates of three of its nearest neighbors. The nearest neighbors will anchor nodes whose coordinates are known, and the distance will be calculated by estimating the round trip time. The simulations are to be performed first on MATLAB and finally on UnetStack. The detailed procedure, along with the background mathematical model, is discussed in this paper.**

***Index Terms*—Beacon, anchors, localized node, blind node, Localization.**

I. Introduction

Recently, there has been increasing interest in underwater wireless sensor networks (UWSNs), a primary source for exploring the ocean environment. The UWSN is being developed by the ad-hoc wireless sensor network (WSN) applications and wireless technologies. With the advances of communication and sensor technology, underwater wireless sensor networks (UWSNs) have become a fast-growing field [1]. They have many exciting applications, such as pollution monitoring, off-shore oil exploration, and oceanography data collection [2]. Localization is a prominent issue in underwater wireless sensor networks because only the sensor node location data can be successfully decoded by the end-users [3]. If we compare it with the terrestrial localization issue, underwater localization has some fundamental differences because the radio waves used in GPS don't work in water. The Global Positioning System (GPS) technology is unavailable in the aquatic environment because of radio waves strong attenuation in water [4]. Thus, the underwater node cannot directly locate itself through the GPS. Second, due to the considerable propagation delay in underwater acoustic channels, the cost of maintaining clock synchronization is high and sunken nodes are usually clocked asynchronously. Third, the submerged sensor node's transmitting energy consumption is several times or even hundreds of times the receiving energy consumption.

For naval [5] defense purposes, UWSNs can provide an immediate deployment option and enhance coastal area surveillance application coverage. UWSNs can be mounted on the bottom of the ocean with underwater sensor nodes that can detect tsunami formations and earthquakes before entering the residential areas. A rough drawing of the underwater node operation is shown in Figure 1. UWSNs can track polluted waters for water pollution detection devices to propagate clean water from their source and warn authorities to take action. UWSNs could be used to monitor coastal creatures and coral reefs, where there is limited data about human activity. Environmental monitoring is also an essential aspect of determining safety and health problems for the environment or human health.

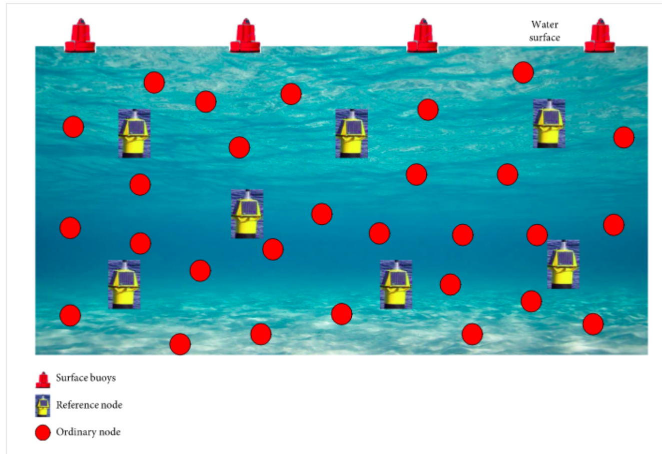

Fig. 1: Underwater node operation

Underwater sensing applications The need to sense the underwater world paves the way for the development of underwater wireless sensor networks. The applications may have a wide variety of requirements such as fixed or mobile, short or long-lived, best-effort, or life-or-death; these requirements further results in various design paradigms.

Underwater networks are often static, i.e., individual nodes attached to docks, too anchored buoys, or the seafloor (as in the cabled or wireless seafloor sensors in Fig. 1). Along with this, there are semi-mobile underwater networks that can be suspended from buoys and deployed by a ship and used temporarily but then left in place for hours or days. The topologies of these networks are static for long durations, allowing the network topology engineering to promote connectivity. However, network connectivity still may change due to the small-scale movement (as a buoy precesses on its anchor) or water dynamics (as currents, surface waves, or other effects change). When battery-powered, static deployments may be energy-constrained.

