Range-Free Nodes Localization in Mobile Wireless Sensor Networks

Slim Zaidi*, Ahmad El Assaf*, Sofiène Affes*†, and Nahi Kandil*†
*INRS-EMT, Université du Québec, Montreal, QC, H5A 1K6, Canada, Email: {zaidi,elassaf,affes}@emt.inrs.ca
†LRTCS, University of Quebec in Abitibi-Témiscaming, Rouyn-Noranda, QC, J9X 5E4, Canada, Email: nahi.kandil@uqat.ca

***Abstract*—In this paper, we propose a novel range-free localization algorithm tailored for mobile wireless sensor networks (MWSN)s. In contrast to the most existing range-free algorithms, the nodes mobility is taken into accounts when designing our algorithm. We show that nodes are able to estimate their positions using solely their locally-available information, thereby avoiding any unnecessary overhead and power costs incurred if information exchange between nodes was required. Furthermore, we show that the proposed algorithm outperforms in accuracy the best representative range-free algorithms. In contrast to the latter, it is able to compensate the nodes mobility effects when the nodes' speeds are moderate.**

***Index Terms*—Mobile wireless sensor networks (MWSN)s, localization accuracy, range-free techniques.**

I. Introduction

Due to their reliability, low cost, and ease of deployment, wireless sensor networks (WSNs) are emerging as a key tool for many applications such as environment monitoring, disaster relief, and target tracking [1]-[3]. A WSN is a set of small battery-powered sensors able to collect data from the surrounding environment and transmit it to a base station or an access point [4]. However, the sensing data are very often useless if the location from where they have been measured is unknown, making their localization a fundamental and essential issue in WSNs. So far, several localization algorithms have been proposed in the literature. These algorithms can be roughly classified into two categories: range-based and range-free.

To properly localize the regular or position-unaware nodes, range-based algorithms exploit the measurements of the received signals' characteristics such as the time of arrival (TOA) [5], the angle of arrival (AOA) [6], or the received signal strength (RSS) [7]. These signals are, in fact, transmitted by nodes with prior knowledge of their positions called anchors (or landmarks). Although range-based algorithms are very accurate, they are unsuitable for WSNs. Indeed, these algorithms require high power to ensure communication between anchors and regular nodes which are small battery-powered units. Furthermore, additional hardware is usually required at both anchors and regular nodes [8], thereby increasing the overall cost of the network. Moreover, the performance of these algorithms can be severely affected by noise, interference, and/or fading.

Unlike range-based algorithms, range-free algorithms, which rely on the network connectivity to estimate the regular node positions, are more power-efficient and do not require any additional hardware and, hence, are suitable for WSNs. Due to these practical merits, range-free localization algorithms have garnered the attention of the research community. So far, many range-free algorithms have been proposed in the literature [9]- [16]. These algorithms mainly fall into two categories: heuristic and analytical. The majority of heuristic algorithms are based on the DV-Hop algorithm [12]. The latter allows derivation of the network's average hop size from the global information of the WSN in a nonlocalized manner, thereby resulting in a prohibitive overhead and, hence, unnecessary high power consumption. Analytical range-free algorithms are, in contrast, more power efficient [14]-[26]. Indeed, these algorithms are based on an accurate analytical evaluation of the average hop size which can be locally computed at each node, thereby avoiding unnecessary power consumption. In spite of their valuable contributions, the algorithms developed in [14]-[26] are based on the assumption that all WSNs nodes are static. However, these nodes are usually designed to be mobile to provide the network with the sufficient flexibility and, hence, the capability to achieve advanced tasks such as in military and underground-mines application. This mobility, if not taken into account when designing the localization algorithm, may severely hinder its accuracy.

To fill this gap, we propose, in this paper, a novel range-free localization algorithm tailored for mobile wireless sensor networks (MWSN)s. In contrast to the most existing range-free algorithms, the nodes mobility is taken into accounts when designing our algorithm. We show that nodes are able to estimate their positions using solely their locally-available information, thereby avoiding any unnecessary overhead and power costs incurred if information exchange between nodes was required. Furthermore, we show that the proposed algorithm outperforms in accuracy the best representative range-free algorithms. In contrast to the latter, it is able to compensate the nodes mobility effects when the nodes' speeds are moderate.

The rest of this paper is organized as follows: Section II describes the system model and discusses the motivation for this work. Section III derives a new average hop size. A novel

Work supported by the CRD, DG, and CREATE PERSWADE <www.create-perswade.ca> Programs of NSERC and a Discovery Accelerator Supplement Award from NSERC.

range-free localization algorithm is proposed in section IV. Simulation results are discussed in Section V and concluding remarks are made in section VI.

Notation: Uppercase and lowercase bold letters denote matrices and vectors, respectively. $[\cdot]_{il}$ and $[\cdot]_i$ are the (i,l)-th entry of a matrix and i-th entry of a vector, respectively. $\mathbf{I}$ is the identity matrix. $(\cdot)^T$ denotes the transpose. $D\left(i,x\right)$ denotes the disc having the i-th node as a center and x as a radius.

