Advances in Computational Sciences and Technology
ISSN 0973-6107 Volume 10, Number 4 (2017) pp. 481-500

http://www.ripublication.com

A Hop Based Distance Estimation Approach for Localization of Nodes in Tree Structured Wireless Adhoc Sensor Networks

Arun Saxena$^{\#1}$ and Manuj Darbari^{*2}

#Research Scholar, Sainath University, Jharkhand, India;
Assistant Professor, Amity Institute of Information Technology,
Amity University, Mumbai, Maharashtra, India

**Associate Professor, Department of Computer Science,*
BBD University, Lucknow, India

Abstract

Proliferation of wireless sensor network applications has led towards identification of a large number of techniques which can be used to locate the position of nodes in adhoc wireless sensor networks. Localization in wireless sensor networks pertains to computation of position of nodes with respect to some coordinate system. Localization is needed for reporting the origin of events, assisting group querying of sensors, routing and answering questions on the network coverage. Even collaboration depends upon node localization so that several other functions like signal processing, communication, time synchronization, sensors repositioning, energy savings, calibration, heuristics etc. could be accomplished. In most of the applications, the collected data by the network without the information of the location is not useful. There are a set of constraints related to localization which include: small size and limited power of sensor nodes, unreliable communication between nodes and limited communication between nodes in a close vicinity defined by the range of communication media. A range free, energy efficient technique for localizing nodes in a tree based structure has been suggested, in this paper. The proposed method utilizes two way message exchange distance estimation process for calculating the distance between anchors and the root node. This distance estimation is further utilized for localizing all nodes in the tree based structure.

Keywords: wireless sensor networks; localization; hop based; range free.

I. INTRODUCTION

A wide variety of applications make use of wireless sensor networks to collect and manipulate sensitive data. These include target tracking, predictive maintenance, healthcare, environment monitoring and smart-homes [1-5]. WSNs are deployed in an adhoc manner and do not have any infrastructure requirements. Consequently, one of the major issues associated with such networks is identification of accurate position of nodes in the network. Localization in wireless sensor networks pertains to computation of position of nodes with respect to some coordinate system.

The use of GPS receivers is one of the simplest ways of finding the accurate location of a node. However, it is not a feasible solution as it is expensive, requires high power consumption and is incompatible with small size of sensor nodes. Moreover, indoor and outdoor deployment issues can adversely affect GPS reception. Due to these limitations of GPS, a large number of methods to identify the location of nodes in WSNs have been suggested. The localization techniques in wireless sensor networks can be mainly classified as: centralized or distributed, range based or range free, anchor based or anchor free and fine grained or coarse grained. In centralized techniques, all calculations related to node localization are computed at a central place and then the results are sent to all the nodes in the network. On the contrary, in techniques which are distributed in nature, all the computations required to find the location of a node is done by the node itself. In anchor based approaches, position of few anchor nodes are known and these positions are used to find the locations of unknown nodes, whereas, in anchor free approaches, only relative position of unknown nodes is estimated. Range based approaches estimate either the distance or the angle between two nodes in order to find the location of sensor and require special hardware for the same, whereas, range free methods estimate the location of nodes using connectivity information. Fine grained approaches make use of received signal strength features for position estimation contrary to the coarse grained approaches [7-11].

Most of the localization algorithms are application specific and each has its advantages and limitations. Range free localization methods utilize network topology information for node localization. Although, range based approaches may be more accurate but range free approaches are preferred over them in many applications as they are economical and require no extra hardware [12]. The energy consumption is either same or better than the most of the other range based methods [10]. In this paper, a hop based distance estimation method for tree based networks is proposed. The method is suitable for tree based networks and is more energy efficient as compared to other methods which are based on DV-Hop localization method.

The organization of the remaining paper is as follows. The related work done in this area and the comprehensive explanation of the proposed system are covered under section II and section III, respectively. Section IV demonstrates working of the system with the help of an example followed by conclusion.

II. RELATED WORK

Due to extensive research conducted in the area pertaining to localization of nodes in wireless sensor networks, the amount of literature available is abundant. The categorization of the proposed work can be done under range free localization scheme. Consequently, literature review covers an overview of range free approaches. The popular range free distance measurement techniques include distance vector (DV) hop, hop terrain, centroid localization algorithms, approximate point in triangulation (APIT) and gradient algorithm [9]. In DV hop, firstly, each anchor node calculates the average hop distance from other two anchors and broadcasts it. The average hop distance is received by an unknown node from its nearest anchor. Using this distance and its hop distance from all the anchors, it calculates its distance from each one of them. Lastly, it estimates its coordinates through triangulation [12]. Hop terrain uses a similar approach as adopted by DV hop for initial distance estimation. After attaining an initial location on the basis of this initial distance estimation, it refines the estimated location by interchanging information amongst its neighbors in order to ensure location accuracy [12].In centroid localization algorithms, the location information of the anchors is broadcasted to the unknown nodes. An unknown node in the vicinity of anchors is the centroid of the polygon formed by several anchors [14-16]. In APIT, upon receiving beacon message from anchor node, the unknown node continuously performs a point in triangulation (PIT) in order to identify whether the anchors form a triangle or not. Once the triangle is formed, its center of gravity is calculated for estimating the position of the unknown node [17]. In gradient algorithm, the unknown node calculates the shortest path distance between itself and the anchor node, after receiving the beacon message from the anchor. Subsequently, this unknown node uses error estimation and multilateration to determine its location [18].

