Half-Symmetric Lens based Localization Algorithm for Wireless Sensor Networks

Noureddine Lasla * , Abdelouahid Derhab * , Abdelraouf Ouadjaout*, Miloud Bagaa *, Adlen Ksentini § and Nadjib Badache *

*Research Center on Scientific and Technical Information (CERIST)
Rue des 3 freres Aissou,Ben-Aknoun, Algiers, Algeria
Email: {nlasla, aderhab, aouadjaout, bagaa, badache}@mail.cerist.dz
§ IRISA - University of Rennes 1
Campus Universitaire de Beaulieu, Rennes, France
Email: adlen.ksentini@irisa.fr

***Abstract*—The area-based localization algorithms use only the location information of some reference nodes, called anchors, to give the residence area of the remaining nodes. The current algorithms use triangle, ring or circle as a geometric shape to determine the sensors' residence area. Existing works suffer from two major problems: (1) in some cases, they might issue wrong decisions about nodes' presence inside a given area, or (2) they require high anchor density to achieve a low location estimation error. In this paper, we deal with the localization problem by introducing a new way to determine the sensors' residence area which shows a better accuracy than the existing algorithms. Our new localization algorithm, called HSL (Half Symmetric Lens based localization algorithm for WSN), is based on the geometric shape of *half-symmetric lens*. We also uses the Voronoi diagram in HSL to mitigate the problem of unlocalizable sensor nodes. Finally, we conduct extensive simulations to evaluate the performance of HSL. Simulation results show that HSL has better locatable ratio and location accuracy compared to representative state-of-the-art area-based algorithms.**

***Index Terms*—Localization, sensor network, area, accuracy**

I. Introduction

The area-based localization algorithms use only the location and proximity information of some special nodes with known locations, labelled as anchors, to give the *residence area* of the remaining nodes. The residence area represents a geographical region containing the sensor node. The construction process of this region consists in drawing a set of geometric shapes using the positions of its neighboring anchors. Afterward, the sensor tests whether it is inside (positive information) or outside (negative information) these shapes. The overlapping of the resulting shapes determines the sensor's residence area. If the sensor does not belong to any shape, it is called *unlocalizable*. The final location of the sensor can be estimated by computing the centroid of its residence area. The efficiency of the area-based algorithms depends on the size of the calculated residence area; the smaller the size of the area is, the better the accuracy is likely to be.

Existing algorithms employ three types of primitive geometric shapes to draw the sensors' residence areas: (1) *Triangle* [2], (2) *Ring* [4], and (3) *Circle* [7]. However, these algorithms suffer from two major issues: (i) in some cases, the decision about the presence of a sensor inside a given area is misleading, which affects the correctness of the obtained residence areas or (ii) they require high anchor density to achieve a low location estimation error.

To address the above issues, we propose in this paper a new area-based localization algorithm, called HSL (Half Symmetric Lens based localization algorithm for WSN). HSL is based on the geometric shape of *half-symmetric lens* which can be simply drawn using only the location information of two anchors. To resolve the problem of unlocalizable sensors, HSL uses the Voronoi diagram [6] to divide the network into a set of cells, and locates each sensor within one Voronoi cell. This allows an unlocalizable sensor to use the negative information to narrow-down its residence area, initially defined as the Voronoi cell where it resides in.

The rest of the paper is organized as follows: Section II presents related work on area-based algorithms. A detailed description of HSL is presented in Section III. Section IV presents simulation results. Section V concludes the paper.

II. Related work

He et. al [2] have proposed an algorithm called APIT, which is a triangular area-based algorithm. In APIT, each sensor s constructs a set of triangular regions made up of vertices formed by all the possible set of three neighbouring anchors. The sensor's presence inside or outside of a triangle can be checked using an Approximative Point In Triangle (APIT) test. However, APIT requires a dense network to perform the test, and can make an incorrect decision in sparse network. Therefore, because the test can fail, both the certainly of the determined sensor's residence area and the accuracy of the estimated position are affected.

Liu et al.[4] have proposed a localization algorithm called ROCRSSI, which draws the sensor residence area as the overlapping of different rings. The ring is formed as the intersection of two concentric circles centred at one anchor. The radius of the inner and the outer circles are the real distances between the first anchor and two other anchors. Based on the comparison of RSSI values between the anchors, the sensor can check whether it is inside or outside the formed ring. In ROCRSSI, each anchor should inform the sensor of the measured RSSI values with its neighbours anchors, which incurs extrat RSSI message exchanges.

