Accurate Range-Free Localization in Multi-Hop Wireless Sensor Networks

Slim Zaidi, *Member, IEEE*, Ahmad El Assaf, *Student Member, IEEE*, Sofiène Affes, *Senior Member, IEEE*, and Nahi Kandil

***Abstract*— To localize the wireless sensor networks nodes, only the hop-based information (i.e., hops' number, average hop size, and so on) has been, so far, exploited by range-free techniques, with poor-accuracy, however. In this paper, we show that localization accuracy may greatly benefit from joint exploitation, at no cost, of the information already provided by the forwarding nodes (i.e., relays) between each anchor (i.e., position aware) and sensor nodes pair. As such, we develop a novel range-free localization algorithm, derive its average location estimation error (LEE) in closed-form, and compare it in LEE performance with the best representative algorithms in the literature. We show that the proposed algorithm outperforms them in accuracy. In contrast to the latter, we further prove that it is able to achieve an LEE average and variance of about 0 when the number of sensors is large enough, thereby achieving an unprecedented accuracy performance among range-free techniques.**

***Index Terms*— Wireless sensor networks (WSN)s, multi-hop, localization, low-cost, location estimation error (LEE).**

I. Introduction

RECENT advances in wireless communications and low-power circuits technologies have led to proliferation of wireless sensor networks (WSNs). A WSN is a set of small and low-cost sensor nodes often equipped with small batteries. The latter are often deployed in a random fashion to sense or collect from the surrounding environments some physical phenomena such as temperature, light, pressure, etc. [1]–[4]. Since power is a scarce resource in such networks, sensors usually recur to multi-hop transmission in order to send their gathered data to an access point (AP). However, the received data at the latter are often fully or partially meaningless if the location from where they have been measured is unknown [5], making the sensors' localization an essential task in multi-hop WSNs. Designed to comply with such networks, many localization algorithms exist in the literature [6]–[30]. To properly localize each sensor, most of these algorithms require the distance between the latter and at least three position-aware nodes called hereafter anchors.[1] Since it is very likely in multi-hop WSNs that some sensors be unable to directly communicate with all anchors, the distance between each anchor-sensor pair is usually estimated using their shortest path. This distance is in fact approximated by the sum of the distances between any consecutive intermediate nodes located on the shortest path between the two nodes. The localization algorithms based on such an approximation are commonly known in the literature as connectivity-based or range-free algorithms [6]–[30]. Depending on the process used to estimate the distances between the intermediate nodes, range-free algorithms may fall into three categories: measurement-based, heuristic, and analytical [6]–[30].

Measurement-based algorithms exploit the measurements of the received signals' characteristics such as the received signal strength (RSS) [6] or the time of arrival (ToA) [7], etc. Using the RSS measurement, the distance between any sensors' pair could be obtained by converting the power loss due to propagation from a sensor to another based on some propagation laws. Unfortunately, due to the likely presence of noise and interference, the distance's estimate would be far from being accurate, thereby leading to unreliable sensor localization accuracy. Using the ToA measurement, both sensor and anchor nodes require high-resolution clocks and extremely accurate synchronization between them.[2] While the first requirement may dramatically increase the cost and the size of every node, the second results in severe depletion of their power due to the additional overhead required by such a process. Furthermore, in the presence of noise and/or multipath, the ToA measurement is severely affected thereby hindering sensors' localization accuracy.

As far as heuristic algorithms [8]–[12] are concerned, most of them are based on variations of the DV-HOP technique [8],

Manuscript received August 11, 2015; revised December 21, 2015, April 1, 2016, and June 14, 2016; accepted June 15, 2016. Date of publication July 12, 2016; date of current version September 14, 2016. This Work was supported by the DG and CREATE PERSWADE <www.create-perswade.ca> Programs of NSERC, a Discovery Accelerator Supplement Award from NSERC, and the Collaborative R&D (CRD) Grants Program of NSERC, Bell Aliant, and Newtrax. The associate editor coordinating the review of this paper and approving it for publication was Y.-C. Wu.

S. Zaidi is with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada (e-mail: slim.zaidi@utoronto.ca).

A. El Assaf, S. Affes, and N. Kandil are with the Centre Énergie Matériaux Télécommunications, Institut National de la Recherche Scientifique, Université du Québec, Montreal, QC H5A 1K6, Canada, and also with the Télébec Underground Communications Research Laboratory, Université du Québec en Abitibi-Témiscamingue, Rouyn-Noranda, QC J9X 5E4, Canada (e-mail: elassaf@emt.inrs.ca; affes@emt.inrs.ca; nahi.kandil@uqat.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCOMM.2016.2590436

[1] In practice, an anchor node refers to a sensor, base station, or a nearby access point (AP) with known position. This information is usually acquired using global positioning system (GPS) technology, configured or manually entered into the node memory prior to deployment.

[2] Please note that advanced ToA-based algorithms known as round-trip (i.e., two-way) ToA algorithms do not require any synchronization between nodes [31],[32]. However, this advantage comes at the cost of additional overhead which becomes prohibitive especially in multi-hop WSNs, making the round-trip ToA algorithms unsuitable for such networks.

whose implementation in multi-hop WSNs requires the computation of the average hop size (i.e., average distance between any two consecutive intermediate nodes) h_{av} to estimate the distance between a sensor and an anchor as $n_h h_{\text{av}}$ where n_h is the number of hops between the two nodes. Such algorithms have, however, a major drawback. Indeed, h_{av} is computed in a non-localized manner and broadcasted in the network by each anchor. This incurs undesired prohibitive overhead and power consumption, thereby increasing the overall cost of the localization process.

More popular alternatives suitable for multi-hop WSNs are the analytical algorithms [13]–[30] which evaluate theoretically h_{av} using the statistical characteristics of the network deployment. The obtained h_{av} is actually locally computable at each regular node, thereby avoiding the unnecessary overhead and power consumption incurred by heuristic techniques if, likewise, it had to be broadcasted in the network. In spite of their valuable contributions, the localization algorithms developed so far in [13]–[30] do not provide unfortunately sufficient accuracy, due to large errors occurred when mapping n_h into distance units. This is primarily caused by the lack of information provided by both h_{av} and n_h. Actually, the distance between an anchor-sensor pair depends not only on the latter hop-based information, but also on the number m of forwarding[3] nodes (i.e., which forward any data between the two nodes). Indeed, when n_h and the total nodes' number are fixed, the distance increases (decreases) if m increases (decreases). Consequently, if this easily-obtained information is taken into account when designing a localization algorithm, its accuracy would definitely be improved.

Hence we propose in this paper, a novel analytical localization algorithm that properly exploits m alongside the hop-based information, derive its average location estimation error (LEE) in closed-form, and compare it in LEE performance with the best representative algorithms in the literature. We show that the proposed algorithm outperforms them in accuracy. In contrast to the latter, we further prove that it is able to achieve a LEE average and variance of about 0 when the number of sensors is large enough, thereby achieving an unprecedented accuracy performance among range-free techniques.

The rest of this paper is organized as follows: Section II describes the system model and discusses the motivation for this work. Section III proposes a new approach aiming to estimate the distance between any anchor-sensor pair. A novel localization algorithm for multi-hop WSNs is introduced in section IV. Its accuracy is analyzed in Section V. Simulation results are discussed in Section VI and concluding remarks are made in Section VII.

II. Network Model and Motivation

Fig. 1 displays the system model of M anchor and N sensor nodes deployed in a 2-D square area S. The anchors[1] are aware of their positions while the sensors are oblivious to this information. These sensors are assumed to be uniformly distributed in S. All anchor and sensor nodes are assumed to have the same range (i.e., transmission capability) denoted by R. Each node is then able to directly communicate with any other node located in the disc having that node as a center and R as a radius, while it communicates in a multi-hop fashion with the nodes located outside it. As shown in Fig. 1, the anchors are marked with red triangles and the sensors are marked with blue discs. If two nodes are able to communicate directly, they are linked with a dashed line that represents one hop. Let (x_i, y_i), $i = 1, \ldots, N$ be the coordinates of the sensors and (a_k, b_k), $k = 1, \ldots, M$ be those of the anchors.

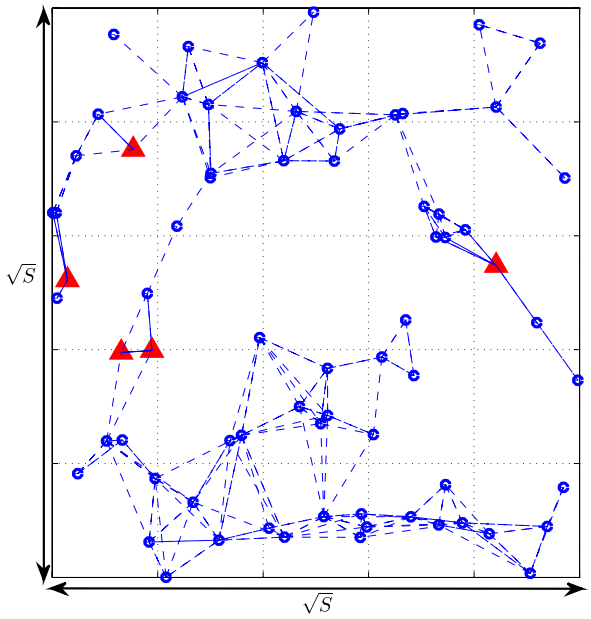


Fig. 1. Network model.

In what follows, we propose an efficient anchor-based localization algorithm aiming to accurately estimate the sensors' positions. Such an algorithm requires that the latter estimate their distances to at least 3 anchors and be aware of their coordinates. The k-th anchor should then broadcast its coordinates (a_k, b_k) through the network. If the i-th sensor is located at a distance less than or equal to R from that anchor, it receives the coordinates in $n_h = 1$ hop. Otherwise, it receives them after $n_h > 1$ hops. So far, in most previous algorithms, the i-th sensor estimates its distance to the k-th anchor d_{i-k} using only the information n_h as

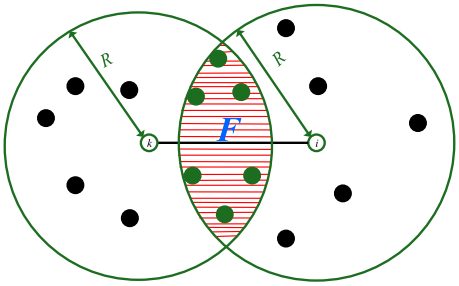
$$\hat{d}_{i-k} = n_h h_{\text{av}}, \tag{1}$$

where h_{av} is a predefined average hop size. Note that this distance estimation (DE) approach relies on the fact that in highly dense WSNs,

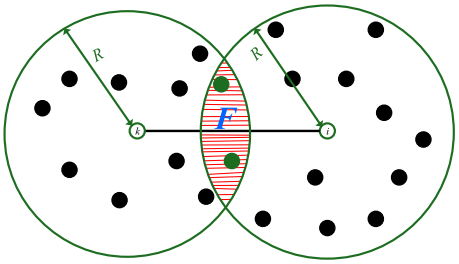
$$d_{i-k} \approx \sum_{l=1}^{n_h} h_l, \tag{2}$$

holds. In (2), h_l is the l-th hop's distance. Unfortunately, this approach exhibits a major drawback. Indeed, h_{av} is usually derived either analytically (i.e., $h_{\text{av}} = \text{E}\{h_l\}$) by exploiting the Poisson Limit Theorem valid for high nodes

[3]Note that a forwarding node between an anchor-sensor pair is a node able to forward any data between the two nodes without being necessary on the shortest path. An intermediate node on this path is then a forwarding node but the reciprocal does not hold true.



(a) Small distance.



(b) Large distance.