II. Network model and motivation

We consider a system model of N MWSN nodes deployed in a 2-D square area S. Let v_i and $\theta_i(t)$ denote the average speed and the motion direction at t seconds (s) of the i-th node, respectively. All nodes are assumed to have the same range (i.e., transmission capability) denoted by R. Each node is hence 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 [15]. Due to the high cost of the global positioning system (GPS) technology, only a few nodes commonly known as anchors are equipped with it and, hence, are aware of their positions. The other nodes, called hereafter position-unaware or regular nodes for the sake of simplicity, are oblivious to this information. Let N_a and $N_r = N - N_a$ denote the number of anchors and regular nodes, respectively. Without loss of generality, let $\left(x_i(t), y_i(t)\right),\ i = 1,\ldots,N_a$ be the coordinates of the anchor nodes and $\left(x_i(t), y_i(t)\right),\ i = N_a+1,\ldots,N$ those of the regular ones. Fig. 1 illustrates a snapshot of the MWSN of our concern. The anchor nodes are marked with red triangles and the regular ones are marked with blue circles. If two nodes are able to directly communicate, they are linked with a dashed line that represents one hop. In the following, we propose an efficient range-free localization algorithm aiming to accurately estimate the regular nodes' positions.

In order to localize the $(i-N_a)$-th regular node (i.e., i-th node), the distances between it and at least 3 anchors are usually required. All anchors should then broadcast at $t = 0$ their coordinates through the network. Let $n_k(t_k)$ be the number of hops between the k-th anchor and the $(i-N_a)$-th regular node where t_k is the time at which the latter receives the first's coordinates. The $(i-N_a)$-th regular node estimates then its distance to the k-th anchor $d_{k-i}(t_k)$ as follows [17]

$$\hat{d}_{k-i}(t_k) = n_k(t_k)\bar{h}_{\rm s} \tag{1}$$

where $\bar{h}_{\rm s}$ is the average hop size throughout the network. Let us assume, without loss of generality, that the k-th received coordinates at the $(i-N_a)$-th regular node are those of the k-th anchor (i.e., $t_1 < t_2 < \ldots < T_{n_a}$). In such a case, the to-be-estimated coordinates are $(x_i(t_{N_a}), y_i(t_{N_a}))$ while the information available at the $(i-N_a)$-th regular node are $\hat{d}_{k-i}(t_k),\ k = 1,\ldots,N_a$. These information are unfortunately

Please note that $t = 0$ seconds corresponds to the beginning of the localization process.

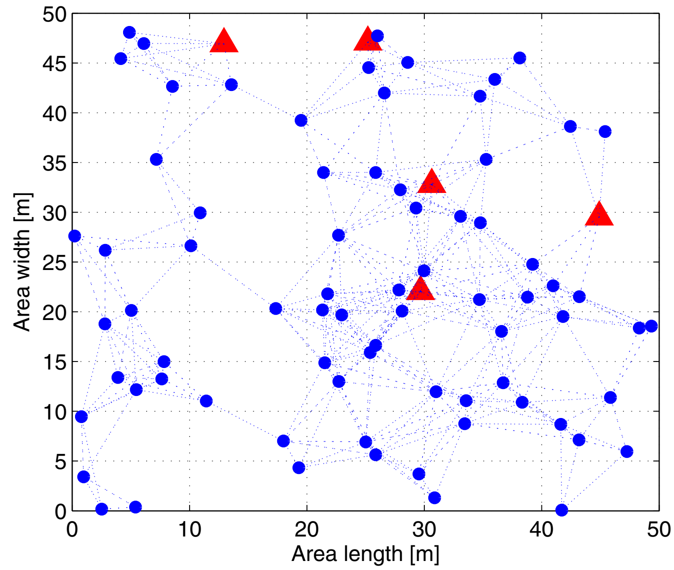

Fig. 1. Network model.

outdated, since the latter node is continuously in motion and, hence, $n_k(t_k) \neq n_k(t_{N_a})$ (i.e., $\hat{d}_{k-i}(t_k) \neq \hat{d}_{k-i}(t_{N_a})$) for $k = 1,\ldots,N_a-1$ may probably occur. If the latter situation is not taken into account, it would definitely hinder the $(i-N_a)$-th regular node localization accuracy. This motivate us to propose a new localization algorithm able to handle the WSNs nodes mobility. First let us derive in the next section the $\bar{h}_{\rm s}$'s expression that will be exploited later in our algorithm.

