IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 1, 2019, accepted February 20, 2019, date of publication March 13, 2019, date of current version April 9, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2904731*

# LEMOn: Wireless Localization for IoT Employing a Location-Unaware Mobile Unit

**CAM LY NGUYEN[1,2] AND USMAN RAZA[3]**

[1]Network System Laboratory, Corporate Research & Development Center, Toshiba Corporation, Kawasaki 212-8582, Japan
[2]Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-0033, Japan
[3]Toshiba Research Europe Ltd., Bristol BS1 4ND, U.K.

Corresponding author: Cam Ly Nguyen (camly.nguyen@toshiba.co.jp)

This work was supported by the Corporate Research & Development Center, Toshiba Corporation.

**ABSTRACT** In recent years, much attention has been paid to wireless localization schemes that exploit receptions of messages sent by a mobile unit. However, existing methods assume an accurate knowledge of the location of the mobile unit and a precise propagation model of the actual radio environment. By getting rid of these two requirements, our proposed localization algorithms make mobility-assisted localization far more practical as we do not need to equip the mobile unit with a global positioning system or run a time-consuming campaign to survey radio environment. *LEMOn* estimates the position of target nodes by using known locations of a small set of fixed anchor nodes while receiving messages sent from a mobile unit from unknown arbitrary locations. *LEMOn-M*, on the other hand, solves the localization matching problem by mapping an arbitrary number of target nodes to the known set of locations. Both algorithms first estimate an inter-node distance using a similarity between received signal strength indicators of beacons received from the mobile unit. The conventional location estimators are then employed to localize target nodes with an unknown location. The obvious examples of real-world applications include but are not limited to unmanned aerial vehicle assisted wireless sensor networks and indoor IoT systems. The various simulations show that the two algorithms achieve a very high localization accuracy even in harsh radio environments while static localization techniques fail.

**INDEX TERMS** Wireless localization, UAV, WSNs, mobility-assisted localization, IoT, sensor networks, RSSI, cooperative localization, localization matching.

## I. INTRODUCTION

With improvements in wireless communication technology over the last decades, the Internet of Things (IoT) is now widely deployed in smart cities, buildings, and houses [1]. Typically, an IoT system consists of low cost and low power devices which interact with each other through the Internet. The main goal of IoT is to ensure every device, including sensors, smart-phones, wearable sensors, tablets, transportation system, etc., can connect with each other through a common interface. This allows machine-to-machine (M2M) communication without human intervention [2], thus can reduce manual cost. There are numerous key issues in IoT including wireless localization. Wireless localization, which refers to extracting geo-location information of an object, has therefore been well researched and developed [3]. Although a Global Positioning System (GPS) module can provide location information of a wireless device, this incurs additional cost and huge power consumption. GPS is thus not always suitable for many IoT applications. To alleviate such problems, several radio-frequency (RF) based localization methods have been developed to estimate the devices' location. These methods measure either devices' proximity (i.e. connectivity) [4], Angle of Arrival (AoA) [5], Time of Arrival (ToA) [6], Time Difference of Arrival (TDoA) [7], or Received Signal Strength Indicator (RSSI) [8], [9] to localize target devices deployed at unknown locations. In contrast to AoA measurements and time-based measurements that require additional hardware and techniques, RSSI can be obtained from almost all wireless hardware. Therefore, RSSI based localization is an cost-effective solution for most IoT applications. This paper, therefore, considers localization problems using RSSI. Typically, an RSSI value $r$ is estimated through the log-distance propagation

The associate editor coordinating the review of this manuscript and approving it for publication was Zhen Li.

model [10].

$$r = P - 10\eta \log_{10} d + X \quad [\text{dBm}] \tag{1}$$

where $d$ is the distance between the transmitter and the receiver, $\eta$ is the path loss exponent, $P$ is a reference power value measured in dBm at a distance of one meter from the transmitter, and $X$ is a random variable characterizing the ranging error caused by multi-path fading and shadowing.

Depending on network paradigms, RF-based localization can be broadly divided into two categories: inter-node communication based localization schemes and mobility-assisted localization schemes. The former type of schemes (referred as *static localization* schemes in this paper), first estimate the Euclidean distances between communicating nodes using inter-node RF measurements, then localize target devices using either multi-lateration methods [11] or cooperative localization methods [8], [12]–[14]. These schemes are deemed suitable for mesh network paradigms. On the other hand, mobility-assisted localization schemes use a wireless mobile unit, which is aware of its location (e.g., equipped with GPS), to assist in estimating the location of target wireless nodes [15], [16]. A mobile unit can use an Unmanned Aerial Vehicle (UAV), a drone, a ground vehicle, etc.

A node could be any device that equipped with a wireless hardware, position of which is often assumed to be fixed or changed only infrequently. In this paper, we call these devices as *sensor nodes* to distinguish from the mobile unit. Unlike the power-constrained sensors, the energy of the mobile unit is typically assumed to be unconstrained. Therefore, these mobility-assisted localization methods are more advanced and practical than static localization methods as reliable inter-node distance is hard to obtain for the latter schemes [16]. For instance, while static localization techniques suffer from the problem of non-line-of-sight (NLoS), e.g. due to obstruction, line-of-sight (LoS) communication links can be established between the UAV and sensor nodes over time when UAV moves. Besides, in sparse node deployments, a nodes may not be able to inter-connect due to their energy constraints. A more powerful UAV, however, can often connect to every sensor node and can thus help localize these nodes. With the rapid development of UAV-assisted WSNs [17], mobility-assisted localization systems are attracting greater attention [18].

However, all of the above localization techniques suffer from the following limitations.

- *Difficulty in Estimating Accurate path loss exponent*: The value of path loss exponent $\eta$ depends on the measured environment. For instance, it is approximately 2 for free space, and varies from 3 to 5 in shadowed urban cellular radio [19]. Estimating accurate values of these parameters is often prohibitively expensive in practice, and is even impossible in human-inaccessible environments, for instance, WSNs for post-disaster monitoring.
- *Difficulty in the Calibration Process*: An RSSI value is a function of the calibration of both the transmitter and receiver. Transmitted powers and received powers vary from device to device due to use of different hardware components. Wireless nodes might be designed to measure and report their own calibration data to each other. This, however, complicates the design [14].
- *Large Ranging Error*: Even if calibration process can be reliably done and path loss exponent is estimated accurately, there is a ranging error caused by multi-path fading and shadowing. These effects can be reduced only if sensor nodes are equipped with specific hardware or technologies. For instance, multiple antennas can mitigate the effect of shadowing [20] and a spread-spectrum method can reduce the effect of frequency-selective fading [14]. However, an ordinary wireless node is not always equipped with such hardware. A calculation in Appendix illustrates that ranging error can be up to six times larger than the distance itself (see Appendix).
- *In Mobility-assisted Localization: Uncertainty of the Mobile Unit's Position*: Certain application deployment areas or environmental conditions can prevent from accurately determining position of the mobile unit [21]. These factors may include unavailability of the GPS in indoor environments or under poor weather conditions. Even if GPS is available, determining accurate position of the mobile unit is challenging in real-time, especially for a high-speed mobile unit. A standard non-differential GPS receiver has an error of 5 m to 10 m. In addition, delay caused by a low update rate, which varies from 1 to 10 Hz on average, and GPS response further increases the positioning error of a high-speed UAV. Moreover, the GPS position accuracy is known to deteriorate when fewer satellites are reachable [22]. To the best of our knowledge, these issues have not been considered and solved in mobility-assisted techniques.

