ORIGINAL RESEARCH

Geometric and decentralized approach for localization in wireless sensor network

Wahiba Larbi-Mezeghrane[1] · Ali Larbi[1] · Louiza Bouallouche-Medjkoune[1] · Djamil Aissani[1]

Abstract
The main function of a sensor node is to collect data from its environment and forward it to base station. In the absence of further information concerning their locations, those data will be unnecessary. Hence, developing algorithms for localizing all nodes of wireless sensor network is extremely important. We present in this paper, a new approach to determine geographical coordinates of unknown nodes, by using mobile anchor. The mobile anchor adopts a spiral trajectory, and diffuses its position periodically during its travel. The proposed approach uses Received Signal Strength Indicator to estimate distance with all broadcast messages received from mobile anchor. To calculate position, our approach determines a cloud of points that surround the solution; these points are selected from the set of intersection points of all beacons received by unknown node, by considering some constraints. The estimated position of unknown node represents the geometric center of this cloud. The behavior of our algorithm was studied by varying some metrics; the average error was minimized compared to those proposed in literature.

1 Introduction

The random deployment of sensors in area of interest generates non-uniform distribution of sensors, but also non-identification of their positions. Equipping nodes with GPS module is a very expensive solution and inefficient in some areas (Karl and Willig 2006; Mao et al. 2007; Kulaib et al. 2011). Hence, research for better solutions to determine nodes position without using GPS is a real challenge for researchers, these solutions should be implemented with low cost, low localization error, energy efficiency and a coverage of all nodes of the network. Localization represents solutions and techniques proposed to define position of all nodes belonging to WSN after their deployment (Sichitiu and Ramadurai 2004). Localization is very important, especially, to know the position of sensor signaling the appearance of an event, and trigger a quick action if necessary, or to control the coverage of area of interest. In order to locate different nodes of network, solutions proposed in literature are essentially based on the use of anchors (beacon, landmark, or seed) which represent nodes whose positions are known, either by the use of GPS module or by manual location (Mao et al. 2007) (Chelouah et al. 2017). These solutions are classified into two large families, according to whether they are based on the estimation of distances (angles) or not, called respectively: range based methods and range free methods (Nazir et al. 2012; Mistry and Mistry 2015; Zhang et al. 2018; Sheltami et al. 2017).

Among the numerous techniques proposed in literature belonging to Range based family, we quote: RSSI, AoA, ToA, TDoA (Yang and Wu 2015). In RSSI (Received Signal Strength Indicator), attenuation of signal during its propagation is exploited to calculate the distance between nodes. This method is characterized by its low computational complexity, and its low cost since no additional hardware investment is required. AoA (Angle of Arrival) consists

✉ Wahiba Larbi-Mezeghrane
 w.mezeghrane@mail.com

 Ali Larbi
 larbiali2@mail.com

 Louiza Bouallouche-Medjkoune
 Louiza_medjkoune@yahoo.fr

 Djamil Aissani
 djamil_aissani@hotmail.com

[1] Research Unit LaMOS, "Modeling and Optimization of Systems" Bejaia University, Targa Ouzemmou, 06000 Bejaïa, Algeria

in calculating the direction (angle) between two nodes. This approach is characterized by an additional hardware required (several spatially separated microphones and a speaker). ToA (Time of Arrival) is based on the calculation of signal reception time to estimate the distance between the transmitter and the receiver, which must be synchronized. TDoA (Time Difference of Arrival) requires ultra sound or audible frequencies in addition to radio signal. This method is known for its accuracy but it is very costly. Among the techniques proposed belonging to Range free methods, we quote: Centroid, APIT, DV HOP (He et al. 2011). The first method is based on connectivity to define the polygon composed of beacons belonging to unknown node communication range. The geometric center of this polygon represents its estimated position. The accuracy of this technique is related to density and distribution of anchors. In APIT (Approximate Point-In-Triangle), unknown node determines all possible combinations of triangles that can be formed by its proximate anchor nodes; the estimated node location is the center of gravity of all triangles where the node resides. DV HOP technique uses a Distance Vector exchange, each node of network obtains the distance separating them from anchors in number of hops, and exchanges updates with its neighbors only. A landmark estimates an average size of one hop per correction, and then uses trilateration to calculate its position.

In order to reduce the number of expensive GPS equipped node, and to address a problem of bad deployment of anchors, mobile anchor seems to be a good solution (Halder and Ghosal 2016). Mobile anchor is a node which knows its own position, by equipped it with GPS for example, and is able to traverse area of interest for assisting deployed sensors to determine their locations. A fundamental research issue of localization based on mobile anchor is to determine the trajectory of mobile anchor, which could be either static or dynamic. In static path planning scheme, the trajectory is known before start the movement of mobile, where as in dynamic path planning scheme, the trajectory is determined during the displacement considering the real distribution of unknown node. Many trajectories are proposed such as: CIRCLE, SCAN, Double SCAN, Spiral, Hilbert, Z-curve (Koutsonikolas et al. 2006; Yue et al. 2015; Hu et al. 2008; Rezazadeh et al. 2015; Bala Subramanian et al. 2020).

The works dealing with localization problems and other issues relating to WSN with decentralized methods have proved their efficiency comparing to centralized approaches, namely in terms of energy, time and communication flow (Mitici et al. 2014; Al-Hattab et al. 2017; Cascone et al. 2010; D'Apice et al. 2011).

The main contribution of this paper is to find locations of all sensors of the network with estimated positions as close as possible to the real positions; the method is implemented in a decentralized way. To do so, we have opted to use only one mobile anchor moving around the region of interest with spiral trajectory, in order to avoid having unlocalized sensors due to a low number of static anchors and / or their bad deployment. The first step of our approach consist on periodic diffusions of the mobile anchor positions during its travel. Next, each unknown node uses RSSI to estimate its distance to all mobile anchor positions from which it received a broadcast message. Thereafter, the estimated position is calculated by combining multilateration and the geometric center of the cloud of point that surround the real position of the unknown node, by considering some constraints.

The rest of this paper is organized as follows: Problem statement and related concepts are presented in Sect. 2. Section 3 describes in detail our contribution. Finally, in Sect. 4, we compare its efficiency to other existing algorithms using simulation.