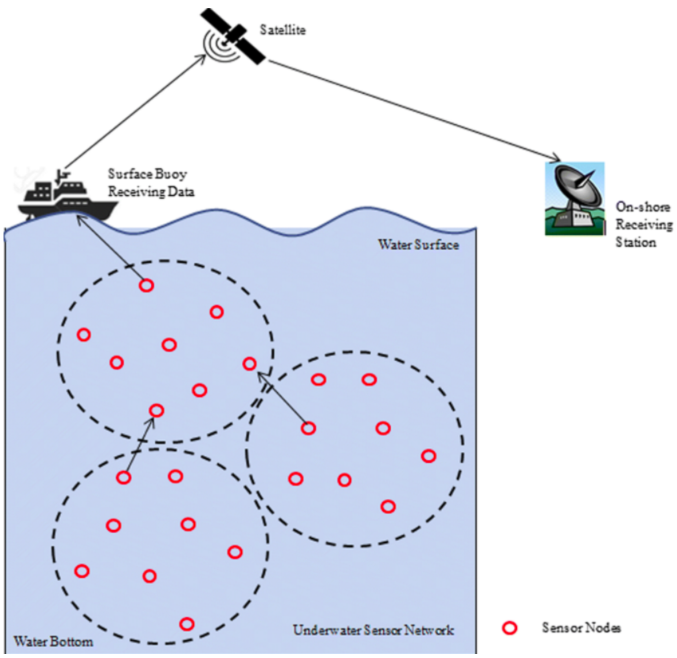

Fig. 2: Communication through UWSNs

The communication in UWSNs is assisted with one or more surface buoys. A surface buoy is a node that is placed partially above the water level and somewhat below the water level. The purpose of this surface buoy is that it sends the data to the satellite, which further forwards the data to the receiving station situated on the land. So the communication system is not entirely underwater; some part of it is completed above the water level.

In underwater sensor networks, determining every node's location is essential to judge the faults or danger present in the network, and the process is known as localization. As GPS does not work in underwater sensor networks, we need to find some method that can leverage the fact that acoustic modems work well in water compared to RF signals of GPS. In simple terms, localization refers to determining the coordinates of a new node that is introduced in the network by taking the reference of the nodes that are already present in the network.

The node placed inside the water needs to know its location because while collecting important data about its surroundings, the receiver must know where that particular data belongs. The process of estimating the coordinates of a node in the network is known as localization. This work is focused on implementing a localization algorithm that is quite popular in wireless sensor networks but not used in underwater sensor networks. In the upcoming section, I will discuss different methods of localization that are currently available in UWSNs. Next, I will discuss the mathematical model, which is the basis of the localization algorithm that I have implemented. Following this, I will present the step-by-step algorithm for the localization of a blind node introduced in the network. Next, I will discuss the Unetstack work after giving its brief overview. Finally, the metrics on which the algorithm is tested are discussed along with the results and conclusion and discuss future work is discussed at last. In underwater sensor networks, determining every node's location is essential to judge the faults or danger present in the network, and the process is known as localization. As GPS does not work in underwater sensor networks, we need to find some method that can leverage the fact that acoustic modems work well in water compared to RF signals of GPS. In simple terms, localization refers to determining the coordinates of a new node that is introduced in the network by taking the reference of the nodes that are already present in the network.

II. Related work

In this section, a brief description of underwater acoustic communication will be discussed. After that, I will review some of the prominently known schemes typically used in underwater localization. Today, the underwater communication system utilizes EM, optics, and acoustic data transmission schemes to transfer data among the nodes' various locations. The conducting nature of the aquatic environment influences EM communication. Simultaneously, optic waves can only move on very short distances because optic waves are more comfortable absorbing underwater environments. Therefore, an acoustic communication scheme is only one scheme with better performance than EM and optical because of less attenuation in the aquatic environment. Acoustic signals also have less attenuation in the deep and thermally stable underwater field. Acoustic signals attenuate more in shallow than deep water because of the temperature, noise, and multipath reflection and refraction. In the aquatic area, the sound speed is not constant; instead, it varies almost at every point. Near to the water surface, the proper pace is nearly 1500, four times higher than the sound rate in air and slower than the EM and optic speed in air.