III. Average hop size's derivation

Let us consider a two-hop communication between the m-th and p-th nodes through an intermediate node n. Please note that, for the sake of simplicity, the effect of nodes' mobility is neglected when deriving $\bar{h}_{\rm s}$'s expression. For the sake of clarity, in what follows, we denote by X and Z the random variables that represent the distances d_{m-p} and d_{m-n}, respectively. In such a case, $\bar{h}_{\rm s}$ could be defined as

$$\bar{h}_{\rm s} = \mathrm{E}\left\{Z\right\}. \tag{2}$$

In order to derive the expectation in (2), we propose to exploit the conditional cumulative distribution function (CDF) $F_{Z|X}(z) = P\left(Z \leq z | x\right)$ of Z with respect to the random variable X. As can be shown from Fig. 2, $Z \leq z$ is guaranteed only if there are no nodes in the dashed area A. Therefore, the conditional CDF $F_{Z|X}(z)$ can be defined as

$$F_{Z|X}(z)\left(z\right) = P\left(Z \leq z|x\right) = P\left(\mathrm{E}_1\right), \tag{3}$$

where $P(\mathrm{E}_1)$ is the probability that the event $\mathrm{E}_1 = \{$no nodes in the dashed area $A\}$ occurs. Assuming that nodes are uniformly deployed in S, the probability of having K nodes in A follows a Binomial distribution $\mathrm{Bin}\left(N,p\right)$ where $p = \frac{A}{S}$. For relatively large N and small p, it can be readily shown that $\mathrm{Bin}\left(N,p\right)$ can be accurately approximated by a Poisson distribution $\mathrm{Pois}(\lambda A)$ where $\lambda = N/S$ is the average nodes

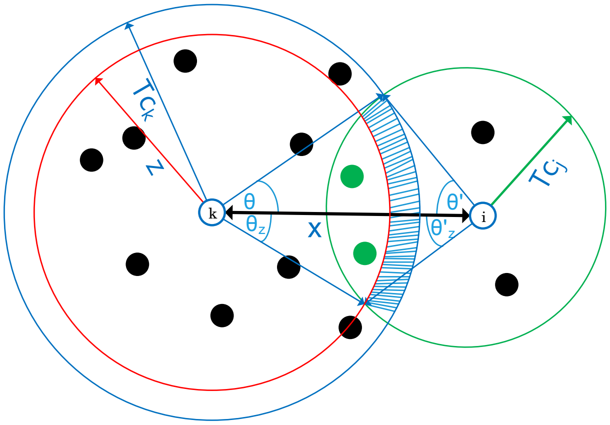

Fig. 2. EHP analysis.

density in the network [21]. Consequently, for a large number of nodes N and small p, we have

$$F_{Z|X}(z) = e^{-\lambda A}. \tag{4}$$

Using some geometrical properties and trigonometric transformations, it is straightforward to show that

$$A = R^2\left(\theta+\theta^{'}+\theta_z^{'}-\frac{\sin(2\theta)+\sin(2\theta^{'})+\sin(2\theta_z^{'})}{2}\right) - z^2\left(\theta_z-\frac{\sin(2\theta_z)}{2}\right), \tag{5}$$

where $\theta = \arccos\left(\frac{x^2}{2Rx}\right)$, $\theta^{'} = \arccos\left(\frac{x^2}{2Rx}\right)$, $\theta_z = \arccos\left(\frac{z^2-R^2+x^2}{2zx}\right)$, and $\theta_z^{'} = \arccos\left(\frac{R^2-z^2+x^2}{2Rx}\right)$.

Finally, the $\bar{h}_s$ is given by

$$\begin{aligned}\bar{h}_s =& \mathrm{E}_x\left(\alpha\left(1-F_{Z|X}(\alpha)\right)+\int_\alpha^R\left(1-F_{Z|X}(z)\right)dz\right)\\ =& \int_R^{2R}\left(\alpha\left(1-F_{Z|X}(\alpha)\right)+\int_\alpha^R\left(1-F_{Z|X}(z)\right)dz\right)f_X(x)dx,\end{aligned} \tag{6}$$

where $\alpha = x - R$ and $f_X(x)$ is the pdf of X. Note that the latter can be considered as a uniform random variable over $[R, 2R]$ and, hence, $f_X(x)$ can be substituted in the latter result by $1/R$. To the best of our knowledge, a closed-form expression for the integral in (6) does not exist. However, $\bar{h}_s$ can be easily implemented since it depends on finite integrals [25].

IV. The Proposed Algorithm

In this section, we propose a two-step localization algorithm. In the first step an initial guess of regular nodes' coordinates are computed using the information broadcasted by the anchors while, in the second step, a correction mechanism which accounts for the nodes' mobility is applied to enhance the localization accuracy.

A. First step: position computation

At $t = 0$, the k-th anchor broadcasts through the network a message containing (x_k, y_k, n) where n is the hop-count value initialized to one. When a node receives this message, 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 an another node, its database information is checked. If the k-th anchor information exists and the received hop-count value n is smaller than the stored n_k, the node updates n_k by n, increments by 1 then broadcasts the resulting message. If n_k is smaller than n, the node discards the received message. However, when the node is oblivious to the k-th anchor position, it adds this information to its database and forwards the received message after incrementing n by 1. This mechanism will continue until the become aware of all anchors' positions and their corresponding minimum hop count.