2 Problem formulation

The present work is an extension of further work reported in Larbi-Mezeghrane et al. (2017), which consists in monitoring the perimeter P of a region of interest with circular shape of radius R. All sensors are deployed randomly around perimeter, over a range of $[R - Depl\_Ring/2, R + Depl\_Ring/2]$ meters (abbreviations are defined in Table 1). In the previous work, positions of sensors are assumed to be known, to focus only on the coverage issue. In this work, the goal is to define (x, y) coordinates of each sensor in WSN. The proposed approach consists of using a single moving anchor that adopts a spiral trajectory and diffusing its position periodically. We believe that this trajectory is more suited to our deployment zone presented above. The proposed approach determines the cloud of points that surround the solution; these points are selected from the set of intersection points of all beacons received by the unknown node, by selecting only points that contribute to define the minimum weight of the cloud of points. The estimated position of the unknown node represents the geometric center of this cloud.

2.1 List of the used abbreviations

The list of abbreviations used in this paper is detailed in Table 1.

2.2 Basic model and assumptions

Our approach relies on the following assumptions:

- Area of Interest is a region with circular shape of radius *R* and perimeter *P*.

Table 1 List of the used abbreviations

Abbreviations	Significations
$Coord(N_i)$	Coordinate of N_i
$NofNode$	Number of deployed nodes
R	Radius of the region of interest
Cc	Center of the region of interest
P	Perimeter to be covered
$Depl\_Ring$	Ring width of deployment zone
Br_{min}	Minimum radius of $Depl\_Ring$
Br_{max}	Maximum radius of $Depl\_Ring$
$NumBeacRec(i)$	Number of message beacons received by node N_i
$NbBroadcast$	Number of broadcasts achieved by mobile anchor
$d(i, j)$	Distance between node N_i and the broadcast position B_j
C_j	Central point of the broadcast position B_j
$Circle(Cj, d(i, j))$	Circle centered in C_j with radius the distance $d(i, j)$,
$Couple_k(IP_1, IP_2)$	The two intersection points IP_1, IP_2 of the kth combination of $Circle(Cj, d(i, j))$ two-by-two
$Distance_k(IP_1, IP_2)$	Euclidean distance between two points IP_1 and IP_2 of the kth combination.
$Couple_{Max}(Cand_1, Cand_2)$	$Couple_k(IP_1, IP_2)$ with greater distance between IP_1 and IP_2
$MatDist(Cand_l)$	Distance matrix of from point $Cand_l$ to all $Couple_k(IP_1, IP_2)$ with $l = \overline{1, 2}$.

- Nodes are deployed randomly around perimeter P, with ring width of deployment zone over a range of $[R - Depl\_Ring/2, R + Depl\_Ring/2]$ meters.
- Each sensor has a detection module, a communication module and a processing module.
- We assume that all nodes have the same communication radius.
- Once sensors are deployed, they are supposed to be static.
- A protocol that supports all necessary informations for a communication within the network is assumed to be present.
- There is only one mobile anchor moving around the region of interest; we assume that it has sufficient energy for moving and broadcasting position informations during the localization process.
- The emission circles are considered as perfect.
- Mobile anchor adopts a spiral trajectory with constant speed.
- After each time interval t, mobile anchor broadcasts its position.
- The trajectory length depends on the number of broadcasts ($NbBroadcast$).

2.3 Illustration of mobile node trajectory

The mobile anchor moves with spiral trajectory according to formulas 1 and 2 :

$$r_n = Br_{min} + n * \Delta d \tag{1}$$

where:

r_n: radius of nth broadcast ;
Br_{min} :   minimum radius of $Depl\_Ring$;
Δd :   variation in radius of each broadcast, it is given by:

$$\Delta d = (Br_{max} - Br_{min})/NbBroadcast \tag{2}$$

The angle turned is given by formula 3:

$$\theta_n = \sum_{i=1}^{n} v * t/r_i \tag{3}$$

where:

θ_n: Angle of nth broadcast ;
v: Speed of mobile anchor ;
t: Time inter broadcast.

Figure 1 illustrates sensors distribution in deployment area, mobile anchor trajectory and positions of broadcasting messages with 120 diffusions. The flowchart describing the general scheme of the proposed solution is shown in Fig. 2. Details of each step are presented in algorithms of Sect. 3.

3 Proposed Localization Algorithm

Algorithm 1 will be executed by the mobile anchor, whereas algorithm 2 and 3 will be executed by all nodes of the network (unknown nodes).

Fig. 1 Environment of the proposed localization solution
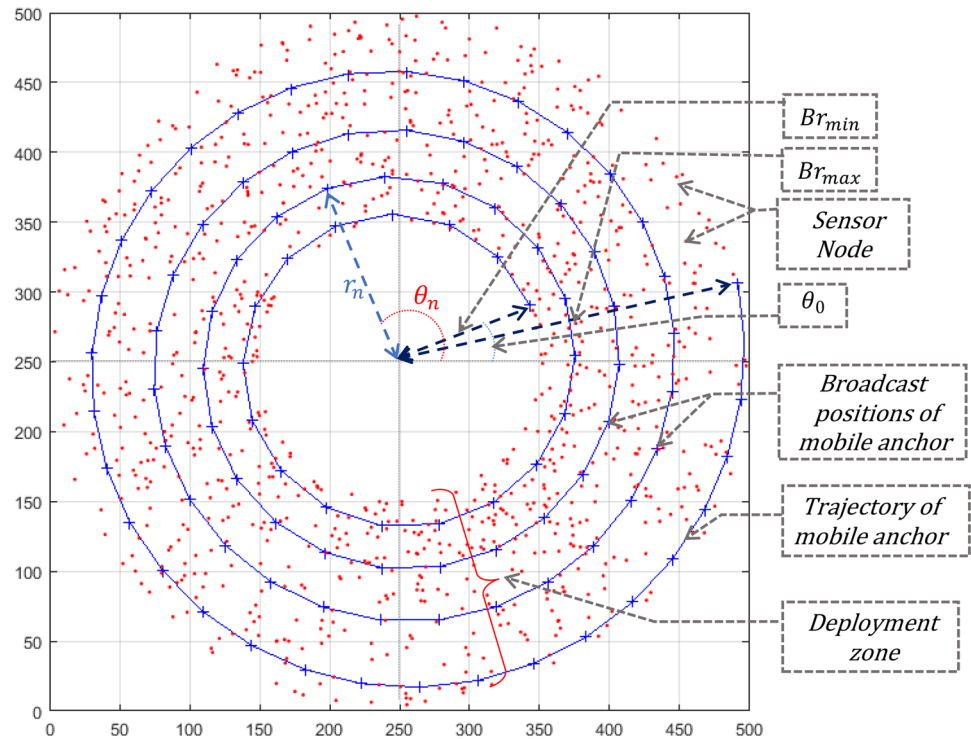
Algorithm 1 Mobile Anchor position broadcasting