Fig. 2. Effect of the distance d_{i-k} on the forwarding area F.

densities [13]–[30] or heuristically by computing the mean hop size of all the shortest paths between anchors as in [8]

$$h_{\text{av}} = \frac{1}{M(M-1)} \sum_{k=1}^{M} \sum_{l=1}^{M} \frac{\sqrt{(a_k - a_l)^2 + (b_k - b_l)^2}}{n_{k,l}}, \quad (3)$$

where $n_{k,l}$ is the number of hops between the k-th and l-th anchors. It is then very likely that h_{av} be different from the mean hop size of the shortest path between the k-th anchor and the i-th sensor (i.e., $h_{\text{av}} \neq \left(\sum_{l=1}^{n_h} h_l\right)/n_h$ and, hence, large DE errors may occur, thereby hindering the i-th sensor's localization accuracy. This motivates us to seek for more efficient DE approach for exploitation by our localization algorithm.

III. Proposed DE Approach

In this work, we propose to exploit, in addition to n_h, another easily obtained information, in order to reduce the distance estimation error, thereby improving the localization accuracy. According to the parity of n_h, we distinguish below between two cases and develop two different approaches suitable for each case.

A. n_h Is Even

For simplicity, let us first assume that $n_h = 2$. Let $D_k(R)$ and $D_i(R)$ be the discs with radius R and having, respectively, the k-th anchor and the i-th sensor as centers. $F = D_k(R) \cap D_i(R)$ is then the forwarding area wherein the forwarding nodes,[4] which forward the messages sent from the k-th anchor to the i-th sensor, are located. An in depth look at this area reveals that it is strongly dependant on d_{i-k}; a fact that could be exploited to estimate the latter. Indeed, as can be

[4]A forwarding node refers to a sensor located in the forwarding area F. Please note that we consider, in this work, that an anchor assists the sensors' localization by only broadcasting its information across the WSN. It is then not involved in forwarding the messages of other anchors.

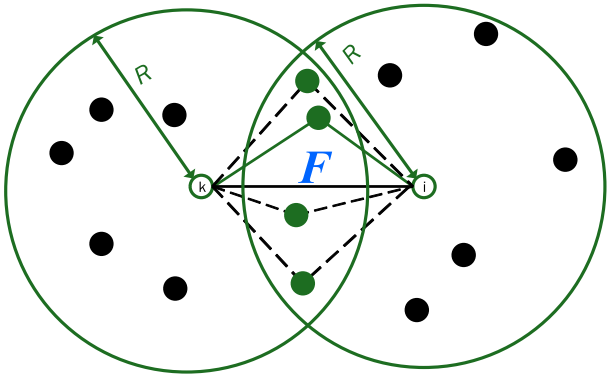


Fig. 3. Two-hop communication.

observed from Fig. 2, if d_{i-k} increases (decreases). then F decreases (increases). Using some geometrical properties and trigonometric transformations, one can even show that

$$F = \Phi(d) = 2R^2 \cos^{-1}\left(\frac{d_{ki}}{2R}\right) - \frac{1}{2} d_{ki} \sqrt{4R^2 - d_{ki}^2}. \quad (4)$$

It follows from (4) that $\Phi(d)$ is a decreasing function of d, which confirms the above observation. As such, computing F is crucial in order to estimate the distance between the k-th anchor and the i-th sensor. From Fig. 3, the latter receives m times the k-th anchor' coordinates, each from a distinct forwarding node. Since nodes are uniformly distributed in S, knowing m, the i-th sensor is able to locally approximate F as $\hat{F} = m/\lambda$ where $\lambda = N/S$ is the WSN density. $\hat{d}_{i-k}$ could then be obtained as

$$\hat{d}_{i-k} = \Psi(\hat{F}), \quad (5)$$

where $\Psi(x) = \Phi^{-1}(x)$ is the inverse function of Φ. Unfortunately, to the best of our knowledge, there is no closed-form expression for $\Psi(x)$. It is then impossible to obtain $\hat{d}_{i-k}$ using (5). In order to circumvent this impediment, a look-up table may be envisaged at each sensor. However, such a table usually requires a large memory space; a scarce resource for these often-primitive devices. Even if it is possible to implement an additional memory space at each node, this would substantially increase the overall cost of the network, especially for large-scale WSNs. Alternatively, one may numerically compute $\hat{d}_{i-k}$. To this end, we propose to equivalently reformulate this problem as a root-finding problem of the function $\tilde{\Phi}(x) = \Phi(x) - \hat{F}$. Many root-finding iterative algorithms already exist in the literature such as Newton-Raphson method, Brent's method, Secant method, etc.. Due to its simplicity, only the latter is of concern in this work. Using the Secant method, $\hat{d}_{i-k}$ is derived by iteratively executing the following instruction:

$$\hat{d}_{i-k}^{p+1} = \hat{d}_{i-k}^{p} - \tilde{\Phi}\left(d_{i-k}^{p}\right) \frac{d_{i-k}^{p} - d_{i-k}^{p-1}}{\Phi\left(d_{i-k}^{p}\right) - \Phi\left(d_{i-k}^{p-1}\right)}, \quad (6)$$

where p refers to the p-th iterations, until convergence (i.e., $p = p^{\max} = \inf_p\{\hat{d}_{i-k}^{p} = \hat{d}_{i-k}^{p+s}, \quad \forall s \in \mathbb{N}^{\star}\}$). From (4), two initial values $\hat{d}_{i-k}^{0}$ and $\hat{d}_{i-k}^{1}$ are required to properly compute $\hat{d}_{i-k} = \hat{d}_{i-k}^{p^{\max}}$. To guarantee fast convergence of the Secant

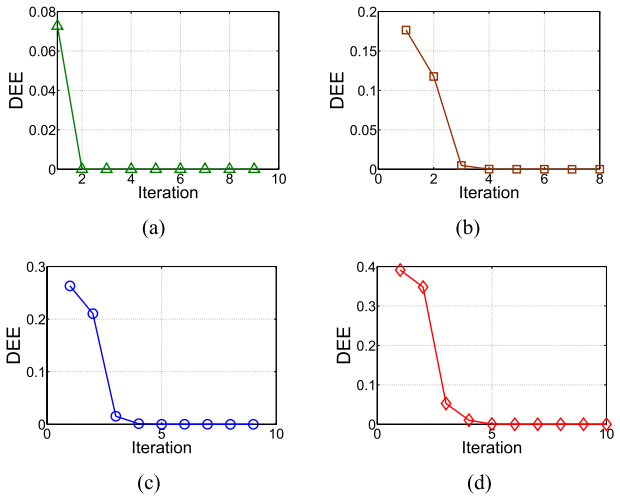


Fig. 4. Distance estimation error (DEE) vs. the number of iterations.

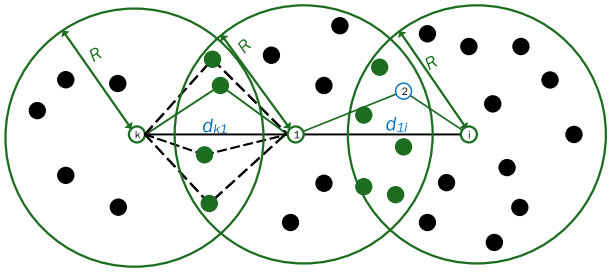


Fig. 5. Four-hop communication.

method, $\hat{d}^0_{i-k}$ and $\hat{d}^1_{i-k}$ must be chosen among the range of possible values of $\hat{d}_{i-k}$ (i.e., [R,2R]). In this work, we opt for $\hat{d}^0_{i-k} = R$ and $\hat{d}^1_{i-k} = 2R$. As can be observed from Fig. 4, using these values, $p^{\max}$ does not exceed 5 iterations. Knowing that the required power to execute one instruction is in the range of 10^{-4} of the power consumed per transmitted bit [33], [34], the power needed to execute the Secant method is then very negligible with respect to the overall power consumed by each sensor. Consequently, the proposed DE approach complies with WSNs where the power is considered as a scarce resource.

Now, let us generalize the proposed DE approach by considering $n_h > 2$. In such a case, d_{i-k} would simply be, as could be observed from Fig. 5, the summation of $n_h/2$ two-hop distances between the k-th anchor and the i-th sensor. $\hat{d}_{i-k}$ is then given by

$$\hat{d}_{i-k} = \sum_{l=1}^{n_h/2} \Psi\left(\frac{m_l}{\lambda}\right), \tag{7}$$

where m_l is the number of forwarding nodes at the l-th 2-hop distance.

B. n_h Is Odd

If n_h is odd, d_{i-k} would be the summation of $(n_h - 1)/2$ 2-hop distances plus the last-hop distance d^{Last}. Using the fact that the minimum square error (MMSE) of the last-hop

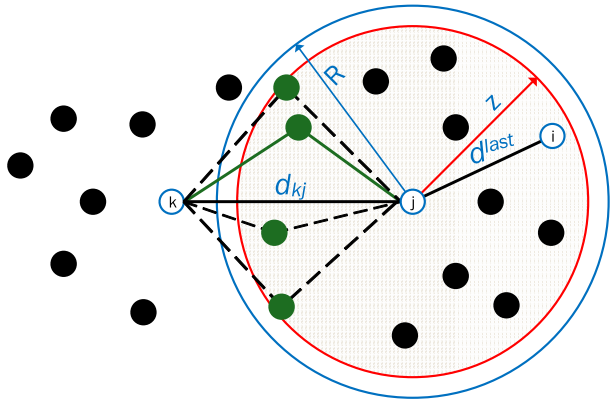


Fig. 6. Last-hop distance estimation.

distance estimation is obtained as $d^{\text{Last}}_{\text{av}} = \text{E}\{d^{\text{Last}}\}$, $\hat{d}_{i-k}$ is given by

$$\hat{d}_{i-k} = \sum_{l=1}^{(n_h-1)/2} \Psi\left(\frac{m_l}{\lambda}\right) + d^{\text{Last}}_{\text{av}}. \tag{8}$$

Now, let us focus on $d^{\text{Last}}_{\text{av}}$. In order to derive it, one should compute the conditional cumulative distribution function (CDF) $F_Z(z) = P\,(Z \le z / Z \le R)$ where, for the sake of clarity, Z refers to the random variable d^{Last}. Actually, as shown in Fig. 6, the probability that the event $\{Z \le z\}$ occurs is the probability that the i-th sensor be in the disc $D_j(z)$ having the j-th sensor as center and z as radius. Therefore, $F_Z(z)$ can be defined as

$$F_Z\,(z) = P\,(\mathcal{A}|\mathcal{B}) = \frac{P\,(\mathcal{A})}{P\,(\mathcal{B})}, \tag{9}$$

where $P\,(\mathcal{A}|\mathcal{B})$ is the probability that the event $\mathcal{A} = \{$the i-th sensor is in the dashed disc $D_j(z)\}$ given $\mathcal{B} = \{$the i-th sensor is in $D_j(R)\}$ occurs. Since the nodes are uniformly distributed in S, we have

$$P\,(\mathcal{A}) = \frac{\pi z^2}{S}, \tag{10}$$

$$P\,(\mathcal{B}) = \frac{\pi R^2}{S}. \tag{11}$$

It follows from (10) and (11) that $F_Z\,(z) = (z/R)^2$ and, hence, the probability density function (PDF) $f_Z(z)$ of Z is given by

$$f_Z\,(z) = \frac{2z}{R^2}. \tag{12}$$

Exploiting (12), we easily show that

$$\hat{d}^{\text{Last}} = \frac{2R}{3}. \tag{13}$$

In what follows, we introduce a new localization algorithm for multi-hop WSN that exploits the proposed DE approach and analytically prove its accuracy. The obtained results will be further verified using Monte Carlo simulations.