To eliminate the above drawbacks of existing mobility-assisted localization methods and static localization methods, we propose two localization techniques that can combine the strength of both schemes. In our network model, we assume that a mobile unit broadcasts beacon messages (referred to as *beacons* for short) periodically to sensor nodes. Unlike conventional mobility-assisted localization techniques, we assume that neither the position of the mobile unit nor the signal propagation model is known. However, we assume that the following information is known: *1)* Position of some of the fixed sensor nodes also called *anchor nodes*, or *2)* A set of node positions, however, we do not know which node locates at which position. The former localization problem is often found in literature [12], [14]. On the other hand, the latter localization problem has been investigated only recently [8] and is called as the wireless *localization matching* problem. Using the RSSI values estimated from the beacon transmissions we propose two methods called *LEMOn*[1] and *LEMOn-M*[2] to respectively resolve the above

[1]LEMOn: Localization Employing a location-unaware MObile unit
[2]LEMOn-M: LEMOn for localization matching

**TABLE 1. Comparison of localization techniques.**

| | Static localization | Mobility-assisted localization | LEMOn, LEMOn-M |
|---|---|---|---|
| **Method** | Measure distance between sensor nodes using inter-node RF measurements | Measure distance between the mobile unit and a sensor node using their RF measurements | Measure distance between sensor nodes using RSSI measurements to the mobile unit |
| **Advantages** | Do not require the mobile unit | Do not require to fix anchor devices. Localization accuracy can be improved by increasing the number of beacons | Do not require calibration, propagation model parameter, and location of the mobile unit |
| **Requirements** | Require calibration, accurate propagation model and positions of anchor nodes | Require calibration, accurate propagation model, and accurate location of the mobile unit | Require a set of anchor nodes or a set of node positions |
| **Applications** | Localization in static WSNs | Localization in UAV assisted WSNs in environments that accurate location of the UAV can be obtained accurately | Localization using a mobile unit whose location is not obtained accurately |

two problems. The main advantages of the proposed LEMOn and LEMOn-M are:

- *Suitable for both indoor and outdoor environments* as the location of the mobile unit is not required,
- *Easily applicable* as these methods do not require system calibration as well as a priori measurements to estimate the path loss exponent,
- *Robust to noise* as these methods use statistical features of RSSI measurements rather than a single or few RSSI measurement that indeed fluctuates a lot.

The characteristics of the above localization techniques are summarized in Table 1. The main contributions of this paper are as follows:

- We define and motivate new localization problems that are hybrid between static localization and mobility-assisted localization.
- We propose and analytically study the performance of LEMOn and LEMOn-M as candidate solutions to the above problems,
- We highlight some potential applications related to real-world scenarios.

The remainder of this paper is organized as follows. In Section II, we describe the system model and problem definitions. In Section III, we highlight conventional work on mobility-assisted localization and localization estimation algorithms with a focus on algorithms that are used in this paper. In Section IV, we present the details of our proposed LEMOn and LEMOn-M techniques. In Sections V and VI, we evaluate the performance of the above algorithms through an extensive simulation-based study. In Section VII, we highlight potential real-world applications. Finally, we conclude this paper with a summary of our results and discuss our future plans in Section VIII.

## II. SYSTEM MODEL AND PROBLEM DEFINITIONS

Consider a wireless network consisting of $N$ *sensor nodes*, $n_1, n_2, \ldots, n_N$ that are deployed in a domain of interest $\mathcal{D}$. A *mobile unit* moving along an arbitrary trajectory inside a domain of interest $\mathcal{V}$ broadcasts beacons periodically to all sensor nodes. Each beacon includes its unique sequence number $k$, $k \in \{1, 2, 3, \ldots, K\}$ ($K$ is the *number of beacons*). We assume that signals transmitted by the mobile unit are strong enough to reach all wireless sensor nodes. Each sensor node $n_i$ receives the beacon transmissions, estimates their RSSI values, then constructs a vector $\boldsymbol{r_i} \in \mathbf{R}^K$ with entries $r_{i,k}$ equal to the RSSI value retrieved from the $k$-th beacon. Node $n_i$ then sends its corresponding vector $\boldsymbol{r_i}$ back to a server (potentially via the mobile unit) for post-processing. The server is then tasked with estimating the locations of the target nodes. Without loss of generality, $\mathcal{D}$ and $\mathcal{V}$ are assumed as two-dimensional spaces, i.e. $\mathcal{D}, \mathcal{V} \subset \mathbf{R}^2$. We assume that either a wireless node or the mobile unit is equipped with an isotropic antenna. RSSI $r_{i,k}$ is, therefore, related with distance $d_{i,k}$ between a wireless nodes $n_i$ and the mobile unit when it sends the $k$-th beacon through the log-distance propagation model [10].

$$r_{i,k} = P_i - 10\eta \log_{10} d_{i,k} + X_{i,k} \quad [\text{dBm}] \tag{2}$$

where $\eta$ is the path loss exponent, which is approximated to 2 for free space, and a value between 3 and 5 for urban environments. Since wireless nodes can be deployed in any environment, we assume that $\eta$ is an unknown constant but is identical for all nodes. $P_i$ is a reference power corresponding to node $n_i$ value at a distance of one meter from the transmitter, and $X$ is a random variable characterizing the noise factors. The reference power $P_i$ corresponding to node $n_i$ is an unknown constant, as sensor nodes are assumed not to be calibrated. The distribution of $X$ depends on the wireless propagation environment. For example, the long-term signal variation is known to follow the Log-normal distribution, whereas the short-term signal variation can be described by several other distributions, such as the Rayleigh distribution. For simplicity, in our analysis, $X$ is assumed to follow the Log-normal distribution, namely Gaussian distribution in dB, i.e. $X \sim \mathcal{N}(0, \sigma_X^2)$, where the standard deviation $\sigma_X$ can be as low as three [8] and as high as 12 [14].

We define two different problems, called the *localization* problem and the *localization matching* problem, which are motivated through real-world applications described in Section VII.

### A. THE LOCALIZATION PROBLEM

The localization problem can be defined in a similar way as in [14]. Given a set $\mathcal{A}$ comprising of $A$ ($A < N$) *anchor nodes* with known-positions (acquired through GPS or or as a result of deployment process), the problem is to estimate the location of other sensor nodes, called *target nodes*.

### B. THE LOCALIZATION MATCHING PROBLEM

The wireless localization matching problem, which is a variation of localization problems [8], is defined as follows. Given a set of $N$ positions $p_1, p_2, \ldots, p_N$ where $N$ sensor nodes are located, the problem is to correctly match each node label $n_i$ with its correct position $p_{i'}$. In the beginning, it is unknown which position $p_{i'}$ node $n_i$ is located at.

## III. BACKGROUND AND RELATED WORK

Since our proposed algorithms adopt a hybrid approach based on mobility-assisted and static localization techniques, this section highlights existing work on mobility-assisted localization as well as localization estimation algorithms for static networks. Section III-A describes a brief history of mobility-assisted localization and differentiates these localization techniques from our proposals. Section III-B provides a background on localization estimation algorithms for static networks, mainly focusing on the representative algorithms used in this paper.

### A. MOBILITY-ASSISTED LOCALIZATION

Mobility-assisted localization schemes locate target nodes using mobile units that are aware of their own location and are capable of moving around the target nodes arbitrarily. The mobile units periodically broadcast beacons with their location, enabling the nearby target nodes to hear this information and estimate their own location [18].

RSSI-based mobility-assisted localization techniques are pioneered by Sichitiu and Ramadurai [15]. They use a wireless device carrying truck that broadcasts messages containing its known location to sensor nodes deployed in an outdoor environment. The nodes estimate RSSI values of the received messages to determine their distance to the truck. The nodes finally calculate their locations using a probabilistic method. Menegatti *et al.* [23] estimate the location of a robot while mapping the nodes simultaneously using RSSI measurements and odometry from the robot. They use a log-distance propagation model, which is calibrated before the experiments to calculate the distance between the robot and sensor nodes. Caballero *et al.* [24] use a robot equipped with Differential GPS (DGPS) moving in an outdoor parking lot to measure RSSI values from neighboring nodes. Similar to the above approaches, the propagation model is calibrated before the experiments.

Besides, mobility can be combined with time-based ranging techniques to localize the sensor nodes. For instance, Sun and Guo [25] use a mobile beacon traversing deployed area of sensor network and broadcasting location-containing packets. On receiving the beacon packets, nodes combine the received locations with the time of arrival of the packets to calculate its own location. Localization is performed using either non-parametric or parametric probabilistic estimation techniques.

In contrast to the ranging techniques described above, some range-free approaches [26]–[28] use connectivity information for locating nodes, thus no extra hardware or data communication is needed for the sensor nodes.

Recently, with the rapid research and development on UAV, there are numerous works on using a UAV to localize unknown devices [29], [30]. For instance, Villas *et al.* [30] use a UAV equipped with GPS broadcasting its geo-location when flying over the monitoring area. Using these 3D geo-locations and the corresponding RSSI values, sensor nodes can calculate its 3D location. Yang *et al.* [29] use a UAV carrying GPS and a camera to collect sensor node images. Locations of non-occluded nodes are then determined using image processing techniques. These nodes are then used as anchor nodes to localize occluded nodes using RSSI ranging localization techniques.

It is, however, worth noting that all of the above techniques require an accurate distance estimation method (i.e. calibration of the propagation model for ranging techniques, or connectivity model for range-free techniques), and an accurate mobile unit position, which can be hard to obtain in practice [21].

Recently, an RSSI-based localization method [31] is proposed that relaxes the requirement for the UAV to know its location. This work, however, can localize sensor nodes that are equipped with the exactly same hardware for the wireless devices. It is, therefore, less practical than the proposals in this paper. To overcome such limitations, the proposed methods in this paper, to the best of our knowledge, are the first ones that require neither calibration nor location of the mobile unit.