DV hop algorithm estimates the coordinates of an unknown node in three phases. In the first phase, each anchor broadcasts its coordinates and hop distance is initialized to 1. At the end of each phase, each node has the minimum hop count to all anchor nodes. Secondly, each anchor calculates its average hop distance from the other two and broadcasts this information. Using the information in first two phases, each unknown node estimates its coordinates using triangulation. Several improvements have been suggested in one or more of these phases of DV hop algorithm to improve its accuracy**.**

Experimental and simulation results have shown that localization accuracy can be improved in three ways. Firstly, by adopting a different average hop calculation method in phase 2, secondly, by using a different coordinate estimation method in phase 3 and lastly, by changing the methods adopted in both phase 2 and phase 3 [19, 20, 23, 26, 27, 30, 33]. However, their energy efficiency is same as classical DV-Hop algorithm as they also use the same flooding process in phase 1 and phase 2, in order to distribute information related to anchor coordinates in phase 1 and average hop size calculated by each anchor node in phase 2. Localization errors can also be reduced by deploying extra regulatory nodes in the network. These nodes calculate an error vector and broadcast hop distance and error vector to neighboring nodes using which they correct estimated coordinate location [21]. Another approach has been proposed to reduce errors by calculating the coordinates of the unknown nodes using centroid formula. This involves three additional steps [24]. By using an optimal number of anchor nodes in the sensing area, the accuracy can be improved. The authors suggest that initially each unknown node should calculate a threshold value say N. Subsequently, information collected from only N anchor nodes is used by each unknown node to estimate its location. Using more than N nodes have little effect on localization accuracy [22]. Several other approaches utilizing particle swarm optimization and mobile beacon node have been suggested in order to reduce localization error [23], [28-32], [34]. Some improved DV-Hop algorithms have also reduced the energy consumption by eliminating the need of broadcasting average hop size by anchors. Thus, network is flooded only once when each anchor broadcasts its coordinates, initializing hop count value to 1.

A close examination of these protocols reveals that energy consumption of APIT and gradient algorithm is low but their localization accuracy is also low [7]. Accuracy of both DV hop and centroid localization algorithms is average but they consume high energy. The reason being, in centroid localization algorithms, a large number of computations are involved in order to locate the position of the unknown node. The anchor nodes in DV hop algorithm flood the network multiple times leading to its high energy consumption and is suitable only for isotropic networks. The improvements in the DV-Hop are focused more on improving accuracy while energy consumption of most of them is same as the original DV-Hop algorithm. This is because they flood the network twice, firstly, to provide the coordinates of anchor nodes and secondly to provide average hop distance information to all nodes [37].

III. PROPOSED WORK

The proposed work is based on the assumption that the sensor network is organized as a hierarchical structure suggested in time synchronization algorithms, TPSN [36], TSRT [35] and TSTSP [37]. This has two implications. Firstly, there is only one node at level 0 which is referred to as root node of the tree. Root node selection process depends on application requirements. Thus, root node may be randomly selected or a node with special characteristics may be designated as the root node. Secondly, node id of its parent is known to each anchor node.

The proposed work utilizes the mechanism adopted for synchronizing the clocks in wireless sensor networks [35-37]. In this mechanism, root and child nodes in a tree synchronize themselves using two way message exchanges. The root node initiates these message interactions to calculate the propagation delay and clock drift between itself and its child nodes. Once these values are calculated, they are communicated to each child node so that their clocks can be synchronized. A brief summary of basic pair wise synchronization process adopted by these protocols is given as follows.

The process is started by a sender node, say X, which sends a message containing its current timestamp T1 to node Y. Node Y receives the message at time T2 and replies with a message containing its current timestamp T3 and the time it had received the message from X i.e. its own T2. The sender node receives the message at time T4. At time T4, node X contains timestamps from T1 to T4 using which it calculates the propagation delay ($\text{Delay}_{\text{current\_level}}$) and clock drift (Clock_Drift) using the equations (1) and (2), respectively.

Node X sends these values to node Y using which node Y calculates its clock offset and corrects its clock.

$$\textbf{Delay}_{\textbf{current\_level}} = ((\textbf{T2} - \textbf{T1}) + (\textbf{T4} - \textbf{T3})) / 2 \qquad (1)$$

$$\textbf{Clock\_Drift} = ((\textbf{T2} - \textbf{T1}) + (\textbf{T4} - \textbf{T3})) / 2 \qquad (2)$$

This process of calculating propagation delay using two way message exchanges, which is used for synchronizing the clocks, can also be adopted to estimate the distance between two nodes in wireless sensor networks. Signal propagation delay between two nodes can be calculated by using (1). Using this, distance between two nodes can be calculated as follows:

$$\textbf{Distance}_{\textbf{current\_level}} = \textbf{v} * \textbf{Delay}_{\textbf{current\_level}} \qquad (3)$$

Here v is the signal propagation speed that is the speed of light in case of radio signals. This method of distance estimation would be referred as Message Exchange Distance Estimation Process (MEDEP), hereafter.