IV. Proposed Localization Algorithm

A. Initialization

As a first step of any anchor-based localization algorithm, the k-th anchor broadcasts through the network a packet

which consists of a header followed by a data payload. The packet header contains the anchor position (a_k, b_k), while the data payload contains $(n, \hat{d})$, where n is the hop-count value initialized to one and $\hat{d}$ is the estimated distance initialized to zero. If the packet is successfully received by a node, it stores the k-th anchor position as well as the received hop-count $n_k = n$ in its database, adds one to the hop-count value and broadcasts the resulting message. Once this message is received by the another node, its database information is checked. If the k-th anchor's position does not exist, the node adds the received information to its database and checks the parity of n. If it is odd, the message is broadcasted after incrementing it by 1. Otherwise, the node creates a variable m_k, which represents the number of received packets from the k-th anchor with the same data payload, and initializes it to one. However, if the node is aware of the k-th anchor's coordinates, it compares n and $\hat{d}$ with the stored ones n_k and $\hat{d}_k$, respectively. If $n > n_k$ or $n = n_k$ but $\hat{d} > \hat{d}_k$, the packet is immediately discarded. If $n < n_k$ or $n = n_k$ and $\hat{d} < \hat{d}_k$, the node updates n_k to n and $\hat{d}_k$ to $\hat{d}$. Otherwise, the parity of n is checked. If it is odd, the packet is broadcasted after incrementing it by 1. If not, m_k is incremented by 1. At this stage, a waiting-time τ, before transmitting the k-th anchor information, is envisaged to ensure that all similar packets are received. Afterwards, using m_k and the approach in Section III-A, the node estimates the last two-hop distance, adds the estimate to $\hat{d}_k$ and broadcasts the resulting packet in the network. This process will continue until each sensor in the network becomes aware of all anchors' position. It is noteworthy that, at this stage, if n_k is even, the sensor is already aware of its distance to the k-th anchor. Otherwise, it is obtained by adding, as discussed in Section III-B, 2/3 to the stored $\hat{d}_k$. Fig. 7 summarizes our algorithm's pseudocode implementable at each sensor.

Input: Number of anchors M, and their positions (x_k, y_k), as well as the received hop-count n_k, and the number of received packets m initialized to zero, where $k = 1, \ldots, M$
for $k = 1 \rightarrow M$ **do**
  Check database
  if Data$\{k\}$= $\emptyset$ **then**
    Save Data$\{k\}$
    $n_k = n_k + 1$
    Broadcast updated Data$\{k\}$
  else if $n \geq n_k$ and $\hat{d} \geq \hat{d}_k$ **then**
    Discard received data
  else if $n \leq n_k$ and $\hat{d} \leq \hat{d}_k$ **then**
    $n_k = n$
    $\hat{d}_k = \hat{d}$
  else if $(n_k \mod 2) = 0$ **then**
    $m = m + 1$
    Wait τ
      $\hat{d}_{k-(n_k-2)} \leftarrow$ Eq. (15) in section IV
  else
    $n_k = n_k + 1$
    Broadcast updated Data$\{k\}$
  end if
end for
$\hat{x}_i, \hat{y}_i \leftarrow$ Eq. (18) in section I
Output (x_i, y_i) ▷ Estimated position of the i-th node

Fig. 7. Proposed algorithm for sensors.

B. Positions' Computation

Once the i-th sensor obtains all the anchors' coordinates and their corresponding distances, it computes its position by solving the following nonlinear equations system:

$$\begin{cases} (a_1 - \hat{x}_i)^2 + (b_1 - \hat{y}_i)^2 = \hat{d}_{i-1}^2 \\ (a_2 - \hat{x}_i)^2 + (b_2 - \hat{y}_i)^2 = \hat{d}_{i-2}^2 \\ \vdots \qquad\qquad\qquad\quad \vdots \\ (a_M - \hat{x}_i)^2 + (b_M - \hat{y}_i)^2 = \hat{d}_{i-M}^2, \end{cases} \tag{14}$$

where $(\hat{x}_i, \hat{y}_i)$ are the estimated i-th sensor's coordinates. After some rearrangements aiming to linearize the system above, we obtain

$$\boldsymbol{\Upsilon}\hat{\boldsymbol{\alpha}}_i = -\frac{1}{2}\boldsymbol{\kappa_i}, \tag{15}$$

where $\hat{\boldsymbol{\alpha}}_i = \left[\hat{x}_i, \hat{y}_i\right]^T$,

$$\boldsymbol{\Upsilon} = \begin{bmatrix} a_1 - a_M & b_1 - b_M \\ a_2 - a_M & b_2 - b_M \\ \vdots & \vdots \\ a_{(M-1)} - a_M & b_{(M-1)} - b_M \end{bmatrix}, \tag{16}$$

and

$$\boldsymbol{\kappa_i} = \begin{bmatrix} \hat{d}_{i-1}^2 - \hat{d}_{i-M}^2 + a_M^2 - a_1^2 + b_M^2 - b_1^2 \\ \hat{d}_{i-2}^2 - \hat{d}_{i-M}^2 + a_M^2 - a_2^2 + b_M^2 - b_2^2 \\ \vdots \\ \hat{d}_{i-(M-1)}^2 - \hat{d}_{i-M}^2 + a_M^2 - a_{(M-1)}^2 + b_M^2 - b_{(M-1)}^2 \end{bmatrix}. \tag{17}$$

Since $\boldsymbol{\Upsilon}$ is a non-invertible matrix, $\hat{\boldsymbol{\alpha}}_i$ could be estimated with the pseudo-inverse of $\boldsymbol{\Upsilon}$ as follows:

$$\hat{\boldsymbol{\alpha}}_i = -\frac{1}{2}\left(\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^T\right)^{-1}\boldsymbol{\Upsilon}^T\boldsymbol{\kappa_i}. \tag{18}$$

Therefore, the i-th sensor is able to obtain an estimate of its coordinates as $\hat{x}_i = [\hat{\boldsymbol{\alpha}}_i]_1$, and $\hat{y}_i = [\hat{\boldsymbol{\alpha}}_i]_2$. It is also noteworthy from (16) and (17) that $\hat{x}_i$ and $\hat{y}_i$ are solely dependent on the anchors' coordinates $(a_k, b_k), k = 1, \ldots, M$ and the estimated distances $\hat{d}_{k-i}, k = 1, \ldots, M$ which are all locally available at the i-th sensor. Therefore, their computation does not require any additional overhead (i.e., additional power cost), making our algorithm compliant with WSNs' power restrictions.

In what follows, the performance of the proposed localization algorithm is analyzed and compared to the most representative benchmarks in the literature.

V. Performance Analysis of the Proposed Algorithm

A. Performance Metrics

One way to prove the efficiency of the proposed localization algorithm is undoubtedly analyzing its accuracy. To this end,

we introduce the following performance metric:

$$\mathcal{E}_{\mathrm{P},i} = \left\| \boldsymbol{\alpha}_i - \hat{\boldsymbol{\alpha}}_i \right\|^2, \tag{19}$$

where $\mathcal{E}_{\mathrm{P},i}$ denotes the i-th sensor's location estimation error (LEE) and $\boldsymbol{\alpha}_i = [x_i, y_i]^T$ is a vector whose entries are the true i-th sensor coordinates. From (19), $\mathcal{E}_{\mathrm{P},i}$ is an excessively complex function of the random variables (x_i, y_i), $i = 1, \ldots, N$, d_{i-k} and $\hat{d}_{i-k}, k = 1, \ldots, M$ and, hence, a random quantity of its own. Therefore, it is practically more appealing to investigate the behavior and the properties of the average LEE $\bar{\mathcal{E}}_{\mathrm{P}}(N) = \mathrm{E}\{\mathcal{E}_{\mathrm{P},i}\}$ achieved using the proposed algorithm. Actually, $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ could be differently defined as

$$\bar{\mathcal{E}}_{\mathrm{P}}(N) = \mathrm{E}\left\{\mathcal{G}_{\mathrm{P}}^{\mathrm{Net}}(N)\right\}, \tag{20}$$

where

$$\mathcal{G}_{\mathrm{P}}^{\mathrm{Net}}(N) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{E}_{\mathrm{P},i}, \tag{21}$$

refers to the global LEE through the network, which is commonly used as a performance metric in the context of localization in WSNs [8]–[22]. Furthermore, using the strong law of large numbers, we show for large N that we have

$$\mathcal{G}_{\mathrm{P}}^{\mathrm{Net}}(N) \xrightarrow{p1} \bar{\mathcal{E}}_{\mathrm{P}}(N), \tag{22}$$

where $\xrightarrow{p1}$ stands for convergence with probability one. From (22), $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ is not only the statistical average of $\mathcal{G}_{\mathrm{P}}^{\mathrm{Net}}(N)$, but also it approaches the latter for any given realization (i.e., any given $(x_i, y_i), i = 1, \ldots, N$). All this proves that $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ is a meaningful and useful performance metric.

In the next section, the average LEE $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ achieved using the proposed algorithm is derived in closed -form and its behavior is analyzed.

B. Proposed Algorithm's Average LEE

It follows from (18) that

$$\mathcal{E}_{\mathrm{P},i} = \frac{1}{4}\left\| \left(\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^T\right)^{-1}\boldsymbol{\Upsilon}^T\boldsymbol{\delta}_i \right\|^2, \tag{23}$$

where $[\boldsymbol{\delta}_i] = \left[\epsilon_1 - \epsilon_M, \ldots, \epsilon_{M-1} - \epsilon_M\right]^T$ with $\epsilon_k = \hat{d}_{i-k}^2 - d_{i-k}^2$ being the squared-distance estimation error. $\mathcal{E}_{\mathrm{P},i}$ is then given by

$$\begin{aligned}\mathcal{E}_{\mathrm{P},i} &= \mathrm{Tr}\left(\left(\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^T\right)^{-1}\boldsymbol{\Upsilon}^T\boldsymbol{\delta}_i\boldsymbol{\delta}_i^T\boldsymbol{\Upsilon}\left(\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^T\right)^{-1}\right)\\ &= \mathrm{Tr}\left(\boldsymbol{\Omega}\boldsymbol{\delta}_i\boldsymbol{\delta}_i^T\right)\\ &= \sum_{k=1}^{M-1}\Omega_{kk}\left([\boldsymbol{\delta}_i]_k\right)^2 + \sum_{k=1}^{M-1}\sum_{l=1,l\neq k}^{M-1}\Omega_{kl}[\boldsymbol{\delta}_i]_l[\boldsymbol{\delta}_i]_k,\end{aligned} \tag{24}$$

where $\mathrm{Tr}(\mathbf{X})$ is the trace of the matrix $\mathbf{X}$ and $\boldsymbol{\Omega} = \boldsymbol{\Upsilon}\left(\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^T\right)^{-2}\boldsymbol{\Upsilon}^T$. Note in the second line of (24) that we exploit the cyclic property of the trace. Since $\epsilon_k, k = 1, \ldots, M$ are i.i.d random variables, we have from (24) the following

$$\bar{\mathcal{E}}_{\mathrm{P}}(N) = \sigma_\epsilon^2\left(2\mathrm{Tr}(\Omega) + \sum_{k=1}^{M-1}\sum_{l=1,l\neq k}^{M-1}\Omega_{kl}\right). \tag{25}$$