Require: Cc, $NofNode$, $Speed$, Br_{min}, Br_{max}, $Nb_{broadcast}$, t.
Ensure: (x_{pos}, y_{pos}) $Broadcast$.
 1: **Initially:** Mobile Anchor (MA) is in position (x_0, y_0) with $\theta_0 = 0, r_0 = Br_{min}$
 2: **for** $i = 1$ to $Nb_{broadcast}$ **do**
 3:      MA moves along the spiral with a constant speed according to formulas 1 and 2;
 4:      After a time interval of t;
 5:      MA gets its position from GPS;
 6:      MA broadcasts its coordinates (x_{pos}, y_{pos});
 7: **end for**

Algorithm 2 Calculating distances by node N_i upon receiving MA coordinates

Require: (x_{pos}, y_{pos}) of MA from B_j .
Ensure: $d(i, j)$ with $i = \overline{1..NofNode}$ and $j = \overline{1..nbrbeacRec(i)}$.
 1: **Begin**
 2:      Save the MA coordinates of B_j;
 3:      Calculating Euclidian distance $d(i, j)$ between N_i and B_j using RSSI Method;
 4:   Save the associated distance $d(i, j)$;
 5: **End**

Fig. 2 Flowchart of the proposed localization solution

Table 2 Positions of broadcasts received by N_i

Broadcast	(x; y) Coordinates of broadcast	Estimated distance / N_i
B_1	(109,59 ; 244,94)	$\hat{d}_1 = 27,40$
B_2	(115,74 ; 203,75)	$\hat{d}_2 = 48,94$
B_3	(76,44 ; 272,40)	$\hat{d}_3 = 34,51$
B_4	(74,56 ; 230,70)	$\hat{d}_4 = 13,10$
B_5	(82,52 ; 189,71)	$\hat{d}_5 = 49,40$

Table 3 Coordinates of the different intersection points

Couple's Number	(x, y) Coordinates of intersection point IP_1	(x, y) Coordinates of intersection point IP_2	Distance Between IP_1 and IP_2
1	(83, 80; 238, 78)	(136, 06; 246, 59)	52, 84
2	(109, 11; 271, 46)	(83, 45; 240, 48)	40, 23
3	(97, 74; 247, 61)	(87, 04; 241, 48)	12, 33
4	(84, 03; 237, 89)	(86, 36; 232, 15)	6, 19
5	(83, 51; 238, 52)	(70, 97; 219, 36)	22, 90
6	(68, 23; 240, 76)	(81, 76; 240, 15)	13, 54
7	(83, 45; 240, 50)	(122, 10; 221, 56)	43, 05
8	**(85, 68;240, 41)**	**(121, 08;156, 64)**	**90, 95**
9	(74, 07; 239, 80)	(83, 55; 240, 50)	9, 50
10	(64, 65; 237, 26)	(81, 29; 240, 50)	16, 96

● : Broadcast Center B_i
× : Intersection points of all circles
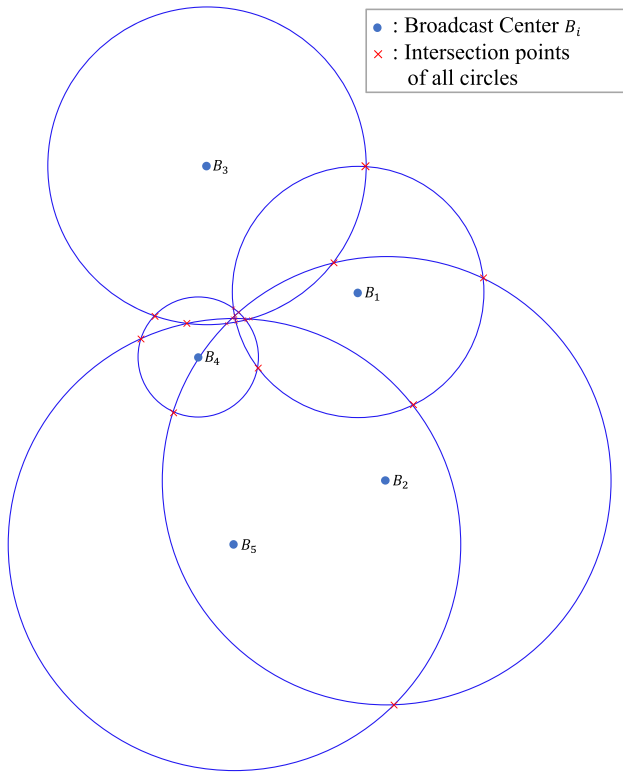
Fig. 3 Geometric illustration of position calculation

Table 4 Coordinates of the selected cloud of points

Intersection point	(x, y) associated Coordinates
IP_1 of couple 1	(83, 80; 238, 78)
IP_2 of couple 2	(83, 45; 240, 48)
IP_2 of couple 3	(87, 04; 241, 48)
IP_1 of couple 4	(84, 03; 237, 89)
IP_1 of couple 5	(83, 51; 238, 52)
IP_2 of couple 6	(81, 76; 240, 15)
IP_1 of couple 7	(83, 45; 240, 50)
IP_2 of couple 9	(83, 55; 240, 50)
IP_2 of couple 10	(81, 29; 240, 50)

