Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, No. 2, April : 2024

# AN IMPROVED DV-HOP PROPAGATION BASED LOCALIZATION APPROACH IN WIRELESS SENSOR NETWORKS

**K Md Saifuddin** School of Electronics and Communication Engineering Reva University Bangalore, India Email: saifu426@gmail.com
**Geetha D Devanagavi** School of Computing and Information Technology Reva University Bangalore, India Email: dgeetha@reva.edu.in

## ABSTRACT
Sensors have a strong connection to the real world, which sets them apart from conventional networks in a major way. Through the use of a variety of sensing devices as well as the processing of raw data, sensor networks may detect & monitor physical phenomena occurring in the locations/regions where even the sensors are located or deployed. Physical phenomena linked with geographic locations/regions are much more important to users of wireless sensor networks over raw data from individual sensor nodes. Geospatial information is becoming more important in sensor networks and applications, and sensor nodes having GPS signal receivers become more widely available. Sensor nodes may also be located without GPS using a variety of localization methods. A location-aware sensor network is now possible thanks to these recent technological advancements. To address the issue of estimating the location and position among wireless sensor nodes, new approaches, techniques, and algorithms must be created. In order to achieve this goal, In this Paper, we are proposing a Distance Vector Hop (DV-Hop) algorithm that Significantly decreases the average of localization error of sensor nodes and which is a solution for localizing nodes having few adjacent anchors. simulation results prove that the proposed algorithm substantially reduces the average of localization error of sensor nodes

**Keywords:**
Localization, Sensor Node, DV-Propagation Method, Wireless Sensor Networks

## INTRODUCTION
Location-aware wireless sensor networks will be focused in this paper. These problems will be addressed in detail: • Localization Technology. Wireless sensor network applications need precise location information. Sensor readings may be retrieved using spatial searches depending on their location. Movement trajectory as well as other moving states is shown by changes throughout object position over time using object tracking applications (i.e., moving speed & moving direction). Wireless sensor networks will indeed be exposed to a range of localization methods (with tradeoffs that cost, complexity, & operational fidelity).
Geographical Routing Protocols. Using position information from sensor nodes, relay nodes, & route destinations, geographical routing systems determine routing decisions. Many research projects have used geo-routing protocols to assist higherlayer information management protocols because of their simplicity & efficiency. Geo-routing protocols that are stateless and connection quality aware would be offered in addition to the conventional stateful protocols [7].
Spatial Queries. A large-scale & dispersed wireless sensor network may be queried using spatial queries (e.g., window queries, k closest neighbor searches, and contour queries). To accomplish efficient, effective, and trustworthy data distribution and collecting, a variety of query processing algorithms have been suggested. To make wireless sensor networks increasingly energy efficient, in-network processing is indeed a key data reduction method [11] The existing algorithms [10] are depending on multiple design ideas, as well as basic query processing methods. Location Tracking of Moving Objects. Moving objects may be tracked using wireless sensor networks. Objects passing through these applications are monitored and tracked by sensor nodes working together. Tracking applications require localized operations, tracking handoff, etc., as well as limited battery power,

computation and storage capacity upon board. Wandering objects inside sensor network settings must be monitored in a way that it maximizes energy efficiency, network performance as well as operational accuracy. This development of wireless sensor networks faces a number of difficulties at different levels and phases. Creating a wireless sensor node that is very small and performs most of its functions, for instance, has many challenges. In the very same vein, new algorithms [1] and protocols must be created from the connection layer to the application layer. Such small nodes will need new operating systems, which will require new programming languages and paradigms.

In some of the wireless sensor network applications [9] It is important that sensor nodes remain aware of their relative position inside the network. To really be meaningful and valuable, sensor node data like temperature, humidity, and pressures must be attributed towards the relative location from which it was acquired. These sensor nodes must've been aware of their locations in order to make that happen. As just a result, the issue of locating or estimating the position between sensor nodes is very often referred to as localization. When it comes to robotics, the phrase "localization" refers to determining the location of a mobile robot inside a particular coordinate system. Nodes may be aware of the importance and direction but rather orientation in respect to either the network or themselves, depending on circumstances. Consider, for instance, the classification of nodes in such a sensor network:

**Dumb Node (D)** It really is the node itself that doesn't understand its true position, not the network, but will ultimately find out from the result of both the localization method under study whether it's really there. Free or unknown nodes were often known as dumb nodes.