Now let us turn our attention to σ_ϵ^2. For the sake of clarity, we first assume that there are exactly two hops between the i-th sensor and each anchor. The obtained results will be thereafter generalized. In such a case, from (4) and (5), the Taylor series expansion of $\Psi(x)$ at F_k yields

$$\hat{d}_{i-k} = d_{i-k} + \sum_{n=1}^{\infty}\frac{\Psi^{(n)}(F_k)}{n!}\Delta F^n, \tag{26}$$

where $\Psi^{(n)}(x)$ is the n-th derivative of $\Psi(x)$ and $\Delta F = m_k/\lambda - F_k$. Assuming that ΔF is small enough to allow approximation of $\hat{d}_{k-i}$ by the first three non-zero terms of the right-hand-side (RHS) of (26), we obtain

$$\epsilon_k \simeq 2d_{i-k}\Psi^{(1)}(F_k)\Delta F + \left(\left(\Psi^{(1)}(F_k)\right)^2 + d_{i-k}\Psi^{(2)}(F_k)\right)\Delta F^2, \tag{27}$$

where $\Psi^{(1)}(x) = \left(4R^2 - \Psi(x)^2\right)^{-1/2}$ and $\Psi^{(2)}(x) = \Psi(x)/\left(4R^2 - \Psi(x)^2\right)^2$. Since the nodes are uniformly deployed in S, the probability of having m_k nodes in F_k follows a Binomial distribution $\mathrm{Bin}(N, p)$ where $p = \frac{F_k}{S}$ and, therefore, the first and second order statistics of m_k are $\mathrm{E}\{m_k\} = \lambda F_k$ and $\mathrm{E}\{m_k^2\} = \lambda F_k\left(1 - \frac{F_k}{S} + \lambda F_k\right)$, respectively. Using the latter along with (27) yields

$$\mathrm{E}_{m_k}\{\epsilon_k\} = \frac{4R^2\lambda^{-1}F_k\left(1 - \frac{F_k}{S}\right)}{\left(4R^2 - \Psi(F_k)^2\right)^2}, \tag{28}$$

where the expectation is taken with respect to m_k. As could be observed from (28), the probability density function $f_{F_k}(F)$ of F_k is crucial to derive σ_ϵ^2 in closed-form. For the sake of mathematical tractability, F_k is assumed to be Uniform in $[0, F_{\max}]$ where $F_{\max} = \Phi(R) = \left(-\frac{\sqrt{3}}{2} + \frac{2\pi}{3}\right)R^2$. Despite this simplifying assumption, we will shortly see in Section VI that the obtained analytical results closely match those obtained empirically by Monte Carlo simulations. $\mathrm{E}\{\epsilon_k\}$ is then given by

$$\begin{aligned}\mathrm{E}\{\epsilon_k\} &= 4R^2\lambda^{-1}\int_R^{2R}\frac{\Phi(x)(1 - \Phi(x)S)}{\left(4R^2 - x^2\right)^2}dx\\ &= \frac{1}{9\left(3\sqrt{3} - 4\pi\right)N}\Big(12\left(-9 + \sqrt{3}\pi\right)S\\ &\quad + \left(27\sqrt{3} + 4\pi\left(9 - 2\sqrt{3}\pi\right)\right)R^2\Big).\end{aligned} \tag{29}$$

In the first line of (29), please note that we resort to the variable change $F = \Phi(x)$. Following similar steps as above, we show that

$$\begin{aligned}\mathrm{E}\left\{\epsilon_k^2\right\} = \frac{1}{27\left(3\sqrt{3} - 4\pi\right)N}\Big(\Big(4\pi\left(-189 + 4\pi\left(9\sqrt{3} + 4\pi\right)\right)\\ - 1053\sqrt{3}\Big)R^4 - 9\left(8\pi\left(3\sqrt{3} + 2\pi\right) - 243\right)R^2S\Big),\end{aligned} \tag{30}$$

TABLE I
CLOSED-FORM EXPRESSIONS OF $\xi_{n,m}$ $n, m = 0, 1, 2$.

Parameter	Closed-form expression
$\xi_{1,0}$	$6\sqrt{3}R \Big/ \big(4(\pi - 3\sqrt{3})\big)$
$\xi_{2,0}$	$\big(8\sqrt{3}\pi + 9\big)R^2 \Big/ \big(8\sqrt{3}\pi - 18\big)$
$\xi_{0,1}$	$\Big(\big(297\sqrt{3} - 8\pi\big(9 + 2\sqrt{3}\pi\big)\big)R^2 + 6\big(4\sqrt{3}\pi - 27\big)S\Big) \Big/ \big(36\big(3\sqrt{3} - 4\pi\big)R\big)$
$\xi_{0,2}$	$\Big(\big(64\pi^3 - 567\sqrt{3} - 216\pi\big)R^2 + 36\big(27 - 4\pi^2\big)S\Big) \Big/ \big(216\big(3\sqrt{3} - 4\pi\big)\big)$
$\xi 1,1$	$\Big(\Big(1215\sqrt{3} - 8\pi\Big(8\pi\big(3\sqrt{3} + \pi\big) - 135\Big)\Big)R^2 + 36\Big(4\pi\big(2\sqrt{3} + \pi\big) - 99\Big)S\Big) \Big/ \big(432\big(3\sqrt{3} - 4\pi\big)\big)$
$\xi_{1,2}$	$\Big(\big(-1701\sqrt{3} + 40\pi\big(13 + 2\sqrt{3}\pi\big)\big)R^3 + 30\big(19 - 4\sqrt{3}\pi\big)RS\Big) \Big/ \big(30\big(3\sqrt{3} - 4\pi\big)\big)$
$\xi_{2,1}$	$\Big(9R\Big(3\sqrt{3}R\big(-1 + 45R\big) - 104S\Big) - 54S - 16\sqrt{3}\big(1 + 3R\big)\pi^2R^2 + 24\pi\Big(3\big(1 - 5R\big)R^2 + \sqrt{3}\big(1 + 6R\big)S\Big)\Big) \Big/ \big(36\big(3\sqrt{3} - 4\pi\big)\big)$
$\xi_{2,2}$	$\Big(9\Big(243 - 8\pi\big(3\sqrt{3} + 2\pi\big)\Big)R^2S + \Big(4\pi\Big(4\pi\big(9\sqrt{3} + 4\pi\big) - 189\Big) - 1053\sqrt{3}\Big)R^4\Big) \Big/ \big(108\big(3\sqrt{3} - 4\pi\big)\big)$

and, hence, σ_ϵ^2 is obtained. It can be then inferred from (25)-(30) that the achieved average LEE $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ using the proposed algorithm linearly decreases with N when S and R are fixed. Furthermore, for sufficiently large N, we have $\bar{\mathcal{E}}_{\mathrm{P}}(N) \simeq 0$. This property is actually a desired feature for any sensor localization algorithm since WSNs are typically dense. It is noteworthy here that the best representative benchmarks in the literature lack such a feature [8], [19]. Recall, however, that the results in (29) and (30) were derived assuming that the number of hops between any anchor-sensor pair is $n_h = 2$ hops. For the sake of generalization, we consider in the sequel that n_h is a random variable with mean $\bar{n}_h$. In such a case, $\mathrm{E}\{\epsilon_k\}$ and $\mathrm{E}\{\epsilon_k^2\}$ could be expressed in (31) and (32), respectively, as shown at the bottom of this page, where $\xi_{n,m}$ $n, m = 0, 1, 2$ are parameter functions of R and S whose expressions are listed in TABLE I.

Proof: See Appendix A.

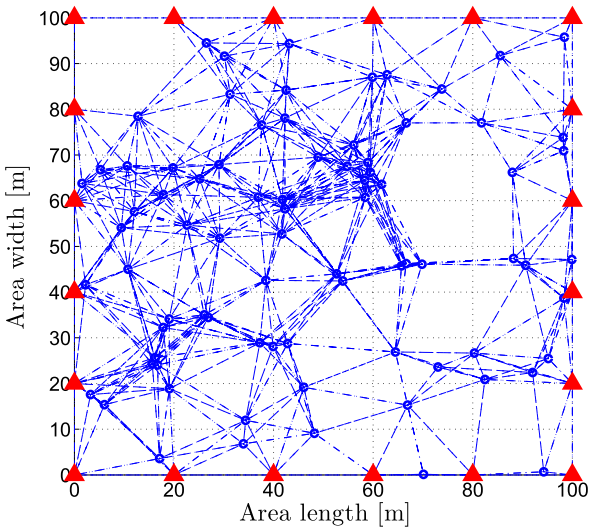
It is noteworthy that the results in (31) and (32) are very interesting in terms of implementation strategy, since they allow, through (25), to easily find the smallest N that keeps $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ below a certain level. They also allow to find the best anchor placement strategy that minimizes $\bar{\mathcal{E}}_{\mathrm{P}}(N)$ for a given N. Moreover, in contrast with the two-hop case, it follows from (31) and (32) that we have

$$\bar{\mathcal{E}}_{\mathrm{P}}(N) \simeq 0.16R^4 + \frac{2R^3}{15}\xi_{1,0} + \frac{R^2}{9}\xi_{2,0} \neq 0, \tag{33}$$

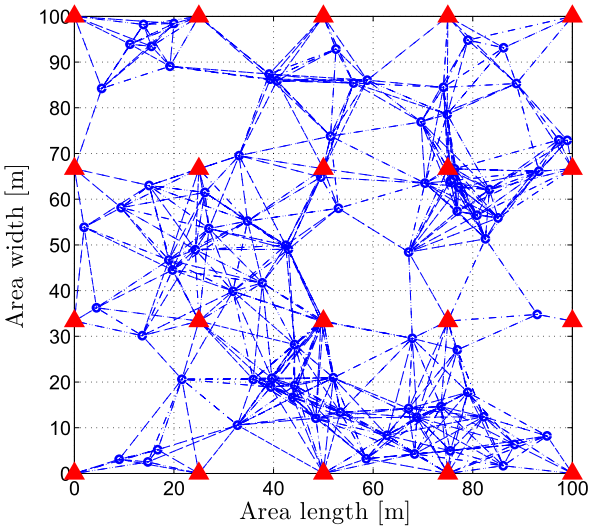
when N is large enough. Note that x is nothing but the error incurred when estimating the last hop of an odd distance between any anchor-sensor pair in the network. A proper anchor selection scheme should then be envisaged to make our proposed algorithm reach its optimal accuracy

$$\mathrm{E}\{\epsilon_k\} = R^2\left(\frac{3\big(2\bar{n}_h - 1\big)\big(\xi_{0,2} + 2\xi_{1,1}\big) + \xi_{0,1}\Big(3\xi_{1,0}\Big(2\bar{n}_h\big(\bar{n}_h - 3\big) + 3\Big) + 4\big(\bar{n}_h - 1\big)\Big)}{12N} + \frac{\xi_{0,1}^2\Big(2\bar{n}_h\big(\bar{n}_h - 3\big) + 3\Big)}{8N^2} - \frac{1}{36}\right). \tag{31}$$

$$\begin{aligned}
\mathrm{E}\left\{\epsilon_k^2\right\} = \frac{R^2}{4N}\Bigg(&\frac{16\xi_{0,2}}{9} - \frac{4\xi_{0,1}}{27} + \frac{(\bar{n}_h - 1)}{72}\Big(2\big(\xi_{0,2} + 2\xi_{1,1}\big) + \xi_{0,1}\big(\bar{n}_h - 3\big)\big(\frac{\xi_{0,1}}{N} + 2\xi_{1,0}\big)\Big)\Big(\frac{12\xi_{0,1}\big(\bar{n}_h - 1\big)}{N} - 1\Big)\\
&+ \big(\bar{n}_h - 1\big)\Bigg(4\xi_{2,2} + \frac{\bar{n}_h - 3}{2}\Bigg(\frac{3\xi_{0,2}^2}{N} + 8\big(\frac{\xi_{1,1}^2}{N} + \xi_{1,2}\xi_{1,0}\big) + \frac{2\xi_{0,1}}{N}\Big(6\xi_{1,2} + 4\xi_{2,1} + \big(3\xi_{0,2} + 4\xi_{1,1}\big)\big(\bar{n}_h - 5\big)\xi_{1,0}\Big)\\
&+ \frac{6\xi_{0,1}^3\xi_{1,0}}{N^2}\big(\bar{n}_h - 7\big)\big(\bar{n}_h - 5\big) + \frac{2\xi_{0,2}\xi_{0,1}^2}{N^2}\Big(\big(\bar{n}_h - 5\big)\xi_{1,0}^2 + 2\xi_{2,0}\Big)\Big(\frac{6\xi_{0,2}}{N} + \big(\bar{n}_h - 5\big)\Big(\frac{12\xi_{1,1}}{N} + \big(\bar{n}_h - 7\big)\xi_{1,0}^2 + \xi_{2,0}\Big)\Big)\Bigg)\Bigg)\\
&+ \bar{n}_h\Bigg(4\xi_{2,2} + \big(\bar{n}_h - 2\big)\Bigg(\frac{3\xi_{0,2}^2}{2N} + 4\big(\frac{\xi_{1,1}^2}{N} + \xi_{1,2}\xi_{1,0}\big) + \frac{\xi_{0,1}}{N}\Big(6\xi_{1,2} + 4\xi_{2,1} + \big(\bar{n}_h - 4\big)\big(3\xi_{0,2} + 4\xi_{1,1}\big)\xi_{1,0}\Big) + 3\frac{\xi_{0,1}^3}{N^2}\big(\bar{n}_h\\
&- 6\big)\big(\bar{n}_h - 4\big)\xi_{1,0} + \xi_{0,2}\Big(\big(\bar{n}_h - 2\big)\xi_{1,0}^2 + \frac{\xi_{2,0}}{2}\Big) + \frac{\xi_{0,1}^2}{N}\Big(\frac{3\xi_{0,2}}{N} + \big(\bar{n}_h - 4\big)\Big(\frac{6\xi_{1,1}}{N} + \big(\frac{\bar{n}_h - 6}{2}\big)\xi_{1,0}^2 + \xi_{2,0}\Big)\Big)\Bigg)\Bigg)\\
&+ R^2\left(\frac{25R^2}{162} + \frac{2R\xi_{1,0}}{15} + \frac{\xi_{2,0}}{9}\right).
\end{aligned} \tag{32}$$