### B. LOCATION ESTIMATION ALGORITHMS

Location estimation algorithms (see [32] for a detailed survey) can be broadly divided into two categories: non-cooperative methods and cooperative methods. In non-cooperative localization, distance measurements are made only between anchor nodes and target nodes. Each target node estimates its distance to the anchor nodes using measured RSSI values, then uses localization algorithm such as multilateration [11] to locate itself. These methods are suitable for either target tracking problem, or mobility-assisted localization.

On the other hand, cooperative localization methods, where distance measurements between target nodes are also made, estimate all node positions simultaneously rather than localizing each target node individually. These methods enhance localization accuracy of non-cooperative techniques by using more measurements. These methods are, therefore, ideal for wireless mesh networks where nodes can communicate with each other [14]. There are numerous localization algorithms such as Multi-Dimensional Scaling (MDS) [33], Semi-Definite Programming (SDP) [12], stochastic optimization (e.g. simulated annealing (SA) [34]), and localization matching [8].

All above methods can determine nodes' location given estimated distances between them. This paper focuses on proposing new distance estimation techniques rather than

new location estimators. Therefore, we use the existing location estimators. We choose SDP localization method for resolving the localization problem because it can output deterministic solutions and can also enhance accuracy compared to other methods [12]. On the other hand, a localization matching method called MLMatch [8] is the unique location estimator that can solve the localization matching problem.

The remainder of this section depicts a high-level description of SDP localization and MLMatch, while more details can be found in [8] and [12].

#### 1) SDP LOCALIZATION METHOD

The problem can be stated as follows: given a set of anchors $\mathcal{A}$ with known locations $\boldsymbol{a}_j$, $(j \in \mathcal{A})$, a set of target nodes $\mathcal{T}$ whose locations are unknown, and some estimated distances $\hat{d}_{i,j}(i, j \in \mathcal{A} \cup \mathcal{T})$, find $\boldsymbol{x}_i$, $(i \in \mathcal{T})$ the location of target nodes, such that:

$$\begin{aligned} \|\boldsymbol{x}_i - \boldsymbol{x}_j\| &= \hat{d}_{i,j}, \quad i, j \in \mathcal{T} \\ \|\boldsymbol{x}_i - \boldsymbol{a}_j\| &= \hat{d}_{i,j}, \quad i \in \mathcal{T}, j \in \mathcal{A} \end{aligned} \tag{3}$$

The problem in (3) can be reformulated as follows.

$$\min_{\boldsymbol{x}_i} \sum_{i=1}^{N} \Big( \sum_{j \in \mathcal{A}} (\hat{d}_{i,j} - \|\boldsymbol{x}_i - \boldsymbol{a}_j\|)^2 + \sum_{j \in \mathcal{T}} (\hat{d}_{i,j} - \|\boldsymbol{x}_i - \boldsymbol{x}_j\|)^2 \Big) \tag{4}$$

To solve the non-convex optimization problem in (4), Biswas *et al.* [12] use relaxation techniques and reformulate it as an SDP problem that can be solved in polynomial time. Note that the solutions may not be global optima.

#### 2) MLMatch METHOD

MLMatch is formulated using a maximum likelihood technique and a statistical model between RSSI values and distances. Each possible matching between nodes and positions is formulated through a permutation $h$, where $h(i) = i'$ if node $n_i$ is guessed to be located at position $p_{i'}$. Given an RSSI matrix $\mathcal{R}$ consisting of RSSI values between sensor nodes, MLMatch tries to find the best matching $h^*$ whose likelihood is biggest. Namely, nodes $n_i$ are most likely to be located at positions $h^*(i)$. Using the correlation between RSSI values and distances, MLMatch reduces the problem into a simple mathematical formulation as follows:

$$h^* = \arg\min_{h \in \mathcal{H}} \sum_{i<j} \Big( r_{i,j} \ln d_{h(i),h(j)} \Big), \tag{5}$$

where $\mathcal{H}$ is the set consisting of all permutation $h$. It then applies an appropriate searching method, for instance, metaheuristic or LP relaxation, to find the best matching $h^*$. Similar to SDP localization, MLMatch may not output the global optima.

## IV. PROPOSED ALGORITHMS LEMON AND LEMON-M

We illustrate algorithms for LEMOn and LEMOn-M. We first derive the relationship between an inter-node separation distance and the similarity between RSSI values in Section IV-A. We then illustrate localization methods using this similarity in Sections IV-B and IV-C. Finally, we analyze factors that affect the localization accuracy in Section IV-D.

### A. DISTANCE ESTIMATION

Inter-node distance estimation is fundamental in any localization method. The proposed algorithms make use of the correlation between inter-node separation distance and the similarity between RSSI values. Intuitively, if nodes $n_i$ and $n_j$ are close to each other, then the distance from the mobile unit to $n_i$ is close to the distance from the mobile unit to $n_j$. In contrast, if nodes $n_i$ and $n_j$ are far from each other, their distances to the mobile unit are different from each other. Since an RSSI value $r_{i,k}$ correlates with the distance between the mobile unit and a sensor node $n_i$, it follows that inter-node distance $d_{i,j}$ between nodes $n_i$ and $n_j$ correlates with some similarity metric between RSSI vectors $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$, where $\boldsymbol{r}_i = [r_{i,1}, r_{i,2}, \ldots, r_{i,K}]$. Using standard deviation of the elements of vector $(\boldsymbol{r_i} - \boldsymbol{r_j})$ denoted as $s_{i,j}$, as a similarity metric, we prove that it is a monotonically increasing function of distance $d_{i,j}$ under some assumptions.

*Theorem 1:* Standard deviation of the elements of vector $(\boldsymbol{r_i} - \boldsymbol{r_j})$ is approximately a monotonically increasing polynomial function of distance $d_{i,j}$ when $K$ and $\mathcal{V}$ are large enough.

*Proof:* The mathematical proof of the above theorem is provided in the Appendix. ∎

Theorem 1 suggests that $s_{i,j}$ can be used to estimate distance $d_{i,j}$. For the following reasons, we assume that $s_{i,j}$ is an approximately linear function of distance $d_{i,j}$, and is expressed through Equation (6). First, the accuracy of the assumption is illustrated through simulations (see Section V). Second, a linear function has a minimal number of unknown parameters compared to other polynomial functions. Thus, these parameters are easy to estimate using measurements between anchor nodes.

$$s_{i,j} = \alpha + \beta \times d_{i,j} + Y \tag{6}$$

where $\alpha$ and $\beta$ are constants, $Y$ is a variable characterizing the error between the linearity, i.e. $\alpha + \beta \times d_{i,j}$, and $s_{i,j}$. $Y$ is a function of the noise that is caused by RSSI noise (see the proof in Appendix) and the difference between $s_{i,j}$ and the linear function of the distance. Since $Y$ is unknown, it is considered as a random variable. Using Equation (6), algorithms for resolving the two localization problems are described next.

### B. LEMON ALGORITHM

LEMOn is designed to solve the localization problem described in Section II-A. It first calculates Euclidean distance $d_{i,j}$ and the similarity $s_{i,j}$ between each pair of anchor nodes $n_i, n_j$ $(\forall i, j \in \mathcal{A})$, to derive values of $\alpha$ and $\beta$ in (6) using the least squared linear regression technique. It then estimates distance $\hat{d}_{i,j}$ between every pair of sensor nodes $n_i$, $n_j$ using the inversion of (6), namely:

$$\hat{d}_{i,j} = \frac{s_{i,j} - \alpha}{\beta} \tag{7}$$

**Algorithm 1** LEMOn Algorithm

**Input:** $\boldsymbol{r_i}, \forall i \in \{1, \ldots, N\}$ $\{\boldsymbol{r_i} = [r_{i,1}, r_{i,2}, \ldots, r_{i,K}]\}$, $\boldsymbol{a_i}, \forall i \in \mathcal{A}$ {positions of anchor nodes}

**Output:** $\boldsymbol{x_i}, \forall i \in \mathcal{T}$ {positions of target nodes}

$\boldsymbol{s} \leftarrow$ new Array; $\boldsymbol{d} \leftarrow$ new Array

**for each** $i, j \in \mathcal{A}$ **do**

  $\boldsymbol{d}$.append($\|\boldsymbol{a_i} - \boldsymbol{a_j}\|$) {distance between two anchor nodes}

  $s_{i,j} \leftarrow \text{std}(\boldsymbol{r_i} - \boldsymbol{r_j})$

  $\boldsymbol{s}$.append($s_{i,j}$)