The entire localization process consists of two phases. In the first phase, an anchor location table is created. This table contains distances of all nodes from the root node. Using this table, root node estimates its location. In the second phase, each unknown node performs MEDEP with its parent node in order to estimate its distance from parent. During MEDEP, the parent node supplies its anchor table to the communicating node. When the MEDEP is complete, the latter updates the distances of all the anchors with itself in the table. Finally, it can estimate its location using the updated table.

Initially, each anchor node starts MEDEP by unicasting ANCHOR message to its parent node at time T1. The parent node receives the message at time T2 and replies with the timestamp it had received the message i.e. T2 and its current timestamp corresponding to the time of its reply i.e. T3. The anchor node receives the message at time T4. At time T4, anchor calculates the propagation delay ($Delay_{current\_level}$) and distance ($Distance_{current\_level}$) using (1) and (3). The anchor node constructs a table and sends it to its parent node.

This parent now sends an Anchor_Found message to its parent node at time T1 and starts MEDEP. Using message exchanges with its parent along with (1) and (3), it calculates the distance delay $Distance_{current\_level}$ between itself and the parent. After the MEDEP is over, this node updates the distances in the anchor table which it had received from its child anchor node and sends it to its parent node. This process of distance calculation and updation of anchor location table continues till table reaches the root node. A similar process for anchor table creation is adopted for other two anchors in the network. When the root node receives three tables from three different sources, it merges them into single anchor location table. The root node has distances from three different anchors, so it can estimate its location on the basis of these distances using triangulation or any other technique. This anchor table creation process and distance estimation of root can be done either during tree construction phase [37, 38] or at a later time depending on the application.

Any child node can estimate its distance with the root node by performing MEDEP with it, after the latter has constructed single anchor table and estimated its location. However, the content of message exchanges between each parent root node and its child nodes is different.

The interaction between root and its immediate child node starts when an immediate child node sends LOC_BEGIN message to root at time T1 for estimating its position. After receiving the message at time T2, the root replies by sending its current timestamp i.e. time of reply T3, the time it had received the message from child node i.e. its own T2 and the anchor location table. Upon receiving the message at time T4, the child node can estimate its distance from its parent using equations (1) and (3). Then using this distance, it updates its anchor location table to find distances from all anchors so that it can estimate its location. All the child nodes at level 1 can estimate their coordinates using this procedure.

All the child nodes at level 1 are also the root nodes for the nodes which are at the next level. Consequently, all the nodes in the next level can estimate their coordinates using the same procedure that was adapted by their respective parent nodes. This process is repeated for all the nodes at all the levels in the tree.

When a child node receives the anchor table there can be two possibilities. First possibility is that an anchor exists in the subtree of the node which is estimating its location. In this case, the child node will already have an anchor table which it had earlier supplied to its parent node. This child node compares child id field of its anchor table with the child id field of all rows in the table sent by root node. If it finds a row in which values of both fields is same, it subtracts the distance that was calculated using MEDEP with its parent node with hop distance from anchor field. For all the other rows where the values of both fields are different, it adds this distance to hop distance from anchor field in the table sent by the root node. Once the anchor table is updated, this child node has distances from all the anchors.

Second possibility is contrary to the previous one. In this case, the child node performs MEDEP with its parent to calculate the distance between them. It then adds this distance with the distance from anchor field in all rows of the table which it had received from its parent node during MEDEP.

IV. DEMONSTRATION OF HOP BASED DISTANCE ESTIMATION METHOD

The working of hop based distance estimation method for tree based networks is explained with the help of a tree shown in figure 1. The number of nodes is larger in wireless sensor networks but for demonstration purpose the depth of the tree and number of nodes is confined to 4 and 16, respectively, in the example given below.

Suppose K, G and D are the anchor nodes. Initially, each of these anchors send ANCHOR message to their respective parent nodes i.e. E, C and A, respectively. Anchor table construction process for all the three anchor nodes is demonstrated in this example.

Initially, node K starts MEDEP with its parent node E in order to calculate the distance D_{EK} between the two nodes. After MEDEP is complete, the anchor location table at K is shown in Table I.

K transfers this table to E which further reconstructs the table after performing MEDEP with B. Anchor location table at E is shown in Table II.

E transfers this table to B. Finally B performs MEDEP with A to reconstruct the table and transfer it to A. Anchor location table at node B and A, are shown in table III and IV, respectively.