2.1 Overview of Localization Approaches

The localization techniques can be classified into two broad categories;range-free and range-based [6]. Range-free methods usually employ the local topology connectivity to estimate the sensor node's location and obtain a coarser site. In contrast, range-based approaches determine sensor nodes' locations using various ranging methods and provide an accurate location. Ranging techniques can be classified into four categories [7]: Time of Arrival (TOA), Time Difference of Arrival (TDOA), Angle of Arrival (AOA), and Received Signal Strength Indicator (RSSI). The distance information obtained by the TOA method is usually used, such as a vast coverage positioning system (WPS), GPS-less localization protocol (GPS-less), motion-aware sensor localization (MASL), sparse underwater positioning (USP), and 3D underwater localization (3DUL). TOA and TDOA methods require exact time synchronization of sensor nodes, which is hard to achieve due to acoustic signal characteristics when traveling in water. In the AOA method, special devices for directional transmission and reception are needed, which increases the cost of localization. The RSSI manner is cost-efficient and convenient because it does not require time synchronization and special devices. Since the RSSI value is vulnerable to environmental disturbances, it isn't easy to get accurate, ranging by directly mapping the absolute RSSI value to the physical distance.

2.2 Range-Based Algorithm In the range-based algorithm, accurate estimation of distance or angle measurement is made, and TDoA, ToA, and AoA algorithms are generally used for this purpose. Because of constraints like the time-varying characteristics, which are negligible in UASNs, RSSI is not beneficial. On the other hand, the time difference between distinct transmission mediums from specific reference nodes used to assess the distance between two objects is employed in TDoA. ToA, on the other hand, operates the time of arrival using distance estimation. In the suggested range-based algorithms, the most frequently employed technique for UASNs is the ToA algorithm, and it is favored in UASNs compared to terrestrial. It is smaller than the radio signal in the atmosphere due to the sound velocity in water. ToA is mainly implemented to UASNs, although ToA needed synchronization among nodes. In the second algorithm, the authors have presented a Circle-Based Range-Free Localization Algorithm (CRFLA) [8] to locate the unlocated nodes that cannot acquire position information through the first phase. For the second phase, finding the location is the conduct of those sensor nodes situated in the first phase. Below is a comprehensive TDoA, ToA, AoA, and RSSI description.

2.2.1 Time Difference of Arrival (TDoA)

The TDoA is a standard method used for localizing underwater; As we know that the GPS signals are extremely attenuating underwater conditions, it is necessary to develop a precise range-based algorithm for locating underwater. The authors define a sequential method for time synchronization, and localization in the acoustic channel [9]. The transmission source time is unknown, and ToA measurements have a favorable prejudice due to the synchronization mistake, leading to a big localization mistake. One method uses TDoA-based measures to fix this issue, which does not rely on a transmitter source's transmission time. They also use the technique of SDP to turn the nonconvex MLE issue into a convex point. The authors consider it a realistic situation in which nodes are not time-synchronized. Underwater sound speed is also unknown, validating localization as a series of two linear estimation issues—the propagation velocity changes with depth, temperature, salinity. The nodes of the water current are continually moving or their self-motion. The authors define a new associated method based on the directional navigation algorithm used in nodes to achieve precise short-term estimates of movement and using continuous nodes sequential algorithm in an underwater setting for joint time synchronization and location. This method achieves a particular localization environment is utilizing only two anchor sensor nodes and exceeds the benchmark systems when node synchronization and propagation speed data are unknown.

2.2.2 Angle of Arrival (AOA)

AoA is a distributed localization algorithm, where it is assumed that all unknown sensor nodes can detect incident signal angles from nearby sensor nodes [10]. As an antenna, a small array is used, and the phase derivative along the axis of the array is used as a measure of the apparent sin of AoA. Perhaps the pronounced DoA lies outside the precise range of arrivals. This is understood from a situation between two array components with a null attenuation. The AoA capability offers nearby nodes about the node's axis for each node bearing; a Radial is an angle from which one can see an object from another point or a reverse bearing soon. In cases where nodes transmit their bearings concerning beacon nodes, AoA-based systems are accountable. In, A scheme used to collect coincident time, and AoA information at some GHz is defined. The algorithm for data processing is also described, and its outcomes are analyzed from data collected in two distinct structures. A model is suggested based on the measurement consequence used by Saleh and Valenzuela (1987) in the clustered double Poisson ToA model. In the time-angle indoor multipath data, a proper clustering shape is determined .

2.3 Range-Free Algorithm

The range-free localization algorithms do not need to use the various bearing information. It only gives a coarse estimation of the sensor position node that is distinct from the range-based algorithm. The author suggested a hybrid localization algorithm with multi-hop, mobile underwater acoustic networks to enhance the localization scheme's effectiveness in a mobile aquatic medium. The network's sensor nodes are divided into multistage nodes for this algorithm, and each stage has a distinct localization operation. Both range-based algorithms and range-free algorithms are employed to enhance precision and decrease localization's communication costs. This algorithm doesn't involve any previous knowledge of the velocity of motion that is readily used in an underwater medium. The range-free algorithm is categorized into the hop count-based, area-based algorithm, and centroid algorithms.