The computation of the $(i - N_a)$-th regular node coordinates could begins when the latter receives all the anchors' information. This means that the to be estimated coordinates are $(x_i(t_{N_a}), y_i(t_{N_a}))$ since $t_1 < t_2 < \ldots < T_{n_a}$ is assumed without loss of any generality. This node starts thus by estimating its distances to all anchors using (1). Exploiting this estimates, the $(i - N_a)$-th regular node (i.e., i-th node) is now able to compute an initial guess $(\hat{x}_i(t_{N_a}), \hat{y}_i(t_{N_a}))$ of its 2-D coordinates as

$$[\hat{x}_i(t_{N_a}), \hat{y}_i(t_{N_a})]^T = -\frac{1}{2}\left(\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^T\right)^{-1}\boldsymbol{\Upsilon}^T\boldsymbol{\kappa_i}. \tag{7}$$

where

$$\boldsymbol{\Upsilon} = \begin{bmatrix} x_1 - x_{N_a} & y_1 - y_{N_a} \\ x_2 - x_{N_a} & y_2 - y_{N_a} \\ \vdots & \vdots \\ x_{(N_a-1)} - x_{N_a} & y_{(N_a-1)} - y_{N_a} \end{bmatrix}, \tag{8}$$

and $\boldsymbol{\kappa_i}$ is a $(N_a - 1) \times 1$ vector with

$$[\boldsymbol{\kappa_i}]_n = \hat{d}_{1-i}(t_n)^2 - \hat{d}_{N_a-i}(t_{N_a})^2 + x_{N_a}^2 - x_1^2 + y_{N_a}^2 - y_1^2, \tag{9}$$

It is noteworthy from (8) and (9) that $\hat{x}_i(t_{N_a})$ and $\hat{y}_i(t_{N_a})$ are solely dependant on the anchors' coordinates $(x_k, y_k), k = 1, \ldots, N_a$ and the estimated distances $\hat{d}_{k-i}(t_k), , k = 1, \ldots, N_a$ which are all locally available at the $(i - N_a)$-th regular node. Therefore, their computation does not require any additional overhead (i.e., additional power cost), making our algorithm compliant with WSNs' power restrictions.

B. Second step: correction mechanism

Unfortunately, errors are expected to occur when estimating the distance between each regular node-anchor pair, thereby hindering localization accuracy. These errors are actually caused by the nodes' mobility, as discussed in Section II, and the fact that the distance estimates are obtained by mapping

Please note that, for the sake of clarity, we substitute, in what follows, $x_k(0)$ and $y_k(0)$ by x_k and y_k, respectively the hops into distance units as in (1). Let $\epsilon_{k-i}^{\mathrm{mob}}$ and $\epsilon_{k-i}^{\mathrm{map}}$ be the errors due to the first and second cause, respectively. Hence, we have

$$\epsilon_{k-i} = \hat{d}_{k-i}(t_k) - d_{k-i}(t_{N_a}), \tag{10}$$

where $\epsilon_{k-i} = \epsilon_{k-i}^{\mathrm{mob}} + \epsilon_{k-i}^{\mathrm{map}}$ Since these errors hinder localization accuracy, we have

$$\begin{cases} x_i(t_{N_a}) = \hat{x}_i(t_{N_a}) + \delta_{x_i} \\ y_i(t_{N_a}) = \hat{y}_i(t_{N_a}) + \delta_{y_i} \end{cases}, \tag{11}$$

where δ_{x_i} and δ_{y_i} are the location coordinates' errors to be determined. Exploiting the Taylor series expansion and retaining the first two terms, the following approximation holds:

$$\begin{aligned} d_{k-i}(t_{N_a}) \;\approx\; & d_{k-i}^{\dagger}(t_{N_a}) + \frac{\hat{x}_i(t_{N_a}) - x_k}{d_{k-i}^{\dagger}(t_{N_a})}\delta_{x_i} + \\ & \frac{\hat{y}_i(t_{N_a}) - y_k}{d_{k-i}^{\dagger}(t_{N_a})}\delta_{y_i}, \end{aligned} \tag{12}$$

where

$$d_{k-i}^{\dagger}(t_{N_a}) = \sqrt{\left(\hat{x}_i(t_{N_a}) - x_k\right)^2 - \left(\hat{y}_i(t_{N_a}) - y_k\right)^2}, \tag{13}$$

for $k = 1, 2, \ldots, N_a$. Therefore, rewriting (12) in a matrix form yields

$$\boldsymbol{\Gamma}_i \boldsymbol{\delta}_i = \boldsymbol{\zeta}_i - \boldsymbol{\epsilon}_i, \tag{14}$$

where

$$\boldsymbol{\Gamma}_{\mathbf{i}} = \begin{bmatrix} \frac{\hat{x}_i(t_{N_a}) - x_1}{d_{1-i}^{\dagger}(t_{N_a})} & & \frac{\hat{y}_i(t_{N_a}) - y_1}{d_{1-i}^{\dagger}(t_{N_a})} \\ \frac{\hat{x}_i(t_{N_a}) - x_2}{d_{2-i}^{\dagger}(t_{N_a})} & & \frac{\hat{y}_i(t_{N_a}) - y_2}{d_{2-i}^{\dagger}(t_{N_a})} \\ \vdots & \vdots & \vdots \\ \frac{\hat{x}_i(t_{N_a}) - x_{N_a}}{d_{N_a-i}^{\dagger}(t_{N_a})} & & \frac{\hat{y}_i(t_{N_a}) - y_{N_a}}{d_{N_a-i}^{\dagger}(t_{N_a})} \end{bmatrix}, \tag{15}$$