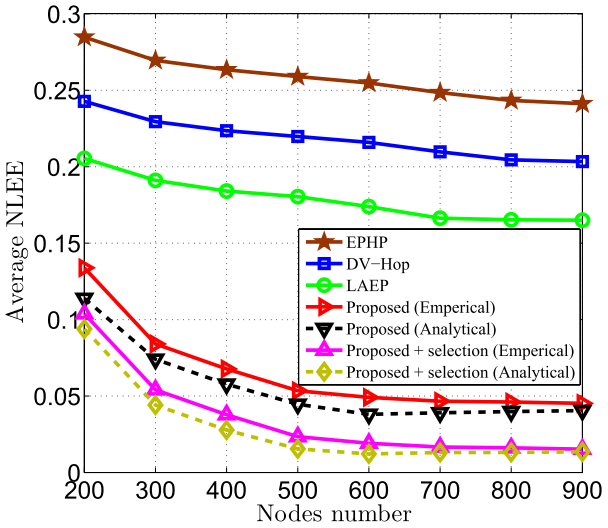


(a) Perimeter anchor placement.

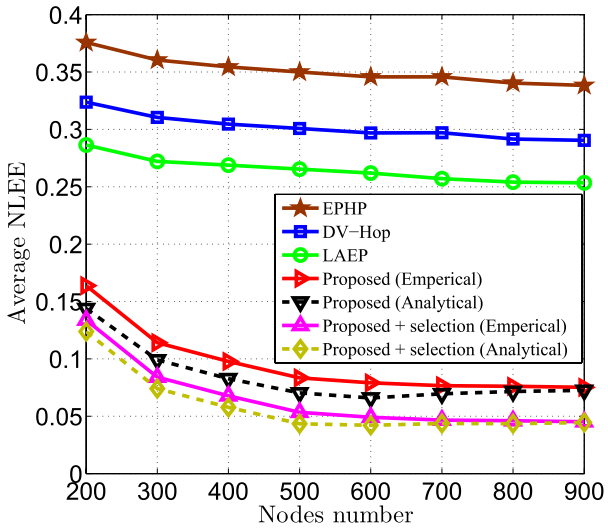


(b) Grid anchor placement.

Fig. 8. Different anchors placements.



(a) Perimeter anchor placement.



(b) Grid anchor placement.

Fig. 9. Average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies versus the nodes number N.

(i.e., $\bar{\mathcal{E}}_{\mathrm{P}}(N) \simeq 0$) at large N. Indeed, if each sensor selects among the list of anchors only those with an even number of hops, its achieved average LEE would approach 0 when N is large enough. This, of course, requires that at least 3 anchors comply with the above criterion. Please note that such a selection scheme could be easily implemented in each sensor without burdening neither the implementation complexity of the proposed localization algorithm nor the overall cost of the WSNs.

C. Proposed Algorithm's Asymptotic LEE

So far, we derived the average LEE achieved by our localization algorithm and studied its behavior and properties. Motivated by the fact that the LEE is a more practical metric than its average, we investigate in this section its statistical properties more thoroughly for the sake of further highlighting the proposed algorithm's accuracy.

Let us consider again the 2-hop case (i.e., two hops between the i-th sensor and the k-th anchor nodes). Exploiting the fact that m_k is a Binomial random variable, we have from the Chebyshev's inequality we have

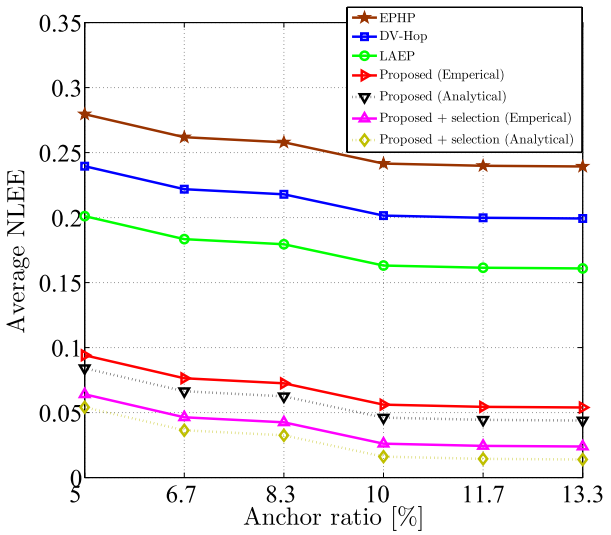
$$1 - P\left(|\Delta F| < \kappa\right) \leq \frac{F_k(S - F_k)}{N\kappa^2}, \tag{34}$$

where κ is any given strictly positive real. If the latter is chosen small enough to guarantee the equivalence $|\Delta F| < \kappa \Leftrightarrow |\Delta F| \simeq 0$, it holds for sufficiently large N that

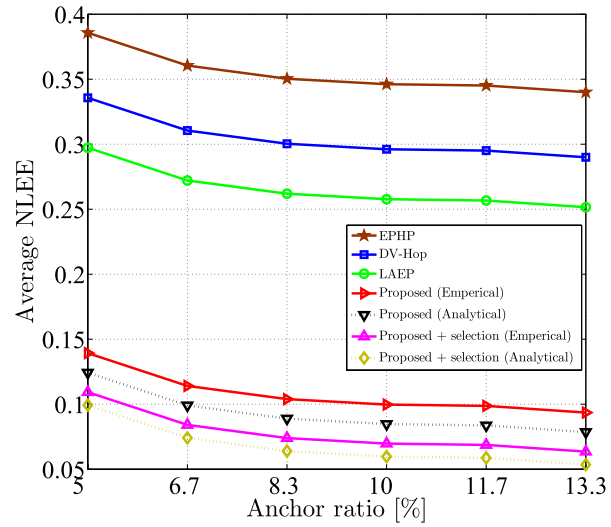
$$P\left(|\Delta F| \simeq 0\right) \simeq 1. \tag{35}$$

Exploiting this result along with (27) we obtain

$$P\left(\epsilon_k \simeq 0\right) \simeq 1, \tag{36}$$

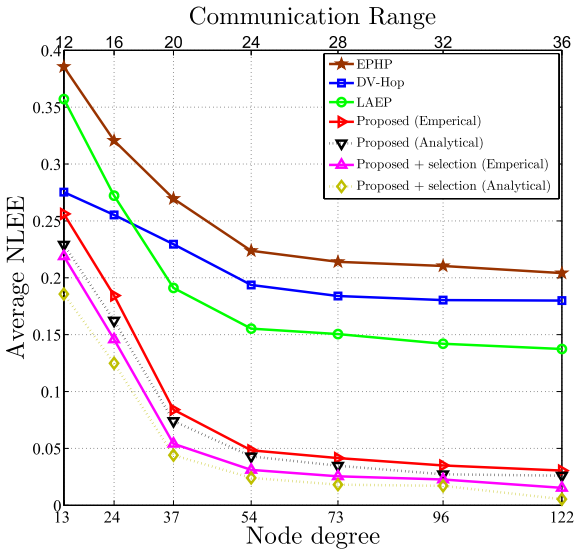


(a) Perimeter anchor placement.

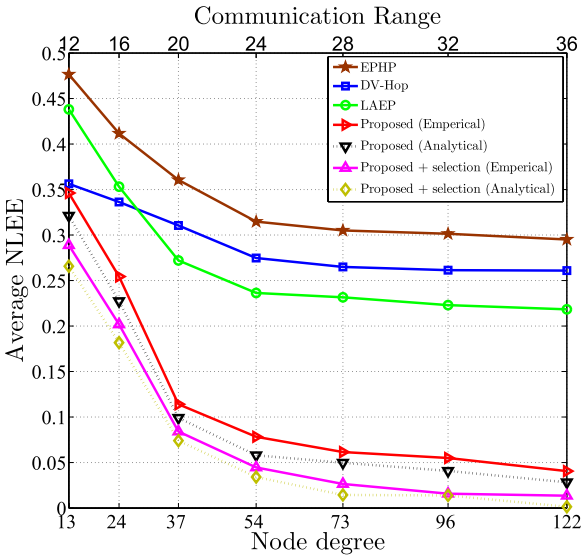


(b) Grid anchor placement.

Fig. 10. Average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies versus the anchor ratio when the nodes number $N = 300$.



(a) Perimeter anchor placement.



(b) Grid anchor placement.

Fig. 11. Average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies versus the node degree and communication range when the nodes number $N = 300$.

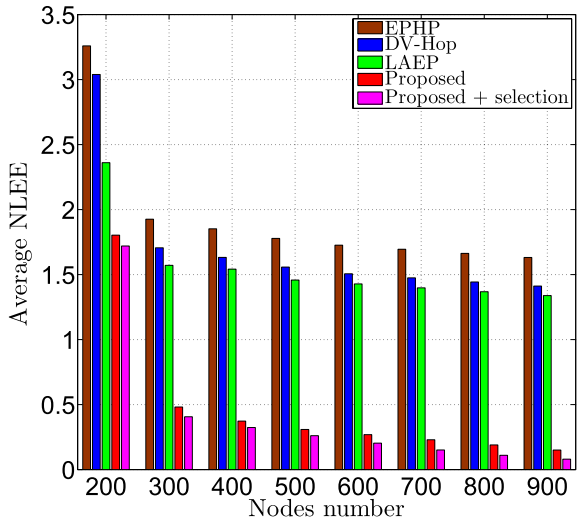
and, hence, for large N we have $\mathcal{E}_{\mathrm{P},i} \simeq 0$. This further proves the accuracy of the proposed algorithm. Furthermore, it is straightforward to show that $\mathcal{E}_{\mathrm{P},i} \simeq 0$ also holds when the number of hops between the i-th sensor and all anchors is even (but not necessarily 2). This emphasizes even more the importance of the anchor selection scheme discussed above.