The circular-area based algorithms [7], [8], [9], [3] calculate the residence area of a sensor based on the assumption that the radio communication coverage area is modeled as a perfect circle with known radius R. So, each sensor can deduce that is within the areas covered by the radio communication of its neighbour anchors. The overlapping area of all the circles, where a sensor is, defines the sensor's residence area. In order to refine the sensor's residence area, the communication regions of anchors within two-hops are discarded from the initial residence area. However, modeling the radio communication coverage area as a perfect circle with known radius is not realistic and does not often hold in practice. Furthermore, it might make a localizable sensor unlocalizable.

III. The proposed algorithm

This section presents our proposed localization algorithm. We first illustrate the basic idea of the approach and we give later the detailed description of the algorithm.

A. Basic idea

In the following, we describe the logical way of our thinking which allowed us to define a new way to draw the sensor residence area. Let $A = \{A_1, A_2, ...A_m\}$ be a set of neighbouring anchors of sensor s in two-dimensional Euclidean plane. For any two anchors A_i and A_j, let $B(A_i, A_j)$ be the perpendicular bisector of segment A_i, A_j, which divides the plane in two half planes HP_{A_i} and HP_{A_j} containing A_i and A_j, respectively. Thus, s is in HP_{A_i} if it is closer to A_i than A_j, or s is in HP_{A_j} if it is closer to A_j than A_i. Let now draw two circles C_{A_i} and C_{A_j} centred at A_i and A_j respectively, with the same radius equals to the distance d_{ij} between A_i and A_j. The intersection of C_{A_i} and C_{A_j} creates a geometric shape called *symmetric lens* or *Vesica piscis* [5]. The bisector $B(A_i, A_j)$ divides the symmetric lens in two equal half symmetric lens $HSL(A_i, A_j)$ and $HSL(A_j, A_i)$ containing A_i and A_j respectively. Therefore, if s is in HP_{A_i} (resp. HP_{A_j}) and is at distance to A_j (resp. A_i) less than the radius of the two circles then s must lie in $HSL(A_i, A_j)$ (resp. $HSL(A_j, A_i)$). Otherwise, s is outside the symmetric lens $SL(A_i, A_j)$ defined by the coordinate of A_i and A_j. We note that the circles drawn to construct HSL are totally different from the circles in the previous presented circular-area based algorithms [7], [8], [9], [3].

RSSI measurement for Near-far relationship: To check its presence inside a half symmetric lens, the sensor needs a mechanism or metric that determines the near-far relationship among nodes in the network. As there is no information about physical distance between a sensor and its anchors, one method that allows to derive the near-far information with no extra hardware is by using RSSI measurement. For instance, the experimental tests on MICA and MICAz motes, presented in [2], [10], show that RSSI values decrease monotonically with increasing distance. In addition, very recent work [1] shows that is also possible to indicate near-far relationship among sensors based on RSSI, even in an environment with obstacles, by using power scanning technique.

B. Unlocalizable node problem

In some situation, it is possible that a sensor s cannot locate itself within any circles or half-symmetric lens of its neighbouring anchors, and it is considered unlocalizable. To overcome this problem, we propose to partition the network area into a set of disjoint sub-areas in such a way to allow sensor s to locate itself within one of those sub-areas. One solution allowing this partitioning is Voronoi diagram. In the following, we give some basic definitions on Voronoi diagram.

Let $A = \{A_1, A_2, ..A_n\}$ be a set of anchors in the two-dimensional bounded network area. The Voronoi cell of an anchor A_i with respect to a set of anchors A, denoted $VN(A_i)$, is the set of points in the plane which are closer to A_i than any anchor in $A \setminus \{A_i\}$. If the Voronoi cell of each anchor is constructed with respect to all other anchors in the network, the set of Voronoi cells will be a partition of the sensor field. From the above definition, we can conclude that each sensor node is located in the Voronoi cell of its closest anchor (using only one anchor). Using further from/closer to information, each sensor node can determine its closest anchor, and hence the Voronoi cell where it resides in.

C. Algorithm description

We make the following assumptions about the nodes and the network.

- A few number of sensor nodes, called anchors, get their own location information via GPS or manual pre-loading.
- Each anchor can get its Voronoi cell with respect to all anchors in the network.