$$\boldsymbol{\zeta}_i = \begin{bmatrix} \hat{d}_{1-i}(t_{N_a}) - d_{1-i}^{\dagger}(t_{N_a}) \\ \hat{d}_{2-i}(t_{N_a}) - d_{2-i}^{\dagger}(t_{N_a}) \\ \vdots \\ \hat{d}_{N_a-i}(t_{N_a}) - d_{N_a-i}^{\dagger}(t_{N_a}) \end{bmatrix}, \tag{16}$$

$\boldsymbol{\epsilon}_i = [\epsilon_{1-i}, \epsilon_{2-i}, \ldots, \epsilon_{N_a-i}]^T$ and $\boldsymbol{\delta}_i = [\delta_{x_i}, \delta_{y_i}]^T$.

Many methods such as the weighted least squares (WLS) might be used to properly derive $\boldsymbol{\delta}_i$. Using WLS, the solution of (14) is given by :

$$\boldsymbol{\delta}_i = \left(\boldsymbol{\Gamma}_i^T \mathbf{P}_i^{-1} \boldsymbol{\Gamma}_i\right)^{-1} \boldsymbol{\Gamma}_i^T \mathbf{P}_i^{-1} \boldsymbol{\zeta}_i, \tag{17}$$

where $\mathbf{P}_i$ is the covariance matrix of $\boldsymbol{\epsilon}_i$. Since ϵ_{k-i} $k = 1, \ldots, N_a$ are independent random variables, $\mathbf{P}_i$ boils down to $\operatorname{diag}\left\{\sigma_{1-i}^2, \ldots, \sigma_{N_a-i}^2\right\}$ where σ_{k-i}^2 is the variance of ϵ_{k-i}. Since $\epsilon_{k-i}^{\mathrm{mob}}$ and $\epsilon_{k-i}^{\mathrm{map}}$ are independent random variable variables, $\sigma_{k-i}^2 = \sigma_{\mathrm{mob}}^2 + \sigma_{\mathrm{map}}^2$.

Now let us focus on σ_{map}^2. Assuming a high node density in the network, d_{k-i} could be approximated as follows

$$d_{k-i} \approx \sum_{j=1}^{n_k} h_j, \tag{18}$$

where h_j is the real size of the j-th hop which is a random variable itself. Using (18) and (10), we easily show that $\sigma_{\mathrm{map}}^2 = n_k \sigma_h^2$ where σ_h^2 is the variance of h_j given by

$$\begin{aligned} {\sigma_h}^2 = & \int_R^{2R} \left(\alpha^2 \left(1 - F_{Z|X}(\alpha)\right) + 2\int_\alpha^R z\left(1 - F_{Z|X}(z)\right) dz\right) f_X(x)\, dx - \\ & \left(\int_R^{2R} \left(\alpha \left(1 - F_{Z|X}(\alpha)\right) + \int_\alpha^R \left(1 - F_{Z|X}(z)\right) dz\right) f_X(x) dx\right)^2. \end{aligned} \tag{19}$$

On the other hand, $\epsilon_{k-i}^{\mathrm{mob}}$ could be considered as Uniform random variable over the interval $[-\left(t_{N_a} - t_k\right) V_i, \left(t_{N_a} - t_k\right) V_i]$. This is due to the fact that the $(i - N_a)$-th regular node motion direction could be in any direction $\theta_i \in [0, 2\pi]$ with the same probability. Therefore, we have

$$[\mathbf{P}_i]_{kk} = n_k \sigma_h^2 + \frac{\left(\left(t_{N_a} - t_k\right) V_i\right)^2}{3}. \tag{20}$$

A straightforward inspection of (20) and (17) reveals that $\boldsymbol{\delta}_i$ solely depends on the information locally available at the $(i - N_a)$-th regular node and, therefore, is locally computable at this node and does not require any additional information exchange between nodes. Moreover, since $\boldsymbol{\Gamma}_i^T \mathbf{P}_i \boldsymbol{\Gamma}_i$ is a 2-by-2 matrix, the entries of its inverse can be analytically and easily derived. Thus, the computation of $\boldsymbol{\delta}_i$ does not burden neither the implementation complexity of the proposed algorithm nor the overall cost of the network. Once we get $\boldsymbol{\delta}_i$, the value of $(\hat{x}_i(t_{N_a}), \hat{y}_i(t_{N_a}))$ is updated as $\hat{x}_i(t_{N_a})$=$\hat{x}_i(t_{N_a}) + \delta_{x_i}$ and $\hat{y}_i(t_{N_a}) = \hat{y}_i(t_{N_a}) + \delta_{y_i}$. The computations are repeated until δ_{x_i} and δ_{y_i} approach zero. In such a case, we have from (11) that $x_i(t_{N_a}) \approx \hat{x}_i(t_{N_a})$ and $y_i(t_{N_a}) \approx \hat{y}_i(t_{N_a})$ and, hence, more accurate localization is performed.