VI. Simulations Results

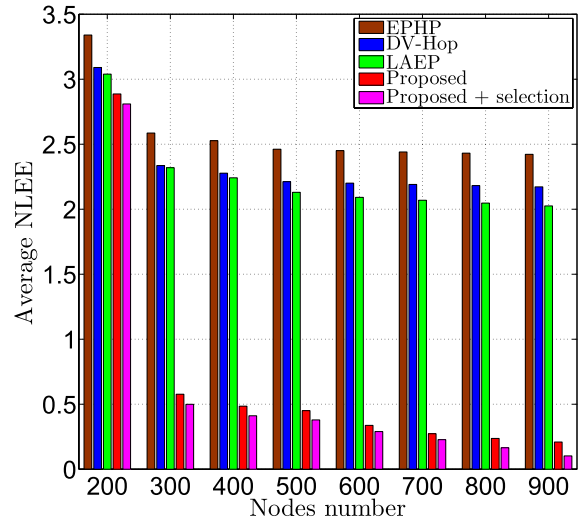
In this section, we validate and illustrate our theoretical results by Monte Carlo simulations. These are conducted to compare under the same network settings the R^2-normalized LEEs (NLEE)s achieved by the proposed algorithm and three of the best representative localization algorithms currently available in the literature, i.e., DV-Hop [8], LAEP [19], and EPHP [20].

All simulation results are obtained by averaging over 600 trials. In all simulations, sensors are uniformly deployed in a 2-D square area $S = 10^4\ m^2$. M is set to 20, expect in Fig. 10 where it varies from 15 to 40. R is set to 20 m, expect in Fig. 11 where it varies from 12 m to 36 m. Two commonly used anchor placement strategies in the context of WSNs are considered: the perimeter and grid placements as depicted in Figs. 8 and 8, respectively.

Fig. 9 plots the average NLEE achieved by the proposed algorithm, DV-Hop, and LAEP versus N with two anchor placement strategies: perimeter in Fig. 9(a) and grid placement in Fig. 9(b). From these figures, the proposed localization algorithm always outperforms in accuracy its counterparts. It is, for instance at $N = 700$, until 12 times more accurate than DV-Hop and until 10 times more accurate than LAEP. This further proves the proposed algorithm's efficiency in WSNs and highlights its advantage over its counterparts.

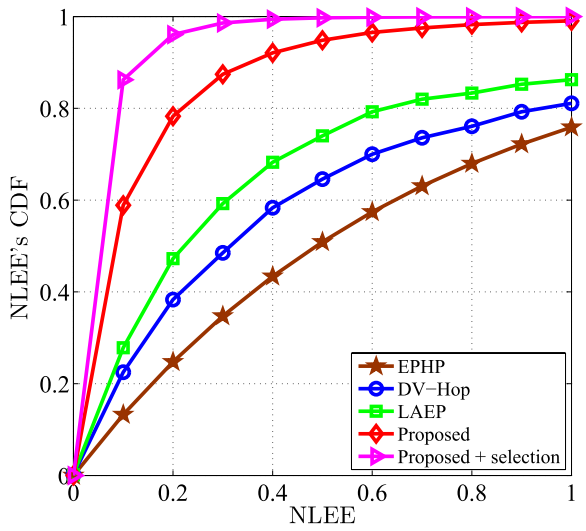


(a) Perimeter anchor placement.

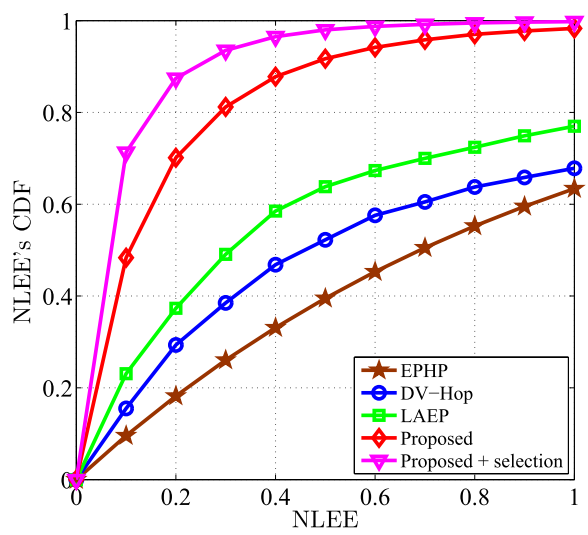


(b) Grid anchor placement.

Fig. 12. NLEE's standard deviation achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies versus the nodes number N.



(a) Perimeter anchor placement.



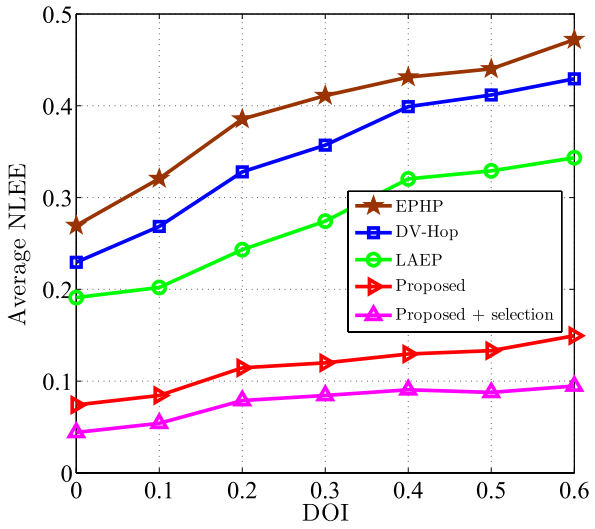
(b) Grid anchor placement.

Fig. 13. NLEE's CDF achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies when the nodes number $N = 300$.

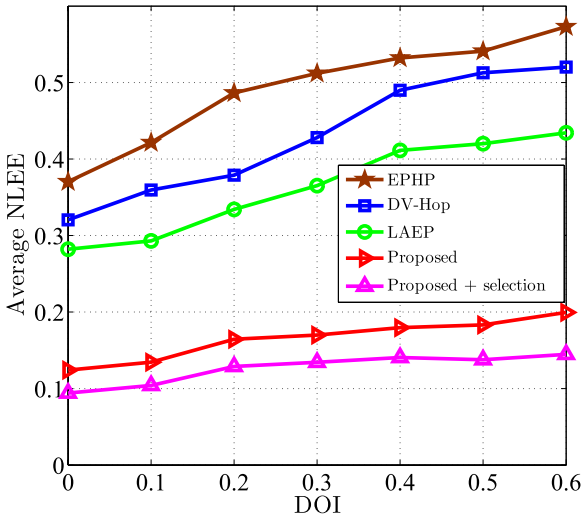
Fig. 10 plots the average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP versus the anchor ratio with two anchor placement strategies: perimeter in Fig. 10(a) and grid placement in Fig. 10(b). From these figures, all algorithms benefit, as expected, from larger anchor ratios. However, the proposed algorithm remains more accurate than its counterparts thereby proving once again its superiority.

Fig. 11 displays the average NLEE achieved by all localization algorithms versus both the node degree and communication range with both the perimeter and grid anchor placement strategies. As can be observed from this figure, the average NLEE of each algorithm decreases, as expected, with both the node degree and communication range. However, the accuracy gain of the proposed algorithm is much more important than those of its counterparts. In contrast to the latter, its average NLEE approaches 0 when the node degree and/or the communication range increase(s).

Fig. 12 plots the NLEE's standard deviation achieved by all localization algorithms versus N, for two anchor placement strategies: perimeter in Fig. 12(a) and grid placement in Fig. 12(b). As it can be observed from these figures, regardless of the anchor placement strategy, the one achieved by the proposed algorithm substantially decreases when N increases while those achieved by the other algorithms slightly decrease. Furthermore, the NLEE's standard deviation achieved by the proposed algorithm with or without anchor selection approaches 0 for any placement strategy. This is due to the fact that the LEE itself being around 0 occurs almost certainly (i.e., with almost probability 1) as stated in Section V-C. On the other hand, Figs. 12(a) and 12(b) suggest that the proposed algorithm's performance is further improved if the

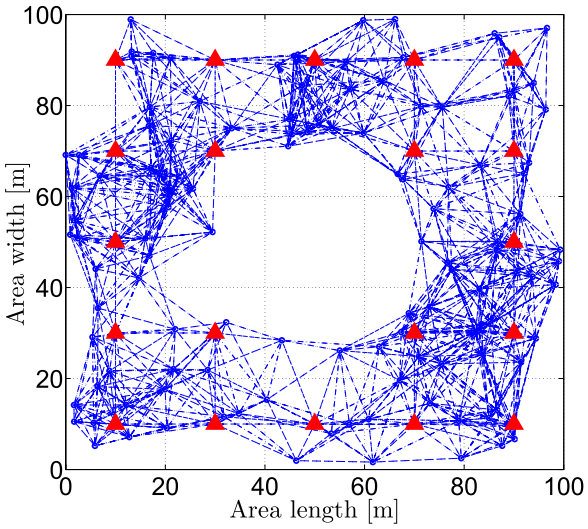


(a) Perimeter anchor placement.

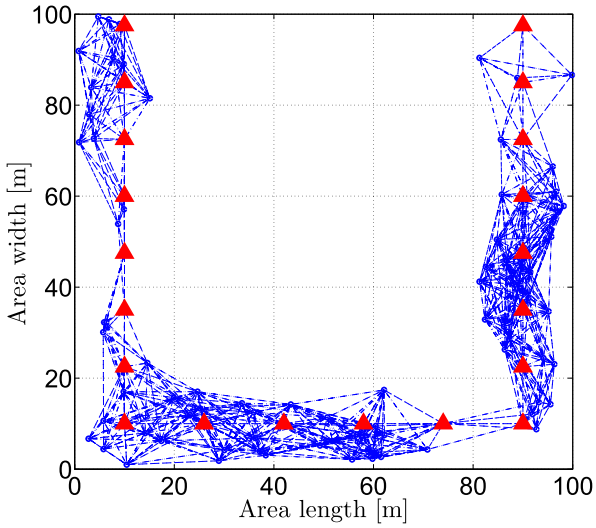


(b) Grid anchor placement.

Fig. 14. Average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP versus the DoI with both perimeter and grid anchor placement strategies when the nodes number $N = 300$.



(a) O-shaped topology.



(b) U-shaped topology.

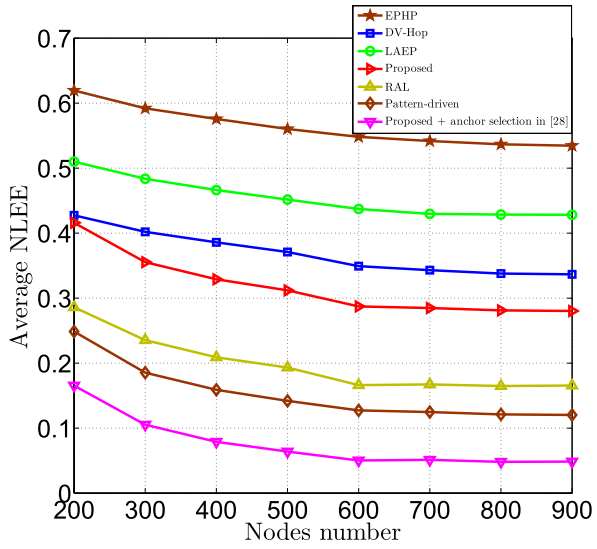
Fig. 15. Different network topologies.

low-cost anchor selection scheme introduced in Section V-B is implemented at each sensor. All these observations corroborate the results and discussions disclosed in Section V.

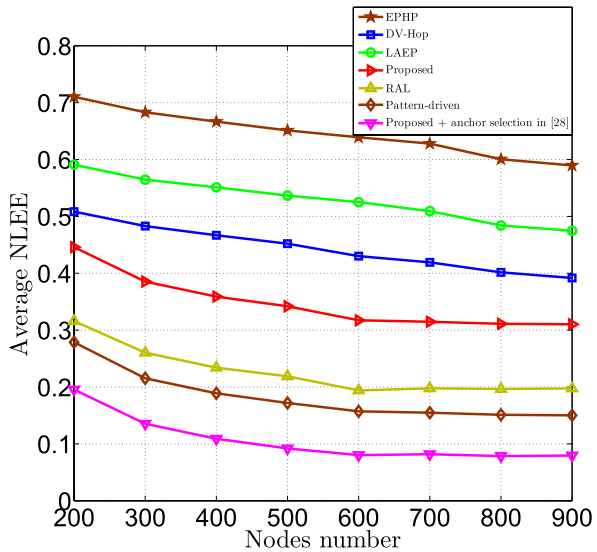
Fig. 13 illustrates the NLEE's CDF achieved by our proposed localization algorithm with and without the anchor selection scheme as well as that achieved by the other algorithms for two anchor placement strategies: perimeter in Fig. 13(a) and grid placement in Fig. 13(b). From these figures, using the proposed algorithm, 80% (98% with anchor selection) of the sensors could estimate their position with NLEE less than 0.2. In contrast, 45% of the nodes achieve the same accuracy with LAEP and only about 38% with DV-Hop using the perimeter anchor placement strategy. This proves even more the accuracy of the proposed localization algorithm.

