J. Basic. Appl. Sci. Res., **2(11)10945-10950, 2012**

© 2012, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Evaluation of Several Anchor Placement Scenarios and Positioning Methodologies in Wireless Sensor Networks

Vahid Vahidi, Abbas Alighanbari*, Alireza Haghnegahdar

School of Electrical and Computer Engineering
Department of Communications and Electronics
Shiraz University, Shiraz, Iran

ABSTRACT

Growing advances in Wireless Sensor Networks (WSN) have provoked a great interest among researchers. WSN offers a wide range of applications, in which localization capabilities play essential roles. There are some algorithms, proposed to tackle the problem of positioning. A major factor, affecting the accuracy of such algorithms, is the anchor placement. In this paper, several anchor placement scenarios are examined and their performances are compared, in conjunction with a few distributed positioning algorithms. Simulation results indicate that increasing the number of anchors in a network and/or the radio range of the sensors, does not necessarily results in better positioning accuracy of the network. Instead, the radio range of sensors and the number of deployed anchors in a specific topology must be selected optimally. In addition, some types of anchor placement such as cross form, lead to acceptable accuracy since some types of anchor placement such as circular placement do not have the potential for good accuracy.

Keywords: Wireles Sensor network; Positioning; anchor placement; Raidio range, Dv Hop, Dv Dis, Bounding Box, Multilateration, Mass Spring.

I. INTRODUCTION

A WSN is composed of several sensor nodes [1]. Each node has the ability of sensing, computing, and communicating with neighboring nodes. Due to its versatility and low costs, the WSN has been widely used in many applications, such as habitat monitoring, health care, environmental, and forecasting. Localization capabilities are essential in most WSN applications. However, precise location information may be unavailable due to the constraints in energy, computation, or terrain [2]. Positioning techniques are used to estimate the location of the nodes. To achieve this, *a priori* knowledge about the position of some sensors, *anchors,* is required [3]. The location of the anchors can be obtained, using a global positioning system (GPS), or by installing them at points with known coordinates.

Most localization algorithms usually adopt one or more anchor deployment strategies [4]-[6]. However, a thorough study of the relationship, between the anchor placement and localization performance, has not yet been attempted. For a square network, it has been suggested that the four corners of the square are the best anchor positions [6]. Also, a specialized circular anchor placement scheme is presented in [7]. The most important difference of this paper with the previously published work is the comprehensiveness of this paper. In the previous work, only the proposed anchor placement methods are tested individually and no comparisons with various placement strategies are made. Different positioning algorithms are considered in this paper and it is found that some topologies lead to better positioning accuracy if specific positioning algorithms are used.

The goal of this paper is to evaluate anchor placement topologies that lead to appropriate positioning accuracy with minimum number of deployed anchors. Therefore, the contribution of this work is: 1) a rigorous investigation of several anchor placement strategies is presented. 2)Comparisons are made, with respect to the localization accuracy and network coverage 3) the combination of some localization methods is considered and a wide range of scenarios are deployed 4) the best radio range which results in a minimum positioning error is presented for each anchor placement scenario 5) some of the best topologies and positioning method are discussed.

The rest of the present paper is organized as follows. Section II provides a brief review of positioning algorithms. Sections III and IV present the methodology and simulation results, respectively and, finally, the conclusions are made in Section V.

II. POSITIONING ALGORITHMS

Localization can be performed in distributed or centralized manners. In a distributed localization, each node is required to locate itself using information available to it. Depending on the network topology and structure, each node may have some information about its neighbors. This information can be the distance to them, and their position. The positioning algorithms consist of three

***Corresponding Author:** Abbas Alighanbari, School of Electrical and Computer Engineering Department of Communications and Electronics, Shiraz University, Shiraz, Iran. alighanbari@ieee.org

steps: a) Finding neighbors and calculating the distance between nodes and anchors, b) Calculation of unknown positions, and c) Refinement of new positions [12].