**Settled Node (S)** In the localization methods, a previously dumb node will be treated as a settled node.

**Beacon Node (B)** This node knows this location from the very beginning, and it continues to know it even after any localization method has been used to locate it. This device uses a different method to determine its location than a standard localization algorithm. Alternatively, it may well be equipped with either a GPS device or placed at a recognized coordinate position, whichever is more convenient. Other names with beacon nodes were reference nodes, anchor nodes, as well as landmark nodes. This should be noted that sensor nodes could have symmetric or asymmetric communication links. A symmetric pair of nodes may reach of another, whereas a non-symmetric pair of nodes cannot. U or v can reach one another in the event of asymmetric communication connections, not both at the same time. Examining a square-shaped, symmetric, twodimensional sensor network. The collection of sensor nodes may be understood as a sequence of vertices in the graph G (V, E) shown below.

$$V = \{v1, v2, \ldots, vn\} \quad (1)$$

The graph G (V, E) consumes a set of edges E that includes all the edges e = (i, j) Eiff , vi extents vj i.e., is the greatest difference amid the two nodes beyond which direct communication between them is impossible. A node and its neighbor are not neighbor nodes if their distance exceeds the value r. This weight w(e) r of the edge e = i j) connecting two neighbor nodes vi and vj is used to calculate the distance connecting them.

Notably, the issue of localization was typically only addressed in two dimensions, with both the assumption that it may be expanded to three dimensions if required or deployed. We have thus described graph G (V, E) as two-dimensional. As illustrated in Figure 1, G appears to be a Euclidean network, with every sensor node having a coordinate (xi, yi). i's location either in that a sensor field is indicated by its coordinates (xi, yi).

The following is a new formulation of the sensor node localization problem: A sensor network with many hops is represented either by graph G = (V, E). Graph with known locations of beacon nodes B (xb, yb) If feasible, locate many more dumb nodes as possible (xd, yd) for the localization issue to be solved. Every latitude, longitude, and altitude of such a node must be determined. In such a sensor network, the problem regarding node localization and placement may be addressed if another node has a GPS device. For sensor networks, it was not a possibility because of a variety of factors. There are no energy-saving or energy-aware GPS receivers, nor are there any energy-aware GPS protocols. Because of the scarcity of energy resources, it is possible to install sensor networks for several years without having to replace any batteries. Wireless sensor networks cannot solve the localization issue using GPS devices. Although beacon nodes make only a small percentage of the overall number of nodes, it's indeed conceivable that they are outfitted integrated GPS devices, allowing them to serve because reference nodes again for localization process. It is considerably more costly to purchase GPS gadgets. Sensor network solutions may become impractical if these components are introduced to every sensor node sometimes in way. Sensor nodes must have a very tiny size. GPS systems rely on satellites to operate, which increase the size for sensor nodes, therefore contradict one of several main criteria of a sensor node. GPS stops working when there's no satellite connection. Indoor applications and Mars exploration are examples of situations where this is true. A proportion of nodes serve provide reference nodes for other nodes; thus, GPS devices are often utilized in a small number of nodes. Beacon nodes are yet another name for these kinds of nodes. By placing a few nodes at established locations, their positions may be known allowing them to act as beacon nodes, rather than using GPS. By utilizing the radio frequency (RF) capabilities from both sensor nodes, every sensor field can then utilize a localization technique to determine its position. There are two factors that affect dumb nodes position-finding ability transmission range between beacons and density of beacons. Wireless sensor networks may be localized using a variety of methods.