V. Simulations results

In this section, we evaluate the performance of the proposed algorithm in terms of localization accuracy by simulations using Matlab. These simulations are conducted to compare, under the same network settings, the proposed algorithm with some of the best representative range-free methods currently available in the literature, i.e., DV-Hop [12], LEAP [14]. All simulation results are obtained by averaging over 100 trials. In these simulations, nodes are deployed in a 2-D square area $S = 50 * 50\ m^2$. Furthermore, R and N_a are set to 18 and 20, respectively. We assume for simplicity that all nodes have the same speed V (i.e., $V_i = V$ $i = 1, \ldots, N$).

As an evaluation criterion, we opt to the normalized root mean square error (NRMSE) defined as follows

$$e = \frac{\sum_{i=1}^{N_u} \sqrt{\left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2}}{N_u T c_i}. \tag{21}$$

In Fig. 3, we plot the NRMSE versus $\alpha = V_i / V_{\mathrm{packet}}$ where V_{packet} is the packet propagation speed through the network. In Figs. 4, 5, and 6, we plot the NRMSE, its standard deviation, and its CDF versus the node density λ, respectively.

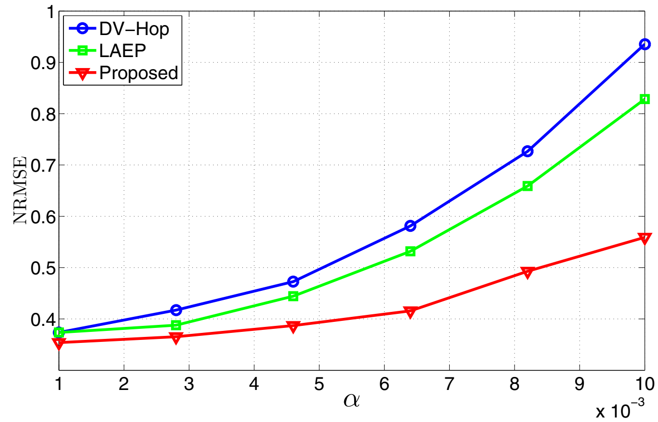

Fig. 3. Localization NRMSE vs α for $\lambda = 0.02$.

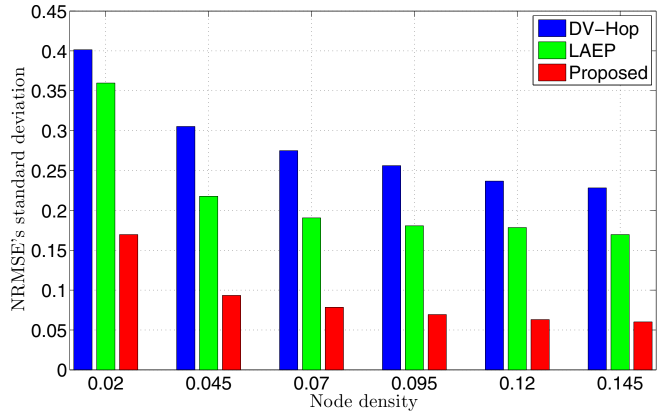

Fig. 5. Localization NRMSE's standard deviation for $\alpha = 4.510^{-3}$.

Fig. 3 plots the localization NRMSE achieved by DV-Hop, LEAP and the proposed algorithm for versus α for $\lambda = 0.02$. As can be shown from this figure, the three algorithms accuracies deteriorate as expected when α increases. However, the proposed algorithm outperforms its counterparts. The NRMSEs achieved by the latter rapidly decreases with α while that achieved by the proposed algorithm slightly decreases. Furthermore, from this figure, our algorithm its able to provide a maximum accuracy when α is small. This means that it is able to compensate the nodes mobility effects for moderate nodes speed. All this proves the superiority of the proposed algorithm over its counterparts.

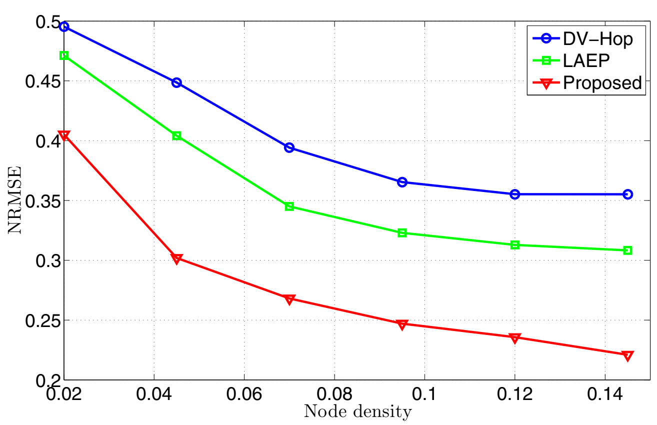

Fig. 4. Localization NRMSE vs the node density for $\alpha = 4.510^{-3}$.