A. Step 1:

In the first step, anchors send their positions to their one-hop neighbors (nodes in their radio range), and then calculate the distance to them. This information is flooded in the network using one of the following methods:

1) DV-Hop Propagation Method

The method is performed in two flood waves [8]. During the first wave, the anchors send their identifications (IDs) and positions to their neighbors, which are in their radio range. Then, they will send this information to all their one-hop neighbors, incrementing the hop count by one. When a node receives information from an anchor, whose ID is unknown to it, it stores this information in its memory and forwards it to its one-hop neighbors, again by incrementing the hop count. When a node receives information regarding an anchor, whose ID is known, it only updates the corresponding hop count and forwards it to its one-hop neighbors when the newly received hop count is smaller than the one in its memory. The nodes continue exchanging information until a predefined 'flood limit' has been reached. At the end of this procedure, each node knows the positions of - and its hop count distance to - at least 'flood limit' anchors. Note that this will also be the case DV-Hop for the anchors that are responsible for generating a second flood wave giving estimations of the average hop distance. When an anchor *i* receives the position and hop count distance of another anchor *j*, it updates its estimation of the average one hop distance c_i, by computing the sum of its Euclidean distances to all known anchors divided by the sum of all the hop counts to these anchors (h_i):

$$c_i = \frac{\sum\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum h_i} \quad i \neq j \qquad (1)$$

2) DV-Distance Propagation Method

The method is similar to the DV-Hop propagation [6]. Each anchor initiates a flood, by sending its position to its one-hop neighbors. An initial distance estimate of zero is chosen and updated through future propagations. The receiving nodes increase the distance measure with an estimation of their distance to the transmitting node, and send the new distance to all their one-hop neighbors. This procedure continues until no more messages are generated or a predefined 'flood limit' is reached. At the end of this flooding, each node knows the position of at least 'flood limit' anchors and has an estimation of its distance (shortest path) to them.

B. Step 2:

In the second phase nodes determine their position based on the distance estimates to a number of anchors provided by Step1. Two positioning approaches are:

1) Multi_Lateration Method

This method is a form of triangulation [10]. From the estimated distances d_i and known positions (x_i, y_i) of the anchors, the method leads to solving a linear equation:

$$\hat{x} = (A^T A)^{-1} A^T b \qquad (2)$$

where:

$$A = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \qquad (3)$$

$$b = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix} \qquad (4)$$

and $\hat{x}$ is the location estimation .

2) Bounding Box (min-max) Method :

The main idea is to construct a bounding box for each anchor using its position and distance estimate, and then to determine the intersection of these boxes. The position of the node is set to the center of the intersection box. Note that the estimated position by Min-max is close to the true position computed through lateration (i.e. the intersection of the three circles) [12]. The bounding box of anchor 'a' is created by adding and subtracting the estimated distance d_i from the anchor position(x_a, y_a):

$$[x_a - d_a, y_a - d_a] \times [x_a + d_a, y_a + d_a] \qquad (5)$$

The intersection of the bounding boxes is computed by taking the maximum of all the coordinate minimums and the minimum of all maximums:

$$[\text{Max}\,(x_i - d_i), \max\,(y_i - d_i)] \times [\min\,(x_i + d_i), \min\,(y_i + d_i)] \qquad (6)$$

The final position is set to the average of both corner coordinates.

C. Step 3:

The objective of the third phase is to refine the (initial) node positions computed during Step2. These positions are not very accurate, even under good conditions (high connectivity, small range errors), because not all available information from previous phases is error free.

III. METHODOLOGY

In this paper, localization process is done as follow.

1. As the first step, each node starts finding its neighbor and computing distance to anchors using DV-Hop or DV-Distance method as described, previously.
2. Then, the nodes, which have more than three anchors in their neighborhood, estimate their coordinates using the Bounding Box or Multi-lateration method. Therefore, four different positioning algorithms, DV-Hop/Bonding Box, DV-Hop/Multi-lateration, DV-Distance/Bounding Box, and DV-Distance/Multi-lateration are considered.
3. Finally, positions obtained at the previous step, are refined, using the refinement method, similar to the mass spring approach [7].