**DV-HOP PROPAGATION METHOD**

This really is the APS algorithm in its simplest version. (Xi, Yi) seems to be the location of both the beacon node i while $h_i$ is indeed the number of nodes between the sensor field node that maintains that table and beacon node i. A message has been sent at each hop, from the beacon node to its near neighbors, and many more., a message's count field may be incremented. Even the beacon i maintains this table. A beacon node i could compute the average hop size after obtaining the locations and hop counts for every of the additional beacon nodes j.: for all beacon nodes *j* where $i \neq j$.

$$c_i = \frac{\sum\sqrt{\left(X_i - X_j\right)^2 + \left(Y_i - Y_j\right)^2}}{\sum h_i}$$

Mostly in network, that average hop size $c_i$, calculated either by beacon node i is transmitted via controlled flooding, as described previously. After memorizing the locations of beacon nodes and the correction factor $c_i$, a dumb node may estimate one's own position via multiple-lateration. That’s an overview of the APS DV-Hop algorithm's many phases of development. beacon nodes were regarded the algorithm's first beginning point since they broadcast their position information among their nearby neighbors. As with the distance vector method, all other nodes use the same approach to receive and transmit location information from beacon nodes.

To establish their ultimate location, all nodes include the locations of any and all beacon network nodes of hops to any of these beacon nodes. Following receipt of several other beacon nodes' positions and hop counts, a beacon node may calculate the average hop length. Using controlled flooding, every other node receives the average hop duration as both a correction factor. That number of hops between all dumb nodes and the beacon nodes is already known. After obtaining the correction factor, any dumb node could convert its hops into characteristic length using the conversion factor. Eventually, every node location is computed via multilateration.

Each beacon node's correction factor would be different. This means that the correction factors for each dumb node in a network will differ. Instead, they should use the first adjustment factor those who get to estimate where they are and toss out any others. As a result of this rule, a dumb node will, in most circumstances, use the correction factor received from its closest beacon node. Additionally, unless the network is big, a TTL field may be included to packets used only to distribute correction factors to minimize signaling and congestion. Throughout this way, a beacon node's correction factor is utilized by dumb nodes within its immediate vicinity. TTL field-based correction factor propagation limitation complements the very first correction factor strategy.

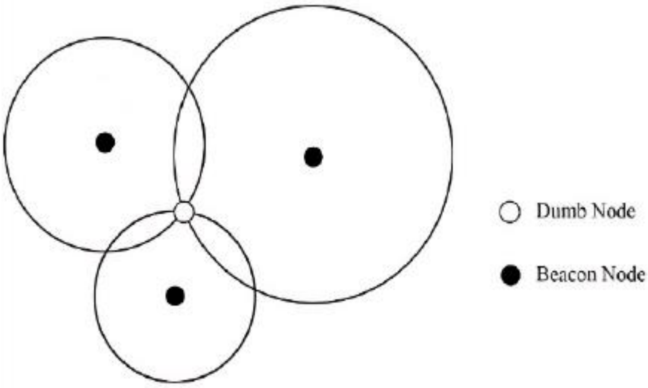



Figure 1. Trilateration – Intersection of three circles around three beacon nodes gives position of the dumb node.

With the assistance of such an example, the correction factor inside the DV-Hop propagation technique may be used to calculate distance and estimate location. Let's look at Figure1, which shows a section of sensor network with 3 beacon nodes, B1, B2 and B3, and six dumb nodes.

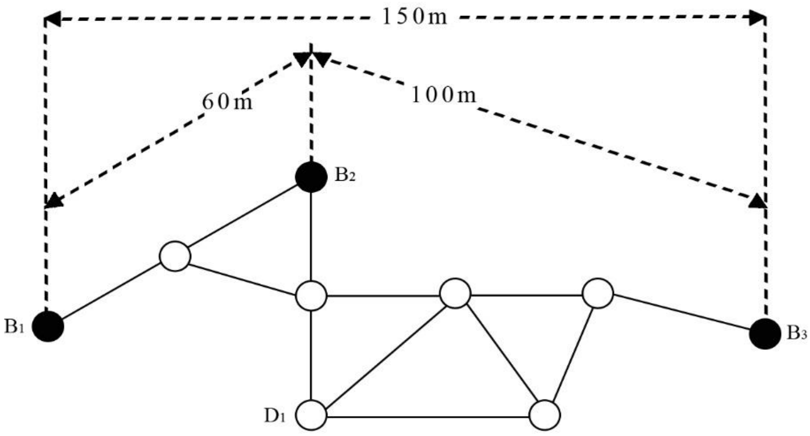



Figure 2: Node localization using DV-Hop propagation method.