Algorithm 3 Position derivation of node N_i

Require: $d(i,j)$ with $i = \overline{1..NofNode}$ and $j = \overline{1..nbrbeacRec(i)}, (x_{pos}, y_{pos})$ Broadcast.
Ensure: $Coord(N_i)$ with $i = \overline{1..NofNode}$.
1: **Begin**
2: **for** each saved position of B_j **do**
3:    Construct $Circle(C_j, d(i,j))$;
4: **end for**
5: Calculate all $Couple_k(IP_1, IP_2)$;
6: Exclude $Circle(C_j, d(i,j))$ with Distance between their centers $\leqslant Dist_{min}$;
7: Select $Couple_{Max}(Cand_1, Cand_2)$;
8: Construct $MatDist(Cand_1)$ and $MatDist(Cand_2)$;
9: Calculate $Weight(Cand_l) = \sum Min\ of\ columns\ of\ MatDist(Cand_l)$, with $l = \overline{1,2}$;
10: Select $Cand_l$ with minimum $Weight(Cand_l)$, with $l = \overline{1,2}$;
11: Construct Cloud of points used for the calculation of Minimum $Weight(Cand_l)$, with $l = \overline{1,2}$;
12: Calculate $Coord(N_i)=$ Geometric center of the cloud of points;
13: **End**

To better understand the functioning of our solution, we present an example explaining step by step how an unknow node determines its position, by executing our algorithm.

Let N_i an unknow node, and let B_1, B_2, B_3, B_4, B_5 be the broadcasts received by N_i (Outputs of algorithm 1).

Node N_i starts with saving broadcasts positions, and then calculates all distances separating it to each diffusion (Lines 3, 4 of algorithm 2). This allows us to draw up the table 2.

From calculated distances (Outputs of algorithm 2) and send positions of beacon message (Outputs of algorithm 1), we construct following circles (Lines 2-4 of algorithm 3):

- $Circle\left(B_1, \widehat{d_1}\right)$
- $Circle\left(B_2, \widehat{d_2}\right)$
- $Circle\left(B_3, \widehat{d_3}\right)$
- $Circle\left(B_4, \widehat{d_4}\right)$
- $Circle\left(B_5, \widehat{d_5}\right)$

Figure 3 illustrates the different constructed circles, next we build the Table 3 from circles previously defined by treating the lines 5–7 of algorithm 3. We seek for pairs of intersection points $(Couple_k(IP_1, IP_2))$ of all circles two by two (the abbreviations are explained in Table 1), by excluding combinations of circles having distance between their centers less than 10 m. Given that no distance is less than 10 m, all couples will be kept. Subsequently, we calculate the distance between intersection points composing each pair. The pair with the largest distance will be then selected as the candidate point (pair highlighted in bold)

$Couple_{Max}(Cand_1, Cand_2) = Couple_k(IP_1, IP_2)$ with distance between IP_1 and IP_2 is the greatest one (couple number
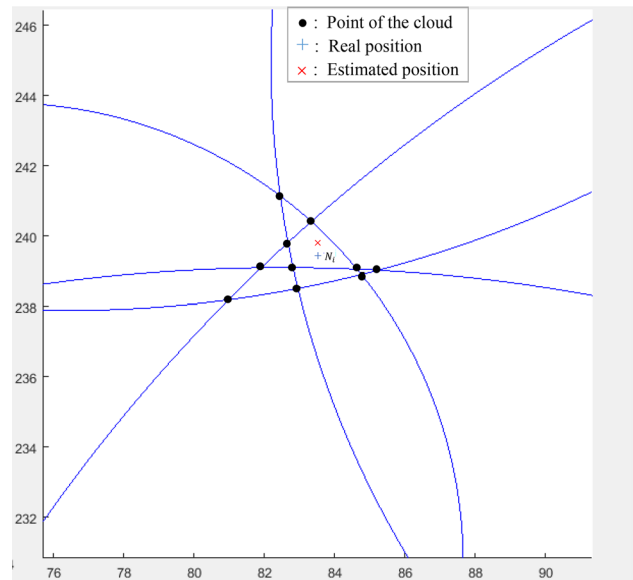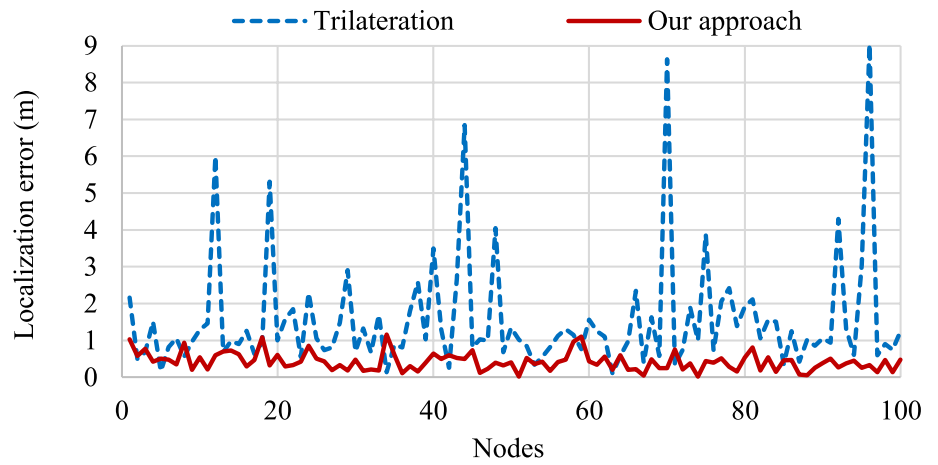

**Fig. 4** Geometric illustration of the cloud of point

Table 5 Setting of simulation parameters

| Parameter	Value
Size of the region	$500 \times 500\ (m^2)$
Ccx (X coordinate of center of region)	$250\ (m)$
Ccy (Y coordinate of center of region)	$250\ (m)$
Depl_Ring	$150\ (m)$
R	$125\ (m)$
NofNode	$100 - 1000$
Br_{min}	$100\ (m)$
Br_{max}	$250\ (m)$
$Dist_{min}$	$10\ (m)$