To perform localization, each anchor starts broadcasting a beacon message containing its location information and its Voronoi cell (polygon vertices) to its reachable sensors. Upon receiving the beacon, each sensor s, constructs its neighbour anchors list denoted by AL_s. Each row in the AL_s includes; the anchor's ID, the anchor's location, the anchor's Voronoi cell and the RSSI corresponding to the received beacon message from the anchor.

The Voronoi cell of the nearest anchor, which has the strongest RSSI in AL_s, represents the initial sensor's residence area. According to the presence of the sensor to each of the half symmetric lens, defined based on the coordinates of its neighbour anchors, the initial residence area is refined.

D. Symmetric lens presence test

The symmetric lens presence test consists in checking if the sensor s is inside or outside the symmetric lens area defined by the coordinate of any two neighbouring anchors. Formally: a sensor s is inside the symmetric lens formed by the coordinates of two anchors A_i and Aj, if and only if:

$$RSSI_{A_i s} > RSSI_{A_i A_j} \quad and \quad RSSI_{A_j s} > RSSI_{A_i A_j}$$

If the symmetric lens presence test for a node s with respect to two neighbour anchors A_i and A_j succeeds, then s performs a second test, called *Half-symmetric lens prensence test*. The goal of this test is to find in which sub-area the sensor node resides. Accordingly, if $RSSI_{A_i s} > RSSI_{A_j s}$ then s concludes that is in the sub-area including A_i. Otherwise, it is in the sub-area including A_j.

The sub-area that contains s is called the *Half-symmetric lens residence area*. We denote this area by $HSL(s, A_{in}, A_{out})$, where A_{in} is the anchor residing in the same half symmetric lens area as s, and A_{out} is the other anchor forming the symmetric lens.

In addition, if the sensor is not within the symmetric lens area $SL(A_i, A_j)$, two conclusions about its possible residence area can be made:

1) The sensor s is inside one of the circles centred at A_i and A_j and not in the symmetric lens $SL(A_i, A_j)$, this area have a *crescent* geometric shape and we denote it by CR. This can be checked by simply comparing if the $RSSI_{A_i s} > RSSI_{A_j A_j}$ or if $RSSI_{A_j s} > RSSI_{A_j A_j}$. In the first case, s is within the circle centred at A_i minus $SL(A_i, A_j)$, noted by $CR(A_i, A_j)$. in the second case, s is within the circle centred at A_j minus $SL(A_i, A_j)$, noted by $CR(A_j, A_i)$.
2) Otherwise, the sensor is outside the area defined by the union of the two circles centred at A_i and A_j with the same radius equals to the distance between them. We refer to this area as *foreign area* and we denote it by $F(A_i, A_j)$.

An example on how the sensor constructs its residence area is illustrated in Figure 1. The three anchors A_1, A_2 and A_3 are neighbours of sensor s, and according to the Voronoi diagram, the network is divided into three sub-areas $VN(A_1)$, $VN(A_2)$ and $VN(A_3)$. Initially, the sensor s locates itself within the Voronoi cell of its closest anchor $V(A_1)$ (see Figure 1(a)). Then, the sensor s finds itself inside $HSL(A_1, A_2)$, as depicted in Figure 1(b), and refines its initial residence area by executing Algorithm 1 (line 7 or 9). After that, sensor s finds itself outside $HSL(A_1, A_3)$ but inside $CR(A_1, A_3)$ (see Figure 1(c)). So, it refines further its residence area by executing Algorithm 1 (line 13 or 16).

Each sensor node s, after executing Algorithm 1, estimates its position as the centroid of the obtained residence area R_S. However, because the low capability of sensors in terms of power and computation, we must simplify the process of determining the coordinate of the finale residence area R_s using only basic arithmetic operations. Thus, we propose the use of grid scan algorithm composed of three steps: (1) dividing the initial residence area, which is defined by the vertexes of the Voronoi cell, into grid array, (2) scanning the grids and mark those belong to the sensor residence area as valid grids (3) estimate the sensor's location as the centroid of the valid grids.