2.3.1. Hop Count-Based Algorithms

In this method, a square grid is taken, and anchor nodes are placed along with the grid's boundaries. Three algorithms are DV-Hope, stable positioning algorithm, and DHL. The DV-Hop utilizes an average estimation of hops' spectrum and the counted number of hops to estimate the anchor node's distance. A robust positioning algorithm is used to raise DV-Hop by inserting an extra refinement step, while DHL may use density consciousness rather than statically estimate distance dynamically. This method is ideal for real-life deployments, where sensor node distribution is more likely to be nonuniform and concentrated in some areas where it is uneven and sparse. DHL has been recommended to improve position estimation accuracy when the network distribution is not uniform [11]. This program needs to consider the density of the node's neighborhood calculates the average hop distance, and the wrong facts distance estimates tend to accumulate with increasing path length.

2.3.2 Area-Based Algorithm

Area-Based Localization Scheme (ALS) [12] and approximate point in a triangle (APIT) are the two area-based algorithms. ALS is a centralized range-free system whose primary benefits are the resistance and simplicity underwater to the variable sound speed. They can measure a sensor node's location within a specific functional area. The 3D multipower area localization scheme (3D-MALS) is an extended version of 2D-ALS for 3D, whereas APIT requires a heterogeneous network. Anchors are fitted with high-power transmitters and can accurately acquire location data using GPS coordinates. The underwater acoustic method is generally nonlinear and hard to evaluate, so a correct nonlinear algorithm is required.

2.4 Prediction-Based Localizations

This type of localization algorithms [13] is used to predict an estimated location of the required node. The exact location of the node is not available through these methods. These localization algorithms use information from more than one node to accurately estimate the location.

2.4.1 Collaborative Localization (CL) A Collaborative Localization (CL) scheme considering mobile UASN applications is one where underwater sensor nodes are accountable for gathering ocean depth information. They are responsible for transferring it to the water surface. This architecture uses profilers and two types of sensor nodes. These two kinds of sensor nodes come down underwater, but profilers are more resonant than other nodes. The distance between profilers and followers is periodically evaluated using the ToA to place the profilers to followers. The target position is estimated using maximum likelihood (MLL) techniques based on arrival distance (DoA). The statistical model between the real and measuring location is based on the marine channel signal envelope. The suggested algorithm also follows the distributed-centralized computing operation that minimizes the energy transmitting node in USNs.

2.4.2 Distributed Localization In this localization method, each sensor node is located separately. The nodes are free to identify neighborhood distance, anchor position, and connectivity data and then send them individually to the reference or supernode. On the other hand, instead of being placed in one central entity, the function of location finding is distributed to the sensors themselves in the distributed localization algorithm. In a distributed network, sensor nodes communicate through peer-to-peer (P2P).

In this project, I am assuming that to determine the location of a sensor node; it needs to be in the range of some other nodes whose location in terms of coordinates is known to us. The nodes whose location is known to us are called anchor nodes, and the one whose location needs to be derived is called a blind node. When the node is in the range of at least three other nodes known to us, using the location of at least three anchor nodes and their range from the blind node, the location of the blind node is derived.

III. Mathematical Modelling

The concept [14] is based on the principle of the intersection of three circles: three circles will have only one unique point of intersection; there is no way that a combination of three circles will have more than one point of intersection.

The idea here is that if we can get the coordinates of the neighboring three nodes and their distance from the blind node, then we can assume that three circles are drawn with the adjacent nodes as a center, and the blind node must lie on the three circles. Hence it must satisfy the equation of all three circles.