Fig. 5 Variation of localization errors for the first 100 sensor nodes

Fig. 6 Average localization error vs number of broadcasts
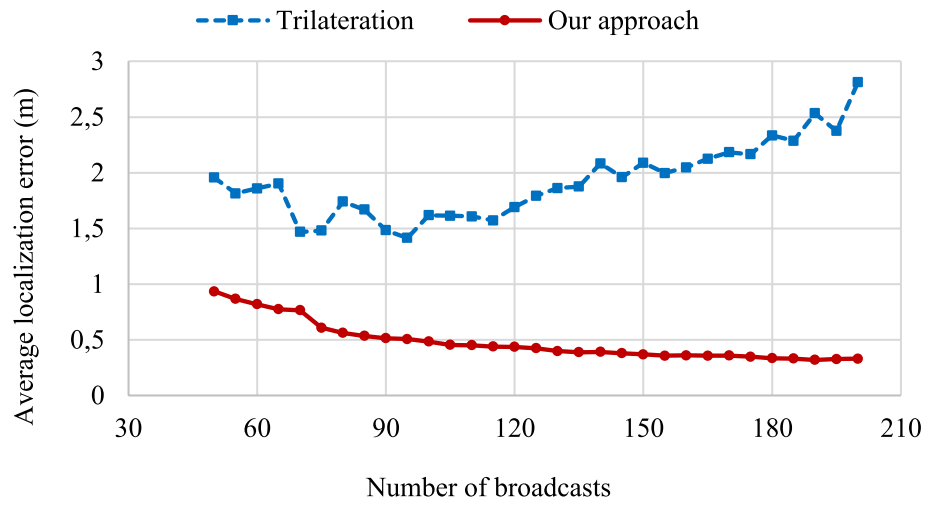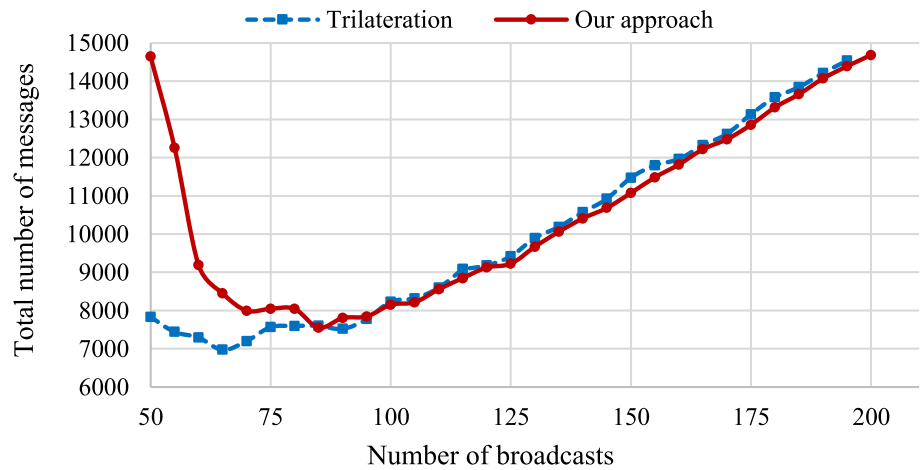

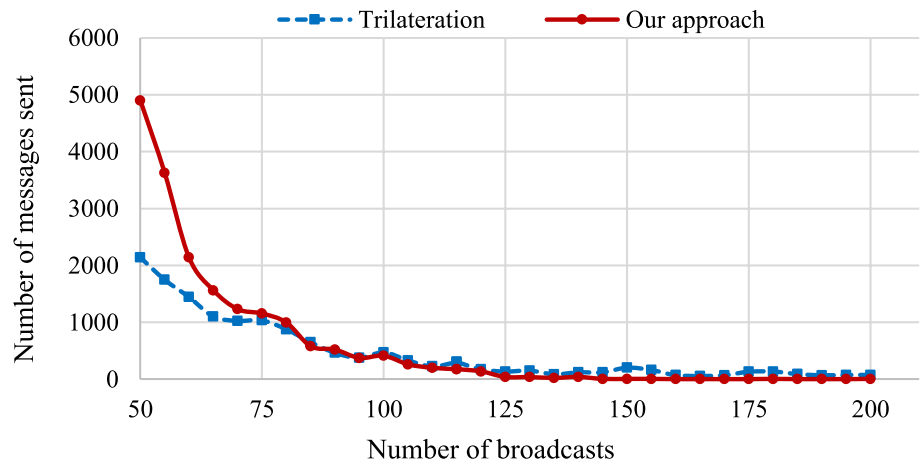**Fig. 7** Number of pseudo anchors participating in localization
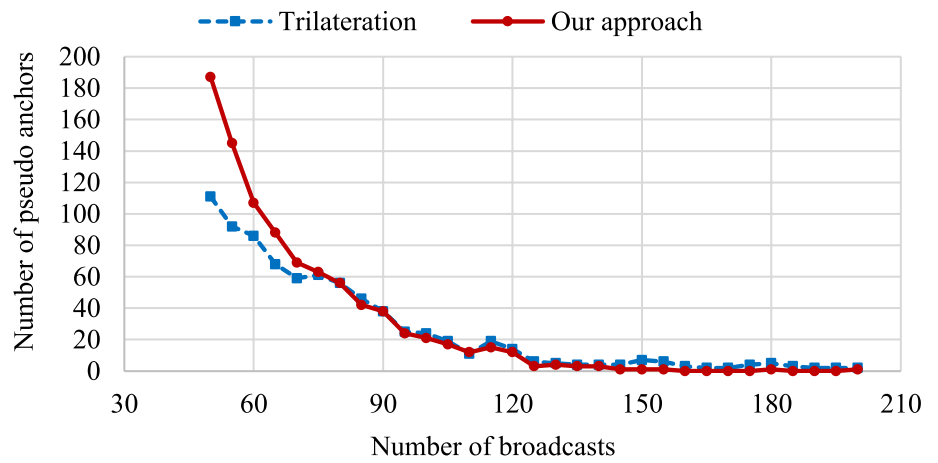
Fig. 8 Total number of messages exchanged vs Number of broadcasts

Fig. 9 Number of messages sent vs Number of broadcasts

8 in Table 3). Consequently, the first candidate point, $Cand_1$ is the point whose coordinates are $(85, 68; 240, 41)$, and the second candidate point $Cand_2$ is the point whose coordinates are $(121, 08; 156, 64)$

The distance matrix $MatDist(Cand_1)$ associated to the first candidate point is as follows: (by treating the line 8 of algorithm 3)

$$MatDist(Cand_1) = \begin{pmatrix} 2,49 & 38,90 & 14,04 \\ 3,02 & 2,88 & 17,45 \\ 2,24 & 11,62 & 21,27 \\ 50,76 & 2,23 & 1,73 \\ 8,29 & 25,68 & 3,93 \\ 41,01 & 2,13 & 4,39 \end{pmatrix}$$