**end for**

$\alpha, \beta \leftarrow$ Linear_Regression($\boldsymbol{s}, \boldsymbol{d}$)

**for each** $i \in \mathcal{A}, j \in \mathcal{A} \cup \mathcal{T}$ **do**

  $s_{i,j} \leftarrow \text{std}(\boldsymbol{r_i} - \boldsymbol{r_j})$

  $\hat{d}_{i,j} \leftarrow \frac{s_{i,j} - \alpha}{\beta}$

**end for**

**for each** $i \in \mathcal{T}$ **do**

  $\boldsymbol{x_i} \leftarrow$ position of node $n_i$ derived by SDP localization method

**end for**

**Algorithm 2** LEMOn-M Algorithm

**Input:** $\boldsymbol{r_i}, \forall i \in \{1, \ldots, N\}$ $\{\boldsymbol{r_i} = [r_{i,1}, r_{i,2}, \ldots, r_{i,K}]\}$, $p_{i'}, \forall i' \in \{1, \ldots, N\}$ {positions of sensor nodes in an arbitrary order}

**Output:** Permutation $h^*$ that node $n_i$ locates at position $p_{h^*(i)}, \forall i \in \{1, \ldots, N\}$

$\mathcal{H} \leftarrow$ set of all permutations of a set of elements $1, 2, \ldots, N$

$h^* \leftarrow \text{argmax}_{h \in \mathcal{H}} \sum_{i<j} \left( s_{i,j} d_{h(i),h(j)} \right)$ {$h^*$ is found using searching methods as in Reference [8]}

It finally applies SDP localization method [12] (see Section III-B) to estimate location of target nodes. A pseudo-code of LEMOn is described in Algorithm 1.

### C. LEMON-M ALGORITHM

LEMOn-M is designed to solve the localization matching problem described in Section II-B. It is designed similar to the MLMatch algorithm described in Section III-B. While MLMatch use a logarithmic relationship between an RSSI value and the corresponding distance, in LEMOn, this relation should be substitute by the linear relationship between the similarity $s_{i,j}$ and distance $d_{i,j}$. Consequently, the formula (5) (cf. Section III-B) is substituted by (8).

$$h^* = \underset{h \in \mathcal{H}}{\text{argmax}} \sum_{i<j} \left( s_{i,j} d_{h(i),h(j)} \right). \tag{8}$$

LEMOn-M then uses the same searching methods as in MLMatch to find the best matching. A pseudo-code of LEMOn-M is described in Algorithm 2.

### D. LIMITS ON LOCALIZATION ACCURACY

Similar to other RSSI-based localization methods, the proposed algorithm cannot guarantee 100% accuracy. It is meaningful to understand factors causing localization error, and the impact of those factors on localization accuracy. This can give insights to optimize the localization accuracy.

As discussed, the proposed algorithms consist of two phases namely distance estimation phase and location estimation phase. Therefore, localization error can accumulate in both phases. The location estimation error is a function of number target nodes and anchor nodes, sensor geometry, and error of approximate solutions due to the location estimators. We recommend the readers to refer to [8] and [14] for further details.

As this paper focuses on distance estimation techniques, we analyze factors that affect the distance estimation error. As discussed in Section IV-A distance estimation error (which is denoted by the variable $Y$ in Equation (6)) is a function of the difference of the similarity $s_{i,j}$ and the linear function of the distance $d_{i,j}$ and the noise due to the RSSI fluctuation. Theoretically, due to the proof of Theorem 1, the difference can be minimized if we increase the area of $\mathcal{V}$, i.e. making the mobile unit move in a large domain. In practice, however, the trajectory of the mobile unit is often pre-determined. In many applications, $\mathcal{V}$ typically equals to $\mathcal{D}$ which is the deployment area of sensor nodes (cf. Section VII). Besides, simulations in Sections V and VI will show that $V = D$ is enough to realize accurate localization.

On the other hand, distance estimation error can be reduced by reducing the effect of RSSI noise, i.e. reducing the variation of random variable $Z$ in Equation (12). Its variation can be reduced by increasing the number of beacons $K$. Note that, the distance estimation error cannot be reduced to zero even if $K$ approaches infinity because the error is also affected by the difference described above. We call the localization accuracy that can be achieved when $K$ approaches infinity the limitation of localization accuracy. In fact, there is a finite value of $K$, often called a threshold, that can achieve the limitation of localization accuracy. Finding this value can not only optimize the localization accuracy but can also optimize the number of beacons. This value, however, is difficult to be derived mathematically as it is dependent with other parameters. The value can be estimated through simulations as in Sections V and VI. All of the above arguments will be confirmed in the two followed sections.

## V. PERFORMANCE EVALUATION OF LEMON THROUGH SIMULATIONS

In order to substantiate the performance of LEMOn, we perform and analyze three simulations in various environments. Further, for reference, we also perform static cooperative localization methods for which RSSI values between individual nodes are used. Note that we do not benchmark our

**TABLE 2. Description of parameter settings.**

| | LEMOn | | | LEMOn-M | | |
|---|---|---|---|---|---|---|
| | **Sim. 1** | **Sim. 2** | **Sim. 3** | **Sim. 4** | **Sim. 5** | **Sim. 6** |
| **Localization algorithm** | LEMOn | LEMOn | LEMOn | LEMOn-M | LEMOn-M | LEMOn-M |
| **Area of $\mathcal{D}$ ($m^2$)** | $100 \times 100$ | $100 \times 100$ | $10 \times 14$ | $10 \times 14$ | $10 \times 14$ | $10 \times 14$ |
| **Area of $\mathcal{V}$** | $100 \times 100$ | $100 \times 100$ | $10 \times 14$ | $10 \times 14$ | $10 \times 14$ | $10 \times 14$ |
| **Height of the Mobile trajectory $h$ (m)** | 20 | 20 | 3 | 3 | 3 | 3 |
| **Path loss exponent $\eta$** | 3 | 3 | 2.5 | 2.5 | 2.5 | 2 & 3 |
| **Propagation model** | Gaussian &Rayleigh | Gaussian | Rayleigh | Gaussian &Rayleigh | Gaussian | Rayleigh |
| **Standard deviation $\sigma_X$** | 5.57 | $\{3, 4, \ldots, 12\}$ | 5.57 | 5.57 | $\{3, 4, \ldots, 12\}$ | 5.57 |
| **Number of beacons $K$** | $100, 200, \ldots, 1000$ | 200 & 800 | 500 | $100, 200, \ldots, 1000$ | 200 & 800 | 500 |
| **Number of nodes $N$** | 35 | 35 | $15, 20, \ldots, 45$ | 15 | 15 | $11, 13, \ldots, 27$ |
| **Number of anchors $A$** | 10 | 10 | 5 &10 | - | - | - |

results against those of static methods because of significant differences in the structure and assumptions of our system. In static cooperative localization methods, we assume that all sensor nodes are perfectly calibrated and the path loss exponent is accurately estimated. We also assume that RSSI values between individual nodes are measured or known. In both cases, the SDP method is used as a localization estimator [12].

### A. PROPAGATION MODELS

Although our analysis described in Section IV-A assume that the RSSI values follow the log-distance propagation model, we argue that it is also sufficiently accurate even if RSSI values follow other models. We substantiate our claim through numerical simulations shown below using a more advanced propagation model that is validated through indoor measurements at 2.4GHz [35]. We simulate a propagation environment experiencing Rayleigh fading and a non-singular path loss. The RSSI values $\bar{r}$ under this propagation model are generated via

$$\bar{r} = P_i - 10\log_{10}(\epsilon + d^{\eta}) + \mathcal{X} \quad \text{[dBm]} \tag{9}$$

where $\epsilon > 0$, and $\mathcal{X}$ is a random variable with density

$$\begin{aligned} f_{\mathcal{X}}(x) &= \mathbb{P}[\mathcal{X} = x] = \frac{\mathrm{d}}{\mathrm{d}x}\mathbb{P}\left[10\log_{10}|h|^2 \leq x\right] \\ &= \lambda 10^{x/10}\exp\left(-\lambda 10^{x/10}\right)\frac{\ln 10}{10} \end{aligned} \tag{10}$$

In the rest of this paper, we call this model *Rayleigh* model for short, while *Gaussian* model stands for the log-distance propagation model encapsulated in Equation (2).

### B. PARAMETER SETTINGS

Simulations 1 and 2 consider a UAV assisted WSNs where $N = 35$ sensor nodes are deployed randomly in a square domain $\mathcal{D} = 100 \times 100$ $m^2$. The mobile unit (i.e. the UAV) flies at a height of $h = 20$ m, randomly inside a domain $\mathcal{V} = \mathcal{D} = 100 \times 100$ $m^2$, which is an common case found in practice. The path loss exponent is set as $\eta = 3$, which is a common value noted by [19] for outdoor urban environments.