Fig. 4 plots the localization NRMSE achieved by DV-Hop, LEAP, and the proposed algorithm versus λ for $\alpha = 4.510^{-3}$. As can be shown from this figure, all algorithms accuracies improve as expected when the node density increases. However, the proposed algorithm always outperforms its counterparts. Indeed, our proposed algorithm turns out to be until about two and three times more accurate than LEAP and DV-Hop respectively.

Fig. 5 plots the NRMSE's standard deviation achieved by all localization algorithms when $\alpha = 4.510^{-3}$. As can be observed from this figure, the one achieved by the proposed algorithm substantially decreases when the node density increases while those achieved by the other algorithms slightly decrease. Furthermore, the NRMSE standard deviation achieved by the proposed algorithm approaches zero. This means that implementing our algorithm in MWSNs guarantees a very accurate localization for any given realization. This result is very interesting in terms of implementation strategy, since it proves that the result in Fig. 4 becomes more and more meaningful as λ grows large.

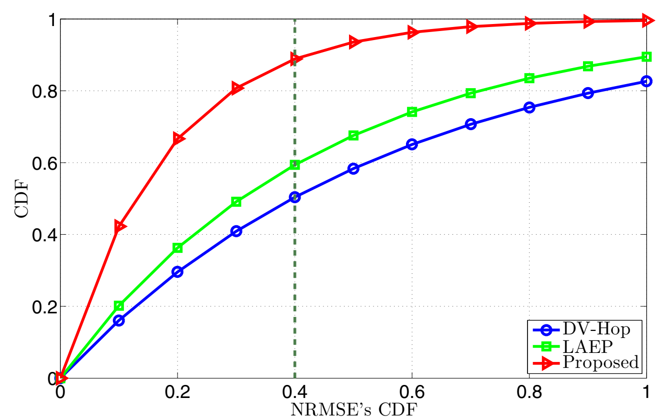

Fig. 6. Localization NRMSE's CDF for $\alpha = 4.510^{-3}$.

Fig. 6 illustrates the localization NRMSE's CDF for $\alpha = 4.510^{-3}$. Using the proposed algorithm, 90% of the regular nodes could estimate their position within $0.4R$. In contrast, 60% of the nodes achieve the same accuracy with LEAP and about 50% with DV-Hop. This further proves the efficiency of the proposed algorithm.

VI. Conclusion

In this paper, a novel range-free localization algorithm suitable for MWSNs was proposed. In contrast to the most existing range-free algorithms, the nodes mobility is taken into accounts when designing our algorithm. It was shown that nodes are able to estimate their positions using solely their locally-available information, thereby avoiding any unnecessary overhead and power costs incurred if information exchange between nodes was required. It was also shown that the proposed algorithm outperforms in accuracy the best representative range-free algorithms. In contrast to the latter, it is able to compensate the nodes mobility effects when the nodes' speeds are moderate.

References

[1] W. Zhang and G. Cao, "DCTC: Dynamic Convoy Tree-Based Collaboration for Target Tracking in Sensor Networks," *IEEE Trans. Wireless Comm.,* vol. 3, no. 5, pp. 1689-1701, Sept. 2004.

[2] S. Zaidi and S. Affes, "Distributed collaborative beamforming in the presence of angular scattering," *IEEE Trans. Commun.*, vol. 62, pp. 1668-1680, May 2014.

[3] F. Gustafsson and F. Gunnarsson, "Mobile Positioning Using Wireless Networks: Possibilities and Fundamental Limitations Based on Available Wireless Network Measurements," *IEEE Signal Process. Magazine,* vol. 22, no. 4, pp. 41-53, July 2005.

[4] D.P. Agrawal and Q.-A. Zeng, *Introduction to Wireless and Mobile Systems,* 3^{rd} edition CL-Engineering June 2010.

[5] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. ODea, "Relative Location Estimation in Wireless Sensor Networks," *IEEE Trans. Signal Process.,* vol. 51, no. 8, pp. 2137-2148, August 2003.

[6] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) Using AOA," *IEEE INFOCOM'2003,* San Francisco, vol. 3, pp. 1734-1743, March 2003.

[7] H. Ren and M. Q.-H. Meng, "Power Adaptive Localization Algorithm for Wireless Sensor Networks Using Particle Filter," *IEEE Trans. Vehicular Techno.,* vol. 58, no. 5, pp. 2498-2508, June 2009.

[8] X. Cheng, H. Shu, Q. Liang, and D. H-C. Du, "Silent Positioning in Underwater Acoustic Sensor Networks," *IEEE Trans. Vehicular Techno.,* vol. 57, no. 3, pp. 1756-1766, May 2008.

[9] Q. Xiao, B. Xiao, J. Cao, and J. Wang,"Multihop Range-Free Localization in Anisotropic Wireless Sensor Networks: A Pattern-Driven Scheme," *IEEE Trans. on Mobile Comp.,* vol. 9, no. 11, pp. 1592-1607, November 2010.

[10] Z. Ziguo and T. He,"RSD: A Metric for Achieving Range-Free Localization beyond Connectivity," *IEEE Trans. Parallel and Distributed Sys.,* vol. 24, no. 11, pp. 1943-1951, November 2011.