The weight of candidate point $Cand_1$, denoted as $Weight(Cand_1)$, is equal to the sum of minimums of columns of matrix $MatDist(Cand_1)$. It is calculated as follows : (by treating the line 9 of algorithm 3)

$Weight(Cand_1) = 2, 49 + 2, 23 + 1, 73 + 3, 02 + 2, 88$
$+3, 93 + 2, 24 + 2, 13 + 4, 39$

$Weight(Cand_1) = 25, 03$

Similarly:

$$MatDist(Cand_2) = \begin{pmatrix} 90,21 & 115,45 & 93,92 \\ 89,30 & 90,09 & 99,35 \\ 91,92 & 95,53 & 98,42 \\ 91,19 & 91,9 & 91,42 \\ 83,12 & 80,28 & 92,3 \\ 64,93 & 91,88 & 92,82 \end{pmatrix}$$

$Weight(Cand_2) = 90, 21 + 91, 9 + 91, 42 + 83,$
$\qquad 12 + 80, 28 + 92, 3 + 64, 93$
$\qquad + 91, 88 + 92, 82$

$Weight(Cand_2) = 778, 87$

We notice that $Weight(Cand_1) < Weight(Cand_2)$, consequently $Cand_1$ will be selected for the creation of the point cloud. This cloud will be composed of points represented on Table 4. (by treating the line 10 and 11 of algorithm 3)

The geometric center is obtained as follows: (the last step of our algorithm)

Fig. 10 Number of messages received vs Number of broadcasts
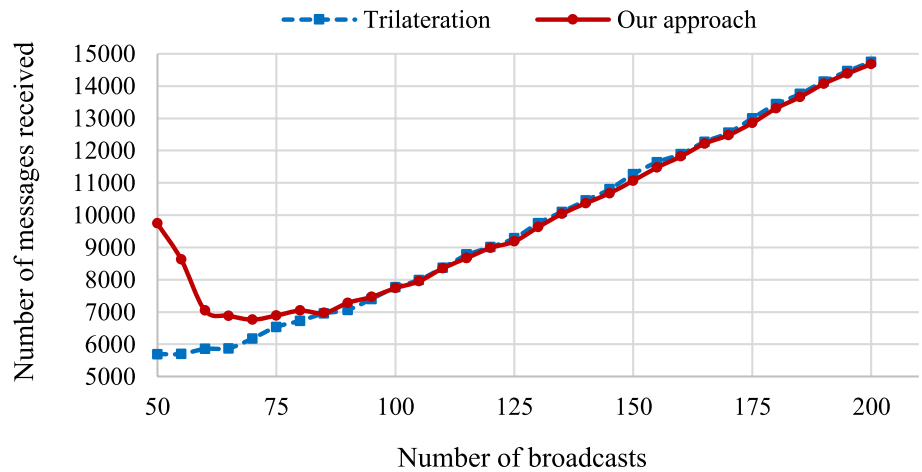
Fig. 11 Average location error vs Communication range
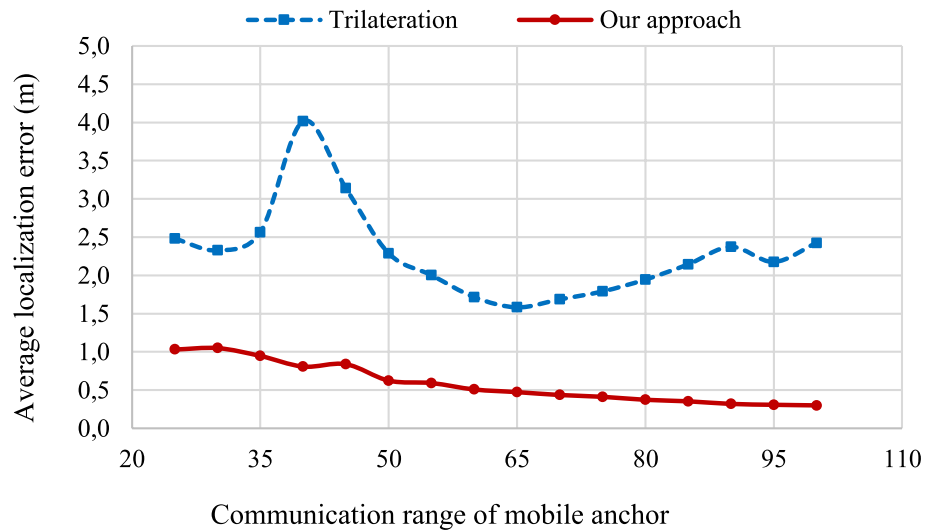
$\hat{x} = (83,8 + 83,45 + 87,04 + 84,03 + 83,51$
$\qquad + 81,76 + 83,45 + 83,55 + 81,29)/9$

$\hat{x} = 83,54$

$\hat{y} = (238,78 + 240,48 + 241,48 + 237,89$
$\qquad + 238,52 + 240,15 + 240,5 + 240,5 + 240,5)/9$

$\hat{y} = 239,87$
$(\hat{x}, \hat{y}) = (83,54 ; 239,87)$
$(\hat{x}, \hat{y})$ represents the estimated coordinates of the position of nœud N_i as shown in the figure 4.