In the refinement stage, the nodes, which have successfully found their positions, measure their new distance from their neighboring anchors($dist_{e_i}$). The absolute difference between the measured distance and the estimated distance, resulting from the first step (DV-hop or DV-distance), is calculated. Finally, nodes change their position towards the neighboring anchor, which has the largest absolute difference amongst all neighboring anchors. In order to find the direction of a neighboring anchor, each node computes the angle ϕ as below:

$$\phi = \tan^{-1}\frac{(x_{max}^{anchor} - x^{est})}{(y_{max}^{anchor} - y^{est})} \qquad (7)$$

where, $(x_{max}^{anchor}, y_{max}^{anchor})$ is the coordinate of the anchor with the largest absolute difference and (x^{est}, y^{est}) is the estimated position of the unknown node. The refined coordinate (x^{new}, y^{new}) is computed as:

$$x^{new} = x^{est} + (\alpha \times dist\_e_{max} \times cos(\phi))$$
$$y^{new} = y^{est} + \left(\alpha \times dist_{e_{max}} \times cos(\phi)\right) \quad (8)$$

where α is the step of refinement, set as 0.1, and $dist\_e_{max}$ is the maximum difference. For clarification, the flowchart of the refinement step is illustrated in Fig. 1.

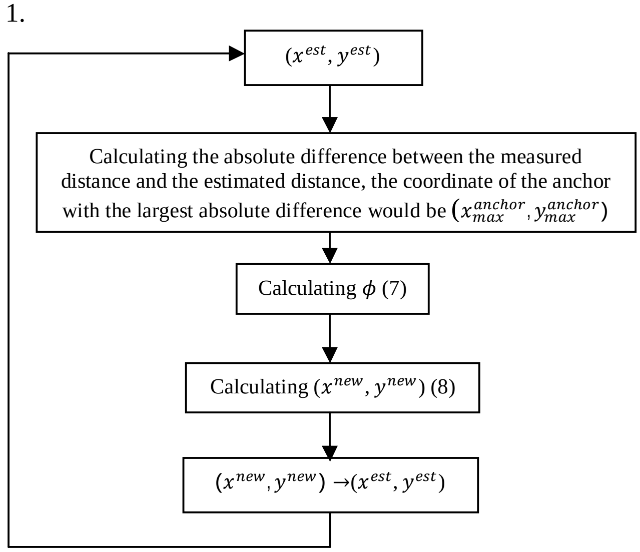

Figure 1. Flowchart of the refinement step

In order to measure the performance of these algorithms, the position error and coverage of the network are defined. The positioning error is the mean difference of the estimated and actual coordinates of the nodes, based on the Euclidean norm [10]. The Position Error (PE) is calculated as follows:

$$PE = \frac{1}{|U|}\sum_{i \in U} \| x_i^{est} - x_i^{act} \| \qquad (9)$$

where U denotes the set of nodes with unknown position and x_i^{est}, x_i^{act} are the estimated and actual positions, respectively.

When the position error of an unknown node, lets say node i, $error_i = (x_i^{est} - x_i^{act})$ is larger than *R*, that node is defined as '***un-localized***'. In this case, the algorithm has failed to produce a useful estimate of the node position. To measure the performance of the algorithms, in regards with this phenomenon, the *Coverage* is defined, as the percentage of the unknown nodes, which the algorithm positioned successfully, with an error smaller than *R*.

$$coverage = {}^{|P|}/_{|U|} \tag{10}$$

where,

$$P = \{i \in U : \left\| x_i^{est} - x_i^{act} \right\| \leq R\} \tag{11}$$

In order to increase the coverage, the localization procedure is repeated, in a few iterations. After the first iteration, the newly localized nodes will be used as anchors. This procedure continues until all the nodes are localized, or no more nodes can be localized, using the method. Any positioning errors, especially in the primary iterations, contribute to the phenomenon of error propagation. Therefore, it is expected that for a large number of iterations, the propagated error significantly render the performance.