Following are the equations of the three circles.

$$(X - X_a)^2 + (Y - Y_a)^2 = d_a^2 \dots\dots\dots\dots\dots\dots(1)$$

$$(X - X_b)^2 + (Y - Y_b)^2 = d_b^2 \dots\dots\dots\dots\dots\dots(2)$$

$$(X - X_c)^2 + (Y - Y_c)^2 = d_c^2 \dots\dots\dots\dots\dots\dots(3)$$

These equations can be furthur eXpanded to:

$$d_a^2 = X^2 - 2X.X_a + X_a^2 + Y^2 - 2Y.Y_a + Y_a^2 \dots\dots\dots\dots\dots\dots(4)$$

$$d_b^2 = X^2 - 2X.X_b + X_b^2 + Y^2 - 2Y.Y_b + Y_b^2 \dots\dots\dots\dots\dots\dots(5)$$

$$d_c^2 = X^2 - 2X.X_c + X_c^2 + Y^2 - 2Y.Y_c + Y_c^2 \dots\dots\dots\dots\dots\dots(6)$$

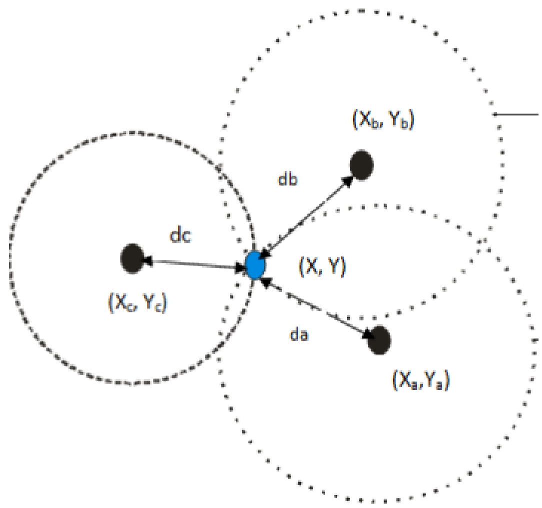

Fig. 3: Intersection of 3 circles at a unique point

The three equations (4),(5), and (6) are independent non-linear simultaneous equations equation (6) was subtracted from equation (5) and equation (4) was subtracted from equation (5) to get the following linear equations:

$$d_b^2-d_c^2 = 2X(X_c-X_b)+X_b^2-X_c^2+2Y(Y_c-Y_b)+Y_b^2-Y_c^2....(7)$$

$$d_b^2-d_a^2 = 2X(X_a-X_b)+X_b^2-X_a^2+2Y(Y_a-Y_b)+Y_b^2-Y_a^2....(8)$$

Rearranging and resolving the above equation, we get the intersection point 'X' and 'Y.'

$$Y = \frac{v_b(X_c - X_b) - v_a(X_a - X_b)}{(Y_a - Y_b)(X_c - X_b) - (Y_c - Y_b)(X_a - X_b)}$$

$$X = \frac{v_a - Y(Y_c - Y_b)}{(X_c - X_b)}$$

where

$$v_a = X(X_c - X_b) + Y(Y_c - Y_b)$$

$$v_b = X(X_a - X_b) + Y(Y_a - Y_b)$$

The values for X and Y gives us an accurate position in two-dimension (2D) for the blind node.

Proposed Methodology

While in terrestrial networks, the distance between the blind node and the anchor node is calculated by judging the RSSI(received signal strength indicator) value [15] of the signal that means that the signal's strength when received. While implementing the same aquatic medium model, the problem is that in underwater sensor networks, the radio waves cannot be employed as they undergo heavy attenuation in water. In the place of radio waves, acoustic signals are used. The acoustic signal is low strength signals whose strength cannot be measured as precisely as the radio waves, So we need some alternative method to calculate the distance. In this project, I use the round trip time to estimate the distance from the blind node to the anchor node. The methodology will be that as soon as the blind node receives information about three neighboring anchor node's coordinates, it will run an agent at a specific interval to judge the round trip time and, thereby, the distance value. The scheme proposed here is already popular [16]in the terrestrial wireless sensor networks, but it is not still employed in the underwater sensor networks. This algorithm aims to determine a specific blind node's location within the distributed nodes along the testbed area. If the blind nodes' positions are not known in a network, these monitors and reports can not be located if need be. The primary obstacles to localization in wireless sensor networks are the sparse anchor node problem; hence, this algorithm is structured to solve it. The proposed algorithm is made up of two phases: the initialization phase and the final phase.