4 Simulation

To evaluate the performance of our proposed approach, the simulation has been implemented using Matlab. We have used the same parameters as those reported in Larbi-Mezeghrane et al. (2017). The detailed experimental parameters are set as Table 5 shows.
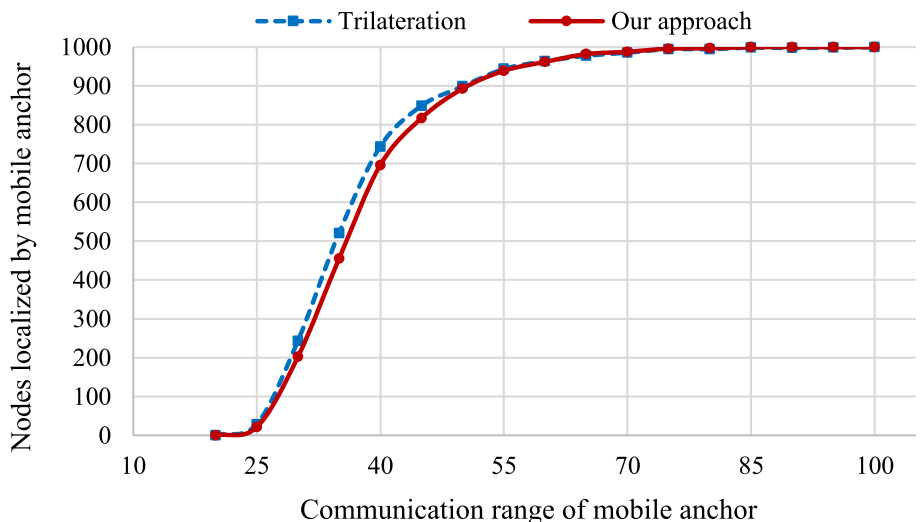
We have compared our approach with trilateration method, both approaches calculate distance with RSSI, we have assumed that all RSSI measurement errors were independent zero-mean, unit-variance Gaussian random variables. In order to ensure a fair comparison, both approaches have been applied on same set of RSSI distance obtained for each node.

Figure 5 compares localization errors obtained by the two approaches for different sensor nodes of network. For the sake of clarity and readability in reading the graph, we reproduced the results of first 100 nodes only. We can clearly observe that our approach gives better results in terms of localization errors for practically all nodes. This can be explained by the fact of associating the multilateration, by exploiting all the received broadcast messages, and the constraints defined for the creation of the cloud of points, as

Fig. 12 Average location error vs Number of nodes
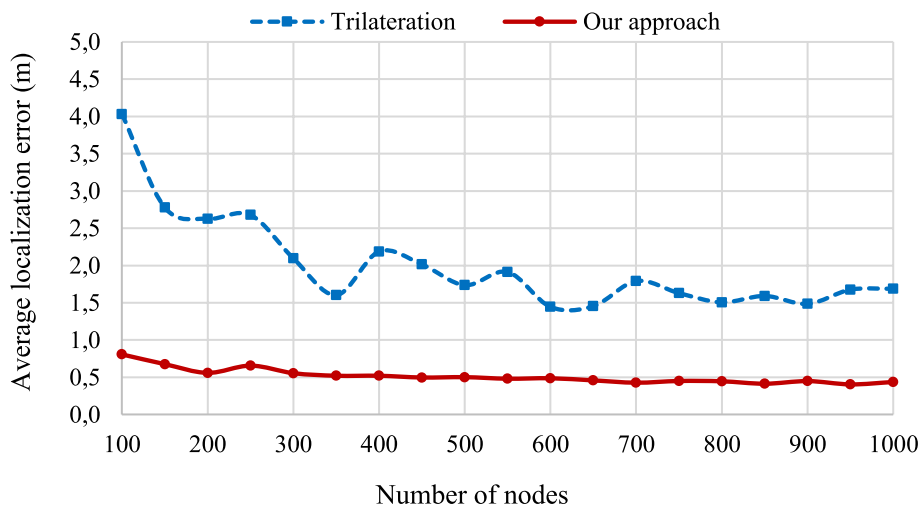
Fig. 13 Number of localized nodes vs Communication range

well as the geometric center makes possible to determine an estimated position very close to the real position.

Figure 6 shows the average localization error (ALE) obtained by our approach comparing with trilateration, by varying the number of broadcasts. It is clear that our approach reduces considerably ALE and has a coherent behavior by augmenting the number of broadcast. By contrary, results obtained by trilateration were weaker with an unstable variation. We also notice the growing increase of ALE with the number of diffusions. This could be explained by the increasing of the number of broadcasts which makes the associated positions very close to each other and tends to make its collinear. This has a negative effect on the estimation of the position. Contrary, in our approach, a bad ALE is obtained when mobile anchor broadcasts a small number of positional messages, but when the number of diffusion increases ALE decreases. This comes down to the multilateration that allows us to exploit all the broadcasts received,

in order to make the cloud of points denser and geometric center closer to the real position. But also to the constraints added in our algorithm to avoid the colinearity of broadcasts positions. When the number of broadcasts is not sufficient to localize all nodes of the network, our approach relies on neighbour nodes that were already localized (we name them pseudo anchor) to help unknown nodes for calculating their positions. This scenario solves a problem of unlocalized nodes, but the position is obtained with error accumulation. This is supported by Fig. 7, which illustrates the number of pseudo anchors participating in localization. We notice that this number is bigger in the first simulations and vanishes after a given number of broadcasts ($\sim$ 125 broadcasts), the numbers coincide in the last two figures.

Figure 8 compares both approaches to messages exchanged. We notice that when the number of broadcasts is small, our approach exchanges an important number of messages comparing to trilateration. This might be explained

by the fact that when a node does not receives enough messages enabling him to calculate his position, trilateration requests at least tree neighbourds. On the other hand, with our approach, the node requests all its neighbors already localized, in order to obtain an estimated position very close to the real position. However, when the number of broadcast increases, unlocalized nodes have to contact their neighbours but use only mobile anchor broadcasts, which is reflected by a linear variation of messages exchanged with the number of broadcasts. Figures 9 and 10 confirm our interpretation in terms of messages exchanged in the network, by separating received messages from those sent. The number of sent messages is very important when the number of diffusion is small, then decrements with the increase of the number of diffusion until its cancellation. As for messages received, they follow the same pace as the messages sent when the number of broadcast is low, thereafter their number varies proportionally with the number of broadcasts.

Figure 11 shows the variation of ALE according to the variation of communication radius of mobile anchor. We note that the results achieved by trilateration method vary with non-uniform way which is difficult to interpret, this stems from the fact that to determine the position of unknow nodes, trilateration method relies on three beacon messages only whatever the position of their sending, since no pre-selection is made. Conversely, with our approach ALE is reduced by increasing the communication range of the mobile anchor. This amounts to the number of nodes receiving the same broadcast that results in the construction of a denser cloud of points, and, consequently, an estimated position closer to the real position.

We note in Fig. 12 that the ALE obtained by our approach is pratically constant (~ 0.5 m) this means that ALE does not vary with the increase of the number of nodes in network. It can be concluded that our approach gives good results even in a dense network.