IV. SIMULATION RESULTS

The simulation consists of 100 nodes deployed in an area of $100 \times 100\ m^2$. For simplicity, a 2-D space is assumed. Some nodes have predefined positions, known as anchors. The rest are randomly, and uniformly, distributed in the area. The positioning algorithm is to estimate the location of the randomly distributed nodes. It is also assumed that the nodes can estimate their distances to their neighbors, with an error which has a normal distribution, as $Error \sim N(0{,}0.05)$ [11]. The radio range of the nodes changes from 1 to 100m, with the incremental steps of 1m. The simulation is run for 100 times, for each scenario, as follows.

Several anchor placement strategies, with different number of anchors, are studied. The arrangements include all-around, cross $(\times)$, plus $(+)$, V-, U-, and M- shapes. The results of the most accurate schemes are presented. The algorithms are studied in two groups. The first group uses a single iteration, while the second group uses more. In each topology, anchors are placed in a symmetrical manner, as the nodes are uniformly distributed in the area. The networks, which use several iterations, involve higher computational complexities, and suffer larger error propagation. Through experiment it was observed that four iterations are sufficient to achieve acceptable accuracy and coverage, therefore, the second-group algorithms are set to use 4 iterations.

Fig. 2 illustrates a typical topology in which 17 anchors are deployed in a cross $(\times)$ form. For the scenario, the simulation results, indicated in Table I, present the range, in which, the coverage is more than 98%, and the PE of that range. The table also shows the optimum range, which has the minimum PE. Using Table I, it is observed that the minimum PE is obtained when a single-iteration DV-hop/Multi-lateration is used. The minimum error is achieves in the range of 68m. Similar producers, for a large number of scenarios, are tested and the ones with the best performance are shown in Table II.

Considering Table I, the coverage of the iterative algorithm reaches one for small ranges, as expected. The optimum range is chosen amongst the ranges in which the coverage is one. Smaller radio ranges may lead to smaller positioning error, however, some nodes will remain undetermined. The results of Table II suggest that the DV-Distance/Bonding-Box combination is best in terms of accuracy, in most topologies. Moreover, it is observed that using larger iterations, better accuracy is achieved, in most cases. In the no-iteration schemes with small errors, the radio range of the nodes is required to be extremely high. This increases the implementation cost. Therefore, in each anchor placement scheme there is a tradeoff between the accuracy, complexity, and the cost of the network.

V. CONCLUSIONS

In this paper, a rigorous evaluation of several anchor placement scenarios, in conjunction with a few distributed positioning algorithms, is presented. Positioning algorithms are done in three phases of data propagating, finding the first step position, and the refinement of the positions. For the first phase DV-dis and DV-hop methods are used and for the second phase, the multilateration and bounding box methods are used. For the refinement step, an algorithm is proposed which is based on the mass spring method. The best radio range which results in a minimum positioning error is calculated for each anchor placement scenario. The results illustrate the essential role of the anchor placement in the performance of the positioning algorithms. It is found that the DV-distance/Bounding Box algorithm achieves the best accuracy, amongst four investigated algorithms. Although, single-iteration algorithms involve lower computational complexity, it is found that significantly better coverage and accuracy are achieved, using larger iterations.

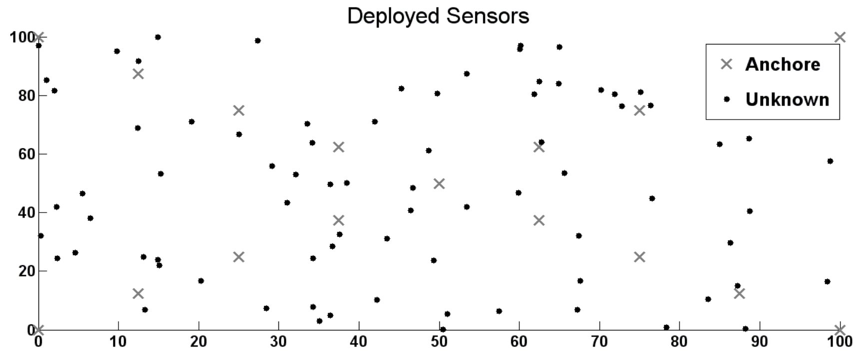

Figure 2. Seventeen anchors are deployed in a cross (×) form.