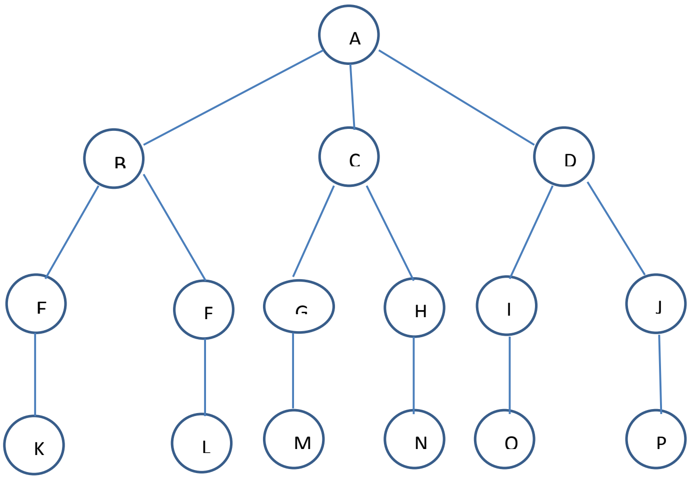

Fig. 1. Example Tree of Depth 3 Consisting of 16 Nodes

TABLE I. ANCHOR LOCATION TABLE AT NODE K

Parent id `	Hop Distance with Anchor	Child id	Hop Distance with Parent Node
E	0	K	DEK

TABLE II. ANCHOR LOCATION TABLE AT NODE E

Parent id	Hop Distance With Anchor	Child id	Hop Distance with Parent Node
B	DEK	E	DBE

TABLE III. ANCHOR LOCATION TABLE AT NODE B

Parent id	Hop Distance With Anchor	Child id	Hop Distance with Parent Node
A	DBE + DEK	B	DAB

TABLE IV. ANCHOR LOCATION TABLE AT NODE A

Parent id	Hop Distance with Anchor	Child Id	Hop Distance with Parent Node
A	DAB + DBE + DEK	B	0

Similarly, G starts MEDEP with its parent node C to calculate the distance D_{CG} between two nodes. After MEDEP is complete, the anchor location table at G is shown in table V.

TABLE V. ANCHOR LOCATION TABLE AT NODE G

Parent id	Hop Distance With Anchor	Child id	Hop Distance With Parent Node
C	0	G	DCG

C also performs MEDEP with A and transfers its table to A. Table corresponding to C is shown in Table VI. The final table corresponding to anchor G available with root node is shown in table VII.

TABLE VI. ANCHOR LOCATION TABLE AT NODE C

Parent id	**Hop Distance With Anchor**	**Child id**	**Hop Distance With Parent Node**
A	DCG	C	DAC

TABLE VII. ANCHOR LOCATION TABLE AT NODE A

Parent id	**Hop Distance With Anchor**	**Child id**	**Hop Distance with Parent Node**
A	DAC+DCG	C	0

The anchor tables corresponding to anchor D at node D and node A are shown in table VIII and IX, respectively.

TABLE VIII. ANCHOR LOCATION TABLE AT NODE D

Parent id	**Hop Distance With Anchor**	**Child id**	**Hop Distance with Parent Node**
A	0	D	DAD

TABLE IX. ANCHOR LOCATION TABLE AT NODE A

Parent id	**Hop Distance With Anchor**	**Child id**	**Hop Distance with Parent Node**
A	DAD	D	0

After obtaining these tables, the root node now merges table IV, VII and IX, changes the fields of the tables it had received and creates a single anchor table which contains distances to all anchors as shown in table X.

TABLE X. ANCHOR LOCATION TABLE CONTAINING DISTANCES TO ALL ANCHORS

Anchor id	**Hop Distance with Anchor**	**Anchor Parent id**	**Hop Distance with Parent Node**
A1	DAB + DBE + DEK	B	0
A2	DAC+DCG	C	0
A3	DAD	D	0

This table is broadcasted by root node to all its child node. Node B receives the table and since one anchor exists in the subtree of this node, it subtracts distance D_{AB} from the hop distance with anchor field in the first row of the anchor location table and adds this distance to the other two rows to its anchor table in its existing anchor location table. Node B obtains this distance from its parent node using its earlier table. The anchor location table at node B is shown in Table XI.

Node E also follows a similar process. The updated anchor table at node E is shown in Table XII. However in case of Node F, the process of updation of anchor table would be different as no anchor node is in the subtree of this node. Node F will not have an existing anchor table so it accepts the entire anchor location table. It then calculates its distance i.e. D_{BF} with its parent node i.e. node B using MEDEP. Subsequently, it adds this distance to all the hop distances with anchor fields in all the rows of the anchor location table. The updated anchor table at node E is shown in table XIII.