Figure 13 illustrates the relationship between the number of nodes localized by the mobile anchor by varying its communication range. We note that for a communication range of less than 25 m no node could be localized by any of the two approaches, but when the communication range increases (between 25 and 55 m) the number of localized nodes increases considerably. We notice that trilateration localizes more nodes compared to our approach, but for the same interval, Fig. 11 shows that the average localization error for trilateration reaches its maximum. In other hand, with our approach, certainly fewer nodes are localized but the accuracy is better (see Fig. 11). When the communication range takes values greater than 60 m, all the nodes of the network are localized by both approaches

5 Conclusion

In this paper, we have studied the problem of localization in WSN; we have proposed an approach by combining two methods, multilateration by exploiting all broadcasting messages to create the cloud of points, and geometric center to select the closer point of the real position. Mobile anchor with spiral trajectory and sufficient broadcasting messages enables all nodes to get enough information to estimate their positions effectively. The approach reduces significantly localization average error, as shown in simulation results. In the present work, few performance parameters are studied, but in the future work, we aim to study more performance parameters and compare the results with other works. It is also interesting to consider the behavior of our approach in real environments with limited resources.

References

Al-Hattab M, Takruri M, Attia H, Al-Omari H (2017) Decentralized localization in wireless sensor networks. In: 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp 1–5, https://doi.org/10.1109/ICECTA.2017.8252071

Bala Subramanian C, Maragatharajan M, Balakannan SP (2020) Inventive approach of path planning mechanism for mobile anchors in wsn. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-020-01752-2

Cascone A, Marigo A, Piccoli B, Rarit L (2010) Decentralized optimal routing for packets flow on data networks. Discrete Contin Dyn Ser B (DCDS-B) 13(1):59–78, https://doi.org/10.3934/dcdsb.2010.13.59

Chelouah L, Semchedine F, Bouallouche-Medjkoune L (2017) Localization protocols for mobile wireless sensor networks: a survey. Comput Electric Eng. https://doi.org/10.1016/j.compeleceng.2017.03.024

D'Apice C, Manzo R, Raritã ÂL (2011) Splitting of traffic flows to control congestion in special events. Int J Math Math Sci https://doi.org/10.1155/2011/563171 (**Article ID 563171):18 pages**)

Halder S, Ghosal A (2016) A survey on mobile anchor assisted localization techniques in wireless sensor networks. Wirel Netw 22(7):2317–2336. https://doi.org/10.1007/s11276-015-1101-2

He Q, Chen F, Cai S, Hao J, Liu Z (2011) An efficient range-free localization algorithm for wireless sensor networks. Sci China Technol Sci 54(5):1053–1060. https://doi.org/10.1007/s11431-011-4351-y

Hu Z, Gu D, Song Z, Li H (2008) Localization in wireless sensor networks using a mobile anchor node. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp 602–607, https://doi.org/10.1109/AIM.2008.4601728

Karl H, Willig A (2006) Localization and positioning. Wiley-Blackwell, Hoboken, pp 231–249. https://doi.org/10.1002/0470095121.ch9 (**chap 9**)

Koutsonikolas D, Das SM, Hu YC (2006) Path planning of mobile landmarks for localization in wireless sensor networks. In: 26th IEEE International Conference on Distributed Computing

Systems Workshops (ICDCSW'06), pp 86–86, https://doi.org/10.1109/ICDCSW.2006.82

Kulaib AR, Shubair RM, Al-Qutayri MA, Ng JWP (2011) An overview of localization techniques for wireless sensor networks. In: 2011 International Conference on innovations in information technology, pp 167–172, https://doi.org/10.1109/INNOVATIONS.2011.5893810

Larbi-Mezeghrane W, Bouallouche-Medjkoune L, Larbi A (2017) Minimum perimeter coverage set based on points of tangency and strong barrier for an extended wsn lifetime. Wirel Pers Commun 97(2):2339–2358. https://doi.org/10.1007/s11277-017-4611-7

Mao G, Fidan B, Anderson BDO (2007) Wireless sensor network localization techniques. Comput Netw 51(10):2529–2553. https://doi.org/10.1016/j.comnet.2006.11.018

Mistry HP, Mistry NH (2015) Rssi based localization scheme in wireless sensor networks: A survey. In: Fifth International Conference on Advanced Computing Communication Technologies, pp 647–652, https://doi.org/10.1109/ACCT.2015.105

Mitici M, Goseling J, Graaf M, Boucherie R (2014) Decentralized vs. centralized scheduling in wireless sensor networks for data fusion. pp 5070–5074, https://doi.org/10.1109/ICASSP.2014.6854568

Nazir U, Shahid N, Arshad MA, Raza SH (2012) Classification of localization algorithms for wireless sensor network: A survey. In: International Conference on Open Source Systems and Technologies, pp 1–5, https://doi.org/10.1109/ICOSST.2012.6472830

Rezazadeh J, Moradi M, Ismail AS, Dutkiewicz E (2015) Impact of static trajectories on localization in wireless sensor networks.

Wirel Netw 21(3):809–827. https://doi.org/10.1007/s11276-014-0821-z

Sheltami TR, Shahra EQ, Shakshuki EM (2017) Perfomance comparison of three localization protocols in wsn using cooja. J Ambient Intell Hum Comput 8(3):373–382. https://doi.org/10.1007/s12652-017-0451-2

Sichitiu ML, Ramadurai V (2004) Localization of wireless sensor networks with a mobile beacon. In: IEEE International Conference on mobile Ad hoc and sensor systems (IEEE Cat. No.04EX975), pp 174–183, https://doi.org/10.1109/MAHSS.2004.1392104

Yang T, Wu X (2015) Accurate location estimation of sensor node using received signal strength measurements. AEU Int J Electron Commun 69(4):765–770. https://doi.org/10.1016/j.aeue.2014.12.007

Yue L, Ding G, Zheng Z (2015) Path planning in sensor localization with mobile anchors: Survey and challenges. In: International Conference on Wireless Communications Signal Processing (WCSP), pp 1–6, https://doi.org/10.1109/WCSP.2015.7341065

Zhang H, Wang Z, Gulliver TA (2018) Two-stage weighted centroid localization for large-scale wireless sensor networks in ambient intelligence environment. J Ambient Intell Hum Comput 9(3):617–627. https://doi.org/10.1007/s12652-017-0458-8