Figure 2: Each beacon node knows its Euclidean distance to the others. A1 and A2 have calculated the correction factors.:

$$C_{B1} = \frac{60+150}{2+5} = 25.714 \text{ m}$$

$$C_{B2} = \frac{60+100}{2+4} = 26.667 \text{ m}$$

$$C_{B3} = \frac{150+100}{5+4} = 27.778 \text{ m}$$

Here From Figure 1 the $C_{B1}$is the Correction factor related to beacon node B1 and $C_{B1}$is the ratio of sum of the Distance from B1 to B2 which is 60m, Distance from B2 to B3 which is 150m and the sum of Euclidean distances of Beacon node B1 which is 2m, Euclidean distances of Beacon node B2 which is 5m and after calculation the Correction factor of Beacon node B1 is 25.714m. similarly $C_{B2}$is the Correction factor related to beacon node B2 and $C_{B2}$is the ratio of sum of the Distance from B1 to B2 which is 60m, Distance from B2 to B3 which is the shortest path is 100m and the sum of Euclidean distances of Beacon node B1 which is 2m,Euclidean distances of Beacon node B3 which is 4m and after calculation the Correction factor of Beacon node B2 is 26.667m. similarly$C_{B3}$is the Correction

factor related to beacon node B3 and $C_{B3}$is the ratio of sum of the Distance from B2 to B3 which is 150m, Distance from B2 to B3 which is the shortest path is 100m and the sum of Euclidean distances of Beacon node B2 which is 5m,Euclidean distances of Beacon node B3 which is 4m and after calculation the Correction factor of Beacon node B2 is 27.778m.
Now let's look at dumb node D1 & find out how far it thinks the 3 beacon nodes to have been from its location. D1 was there are 3 hop counts between beacon node B1 as well as beacon node B2, and 3 hop counts between beacon node B3 and beacon node B4. In Figure 2 Due to its proximity to the nearest node, B2 would receive the first correction factor, which is 26.667 m, from that node.
Therefore, dumb node D1 can now estimate distances from any of the 3 beacon nodes about with 100 kilometers:
Distance of D1 from B1 = 26.667 x 3 = 80 m
Distance of D1 from B2 = 26.667 x 2 = 53.334 m
Distance of D1 from B3 = 26.667 x 3 = 80 m
Every one of the 3 beacon nodes positions is already known to D1. It may now trilaterate its own location and use the distances to all these beacon nodes as additional information. Other benefits include reduced processing time and cost due to the absence of extra equipment on sensor nodes. Measurement errors are indeed minimized due to the lack of a measurement methodology used for range estimation inside the DV-Hop method. In particular, it only works well in networks with consistent hop lengths across all segments. If the network was anisotropic, node density & hop length don't always match the correction factor computed either by beacon node, resulting in unsatisfactory results

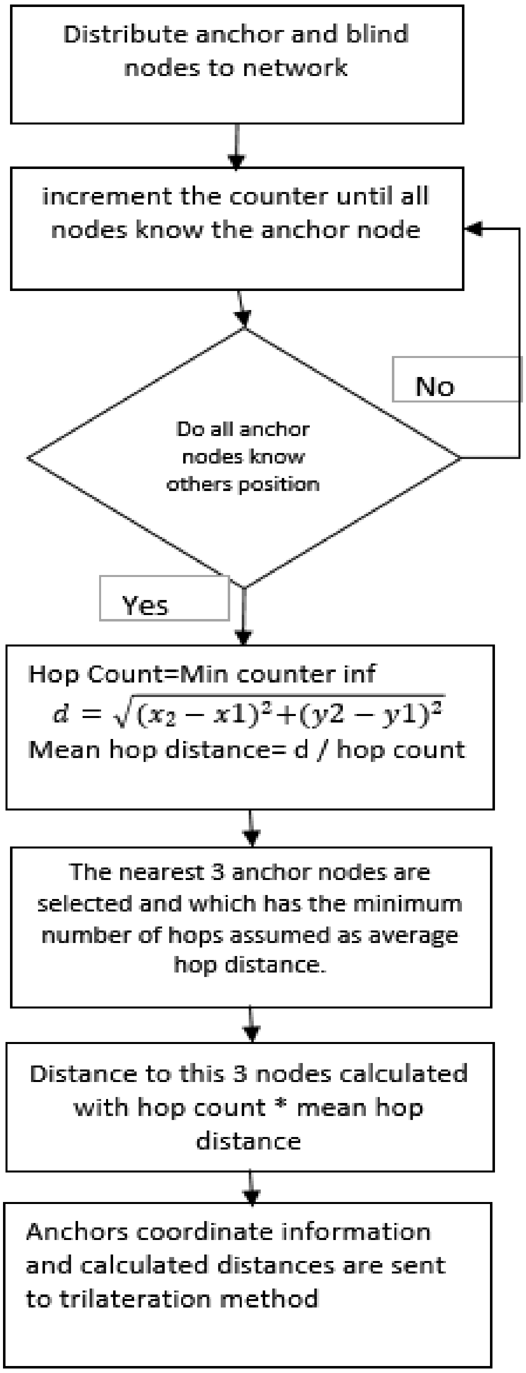
## CHARACTERISTICS OF LOCALIZATION ALGORITHM