A. Initialization Phase Only the anchors will have position data before implementing the positioning algorithm, but all the nodes have identification numbers (IDs). The network is considered for this the algorithm will be scalable to a considerable number of nodes spread over the testbed area. The percentage of anchor nodes will be small. It results in a situation where only a tiny percentage of nodes in the network can establish direct contact with any of the anchor nodes. To overcome this initial information deficiency, the initialization phase is usually initiated at all anchor nodes by making them broadcast their data, which includes their location position and other parameters sensed. The blind node in the direct range of the broadcast will store the anchor locations once for a particular node and estimate the range to anchors based on the estimation of round trip time, after which these also broadcast the anchor locations to other blind nodes. Through this process, all blind nodes will know the location of the anchors and their distance.

B. Final Phase If a blind node can estimate its distance to at least three anchor nodes; then the blind node can perform trilateration to get its accurate location in 2D; this blind node becomes a "converted" anchor node whose position will now be sent to the sink. This process (initialization and final phases) will continue until all blind nodes become converted to anchor nodes.

The following pseudocode explains the procedure in a proper algorithmic manner.

1) When a positioning packet has been broadcast by anchors
2) IF a blind node is within the range of broadcast
3) Then store the positioning packet and compute the broadcast the anchor node position to other blind nodes.
4) Else do nothing
5) IF a blind node receives packets from at least three different anchors

6) Then perform trilateration
7) Else do nothing
8) If the trilateration is successful, the blind node becomes converted anchor node
9) Then Go to 1
10) Else repeat 6
11) End

C. Evaluation of round trip time One of the significant steps in the algorithm mentioned above is estimating the distance between the anchor node and the blind node by estimating the round trip time.

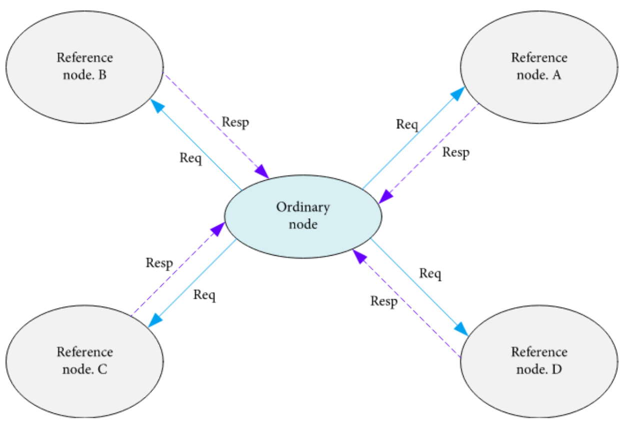

Fig. 4: Representation of estimation of distance using round trip time.

As compared to the symmetric round trip-based algorithm, this procedure consumes less energy because for the localization of k blind nodes, let say there are four anchor nodes in the neighborhood of the blind node, they are A, B, C, and D anchor nodes. To estimate the distance [17] between them, the blind node will send a request and note the time stamp. When it receives the response, it will again note the timestamp, and from both the timestamp, it will estimate the total time taken by the signal in one round trip. As the signal's speed and the time are known, the distance between the nodes can be calculated.

IV. UnetStack Work

This section of the paper presents the implementation of the the proposed mathematical model in one of the industry-standard open-source underwater network simulator UnetStack3 [7].

4.1. Overview of UnetStack UnetStack is an agent-based software stack and simulator used for underwater networks. There are many AUVs deployed with different sensors. These AUVs need to communicate with each other. UnetStack is one such software stack that makes this possible. UnetStack also has a simulator wherein these scenarios can be created and tested before deploying actual nodes in the ocean. UnetStack has different components such as,

- Unet framework: Which provides the core services, messages, Unet API, and Unet Agents for developers to use. These Unet Agents are built upon fjåge agents. These agents are typically written in groovy or java.
- Unet basic stack: The basic stack includes a collection of different essential agents that are needed by almost all the Unets.
- Unet premium stack: These include a collection of unet agents used for providing higher performance and functionality. They have many similar agents compared to the basic ones, but they use optimizations to increase the performance and improve bandwidth efficiency.
- Unet simulator: This is the essential tool in UnetStack. It simulates Unets with many nodes. The simulation can be in real-time or discrete based upon the need of a developer.
- Unet IDE: Unet IDE is provided with UnetStack3 soft- ware bundle. It is helpful for developers to develop, simulate, and test Unet agents and protocols. Unet simulator also has a map feature wherein the developer can see the deployed nodes on a map and interact with his nodes during the simulation.
- Unet Audio: This is a new feature in UnetStack. It uses the sound card of a computer to test Unets based on acoustics. It is a great tool to develop networking proto- cols, test them and develop new acoustic communication techniques.