Simulation 1 investigates the localization error when numbers of beacons $K$ varies. It validates the robustness of the proposed method under different propagation models.Using Rayleigh model, RSSI values between the devices are generated using Equation (9) with common parameters $\sigma_X = 5.57$, $\lambda = e^{-\gamma}$, and $\epsilon = 0.1$ [8]. Using Gaussian model, RSSI values are generated using Equation (2) with $\sigma_X$ also equal 5.57. We also use this parameter settings for all RSSI values that follow the Rayleigh model in the rest of the simulations. This way we can test the robustness of LEMOn against other propagation models. The positions of the mobile unit are also generated randomly in the domain $\mathcal{V}$. The number of beacons $K$ is chosen in the range $\{100, 200, 300, \ldots, 1000\}$. This way we can observe the effect of $K$ on the localization error.

Simulation 2 observes the localization error of LEMOn against different levels of signal noise. RSSI values are generated using a Gaussian model with $\sigma_X$ is chosen in the range $\{3, 4, 5, 6, \ldots, 12\}$ which is as low as in [8] as high as in [36]. The number of beacons $K$ is set as 200 and 800. This way we can test the robustness of LEMOn in a less noisy environment and very noisy environments, and also the effect of $K$ on these scenarios.

Simulation 3 considered an indoor IoT network where sensor nodes are deployed in a room with area $\mathcal{D} = 10 \times 14$ $m^2$ and height $h = 3$ m. The path loss exponent is set as $\eta = 2.5$, which is a common value used for indoor environments [8], [19]. It investigates localization error of LEMOn against different numbers of sensor nodes. The number of nodes $N$ is chosen from the set $\{15, 20, 25, \ldots, 45\}$ among which $A = 5$ or $A = 10$ nodes acted as anchors. RSSI values were generated using the Rayleigh model. The number of beacons $K$ is set to 500.

In each simulation, the reference power $P_i$ (cf. Equations (2), (9)) for each node is generated randomly. For each set of parameters, simulations are performed 20 times to obtain statistical averages.

The values of all other parameters used are detailed in Table 2.

### C. SIMULATION ENVIRONMENT

We run our experiments with Python, a high-level programming language. In each realization, position of $N$ sensor nodes are generated randomly under Poisson distribution in domain $\mathcal{D}$. Among these nodes, $A$ nodes are chosen randomly acting as anchor nodes. Position of the mobile unit when it sends a beacon is also generated randomly in domain $\mathcal{V}$. An RSSI value $r_{i,k}$ is generated using either formulation (2) or (9), where random variable $X$ follows Gaussian distribution or Rayleigh distribution, respectively. In particular, $X$ is generated using *random.py* module, a Python pseudo-random number generator that implements various probability distributions.

### D. IMPLEMENTATION OF SDP LOCALIZATION

In the proposed localization problem, we use [12, Formula (6)], which is designed to localize nodes when distances between some nodes are accurately known. On the other hand, in the static localization problem, namely localization using inter-node RSSI values, we use Formulas (11) and (12) from [12] that solve RSSI-based cooperative localization problems. We use MOSEK optimizer for Python [37] as the SDP solver.

### E. RESULT ANALYSIS

In every simulation, we run LEMOn to estimate the location of target nodes using the mobile unit. For reference, we also run SDP localization scheme using inter-sensor nodes' RSSI, which is a static cooperative localization technique. We then output localization error that is the mean error (in meter). The results are illustrated by Figures 1, 2 indicating that LEMOn performs well in various environments. Localization errors from Simulation 1 are illustrated in Figure 1 a) indicating that the localization error decreases when the number of beacons $K$ are increased to 400. The average localization error, however, does not vary when $K$ varies from 400 to 1000. This confirms the argument on the threshold of $K$ in Section IV-D. Besides, LEMOn performs very well regardless of the distribution of RSSI values such as the Gaussian model or the Rayleigh model. This simulation validates the robustness of LEMOn against other random variable distribution models for wireless fading such as the Rayleigh distribution.

Localization errors from Simulation 2 are illustrated in Figure 1 b) indicating that the localization error increases with the level of noise, especially when the value of $K$ is not large enough, e.g. $K = 200$. However, for a large $K$ (e.g., $K = 800$), LEMOn can achieve accuracy of less than 10 m in a noisy environment characterized by a $\sigma_X = 10$ when static localization technique fails (cf. Figure 2 b). This is because static localization technique that estimates inter-node separation distance directly from RSSI measurement between them suffer a large error. The calculation in Appendix shows that RSSI-based ranging error in this environment is even larger than the inter-node separation distance. On the other hand, LEMOn uses statistical RSSI similarity, thus can

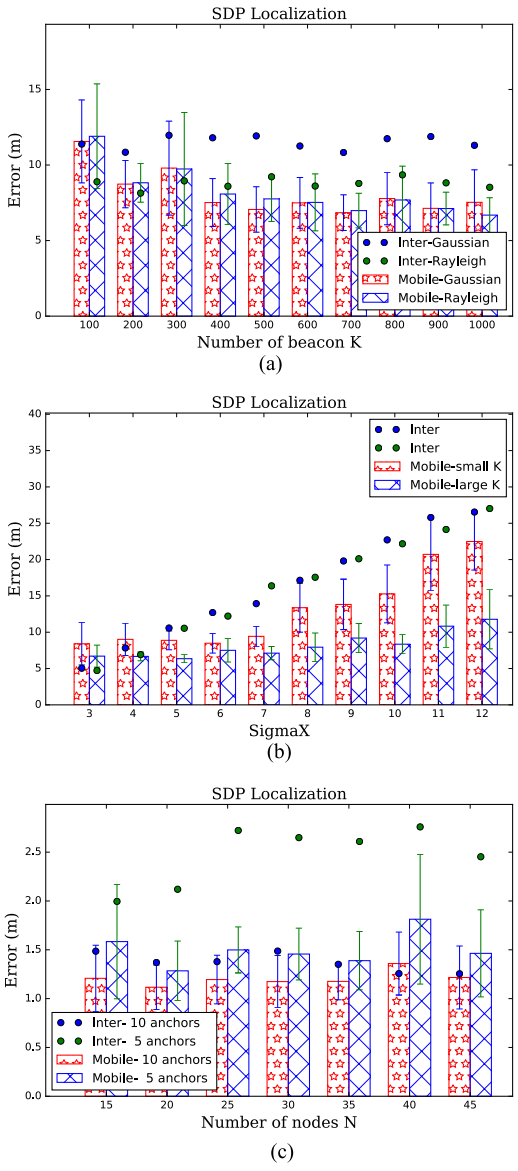


**FIGURE 1. Simulations 1-3 results: Average localization error obtained by LEMOn (bars) and by using static localization, i.e. using inter-node RSSIs (circles). a) Simulation 1: Impact of number of beacons on localization accuracy; b) Simulation 2: Impact of noisy level on localization accuracy; c) Simulation 3: Impact of number of nodes on localization accuracy.**

suppress error caused by a single RSSI value. Simulation 2 validates the robustness of LEMOn in noisy environments. Simulations 1 and 2 confirm the argument in Section IV-D regarding the impact of the number of beacons on localization accuracy. Besides, the localization error can be minimized to 5-10 m even in noisy environments.