It is the primary goal of a localization algorithm to identify a node's location. In order to be practical, the algorithm must satisfy certain requirements. The localization algorithm's requirements are typically determined by the kind of application in which it is developed. The following are the general design goals or desirable features of even an ideal localization algorithm:

- RF-based localization techniques are highly desired. An RF transmitter is included inside the sensor nodes. Throughout addition to its main function of data transmission, an effective localization method makes use of this radio capacity for localization.
- Wireless sensor networks are ad-hoc in nature. The ad hoc character of both the network should be taken into account in the localization method.
- As a result of this, the localization method should have a short reaction time. Sensor nodes may be installed rapidly as a result.
- When to use such an algorithm to locate a sensor node, the location should be precise enough for all the application in which it has been utilized.
- Under order to function in unfavorable circumstances, the algorithm must be resilient.
- Even when sensor items are deployed or removed, the position of sensor nodes should be able to be determined. It will also be able to even provide satisfactory results across sensor networks having moderate to large number of neurons.
- As well as being energy-efficient, this technique must also be energy-conscious since the sensor nodes are often self-contained and doesn't have an external power supply.
- As the number the beacon nodes changes, the localization algorithm should adjust. This algorithm was able to give position estimations even when the amount of possible beacon nodes varies. Overall accuracy of node estimations, on the other hand, will vary as the number active beacon nodes increases. In general, a localization method can calculate more accurate estimates about node locations when there are more beacon nodes.

- So that it can calculate node positions under any circumstances of changing surroundings and weather, the method should really be universal. This should be able to operate in both confined and uncontrolled settings, including inside and outdoors.

**BLOCK DIAGRAM**




**SIMULATION AND RESULTS**

A localization algorithm should be accurate following the requirement of various location-aware applications such as target tracking, rescue, and fire relief. extent of matching between the position estimated by the localization algorithm and the real positions defined the term of localization accuracy. As the estimate position is close to the real position, the localization algorithm is as good as in [1], the requirement on the resolution of the positional accuracy depends on applications. In this section, we discuss the performance of our DV-Hop algorithm.

We Undertake that all nodes in the network are unchanged, and the location of the anchor node has no deviation. Anchors are provided with a GPS and thus they are aware of their positions, while the other sensor nodes are not localized. evaluation of the proposed localization algorithm is performed through simulation using MATLAB R2015a.