UnetStack nodes support the mobility of nodes wherein different parameters such as time, duration, speed, dive rate, heading, and turn rate can be used to define the motion of nodes. Using the mobility parameter given in UnetStack nodes, we can create our scenario, i.e., simulate node movement based on ship movement.

4.2 Localization for static blind node To localize a node two agents are developed each to be installed on blind node and anchor nodes espectively.

In order to localize a static blind node following algorithm is followed.

- The blind node will first start broadcast request for getting the address of the neighbour nodes.
- As soon as the anchor nodes in the neighbourhood receives the request they reply with a PDU conaining their address.
- At the blind node, the agent will sore the address in a list.
- As soon as the blind nodes receives the addressess from three neighbour nodes, it queries them for getting their ranges.
- In the range response packet itself we can extract the coordinates by using the *getpeerlocation()* function.
- To avoid the collision *Waker behavior* is employed in querying the ranges. It triggers the anchor nodes to respond at different times.

4.3 Localization for mobile blind node The problem with mobile nodes is that by the time the localization is completed the node might move to a new position. So during the estimation if we know the speed with which the node is moving, we can get the distance covered by it in a given time frame. Adding this distance to the estimated coordinates we can get the actual coordinates of the nodes at that time.The below figure represents the concept.

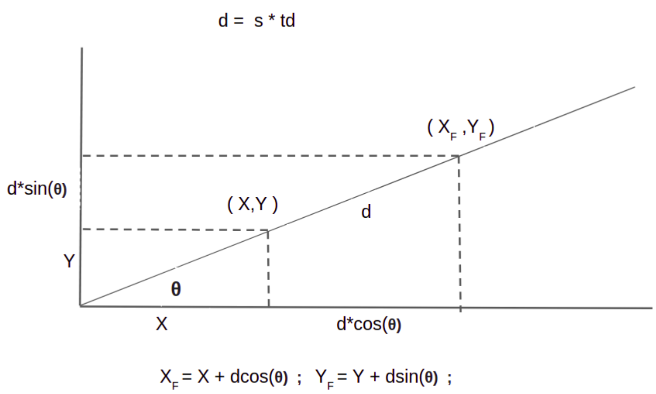

Fig. 5: Localization for mobile blind node

In order to localize a static blind node following algorithm is followed.

- The blind node will start broadcast request for getting the address of the neighbour nodes.
- The current timestamp is saved in a variable 'start'.
- As soon as the anchor nodes in the neighbourhood receives the request they reply with a PDU conaining their address.
- At the blind node, the agent will sore the address in a list.
- As soon as the blind nodes receives the addressess from three neighbour nodes, it queries them for getting their ranges.
- In the range response packet itself we can extract the coordinates by using the *getpeerlocation()* function.
- To avoid the collision *Waker behavior* is employed in querying the ranges. It triggers the anchor nodes to respond at different times.
- After getting the ranges and coordinates information ffrom the anchor nodes, the coordinates of the blind nodes are estimated.
- The estimated coordinates migh be different than the current coordinates because during all this time the blind node has travelled some distance.
- Therefore the estimated coordinates are updated as shown in the Fig.5

4.4 Metrics for Evaluating Algorithm

Following metrics are used to evaluate the performance of the algorithm.

1. Localization Error Localization Error [18] is defined as the distance between the estimated and the actual coordinates of the node. It is computed using the equation below, where

$$(x_e^i, y_e^i)$$

and

$$(x_a^i, y_a^i)$$

are node i's estimated and actual coordinates.

$$Error = \sqrt{(x_e^i - x_a^i)^2 + (y_e^i - y_a^i)^2}$$

2. Total time taken by the algorithm The algorithm's total time to execute should not be very high, as it will lead to low performance. The ideal situation will be that the algorithm completes its execution and updates the coordinates on the map within a reasonable time.

V. Results

5.1. Simulation Setup We simulate a topology of 4 nodes in a sensing area. We assume that 4 nodes are randomly deployed in a 3000m×3000m area. The anchor nodes are static. For mobile blind node I assume that the speed is 1m/s. The interval of reponse to broadcast request varies from 1s to 4s. I set the error threshold as 15m. The mobile node has a direction of 100° measured from north. The motion model for simulation has 3 legs. For the first 2 minutes the blind node will be stationary. Next for 10 minutes it will move in a straight line triggering the localization algorithm after every 2 minutes. At last it will stop after travelling for 10 minutes and stays there forever.