[11] J. Lee, B. Choi, and E. Kim, "Novel Range-Free Localization Based on Multidimensional Support Vector Regression Trained in the Primal Space," *IEEE Trans. on Neural Networks and Learning Systems,* vol. 24, no. 7, pp. 1099-1113, July 2013.

[12] D. Niculescu and B. Nath, "DV Based Positioning in Ad Hoc Networks," *J. Telecomm. Systems,* vol. 22, no. 4, pp. 267-280, 2003.

[13] J-P. Sheu, P-C. Chen, and C.-S. Hsu, "A Distributed Localization Scheme for Wireless Sensor Networks with Improved Grid-Scan and Vector-Based Refinement," *IEEE Trans. Mobile Comp.,* vol. 7, no. 9, pp. 1110-1123, September 2008.

[14] Y. Wang, X. Wang, D. Wang, and D. P. Agrawal, "Range-Free Localization Using Expected Hop Progress in Wireless Sensor Networks," *IEEE Trans. Parallel and Distributed Sys.,* vol. 25, no. 10, pp. 1540-1552, October 2009.

[15] A. El Assaf, S. Zaidi, S. Affes, and N. Kandil, "Efficient Range-Free Localization Algorithm for Randomly Distributed Wireless Sensor Networks," *IEEE GLOBECOM'2013,* Atlanta, GA, December 9-13, 2013.

[16] D. Ma, M. J. Er, and B. Wang, "Analysis of Hop-Count-Based Source-to-Destination Distance Estimation in Wireless Sensor Networks With Applications in Localization," *IEEE Trans. Vehicular Techno.,* vol. 59, no. 6, pp. 2998-3011, July 2010.

[17] A. EL Assaf, S. Zaidi, S. Affes, and N. Kandil, "Range-free localization algorithm for anisotropic wireless sensor networks," *IEEE VTC'2014-Fall,* Vancouver, Canada, Sep. 14-17, 2014.

[18] L. Kleinrock and J. Silvester,"Optimum Transmission Radii for Packet Radio Networks or Why Six is a Magic Number," *in Proc. IEEE Nat. Telecommun. Conf.,* pp. 4.3.14.3.5, 1978.

[19] A. EL Assaf, S. Zaidi, S. Affes, and N. Kandil, "Range-free localization algorithm for heterogeneous wireless sensors networks," *IEEE WCNC'2014,* Istanbul, Turkey, Apr. 6-9, 2014.

[20] J. C. Kuo, and W. Liao,"Hop Count Distribution of Multihop Paths in Wireless Networks With Arbitrary Node Density: Modeling and Its Applications," *IEEE on Trans. Vehicular Technology,* vol. 56, no. 4, pp. 2321-2331, July 2007.

[21] A. EL Assaf, S. Zaidi, S. Affes, and N. Kandil, "Cost-effective and Accurate Nodes Localization in Heterogeneous Wireless Sensor Networks," *IEEE ICC'2015,* London, United Kingdom, June. 8-12, 2015.

[22] J.C. Kuo, W. Liao and T.C. Hou,"Impact of Node Density on Throughput and Delay Scaling in Multi-hop Wireless Networks," *IEEE on Trans. Wireless Commun.,* vol. 8, no. 10, pp. 5103-5111, October 2009.

[23] A. EL Assaf, S. Zaidi, S. Affes, and N. Kandil, "Accurate Nodes Localization in Anisotropic Wireless Sensor Networks". *Inter. J. Dist. Sensor Net.*, vol. 2015, pp. 1-17, April 2015.

[24] B. Xiao, L. Chen, Q. Xiao and M. Li, "Reliable Anchor-Based Sensor Localization in Irregular Areas," *IEEE Trans. on Mobile Comp.,* vol. 9, no. 1, pp. 60-72, Janvier 2010.

[25] A. EL Assaf, S. Zaidi, S. Affes, and N. Kandil, "Low-cost localization for multi-hop heterogeneous wireless sensor networks". Accepted for publication in *IEEE Trans. Wireless. commun.*, August 2015.

[26] S. Zhang, J. Cao, L. Chen, and D. Chen, "Accurate and Energy-Efficient Range-Free Localization for Mobile Sensor Networks," *IEEE Trans. on Mobile Comp.,* vol. 9, no. 6, pp. 897-910, June 2010.

[27] D. E. Manolakis, "Efficient Solution and Performance Analysis of 3-D Position Estimation by Trilateration," *IEEE Trans. Aerospace and Electonic Sys.,* vol. 32, no. 4, pp. 1239-1248, October 1996.

[28] C. Bettstetter, "On The Connectivity of Wireless Multihop Networks With Homogeneous and Inhomogeneous Range Assignment, "*Proc. IEEE VTC'2002-Fall,* vol. 3, pp. 1706-1710, 2002.

[29] C. Bettstetter, "On the Connectivity of Ad Hoc Networks," *The Computer Journal,* vol. 47, no. 4, pp.432-447, 2004