Fig. 14 plots the average NLEE achieved by the proposed algorithm and its counterparts versus the degree of range irregularity (DoI). In this figure, the transmission range is no longer assumed circular. A range irregularity model similar to that in [35] was implemented instead. From Figs. 14(a) and 14(b), the average NLEEs achieved by all algorithms deteriorate due to the range irregularity. This is expected since their designs do not account for such a phenomenon. However, the proposed algorithm remains more accurate than its counterparts. This further proves its superiority over the latter.

Figs. 16 and 17 plot the average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP versus N in two different anisotropic topologies commonly used in the context of WSNs: the O-shaped and the U-shaped illustrated in Figs. 15(a) and 15(b), respectively. In these figures, we also plot the average NLEE achieved by two well-known algorithms whose designs account for such anisotropy: RAL [29] and Pattern-driven [30]. Furthermore, for the sake of fairness, we plot the average NLEE of our algorithm with the anchor selection strategy developed for anisotropic

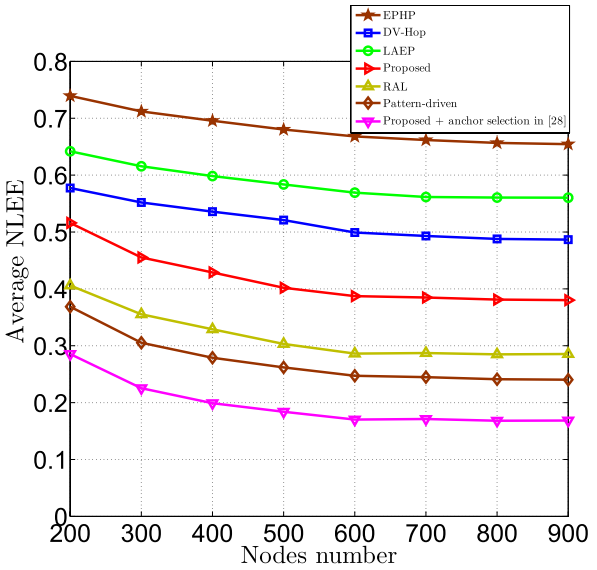


(a) Perimeter anchor placement.

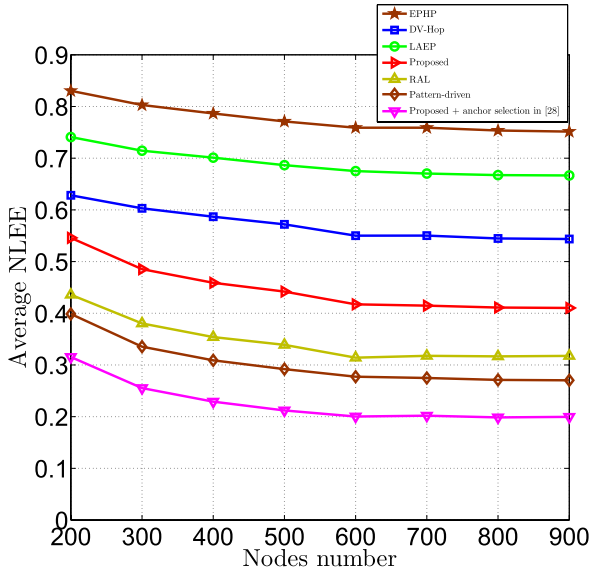


(b) Grid anchor placement.

Fig. 16. Average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies versus the nodes number N in the O-Shaped topology.



(a) Perimeter anchor placement.



(b) Grid anchor placement.

Fig. 17. Average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP with both perimeter and grid anchor placement strategies versus the nodes number N in the U-Shaped topology.

environments in [28]. We observe from Figs. 16 and 17 that the average NLEE achieved by the proposed algorithm, DV-Hop, LAEP, and EPHP deteriorates due to the presence of obstacles (i.e., areas with no nodes, for instance mountains, hills, etc.) in the network. This is hardly surprising since such obstacles, which are not taken into account when designing the latter, cause DE estimation errors, thereby hindering their accuracies. However, as could be observed from Figs. 16 and 17, the proposed algorithm remains more accurate than DV-Hop, LAEP, and EPHP thereby proving its robustness against the latter in anisotropic environments. On the other hand, RAL and Pattern-driven, whose designs account for such environments, outperform our algorithm. Nevertheless, if the latter is implemented with our anchor selection strategy previously developed in [28], it becomes once again more accurate than RAL and Pattern-driven thereby highlighting unambiguously the efficiency of the DE approach proposed in Section III..

Fig. 18 shows the total number of exchanged packets N_{packets} using Pattern-driven, LAEP, and the proposed algorithm with and without the anchor selection strategy developed in [28]. Please note that RAL and DV-Hop require almost the same N_{packets} as Pattern-driven while EPHP requires the same N_{packets} as LAEP. We see from Fig. 18 that the proposed algorithm requires the same number of exchanged packets as LAEP while it requires half the packets exchanged with Pattern-driven and RAL. This is expected since the latter require, in contrast to the proposed algorithm, a second information broadcast in the network. This implies that the overall power required by our algorithm to transmit and receive the exchanged packets is the same as the one required by LAEP

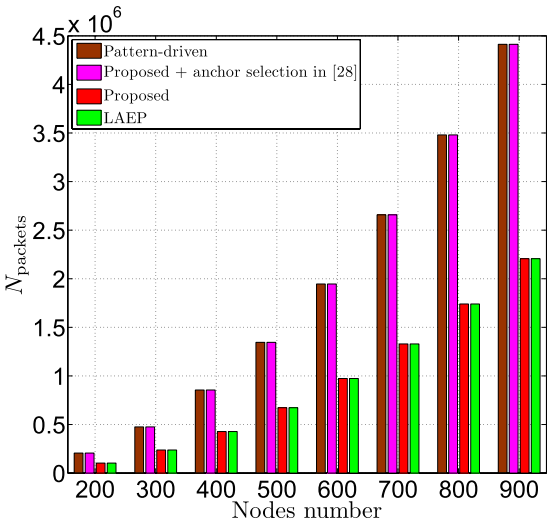


Fig. 18. The total number of exchanged packets N_{packets} versus the node density.

while it is the half of the one needed by Pattern-driven and RAL. On the other hand, if our algorithm is implemented with the anchor selection strategy developed in [28], it would incur the same power cost as the latter. This highlights the flexility of the proposed algorithm in that it is able to adapt both its accuracy and cost to the targeted WSN application. Indeed, by applying an anchor selection strategy such as [28], its accuracy could be easily enhanced on a need basis at the cost of more power consumption. Reciprocally, the proposed algorithm allows precious power savings at the cost of reduced yet acceptable accuracy when implemented without anchor selection such as [28].

VII. Conclusion

In this paper, we proposed a novel localization algorithm which properly exploits, in addition to the hop-based information, the forwarding nodes' number between any anchor-sensor pair. Its average location estimation error (LEE) was derived in closed-form and compared to those of the best representative algorithms in the literature. We showed that the proposed algorithm outperforms them in accuracy. Furthermore, we proved that, in contrast to the latter, our algorithm is able to achieve an average LEE of about 0, when the total sensors' number N is large enough. We also proved in such a condition that any realization of its achieved LEE approaches 0, which confirms unambiguously its high accuracy.

Appendix A

In order to compute $\mathrm{E}\{\epsilon_k\}$ and $\mathrm{E}\left\{\epsilon_k^2\right\}$ for any given n_h, one should distinguish two cases: a) n_h is even and b) n_h is odd. Let us first assume that n_h is even. In such a case, $n_h/2$ 2-hop distances exist between the i-th sensor and the k-th anchor nodes and, hence,

$$d_{i-k} = \sum_{r=1}^{n_h/2} d_r, \tag{37}$$

where d_r denotes the r-th 2-hop distance. ϵ_k is then given by

$$\epsilon_k = \left(\sum_{r=1}^{n_h/2} \hat{d}_r\right)^2 - \left(\sum_{r=1}^{n_h/2} d_r\right)^2, \tag{38}$$

where $\hat{d}_r$ is the estimated r-th 2-hop distance. Applying (26) to $\hat{d}_r$ and retaining the two first non-zero terms, ϵ_k could be equivalently expressed as

$$\epsilon_k = \left(\sum_{r=1}^{n_h/2} \zeta\left(F_{k,r}\right)\right)^2 + 2\sum_{r=1}^{n_h/2} \Psi\left(F_{k,r}\right) \sum_{r=1}^{n_h/2} \zeta\left(F_{k,r}\right), \tag{39}$$

where $F_{k,r}$ is the forwarding area associated with d_r and

$$\zeta\left(F_{k,r}\right) = \Psi^{(1)}\left(F_{k,r}\right)\left(\frac{m_{k,r}}{\lambda} - F_{k,r}\right) + \frac{\Psi^{(2)}\left(F_{k,r}\right)}{2}\left(\frac{m_{k,r}}{\lambda} - F_{k,r}\right)^2, \tag{40}$$

where $m_{k,r}$ is the number of forwarding sensors in $F_{k,r}$. It follows from (39) that

$$\mathrm{E}\{\epsilon_k\} = \frac{n_h}{2N}\left(\xi_{0,2} + 2\xi_{1,1} + 2\xi_{0,1}\left(\frac{n_h}{2} - 1\right)\left(\frac{\xi_{0,1}}{N} + \xi_{1,0}\right)\right), \tag{41}$$

where $\xi_{1,0} = \mathrm{E}\left\{\Psi\left(F_{k,r}\right)\right\}$, $\xi_{2,0} = \mathrm{E}\left\{\Psi\left(F_{k,r}\right)^2\right\}$, and $\xi_{n,m} = N\mathrm{E}\left\{\Psi\left(F_{k,r}\right)^n \zeta\left(F_{k,r}\right)^m\right\}$ for $(n,m) \in \{(i,j) | i,j = 0,1,2\} \setminus \{(1,0),(2,0)\}$. In order to obtain (41), please note that, we resort to the Multinomial theorem to break $\left(\sum_{r=1}^{n_h/2} \zeta\left(F_{k,r}\right)\right)^2$ into several terms.

Now, let us turn our attention to the computation of $\xi_{n,m}$. We first start by $\xi_{0,1} = N\mathrm{E}\left\{\zeta\left(F_{k,r}\right)\right\}$. Since the sensors are uniformly deployed in S, the probability of having $m_{k,r}$ sensors in $F_{k,r}$ follows a Binomial distribution $\mathrm{Bin}\left(N, \frac{F_{k,r}}{S}\right)$ and, hence, we have

$$\mathrm{E}_{m_{k,r}}\left\{\zeta\left(F_{k,r}\right)\right\} = \Psi^{(2)}\left(F_{k,r}\right) F_{k,r}\left(1 - \frac{F_{k,r}}{S}\right), \tag{42}$$

where $\mathrm{E}_{m_{k,r}}$ refers to the expectation with respect to $m_{k,r}$. Using (42) and integrating by parts twice yields

$$\xi_{0,1} = \frac{S}{2F_{\max}}\left[\Psi^{(1)}(F) F\left(1 - \frac{F}{S}\right) - \Psi(F)\left(1 - 2\frac{F}{S}\right)\right]_0^{F_{\max}} - \frac{1}{F_{\max}}\int_0^{F_{\max}} \Psi(F)\, dF. \tag{43}$$

$\xi_{0,1}$'s expression is then obtained by substituting $\Psi(F_{\max})$ with R and $F_{\max}$ with its expression.