5.2. Observations Following observations were made:

- For the stationary blind node the estimations are very accurate resulting an error of 0.00038 m
- For mobile nodes as we increase the speed of the blind node the error increases linearly.
- It happens because of reception of responses from the anchors at different locations.
- For the speed of 1mps It has an error of 10m on average.
- The total time taken for completing the algorithm is between 30s to 40s.
- If we try to furthur reduce the time, it may leads to collision as it is a unicast communication.

VI. Conclusions and Future work

In this paper we addressed the problem of localization for underwater acoustic sensor networks using intersection of circle method that is quite popular scheme in terrestrial wireless sensor networks. Compared with the approaches used in terrestrial sensor networks, underwater localization consumes more energy. First, we discussed the application of underwater sesing. After that the related work about the present localization techniques is presented. A novel approach based on mathematical model is presented next and based on the mathematical model,we proposed solutions for static and mobile blind nodes.The simulation is performed on UnetStack simulater.Finally,we analyze the simulation results and find out that our algorithm can provide accurate localization for sensor nodes. As we have seen that the algorithm gives very accurate results for the case of static blind node.But for the mobile blind node the error increases propotional to the speed of the node. In future , attempts can be made to make the algorithm more accurate with increasing speed of the blind node.This can be done by taking the difference in estimation as a function of speed of the blind node.

References

[1] T. Rahman, X. Yao, G. Tao, Consistent data collection and assortment in the progression of continuous objects in iot (2018).

[2] E. Shakshuki, A. A. Elkhail, I. Nemer, M. Adam, T. Sheltami, Comparative study on range free localization algorithms, the 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops (2019). doi:https://doi.org/10.1016/j.procs.2019.04.068. URL http://www.sciencedirect.com/science/article/pii/S1877050919305307

[3] J. Hightower, R. Want, G. Borriello, Spoton: An indoor 3d location sensing technology based on rf signal strength (2000).

[4] Y. Zhou, K. Chen, J. He, J. Chen, A. Liang, A hierarchical localization scheme for large scale underwater wireless sensor networks (2009).

[5] G. Han, J. Jiang, L. Shu, M. Guizani, An attack-resistant trust model based on multidimensional trust metrics in underwater acoustic sensor network (2015).

[6] M. T. Isik, O. B. Akan, A three dimensional localization algorithm for underwater acoustic sensor networks (2009).

[7] R. P. Vio, R. Cristi, K. B. Smith, Uuv localization using acoustic communications, networking, and a priori knowledge of the ocean current (2017).

[8] J. H. Tay, V. R. Chandrasekhar, W. K. G. Seah, Selective iterative multilateration for hop count-based localization in wireless sensor networks (2006).

[9] A. Alcocer, P. Oliveira, A. Pascoal, Underwater acoustic positioning systems based on buoys with gps (2006).

[10] A. Amar, Y. Buchris, M. Stojanovic, Angle-of-arrival-based detection of underwater acoustic ofdm signals (2015). doi:10.1109/SPAWC.2015.7227053.

[11] S. Yang, J. Yi, H. Cha, Hcrl: A hop-count-ratio based localization in wireless sensor networks (2007).

[12] N. Lasla, M. F. Younis, A. Ouadjaout, N. Badache, An effective area-based localization algorithm for wireless networks (2014).

[13] H. Song, V. Shin, M. Jeon, Mobile node localization using fusion prediction-based interacting multiple model in cricket sensor network, IEEE Transactions on Industrial Electronics 59 (11) (2011) 4349–4359.

[14] D. G. Schaeffer, A mathematical model for localization in granular flow (1992).

[15] K. Benkic, M. Malajner, P. Planinsic, Z. Cucej, Using rssi value for distance estimation in wireless sensor networks based on zigbee (2008).

[16] J. Xu, W. Liu, F. Lang, Y. Zhang, C. Wang, Distance measurement model based on rssi in wsn (2010).

[17] P. Karn, C. Partridge, Improving round-trip time estimates in reliable transport protocols, Vol. 17, ACM New York, NY, USA, 1987, pp. 2–7.

[18] H. Aksu, D. Aksoy, I. Korpeoglu, A study of localization metrics: Evaluation of position errors in wireless sensor networks (2011).