IV. Simulation results

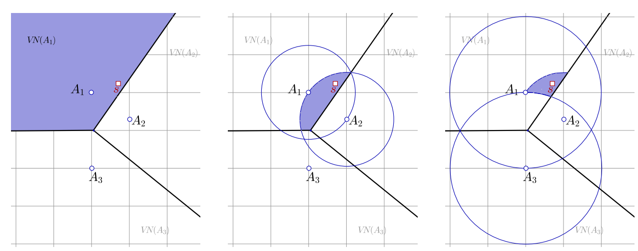
In this section, the performance of HSL is compared with that of APIT, ROCRSSI and a representative circular-area based algorithm called DRLS [7]. The algorithms are evaluated in terms of the following metrics:

1) *Locatable ratio*: is defined as the percentage of nodes successfully located within the residence area by the localization algorithm.
2) *Estimation error*: is defined as the average Euclidean distance between the real location of a node and its estimated location.

In the simulation setting, n sensor and m anchors nodes are randomly deployed with a uniform distribution within an area of size S. We simulate two levels of densities: high ($250m \times 250m$), and low ($400m \times 400m$). The ratio of anchors is defined as $\frac{m}{n}$. In our simulation results, each plotted point represents the average of 100 executions. We plot the 95% confidence interval on the graphs. The number of nodes

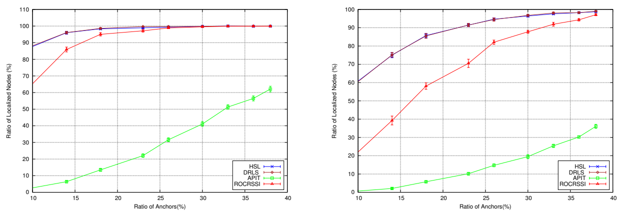
Algorithm 1 Residence area of sensor s

1: let A be the set of neighbour anchors.
2: let $V(A_n)$ be the Voronoi cell of the nearest anchor A_n, where $A_n \in A$.
3: let $R = V(A_n)$ be the initial residence area.
4: **for** each $A_i, A_j \in A$ where $A_i \neq A_j$ **do**
5:   **if** $RSSI_{A_i s} > RSSI_{A_i A_j}$ and $RSSI_{A_j s} > RSSI_{A_i A_j}$ **then**
6:     **if** $RSSI_{A_i s} > RSSI_{A_j s}$ **then**
7:       $R = R \cap HSL(A_i, A_j)$
8:     **else if** $RSSI_{A_j s} > RSSI_{A_i s}$ **then**
9:       $R = R \cap HSL(A_j, A_i)$
10:     **end if**
11:   **else**
12:     **if** $RSSI_{A_i s} > RSSI_{A_i A_j}$ **then**
13:       $R = R \cap CR(A_i, A_j)$
14:     **else**
15:       **if** $RSSI_{A_j s} > RSSI_{A_i A_j}$ **then**
16:         $R = R \cap CR(A_j, A_i)$
17:       **else**
18:         $R = R \setminus C(A_i) \cup C(A_j)$
19:       **end if**
20:     **end if**
21:   **end if**
22: **end for**



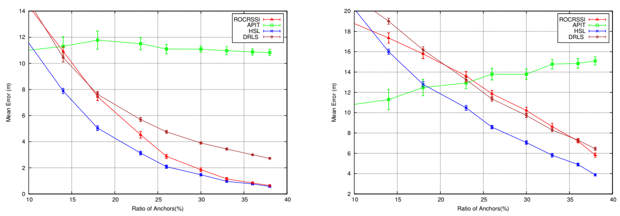
(a) s is within the $VN(A_1)$ (b) s is in half symmetric lens $HSL(A_1, A_2)$ (c) s is in $HSL(A_1, A_2)$ and not in $SL(A_1, A_3)$

Fig. 1: Residence area construction



(a) Network size (250×250) (b) Network size (400×400)

Fig. 2: Locatable ratio



(a) Network size (250×250) (b) Network size (400×400)

Fig. 3: Estimation error

and the node's transmission range are set to 300 and $40m$ respectively.

1) Locatable ratio: Figure 2 shows the *Locatable ratio* as a function of ratio of anchors. From figure 2(a) and figure 2(b), we can notice more nodes can be localized as the ratio of anchors increases. This is due to the fact that nodes that do not have enough or any neighbouring anchors under low ratio of anchors, start having more neighbour anchors when this ratio increases, and hence they become locatable. We can also notice that HSL and DRLS outperform the other localization algorithms, ROCRSSI comes second, whereas APIT shows the worst performance. This can be explained as follows: each node in HSL and DRLS needs at least one neighboring anchor to locate itself within an initial residence area. On the other hand, ROCRSSI requires at least two anchors to construct its residence area, where APIT requires at least three anchors. In addition, the three (resp., two) anchors in APIT (resp., ROCRSSI) have to form a triangle (resp., ring) and the sensor must lie within that triangle (resp., ring) in order to locate itself. These strong constraints lead to low locatable ratio.