TABLE XI. UPDATED ANCHOR LOCATION TABLE AT NODE B

Anchor id	Hop Distance with Anchor	Anchor Parent id	Hop Distance with Parent Node
A1	DBE + DEK	B	DAB
A2	DAB +DAC+DCG	C	-
A3	DAB + DAD	D	-

TABLE XII. UPDATED ANCHOR LOCATION TABLE AT NODE E

Anchor Id	Hop Distance with Anchor	Anchor Parent Id	Hop Distance with Parent Node
A1	DEK	E	DBE
A2	DBE + DAB +DAC+DCG	C	-
A3	DBE + DAB + DAD	D	-

TABLE XIII. UPDATED ANCHOR LOCATION TABLE AT NODE F

Anchor id	Hop Distance with Anchor	Anchor Parent Id	Hop Distance with Parent Node
A1	DBF+ DBE + DEK	B	DBF
A2	DBF +DAB +DAC+DCG	C	-
A3	DBF + DAB + DAD	D	-

V. ANALYSIS

In general, the accuracy of any localization algorithm increases with the increase in the number of anchor nodes in the network. However, this is not always true in case of DV-Hop algorithm. Another variant of DV Hop is DV distance propagation method which utilizes RSSI based distance estimation method. In this method, distance between nodes is measured in meters rather than hops. Although RSSI based methods are economical as they do not require any extra hardware but they are not very accurate as radio propagation is susceptible to various disturbances.

The proposed protocol utilizes the basic ideas of TPSN protocol for distance estimation. TPSN is a popular time synchronization protocol and uses two way message exchanges to synchronize a pair of nodes by calculating clock drift and propagation delay. This propagation delay can also be used to estimate the distance between two nodes. BBLP, continuous ranging protocol and optimized beacon protocol utilize a similar method for distance estimation. These protocols are based on the assumption that three anchors are deployed at the center of a 2D network, contrary to the proposed protocol in which the network is organized as a hierarchical tree based structure. Contrary to this, no assumption related to deployment of anchors is made. Therefore, the performance of BBLP, continuous ranging protocol and optimized beacon protocol is not evaluated.

The performance of TPSN was evaluated on Berkley motes by Ganeriwal et al. [36]. The statistics reported error in magnitude only. The average error was found to be 17.61 µs, 20.91 µs, 23.23 µs, 21.436 µs and 22.66 µs for 1, 2, 3, 4, and 5 hop distances, respectively, whereas, in the worst case, error was 45.2 µs, 51.6 µs, 66.8 µs, 64 µs and 73.6 µs for 1, 2, 3, 4, and 5 hop distances respectively. In the best case scenario, error was 2.8 µs for 3 hops and 0 for all other distances. The basis of distance estimation technique suggested in the proposed protocol is based on TPSN's two way message exchange method for calculating propagation delay. The experimental results of Ganeriwal et al. [36] show that the time estimation error was less than or equal to the average case error in 65% of the times. The accuracy of localization is directly related to time estimation error in the proposed protocol. The reason for this is that it involves propagation delay in order to calculate distance between nodes. So, reduction in timing errors would lead to higher localization accuracy.

The proposed protocol is more energy efficient as compared to DV Hop and other protocols that utilize distance vector exchange to estimate the distance of nodes from anchors. This reduction in energy consumption is due to the fact that all the information related to anchors is routed through the root node. This approach is different from the one adopted by DV Hop and other protocols based on it. DV hop floods the network twice, firstly, to provide anchor coordinates and secondly, to provide correction factor to all nodes in the network. The performance evaluation of the proposed protocol is conducted for various trees. Table X1V along with Fig. 2 show the number of messages exchanged for various trees having depth two and different anchor locations. The results for the tree configuration and anchor locations shown in Table XIV are similar to tree configuration shown in table XV.

Similar evaluation was conducted for trees having depth three. Configuration, location of anchors at various levels in the tree and the number of message exchanges for one tree are shown in table XVI. It was observed that when anchors were located at level 1 and level 2 only, the total number of message exchanges were similar to the results shown in table XIV and XV. However, when the anchors were located at level 2 and level 3, the results found were different as shown in table XVI and Fig. 3. The latter shows the number of messages exchanged for tree having depth three when the anchors were located at level 2 and level 3.

TABLE XIV. CONFIGURATION, LOCATION OF ANCHOR NODES AND TOTAL NUMBER OF MESSAGES INVOLVED IN A TREE OF DEPTH 2

(Level, Subtree No. From Left to Right)	**Total No. of Nodes**	**Total No. of Anchors**	**DVHOP**	**MEDEP**	**Total No. of Anchors**	**DVHOP**	**MEDEP**	**Total No. of Anchors**	**DVHOP**	**MEDEP**	**Total No. of Anchors**	**DVHOP**	**MEDEP**
(1,1)	12	0	222	82	1	220	81	2	218	80	3	216	79
(2,1)	9	1			1			1			0		
(2,2)	3	1			1			0			0		
(2,3)	4	1			0			0			0		
(2,4)	5	0			0			0			0		
(2,6)	2	0			0			0			0		

TABLE XV. CONFIGURATION, LOCATION OF ANCHOR NODES AND TOTAL NUMBER OF MESSAGES INVOLVED IN A TREE OF DEPTH 2

(Level, Subtree Number From Left to Right)	Total Number of Nodes	Total Number of Anchors	DVHOP	MEDEP	Total Number Of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP
(1,1)	19	0	222	82	1	220	81	2	218	80	3	216	79
(2,1)	10	2			1			1			0		
(2,2)	6	1			1			0			0		