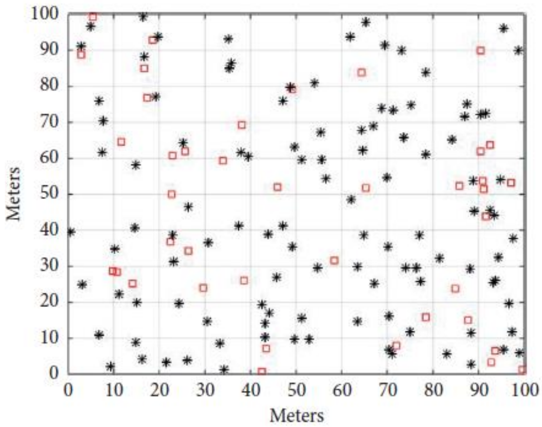



Figure 2: Network model

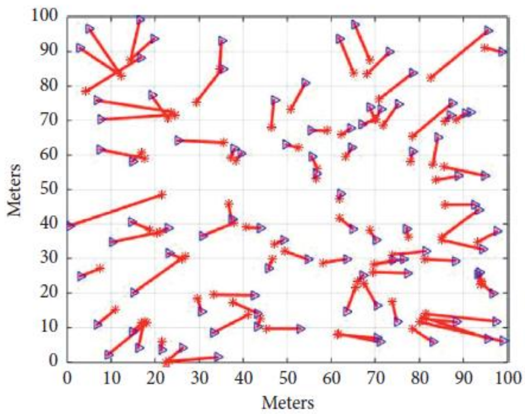



Figure 3: Localization results of DV-Hop algorithm.

***Localization Result:*** Figures 2 and 3 show the final evaluation results of the sensor nodes by the Distance Vector-Hop algorithm respectively. In the two figures, symbol "∗" Point out's the actual location of the unknown node. The obtained estimated location by using anchor nodes is shown by the blue symbol "△". the true node location and its evaluation are connected by the red direct line which shows the estimation error.

## REFERENCES

[1] Gaurav Soni, Mahendra Kumar Jhariya, Kamlesh Chandravanshi, Deepak Tomar "A Multipath Location based Hybrid DMR Protocol in MANET" 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE-2020), 07-08 February 2020, (IEEE Conference Record # 48199)

[2] Waheb A. Jabbar, Wasan Kadhim Saad, and Mahamod Ismail,"MEQSA-OLSRv2A Multicriteria-based Hybrid Multipath Protocol for Energy-Efficient and QoS-Aware Data Routing in MANET-WSN Convergence Scenarios of IoT", Vol. 6, Pp. 76546-76572, IEEE Access 2018.

[3] Swetha M. S, Thungamani M, “A Novel Approach to Secure Mysterious Location Based Routing For Manet”,International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-8 Issue-7 May, 2019
[4] Sibashrit Pattnaik, Arunanshu Mahapatro “Energy Aware Mobility Prediction in Wireless Sensor Networks” 2015 International Conference on Technologies for Sustainable Development (ICTSD-2015), Feb. 04–06, 2015, Mumbai, India
[5] Pan, Shengli, Yu Tang, and Guangmin Hu,Identification of Multipath Routing Based on End-to-End Packet Order, Global Communications Conference (GLOBECOM), 2016 IEEE.
[6] Ruchika Das, N. Rakesh “OMLP: An Efficient Opportunistic Multi-path Routing using Location and Power aware method in Wireless Network.” 2017 International Conference On Smart Technology for Smart Nation 978-1-5386-0569-1$31.00_c 2017 IEEE
[7] Nisha Arora, Ajay Jangra, “GLAAR: Geographic Location Aware Adaptive Routing in Mobile Ad Hoc Networks (MANETs)”, International Journal of Computer Applications, ISSN: 0975 – 8887, Volume 50– No.22, July 2012.
[8] Akyildiz, I. F., & Vuran, M. C. (2010). Wireless Sensor Networks. USA, John Wiley & Sons. [15] Mao, G., & Fidan, B. (Ed.). (2009). Localization Algorithms and Strategies for Wireless Sensor Networks. New York, Information Science Reference: IGI Global.
[9] Dargie, W., & Poellabauer, C. (2010). Fundamentals of Wireless Sensor Networks: Theory and Practice. John Wiley & Sons, Ltd.
[10] R. Niu, R. S. Blum, P. K. Varshney, and A. L. Drozd, “Target localization and tracking in noncoherent multiple-input multipleoutput radar systems,” IEEE Trans. Aerosp. Electron. Syst., vol. 48, no. 2,pp. 1466–1489, Apr. 2012.
[11] M. Z. Win, “Network localization and navigation via cooperation,” IEEE Commun. Mag., vol. 49, no. 5, pp. 56–62, May 2011