2) Estimation error: Figure 3 shows the *estimation error* as a function of ratio of anchors. The first observation we can draw from figures 3(a) and figure 3(b) is that for some small values of ratio of anchors, APIT and ROCRSSI are better than HSL and DRLS in terms of estimation error. However, this observation is misleading as the estimation error of HSL and DRLS in these cases is the average of many locatable nodes, whereas it is the average of small number of locatable nodes in the case of APIT and ROCRSSI. As the ratio of anchors increases, HSL outperforms APIT, ROCRSSI, and DRLS as the voronoi cells become smaller and the number of successful belonging tests increases, and hence lower estimation error is achieved. The curve of APIT is not decreasing as the ratio of anchors increases. This is due to the fact that APIT can make incorrect decision about the presence of sensors inside triangles. As explained in the related work section, the correctness of the presence test in APIT depends on the sensors' neighbours density, which depends on the network topology that changes for each value of anchor ratio. On the other hand, the curve of ROCRSSI is decreasing as a function of ratio of anchors because the probability to have successful presence tests increases, and hence smaller residence areas and lower estimation errors are obtained.

V. Conclusion

In this paper, we have proposed HSL, a new distributed area-based localization algorithm for wireless sensor network. HSL is designed to achieve high locatable ratio and low error location under lower anchor density compared to APIT, ROCRSSI, and DRLS. HSL is based on the geometric shape of half-symmetric lens which can be simply drawn using only the location information of two anchors. The presence tests consist in checking whether a sensor is within the symmetric lens of different combinations of two neighboring anchors. To resolve the problem of unlocalizable sensors, HSL uses the Voronoi diagram to divide the network into a set of cells, and locates each sensor within one Voronoi cell. This allows an unlocalizable sensor to use the negative information to narrow-down its residence area, initially defined as the Voronoi cell where it resides in. Simulation results have shown that HSL offers good performance. HSL and DRLS both have similar performance in terms of locatable ratio. In the other cases, HSL exhibits the best results, it outperforms APIT, ROCRSSI, and DRLS in terms of location accuracy and locatable ratio.

References

[1] C. Bo, D. Ren, S. Tang, X.-Y. Li, X. Mao, Q. Huang, L. Mo, Z. Jiang, Y. Sun, Y. Liu *et al.*, "Locating sensors in the forest: A case study in greenorbs," in *INFOCOM*, 2012.

[2] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. F. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *MOBICOM*, D. B. Johnson, A. D. Joseph, and N. H. Vaidya, Eds. ACM, 2003, pp. 81–95.

[3] T. Kim, M. Son, W. Choi, M. Song, and H. Choo, "Low-cost two-hop anchor node-based distributed range-free localization in wireless sensor networks," in *ICCSA (3)*, 2010, pp. 129–141.

[4] C. Liu, T. Scott, K. Wu, and D. Hoffman, "Range-free sensor localisation with ring overlapping based on comparison of received signal strength indicator," *International Journal of Sensor Networks*, vol. 2, no. 5/6, pp. 399–413, 2007.

[5] D. Pedoe, *Circles: A Mathematical View (Spectrum)*, 2nd ed. The Mathematical Association of America, 1997.

[6] F. P. Preparata and M. I. Shamos, *Computational geometry: an introduction*. New York, NY, USA: Springer-Verlag New York, Inc., 1985.

[7] J.-P. Sheu, P.-C. Chen, and C.-S. Hsu, "A distributed localization scheme for wireless sensor networks with improved grid-scan and vector-based refinement," *IEEE Transactions on Mobile Computing*, vol. 7, no. 9, pp. 1110–1123, sep 2008.

[8] J.-P. Sheu, J.-M. Li, and C.-S. Hsu, "A distributed location estimating algorithm for wireless sensor networks," in *SUTC (1)*. IEEE Computer Society, 2006, pp. 218–225.

[9] Z. Yan, Y. Chang, Z. Shen, and Y. Zhang, "A grid-scan localization algorithm for wireless sensor network," in *CMC (2)*. IEEE Computer Society, 2009, pp. 142–146.

[10] Z. Zhong and T. He, "Rsd: A metric for achieving range-free localization beyond connectivity," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 11, pp. 1943–1951, 2011.