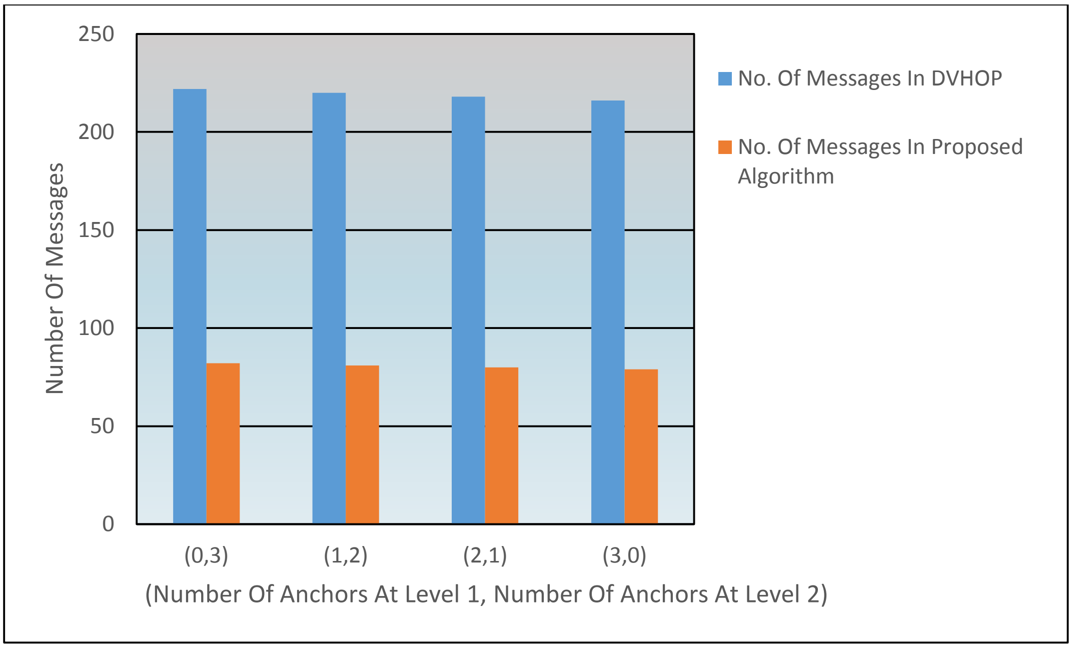

Fig. 2. A Bar Graph Showing the Number of Anchors w.r.t. Number of Messages Exchanged for Distance Estimation in MEDEP and DV Hop for a Tree of Depth 2

TABLE XVI.CONFIGURATION, LOCATION OF ANCHOR NODES AND TOTAL NUMBER OF MESSAGES INVOLVED IN A TREE OF DEPTH 3

(Level, Subtree Number From Left to Right)	Total Number of Nodes	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	TOTAL NUMBER of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP	Total Number of Anchors	DVHOP	MEDEP
(1,1)	10	0	222	82	1	220	81	2	218	80	3	216	76	0	228	85	1	224	82	2	220	79	0	226	84	0	224	83
(2,1)	6	3			1			1			0			0			0			0			0			2		
(2,2)	8	0			1			0			0			0			0			0			1			0		
(3,1)	4	0			0			0			0			2			1			1			1			1		
(3,2)	4	0			0			0			0			1			1			0			1			0		
(3,3)	3	0			0			0			0			0			0			0			0			0		