TABLE I. SIMULATION RESULTS FOR THE 17 ANCHOR SCENARIO, DEPLOYED IN A CROSS (×) FORM

Iteration	Positioning Algorithm	Range(coverage ≈ 1) (meters)	PE for Range(coverage≈1)	Optimum range (meters)	Minimum PE
No	DV-Hop/ Bounding Box	68	3.997	70	3.786
	DV-Hop/Multi-lateration	65	0.5267	68	0.1
	DV-Distance/ Bounding Box	61	4.034	72	3.346
	DV- Distance /Multi-lateration	71	4.351	83	3.584
Yes	DV-Hop/ Bounding Box	28	1.705	29	0.7346
	DV-Hop/Multi-lateration	58	0.3727	66	0.1651
	DV- Distance / Bounding Box	30	1.509	41	0.2628
	DV- Distance /Multi-lateration	44	0.2101	62	0.1843

TABLE II. OPTIMUM CASES FOR DIFFERENT SCENARIOS

Anchor Placement Scenario	Optimum Positioning Algorithm	Iteration	Optimum Range(meters)	Optimum PE
8 around and 1 in the middle	DV- Distance / Bounding Box	Yes	41	0.598
16 around and 1 in the middle	DV- Distance / Bounding Box	Yes	37	0.341
9 in Cross(×) form	DV- Distance / Bounding Box	Yes	54	0.3038
17 in Cross(×) form	DV-Hop/Multi-lateration	No	68	0.0978
9 in Plus(+) form	DV- Distance / Bounding Box	Yes	42	0.4197
9 in Plus(+) form and 4 in the corners	DV-Hop/ Bounding Box	No	83	0.5138
17 in Plus(+) form and 4 in the corners	DV- Distance / Bounding Box	Yes	41	0.1259
11 in U form	DV- Distance / Bounding Box	Yes	50	0.3562
19 in U form	DV- Distance / Bounding Box	Yes	34	0.416
17 in M form	DV- Distance / Bounding Box	Yes	40	0.7017

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communications Magazine, 40(8):102-114, August 2002.

[2] Samira Afzal, A Review of Localization Techniques for Wireless Sensor Networks, *J. Basic. Appl. Sci. Res.*, 2(8)7795-7801, 2012.

[3] Amjad Osmani, Design and Evaluation of New Intelligent Sensor Placement Algorithm to Improve Coverage Problem in Wireless Sensor Networks, *J. Basic. Appl. Sci. Res.*, 2(2)1431-1440, 2012.

[4] K.-P. Shih and S.-S.Wang., “Distributed direction-based localization in wireless sensor networks,” In ISCC, pages 373–378, 2005.

[5] C. Liu and K. Wu., “Performance evaluation of range-free localization methods for wireless sensor networks,” In IPCCC, pages 59–66, 2005.

[6] A. Savvides, W. Garber, S. Adlakha, R. Moses, and M. B. Srivastava., “On the error characteristics of multihop node localization in ad-hoc sensor netwoks,” In 2nd Intl. Workshop on Inform. Proc. in Sensor Networks, 2003.

[7] S.-S. Wang, K.-P. Shih, and C.-Y. Chang., “Distributed direction-based localization in wireless sensor networks,” Computer Commun., 30(6):1424–1439, 2007.

[8] D. Nicolescu and B. Nath, “Ad hoc positioning system (aps),” IEEE GLOBECOM (San Antonio), November 2001, pp. 2926-2931.

[9] B. Aksoy, V. Ustun, and J. S. Smith, “localization using discreet event simulation,” Proceedings of the 2004 Winter Simulation Conference.

[10] L. Doherty, Check, “Algorithms for position and data recovery in wireless sensor networks,” Master's thesis, University of California at Berkeley.

[11] P. Biswas and Y. Ye, “A distributed method for solving semi definite programs arising from ad hoc wireless sensor network localization,” Tech. report, Dept of Management Science and Engineering, Stanford University, 2003.

[12] K. Langendoen and N. Reijers, “Distributed Localization Algorithms,” works. In First Int. Conf. on Embedded Networked Sensor Systems (SenSys), pages 1–13, November 2003.