Localization errors from Simulation 3 are illustrated in Figure 1 c) indicating that the localization error does not vary a lot with the variation in the number of nodes $N$. On the other hand, decreasing number of anchors slightly reduces the localization accuracy. Especially, LEMOn performs much better than the static SDP localization when there is a small number of anchors (for example $A = 5$). This suggests that a high localization accuracy can be achieved using LEMOn

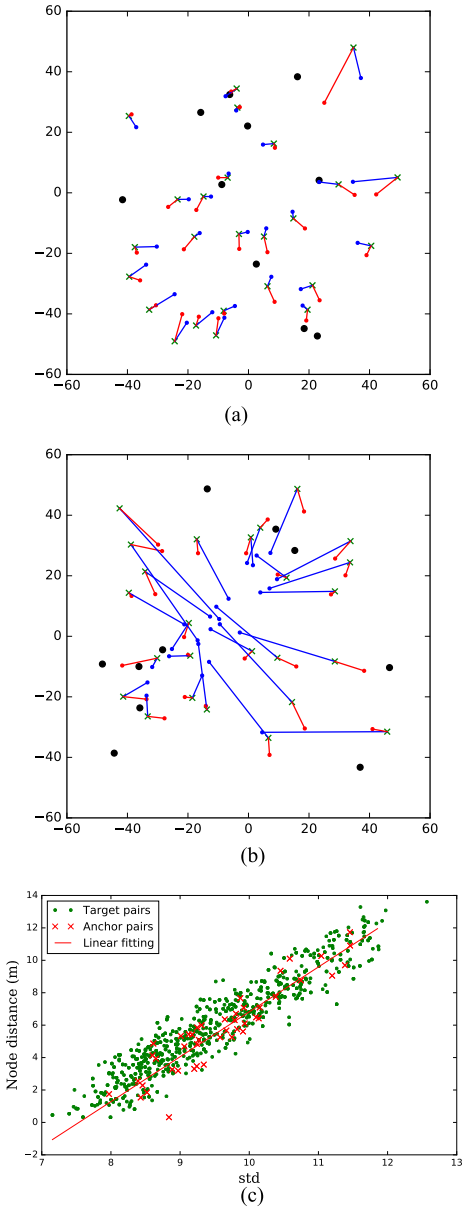


**FIGURE 2.** Simulations 1-3 results: a) A visualization of location estimation results by LEMOn (red triangles) and by using inter-node RSSI (blue dots) in a less noisy environment, and b) in a noisy environment. Black circles depict the position of anchor nodes, and green cross depict the true position of target nodes; c) Relationship between RSSI similarity and distances. a) Less noisy environment, b) Noisy environment, c) Relationship between RSSI similarity and distances.

even for a small number of anchor nodes. Besides, the localization error is around 1.5 m in this indoor environment.

The relationship between the similarity $s_{i,j}$ and inter-node distances $d_{i,j}$ is illustrated in Figure 2 c) indicating that the similarity is approximately linearly related to the distance. This confirms the argument on the linearity discussed in Section IV-A. Figure 2 c) also depicts that the linear fitting of distance between anchor nodes and their similarities $s_{i,j}$ is close to that of the target nodes. This enables node localization without a priori measurements.

All above results confirmed the robustness of LEMOn in various environments, and against different propagation models. Especially, it is robust even in very noisy environments.

## VI. PERFORMANCE EVALUATION OF LEMON-M THROUGH SIMULATIONS

Similar to simulations in Section V, we perform three simulations in different environments. Further, for reference, we also performed the static localization matching method, i.e. inter-sensor nodes RSSI values are used. In the static localization matching method, we assume that all sensor nodes are perfectly calibrated. We also assume that RSSI values between all node-pairs are obtained.

### A. PARAMETER SETTINGS

Similar to Simulation 3 in Section V, we consider an indoor IoT network where wireless nodes are deployed in a room with area $\mathcal{D} = 10 \times 14$ m$^2$ and height $h = 3$ m. The path loss exponent $\eta$ is set to 2, 2.5, and 3 based on the real empirical measurements acquired from multiple indoor environments in one of our earlier studies [8]. Use of multiple path loss exponents thus enables us to evaluate our protocols in different indoor environments. Wireless nodes attached to light bulbs and air conditioners serve as fixed objects, while a ground vehicle that moves around the room acts as a mobile unit. Furthermore, $\mathcal{V}$ is set equal to $\mathcal{D}$. Other parameters are set to exactly same as the three simulations described in Section V to evaluate the performance of LEMOn-M in different scenarios, and whether it performs similarly to LEMOn.

Simulation 4 evaluates the localization accuracy when the number of beacons $K$ varies. It also validates the robustness of the LEMOn-M under different propagation models. The position of the mobile unit is also generated randomly in the domain $\mathcal{V}$. The number of beacons $K$ is chosen from the set $\{100, 200, 300, \ldots, 1000\}$. RSSI values are generated randomly using the Gaussian model or the Rayleigh model. This way we can observe the effect of $K$ on the localization error.

Simulation 5 observes the localization accuracy of LEMOn-M against different levels of signal noise. RSSI values are generated using a Gaussian model with $\sigma_X$ chosen from the set $\{3, 4, \ldots, 12\}$. The number of beacons $K$ is set to 200 and 800. This way we can test the robustness of LEMOn-M in environments characterized by different noise levels, and also the impact of $K$ on these scenarios.

Simulation 6 studies localization accuracy of LEMOn-M against different numbers of sensor nodes. The number of nodes $N$ is chosen in the range $\{11, 13, \ldots, 27\}$. RSSI values are generated using the Rayleigh model. The number of beacons $K$ is set to 500. The path loss exponent $\eta$ is set to 2 or 3, which is different than Simulations 4 and 5, in order to test the performance of LEMOn-M in other environments.

In each simulation, the reference power $P_i$ (cf. Equations (2), (9)) for each node is generated randomly. For each set of parameters, simulations are run 20 times to obtain

statistical averages. The values of all other parameters used are detailed in Table 2.

### B. RESULTS ANALYSIS

In every simulation, we run LEMOn-M and to determine the best matching between nodes and positions using the mobile unit. For reference, we also run MLMatch localization using RSSI values between nodes that is the static localization matching technique. We then output localization accuracy that is defined as the ratio between the number of accurately matched nodes and the number of total nodes [8]. The results are illustrated by Figure 3 indicating that LEMOn-M performs well in various environments. Localization accuracy from Simulation 4 are illustrated in Figure 3 a). Localization accuracy increases with increase in the number of beacons $K$ up to 500. The average localization accuracy, however, does not vary when $K$ increases further from 500 to 1000. Besides, LEMOn-M performs very well regardless the RSSI values follow the Gaussian model or the Rayleigh model. This validates the robustness of LEMOn-M against other random variable distribution models for wireless fading such as the Rayleigh distribution.

Localization accuracy from Simulation 5 are illustrated in Figure 3 b) indicating that the localization accuracy decreases when the level of noise increases, especially when $K$ is not large enough, e.g. $K = 200$. However, when $K$ is large (e.g. $K = 800$), LEMOn-M can achieve localization accuracy even in very noisy environment.

Localization accuracy from Simulation 6 are illustrated in Figure 3 c) indicating that the localization accuracy does not vary a lot with the variation of the number of nodes $N$.

The trend of localization error by LEMOn-M is similar to that in LEMOn. Our results show that both algorithms can outperform static cooperative localization methods when there are sufficient number of beacon nodes $K$. Both methods are resilient against high noise level especially for sufficiently large $K$, which introduces diverse variations of distance between the mobile unit and sensor nodes. It helps in suppressing the effect of noise by shadowing and fading. On the other hand, in static localization techniques, multiple transmissions between two fixed nodes can reduce the effect only if nodes are equipped with specific hardware or technologies (cf. Section I).

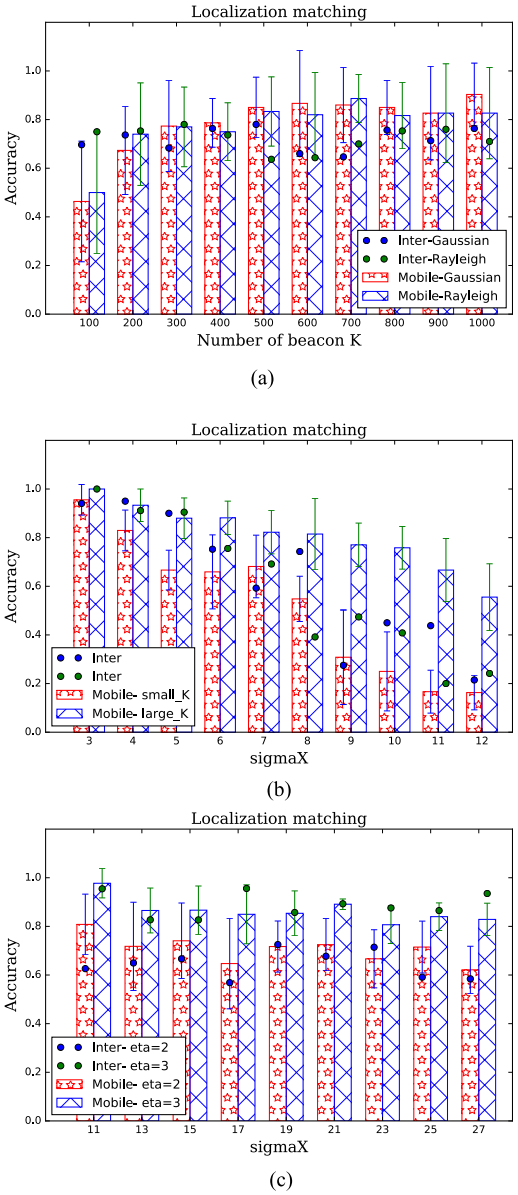


**FIGURE 3.** Average localization error obtained by LEMOn-M (bars) and by using static localization, i.e. using inter-node RSSIs (circles).
a) Simulation 4: Impact of number of beacons on localization accuracy;
b) Simulation 5: Impact of noisy level on localization accuracy;
c) Simulation 6: Impact of number of nodes on localization accuracy.