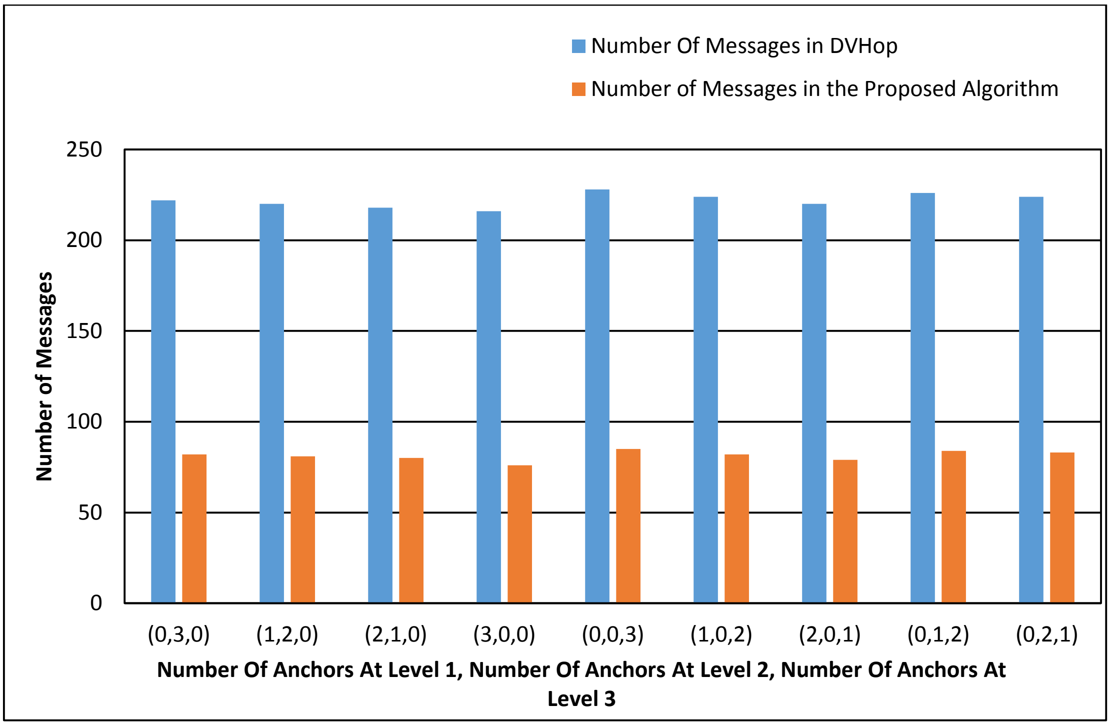

Fig. 3. Bar Graph Showing the Number of Anchors w.r.t. Number of Messages Exchanged for Distance Estimation in MEDEP and DV Hop for a Tree of Depth 3

VI. CONCLUSION

The basic characteristics of DV Hop algorithm is that it utilizes classical distance vector exchange using which all the nodes in the network calculate their distances from all the three anchors and estimate their location. Its major advantages are that it is simple and free from measurement errors. However, it is suitable only for isotropic networks and consumes high energy. The energy consumption of DV Hop is high as it requires the network to be flooded twice in order to provide the coordinates of anchors and the correction factor to all other nodes in the network. The proposed protocol presents a novel localization approach for tree based networks. It calculates the propagation delay using two way message exchanges and then calculates the distance using it. Once the distance of root node from all anchors is calculated, it is forwarded to all other nodes in the network. Thus, the network is flooded only once and requires lesser number of message interchanges, making the proposed protocol more energy efficient as compared to other protocols which make use of distance vector exchange method of estimating distance. Lesser number of message interchanges among nodes also reduces collisions in the network. It does not require extra hardware for localization of nodes and works with non isotropic networks contrary to the DV Hop propagation method.

REFERENCES

[1] T. Alhmiedat, “A Survey on Environmental Monitoring Systems using Wireless Sensor Networks”, Journal of Networks, Vol. 10, No. 11, 2015.

[2] E. Karapistoli , I. Mampentzidou and A. Anastasios, “Environmental monitoring based on wireless sensor network technology: A survey of real world applications”, International Journal of Agricultural and Environmental Information Systems (IJAEIS), Vol. 5, No. 4, 2014.

[3] G. Barrenetxea, F. Ingelrest, G. Schaefer and M. Vetterli, “Wireless sensor networks for environmental monitoring: The sensorscope experience”, 2008 IEEE International Zurich Seminar on Communications, pp. 98–101, Zurich, 12-14 March 2008.

[4] M. F. Othman and K. Shazali, “Wireless sensor network applications: A study in environment monitoring system”, International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012), Vol. 41, pp. 1204–1210, 2012.

[5] F. Ingelrest, G. Barrenetxea, G. Schaefer and M. Vetterli, “Wireless sensor networks for environmental monitoring: The sensor scope experience”, ACM Transactions on Sensor Networks, Vol. 6, No. 2, Article 17, February 2010.

[6] A. Pal, Localization Algorithms in Wireless Sensor Networks: Current Approaches and Future Challenges, Network Protocols and Algorithms, Vol. 2, No.1, 2010.

[7] A. Mesmoudi , M. Feham and N. Labraoui, “Wireless sensor networks localization algorithms: A comprehensive survey”, International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.6, November 2013.

[8] N. A. Alrajeh, M. Bashir and B. Shams, “Localization techniques in wireless sensor networks”, International Journal of Distributed Sensor Networks, Vol. 9, No. 6, 2013.

[9] J. Kuriakose, S. Joshi and V.I. George, “Localization in wireless sensor networks: A survey”, CSIR Sponsored X Control Instrumentation System Conference - CISCON-2013.

[10] Z. Li, R. Li, Y. Wei and T. Pei. “Survey of localization techniques in wireless sensor networks”, Information Technology Journal, Vol. 9, No. 8, pp. 1754-1757, 2010.

[11] D. Niculescu and B. Nath, “DV based positioning in ad hoc networks”, Journal of Telecommunication Systems, Vol. 22, No.14, pp. 267-280.

[12] R. Manzoor, Energy efficient localization in wireless sensor networks using noisy measurements [M.S. thesis], 2010.

[13] H. Chen, “Novel centroid localization algorithm for three dimensional wireless sensor networks”. In Proc. of the 4th international conference on IEEE wireless communications, pp. 1-4, 2008.

[14] X. Lai, J. Wang and G. Zeng, “Distributed positioning algorithm based on centroid of three-dimension graph for wireless sensor networks”, System Simulation Technology, Vol. 20, No.15, pp. 4104-4111, 2008.

[15] J. Li, K. Wang and L. Li, "Weighted centroid localization algorithm based on intersection of anchor circle for wireless sensor network", Journal of Ji Lin University Engineering and Technology, Vol. 39, No. 6, pp. 1649-1653, 2009.