As far as $\xi_{0,2}$ is concerned, it can be readily shown that

$$\xi_{0,2} = \frac{S}{2F_{\max}}\int_0^{F_{\max}} \left(\Psi^{(1)}(F)\right)^2 F\left(1 - \frac{F}{S}\right) dF. \tag{44}$$

In order to compute the above integral, one could apply the variable change $F = \Phi(x)$. From (4), this implies that $dF = \sqrt{4R^2 - x^2}dx$ where $x \in [R, 2R]$. $\xi_{0,2}$ is then easily obtained by integrating over x. It is noteworthy that, using similar approaches as above, all the expressions of $\xi_{n,m}$ n, $m = 0, 1, 2$ could be derived.

Let us focus now on the case where n_h is odd. It follows from (26), (8), and (13) that ϵ_k is given by

$$\epsilon_k = \left(\sum_{r=1}^{(n_h-1)/2} \zeta(F_{k,r})\right)^2 + 2\sum_{r=1}^{(n_h-1)/2} \Psi(F_{k,r}) \sum_{r=1}^{(n_h-1)/2} \zeta(F_{k,r}) + \frac{4R}{3}\sum_{r=1}^{(n_h-1)/2} \zeta(F) + \frac{4R^2}{9} - Z^2. \quad (45)$$

Recall here that Z refers to the random variable d^{Last}. It follows then from (45) that

$$\mathrm{E}\{\epsilon_k\} = \frac{n_h-1}{2N}\left(2\xi_{0,1}\left(\left(\frac{n_h-1}{2}-1\right)\left(\frac{\xi_{0,1}}{N}+\xi_{1,0}\right)+\frac{2R}{3}\right) + \xi_{0,2} + 2\xi_{1,1}\right). \quad (46)$$

Finally, using (41) and (41), (31) is obtained.

Following similar above steps, (32) is also obtained.

REFERENCES

[1] D. P. Agrawal and Q.-A. Zeng, *Introduction to Wireless and Mobile Systems*, 3rd ed. Boston, MA, USA: Cengage Learning, 2010.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[3] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: A survey," *IEEE Wireless Commun.*, vol. 11, no. 6, pp. 6–28, Dec. 2004.

[4] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: Possibilities and fundamental limitations based on available wireless network measurements," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 41–53, Jul. 2005.

[5] V. Lakafosis and M. M. Tentzeris, "From single-to multihop: The status of wireless localization," *IEEE Microw. Mag.*, vol. 10, no. 7, pp. 34–41, Dec. 2009.

[6] J. Rezazadeh, M. Moradi, A. S. Ismail, and E. Dutkiewicz, "Superior path planning mechanism for mobile beacon-assisted localization in wireless sensor networks," *IEEE Sensors J.*, vol. 14, no. 9, pp. 3052–3064, Sep. 2014.

[7] H. Shen, Z. Ding, S. Dasgupta, and C. Zhao, "Multiple source localization in wireless sensor networks based on time of arrival measurement," *IEEE Trans. Signal Process.*, vol. 62, no. 8, pp. 1938–1949, Apr. 2014.

[8] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *Proc. IEEE GLOBECOM*, San Antonio, TX, USA, Nov. 2001, pp. 2926–2931.

[9] 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, Nov. 2011.

[10] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proc. ACM MobiCom*, San Diego, CA, USA, Sep. 2003, pp. 81–95.

[11] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "DV-Loc: A scalable localization protocol using Voronoi diagrams for wireless sensor networks," *IEEE Wireless. Commun.*, vol. 16, no. 2, pp. 50–55, Apr. 2009.

[12] L. Gui, T. Val, and A. Wei, "Improving localization accuracy using selective 3-Anchor DV-hop algorithm," in *Proc. IEEE VTC*, San Francisco, CA, USA, Sep. 2011, pp. 1–5.

[13] C. Bettstetter and J. Eberspacher, "Hop distances in homogeneous ad hoc networks," in *Proc. IEEE VTC*, Jeju Island, South Korea, Apr. 2003, pp. 2286–2290.

[14] C. Buschmann, H. Hellbrück, S. Fischer, A. Kröller, and S. Fekete, "Radio propagation-aware distance estimation based on neighborhood comparison," in *Proc. EWSN*, Delft, The Netherlands, Jan. 2007, pp. 325–340.

[15] B. Huang, C. Yu, B. D. O. Anderson, and G. Mao, "Estimating distances via connectivity in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 14, no. 5, pp. 541–556, Apr. 2014.

[16] X. Ta, G. Mao, and B. D. O. Anderson, "On the probability of k-hop connection in wireless sensor networks," *IEEE Commun. Lett.*, vol. 11, no. 8, pp. 662–664, Aug. 2007.

[17] R. D. Yates and D. J. Goodman, *Probability and Stochastic Processes*, 2nd ed. New York, NY, USA: Wiley, 2004.

[18] X. Ta, G. Mao, and B. D. O. Anderson, "Evaluation of the probability of K-hop connection in homogeneous wireless sensor networks," in *Proc. IEEE GLOBECOM*, Washington, DC, USA, Nov. 2007, pp. 1279–1284.

[19] Y. Wang, X. Wang, D. Wang, and D. P. Agrawal, "Range-free localization using expected hop progress in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 10, pp. 1540–1552, Oct. 2009.

[20] L. Kleinrock and J. Silvester, "Optimum transmission radii for packet radio networks or why six is a magic number," in *Proc. IEEE NTC*, Birmingham, AL, USA, Dec. 1978, pp. 1–4.

[21] A. El Assaf, S. Zaidi, S. Affes, and N. Kandil, "Efficient range-free localization algorithm for randomly distributed wireless sensor networks," in *Proc. IEEE GLOBECOM*, Atlanta, GA, USA, Dec. 2013, pp. 201–206.

[22] A. El Assaf, S. Zaidi, S. Affes, and N. Kandil, "Range-free localization algorithm for heterogeneous Wireless Sensor Networks," in *Proc. IEEE WCNC*, Istanbul, Turkey, Apr. 2014, pp. 2805–2810.

[23] S. Vural and E. Ekici, "On multihop distances in wireless sensor networks with random node locations," *IEEE Trans. Mobile Comput.*, vol. 9, no. 4, pp. 540–552, Apr. 2010.

[24] M. Haenggi, "On distances in uniformly random networks," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3584–3586, Oct. 2005.

[25] J.-C. Kuo and W. Liao, "Hop count distribution of multihop paths in wireless networks with arbitrary node density: Modeling and its applications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 4, pp. 2321–2331, Jul. 2007.

[26] M. Li and Y. Liu, "Rendered path: Range-free localization in anisotropic sensor networks with holes," *IEEE/ACM Trans. Netw.*, vol. 18, no. 1, pp. 320–332, Feb. 2010.

[27] S. Zhang, J. Cao, C. Li-Jun, and D. Chen, "Accurate and energy-efficient range-free localization for mobile sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 6, pp. 897–910, Jun. 2010.

[28] A. El Assaf, S. Zaidi, S. Affes, and N. Kandil, "Accurate Nodes Localization in Anisotropic Wireless Sensor Networks," *Int. J. Distrib. Sensor Netw.*, vol. 2015, Apr. 2015, Art. no. 105682.

[29] B. Xiao, L. Chen, Q. Xiao, and M. Li, "Reliable anchor-based sensor localization in irregular areas," *IEEE Trans. Mobile Comput.*, vol. 9, no. 1, pp. 60–72, Jan. 2010.

[30] Q. Xiao, B. Xiao, J. Cao, and J. Wang, "Multihop range-free localization in anisotropic wireless sensor networks: A pattern-driven scheme," *IEEE Trans. Mobile Comput.*, vol. 9, no. 11, pp. 1592–1607, Nov. 2010.

[31] W. Dargie and C. Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice*, 1st ed. New York, NY, USA: Wiley, 2010.

[32] R. N. Duche and N. P. Sarwade, "Sensor node failure detection based on round trip delay and paths in WSNs," *IEEE Sensors J.*, vol. 14, no. 2, pp. 455–464, Feb. 2014.

[33] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 54–69, Jul. 2005.

[34] J. C. Chen, K. Yao, and R. E. Hudson, "Source localization and beamforming," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 30–39, Mar. 2002.

[35] S. Biaz, Y. Ji, B. Qi, and W. Shaoen, "Realistic radio range irregularity model and its impact on localization for wireless sensor networks," in *Proc. IEEE WiCOM*, Wuhan, China, Sep. 2005, pp. 669–673.

Slim Zaidi received the B.Eng. degree (Hons.) in telecommunications from the National Engineering School of Tunis, Tunis, Tunisia, in 2008, and the M.Sc. and Ph.D. degrees (Hons.) from Centre Énergie Matériaux Télécommunications, Institut National de la Recherche Scientifique, Université du Québec, Montreal, QC, Canada, in 2011 and 2015, respectively. He is currently a Postdoctoral Fellow with the University of Toronto, Toronto, ON, Canada. His research interests include MIMO and cooperative communications, wireless ad hoc networks, cellular technologies, and beyond 4G systems. He received twice the National Grant of Excellence from the Tunisian Government at both the M.Sc. (2009-2010) and the Ph.D. (2011-2013) programs. He also received a top-tier graduate Ph.D. Scholarship from the Natural Sciences and Engineering Research Council of Canada (2013-2015). He had to decline another prestigious Ph.D. Scholarship offered over the same period from the Fonds de Recherche du Québec Nature et Technologies (FRQNT). Recently, he received a prestigious Postdoctoral Fellowship from FRQNT (2016-2018).

Ahmad El Assaf received the B.S. degree from the Computer and Telecommunications Networks Engineering Department, Lebanese University, Saida, Lebanon, in 2007, and the M.Sc. degree from the University of Quebec in Abitibi-Témiscamingue, Val-d'Or, QC, Canada, in 2012.

He is currently pursuing the Ph.D. degree with the Centre Énergie Matériaux Télécommunications, Institut National de la Recherche Scientifique, Université du Québec, Montreal, QC, Canada. His research interests include localization algorithms, machine learning, optimization, and wireless sensor networks.

Sofiène Affes (SM'04) received the Diplôme d'Ingénieur degree in telecommunications and the Ph.D. degree (Hons.) in signal processing from the École Nationale Supérieure des Télécommunications, Paris, France, in 1992 and 1995, respectively. He was a Research Associate with INRS, Montreal, QC, Canada, until 1997, an Assistant Professor until 2000, and an Associate Professor until 2009. He is currently a Full Professor and Director of PERWADE, a unique U.S. $4M research training program on wireless in Canada involving 27 faculty members from 8 universities and 10 industrial partners. From 2003 to 2013, he was a Canada Research Chair in Wireless Communications. He has been a recipient of a Discovery Accelerator Supplement Award twice from NSERC, from 2008 to 2011 and from 2013 to 2016. He is an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS and the *Journal on Wireless Communications and Mobile Computing* (Wiley). He was previously an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He already served as a General Co-Chair of the IEEE VTC'2006-Fall and the IEEE ICUWB 2015, both held in Montreal, QC, Canada. For his contributions to the success of both events, he received a Recognition Award from the IEEE Vehicular Technology Society in 2008 and a Certificate of Recognition from the IEEE Microwave Theory and Techniques Society in 2015. He is currently serving as the General Chair of 28th IEEE PIMRC to be held in Montreal in the fall 2017.

Nahi Kandil has been a Professor with the Engineering School, University of Quebec in Abitibi-Témiscamingue since 2000. He has worked on research dealing with neural networks and their applications to wireless communications and power systems. He is now working mainly with localization and channel modeling in confined areas and mine environments.