## VII. APPLICATIONS

This section highlights some real-world applications of LEMOn and LEMOn-M.

### A. UAV ASSISTED WSNs

The proposed algorithm can be used in a UAV assisted WSN which has various applications. For instance in agriculture, a UAV can control the amount of chemical sprayed over a piece of land [38], or to gather data from deployed wireless sensors [39]. For the purpose of post-disaster monitoring, UAVs are used to deploy a WSN in the area [40]. For data collection, a UAV is designed to collect data from a WSN efficiently [41], or to dispatches mobile agents that are used to collect data [42]. In all of these applications, it is necessary to collect sensed data from nodes and is necessary to relate the stream of data to the location of the corresponding sensor node. The UAV is, therefore, planned to fly along an optimal trajectory to collect the data from the WSN. To cover all sensor nodes without leaving any gap, the UAV is made to fly through an operation area parallel to the sensor node deployment plane [43], [44] (see Figure 4). Because a sensor node should detect whether the UAV is nearby, the UAV is assumed to transmit beacon messages periodically (e.g. every two seconds in [44]). It is often assumed that there is a small

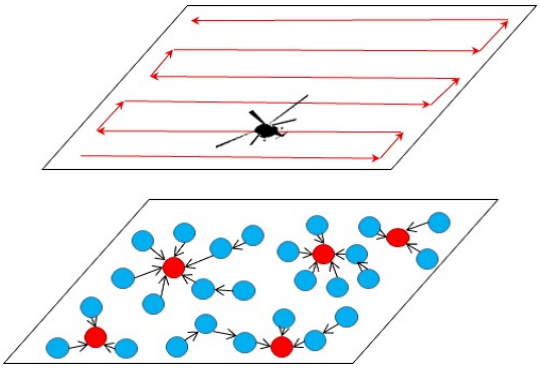

**FIGURE 4.** A UAV traveling around to collect data from sensor nodes.

number of sensor nodes that know their positions either through GPS or deployment time configuration. These nodes are often used as a cluster head that collects data from nearby nodes. Since the energy supply of the UAV is not limited as that of sensor nodes, it is possible to assume that the transmit power of the UAV is large enough so that all the sensor can receive the beacons. Under this network model, we can obtain all necessary information to perform LEMOn. Further, simulation results in Section V show that LEMOn can provide an accuracy of 5-10 m which is similar to the accuracy that one can get from enhanced GPS.

**FIGURE 5.** A wireless mobile unit going around to collect RSSIs from IoT devices.

### B. WIRELESS LOCALIZATION FOR INDOOR IoT

We consider an indoor IoT system where objects, for instance, light bulbs, air conditioners, TVs, fans, etc., are equipped with wireless transceivers (see Figure 5). To control each object, it is necessary to match its location and its ID (e.g., MAC address). Whilst the equipment positions of some fixed objects such as light bulbs, air conditioners, are well known from the floor plan blueprints (often found in out-of-reach positions, e.g., behind ceiling panels or rooftops), the specific equipment ID may not be recorded by installation engineers due to high manual labor cost [8].

Our previous work proposes a new problem called *the wireless localization matching problem (WLMP)* that automatically matches an object's position and its ID [8]. The problem is resolved using RSSI values between all pairs of devices, and the set of positions of devices that are known from floor plan blueprints. However, in some network systems, collecting RSSI values between those devices complicates the network design. This is because wireless devices are often designed to communicate to some peculiar wireless controllers or sink nodes rather than to each other. The proposed solution in [8] is, therefore, inappropriate for typical sensor networked systems.

LEMOn-M can resolve the above problem. Moving the controller or the sink node around the floor and connecting it with a UAV or a ground robot such as vacuum cleaning robot can efficiently collect RSSI values from the wireless devices. A vacuum cleaning robot is often designed to move around a floor, thus making collection process of the RSSI values feasible.

Simulation results in Section VI show that LEMOn-M outperforms in doing the localization matching in the wireless mesh networks, thus being feasible for real-world applications. Consequently, devices such as light bulbs and air conditioners can be matched to their exact position efficiently.

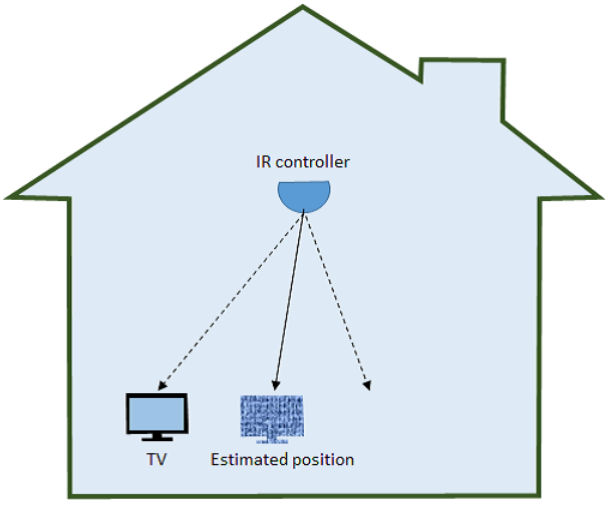


**FIGURE 6.** An example of application of indoor IoT localization.

Furthermore, besides matching known fixed positions to the devices using LEMOn-M, LEMOn can be used to localize devices for which localization is not known at all. Simulation 3 in Section V shows that LEMOn achieves less than 2 meters of accuracy on average. This is sufficient for many indoor applications. Figure 6 illustrates an application use-case common to today's smart homes. Electronic devices are controlled through an Infrared Radiation (IR) controller. The emitting angle of an IR controller is often between $\pm 10°$ and $\pm 60°$. We assume that emitting angle of the IR controller is $\pm 25°$, and the distance between the controller and an electronic device (e.g. a TV) is larger than 5 m. Then the IR

controller can control the TV using an estimated position of TV that is less than 2m away from its real position.

## VIII. CONCLUSIONS

While RF localization has come a long way [45], [46], there are many unconventional localization problems that are still unexplored. This paper proposes two novel localization methods namely LEMOn and LEMOn-M that use a location-unaware mobile unit to help estimating location of other wireless nodes. The mobile unit can be either an UAV, a ground robot, or a mobile access point. The method can, therefore, be used for many IoT systems, for instance, UAV assisted WSNs which have attracted great attention recently.

The proposed methods advance conventional localization methods in several dimensions. First, these can be used in both indoor and outdoor environments because he location of the mobile unit is not needed. Second, the techniques are hardware-independent and do not require complex calibration, making these suitable for a wide variety of emerging IoT applications. Third, we do not need to run lengthy campaigns to estimate parameters (such as path loss exponents) of each individual radio environment wherever the applications are to be deployed.

We extensively evaluated the performance of LEMOn and LEMON-M using simulations. LEMOn achieves a 5-10 m accuracy on average similar to a GPS in outdoor environments. In indoors, LEMOn accurately localize within 2 m on average, which makes it suitable for many indoor applications. LEMOn-M is shown to outperform the static localization matching techniques even in very noisy environments where competing solutions fail. Finally, we highlighted real-world application scenarios such as UAV assisted WSNs and indoor IoT systems where these techniques can be applied.

Despite the encouraging results achieved by LEMOn and LEMOn-M, there is still progress to be made on this front. The performance of these techniques for large network size and deployment area must be investigated. In addition, advanced data processing techniques can be exploited to minimize the inaccuracies arising from signal strength outliers, external interference, non-line of sight propagation, etc. Finally, we like many other authors assume perfectly isotropic antennas deployed on a 2D surface for this study. Generalizing to WSNs equipped with an-isotropic antenna deployed in a 3D space is an interesting research direction.

## APPENDIX

### A. PROOF OF THEOREM 1

Notations, and assumptions used in the proof are first described. $s_{i,j}$ denotes standard deviation of vector $(\boldsymbol{r_i}-\boldsymbol{r_j})$. $\boldsymbol{c_k}$ denotes the position, i.e. coordinate, of the mobile unit when it sends the $k$-th beacons. $\boldsymbol{n_i}$ denotes the position of node $n_i$. $\|\boldsymbol{x}-\boldsymbol{y}\|$ is the Euclidean distance between two points locating at positions $\boldsymbol{x}$ and $\boldsymbol{y}$, e.g., $d_{i,j} = \|\boldsymbol{n_i}-\boldsymbol{n_j}\|$. $\ln\cdot$ denotes the nature logarithm. $x$ is the magnitude of vector $\boldsymbol{x}$. $V$ and $D$ are the areas of $\mathcal{V}$ and $\mathcal{D}$. For simplicity, $\mathcal{V}$ and $\mathcal{D}$ are assumed to be parallel to each other; $h$ denotes the distance between them.