[16] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher, "Range-free localization schemes in large scale sensor networks", In Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking (MobiCom'03), September 2003, San Diego, CA, USA, pp. 81-95.

[17] J. Bachrach, R. Nagpal, M. Salib and H. Shrobe, "Experimental results for and theoretical analysis of a self-organizing a global coordinate system from ad hoc sensor networks", Telecommunications System Journal*,* Vol. 26, No. 2-4, pp. 213-233, June 2004.

[18] W. W Ji and Z. Liu, "An improvement of DV-Hop algorithm in wireless sensor networks", International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1-4, 22-24 Sept. 2006.

[19] H. Chen, K. Sezaki, P. Deng and H. C. So, "An improved DV-Hop localization algorithm for wireless sensor networks", 3rd International Conference on Industrial Electronics and Applications, pp. 1557-1561, 3-5 June 2008.

[20] Y. Wang, W. Tian, Y. Pang and Y. Liu, "A new improved DV-Hop localization algorithm", IET International Conference on Communication Technology and Applications (ICCTA 2011), pp. 772-776, 14-16 October, 2011.

[21] Q. Qian, X. Shen and H. Chen, "An Improved node localization algorithm based on DV-Hop for wireless sensor networks", Comput. Sci. Inf. Syst., pp. 953-972, 2011.

[22] X. Chen and B. Zhang, "Improved DV-Hop node localization algorithm in wireless sensor networks", International Journal of Distributed Sensor Networks, Vol. 8, No. 8, 2012.

[23] S. Tomic and I. Mezei, "Improved DV-Hop localization algorithm for wireless sensor networks", International Symposium on Intelligent Systems and Informatics, pp. 384-394, September 20-22, 2012, Subotica, Serbia.

[24] J. Ding, L. Zhang, G. Cheng, Z. Ling and Z. Zhang, "Study on DV-Hop algorithm based on modifying hop count for wireless sensor networks", International Journal of Computer Science & Engineering Technology (IJCSET), Vol. 2, No. 10, pp. 1452-1456, October 2012.

[25] Y. Liu, X. Luo, C. Long and N. Zhou, "Improved DV-Hop localization algorithm based on the ratio of distance and path length", Journal of Information & Computational Science Vol. 9,

No. 7, pp. 1857-1882, 2012.

[26] H. Yang, Q. Wang and Z. Liu, "A wireless sensor networks DV-Hop localization algorithm", International Conference on Information Science and Computer Applications (ISCA 2013), pp. 281-286, September 2013.

[27] C. Cai and L. Yuan, "DV-Hop localization algorithm improvement of wireless sensor networks", Journal of Theoretical and Applied Information Technology, Vol. 28, No. 3, pp. 1546-1551, February 2013.

[28] F. Zhang, "Positioning research for wireless sensor networks based on PSO algorithm", ELEKTRONIKA IR ELEKTROTECHNIKA, Vol. 19, No. 9, pp. 7 -10, 2013.
[29] Y. Wang, Z. Fang and L. Chin, "A new type of weighted DV-Hop algorithm based on correction factor in WSN's", Journal of Communications, Vol. 9, No. 9, September 2014.
[30] T. Zhang and N. Zhu, "A novel improved node localization algorithm of wireless sensor network", Journal of Chemical and Pharmaceutical Research, Vol. 6, No.6, pp. 1136-1144, 2014.
[31] X. Qin, "An improved DV-Hop wireless sensor network localization algorithm", The Open Automation and Control Systems Journal, Vol.6, pp. 215-219, 2014.
[32] J. Kun and Y. Li, "Wireless sensor networks target localization based on least square method and DV-Hop algorithm", Journal of Networks, Vol. 9, No. 1, January 2014.
[33] H. Sassi, T. Najeh and N. Liouane, "A selective 3-anchor DV- Hop Algorithm based on the nearest anchor for wireless sensor network", World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, Vol. 8, No. 10, pp. 1751-1755, 2014.
[34] X. Yang and K. Rui, "An improved DV-Hop algorithm based on artificial swarm algorithm", Chemical Engineering Transactions, Vol. 46, pp. 223-228, 2015.
[35] S. Rahmatkar and A. Agarwal, "Reference based tree structured time synchronization approach and its analysis in WSN", International Journal of ADHOC, Sensor and Ubiquitous Computing (IJASUC) Vol. 2, No. 1, March 2011.
[36] S. Ganeriwal, R. Kumar and M. B. Srivastava, "Timing-sync protocol for sensor networks", First ACM Conference on Embedded Network Sensor System (SenSys), 2001, pp. 138-149.
[37] A. Saxena, S. Rahamatkar and M. Darbari "A robust single hop tree structured time synchronization protocol for wireless Ad-Hoc sensor networks", International Review on Computers and Software (IRECOS), Vol. 10, No.10, October, 2015.
[38] A. Saxena and M. Darbari, "Multi hop time synchronization protocol for wireless ad-hoc sensor networks", International Review on Computers and Software (IRECOS), Vol. 11, No. 6, June, 2016.