Using Equation (2), we have

$$\begin{aligned} r_{i,k} - r_{j,k} &= P_i - P_j - 10\eta \log_{10}\frac{d_{i,k}}{d_{j,k}} + X_{i,k} - X_{j,k} \\ &= P_i - P_j - \frac{10\eta}{\ln 10}\big(\ln\frac{d_{i,k}}{d_{j,k}} + X_k\big) \end{aligned} \tag{11}$$

where, $X_k = \frac{(X_{i,k}-X_{j,k})\ln 10}{10\eta}$. Let $\delta = \frac{10\eta}{\ln 10}$, then $X_k \sim \mathcal{N}(0, 2(\frac{\sigma_X}{\delta})^2)$. Since $(P_i - P_j)$ and $\delta$ are constant, the standard deviation $s_{i,j}$ of the vector $\boldsymbol{r_i} - \boldsymbol{r_j}$ divided by $\delta$ equals the standard deviation of a vector $\boldsymbol{r}$ whose entries equals $r_k = \ln\frac{d_{i,k}}{d_{j,k}} + X_k$. Namely,

$$\begin{aligned} (\frac{s_{i,j}}{\delta})^2 &= \frac{1}{K}\sum_{1\le k\le K}\big(\ln\frac{d_{i,k}}{d_{j,k}} + X_k - \ln\frac{\bar{d}_i}{\bar{d}_j} - \bar{X}\big)^2 \\ &= \frac{1}{K}\sum_{1\le k\le K}(X_k - \bar{X})^2 \\ &\quad + \frac{1}{K}\sum_{1\le k\le K} 2(X_k - \bar{X})(\ln\frac{d_{i,k}}{d_{j,k}} - \ln\frac{\bar{d}_i}{\bar{d}_j}) \\ &\quad + \frac{1}{K}\sum_{1\le k\le K}(\ln\frac{d_{i,k}}{d_{j,k}} - \ln\frac{\bar{d}_i}{\bar{d}_j})^2 \end{aligned} \tag{12}$$

where $\bar{d}_i$, $\bar{d}_j$, and $\bar{X}$ denote, respectively, the average value of distance $d_{i,k}, \forall k$, $d_{j,k}, \forall k$, and $X_k, \forall k$. The first term of the right side of (12) is the variance of $X_k$ as $K$ being large enough, thus equaling $2(\sigma_X/\delta)^2$. We denote $Z$ the second term of the right side of (12). Due to Central Limit Theorem (CLT), $Z$ is a random variable following zero mean Gaussian distribution. Consequently, (12) equals

$$\begin{aligned} (\frac{s_{i,j}}{\delta})^2 &\approx Z + \frac{2\sigma_X^2}{\delta^2} + \frac{1}{K}\sum_{1\le k\le K}(\ln\frac{d_{i,k}}{d_{j,k}} - \ln\frac{\bar{d}_i}{\bar{d}_j})^2 \\ &\approx Z + \frac{2\sigma_X^2}{\delta^2} + \lim_{K\to+\infty}\frac{1}{K}\sum_{1\le k\le K}(\ln\frac{\|\boldsymbol{c_k}-\boldsymbol{n_i}\|}{\|\boldsymbol{c_k}-\boldsymbol{n_j}\|} - \ln\frac{\bar{d}_i}{\bar{d}_j})^2 \\ &\approx Z + \frac{2\sigma_X^2}{\delta^2} + \int_{\mathcal{V}}(\ln\frac{\|\boldsymbol{x}-\boldsymbol{n_i}\|}{\|\boldsymbol{x}-\boldsymbol{n_j}\|} - \ln\frac{\bar{d}_i}{\bar{d}_j})^2 d\boldsymbol{x} \end{aligned} \tag{13}$$

Using the assumption that $\mathcal{V}$ is large enough, $\mathcal{V}$ can approximately be assumed to be a disc having radius of $R = \sqrt{V/\pi}$, and centered at the projection of the midpoint of $\boldsymbol{n_i}$ and $\boldsymbol{n_j}$ onto the plane $\mathcal{V}$ (see Figure 7). Under this assumption, $\bar{d}_i$ equals $\bar{d}_j$ therefore $\ln\frac{\bar{d}_i}{\bar{d}_j}$ can be ignored. Let $\theta$ be the angle between vector $\boldsymbol{x}$ and the projection of vector $\boldsymbol{n_j} - \boldsymbol{n_i}$ onto domain $\mathcal{V}$, distances between $\boldsymbol{x}$ and $\boldsymbol{n_i}, \boldsymbol{n_j}$ are formulated as (14)

$$\begin{aligned} \|\boldsymbol{x}-\boldsymbol{n_i}\|^2 &= h^2 + x^2 + (d_{i,j}/2)^2 + x d_{i,j}\cos\theta \\ \|\boldsymbol{x}-\boldsymbol{n_j}\|^2 &= h^2 + x^2 + (d_{i,j}/2)^2 - x d_{i,j}\cos\theta \end{aligned} \tag{14}$$

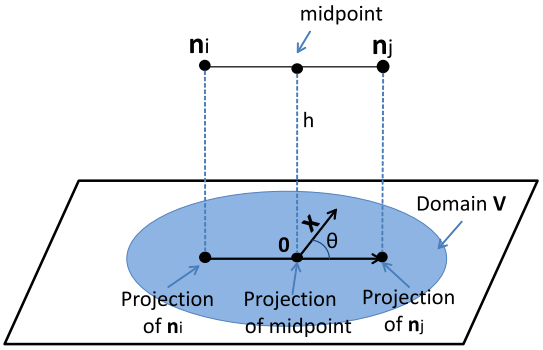


**FIGURE 7.** An example of geometric relationship between domain $\mathcal{V}$ and sensor nodes.

Using Taylor series expansion of $\ln(a+x)$ when $x < a$, i.e. $\ln(a+x) \approx \ln a + x/a$, (15) is obtained.

$$\begin{aligned}\frac{\ln\|\boldsymbol{x}-\boldsymbol{n_i}\|}{\ln\|\boldsymbol{x}-\boldsymbol{n_j}\|} &= \frac{1}{2}(\ln\|\boldsymbol{x}-\boldsymbol{n_i}\|^2 - \ln\|\boldsymbol{x}-\boldsymbol{n_j}\|^2)\\ &\approx \frac{1}{2}\Big(\ln(h^2+x^2+(d_{i,j}/2)^2)+\frac{xd_{i,j}\cos\theta}{h^2+x^2+(d_{i,j}/2)^2}\\ &\quad -(\ln(h^2+x^2+(d_{i,j}/2)^2)-\frac{xd_{i,j}\cos\theta}{h^2+x^2+(d_{i,j}/2)^2})\Big)\\ &\approx \frac{xd_{i,j}\cos\theta}{h^2+x^2+(d_{i,j}/2)^2} \qquad (15)\end{aligned}$$

By substituting (15) into (13) we have:

$$\begin{aligned}&(\frac{s_{i,j}}{\delta})^2\\ &\approx Z+\frac{2\sigma_X^2}{\delta^2}+\frac{1}{\pi R^2}\int_0^R xdx\int_{-\pi}^{\pi}(\frac{xd_{i,j}\cos\theta}{h^2+x^2+(d_{i,j}/2)^2})^2d\theta\\ &\approx Z+\frac{2\sigma_X^2}{\delta^2}+\frac{1}{\pi R^2}\int_0^R\frac{\pi x^3d_{i,j}^2}{(h^2+x^2+(d_{i,j}/2)^2)^2}dx\\ &\approx Z+\frac{2\sigma_X^2}{\delta^2}+\frac{d_{i,j}^2}{2R^2}\int_0^{R^2}\frac{y}{(y+h^2+(d_{i,j}/2)^2)^2}dy\\ &\approx Z+\frac{2\sigma_X^2}{\delta^2}+\frac{d_{i,j}^2}{2R^2}\int_0^{R^2}(\frac{1}{y+h^2+(d_{i,j}/2)^2}\\ &\quad -\frac{h^2+(d_{i,j}/2)^2}{(y+h^2+(d_{i,j}/2)^2)^2})dy\\ &\approx Z+\frac{2\sigma_X^2}{\delta^2}+\frac{d_{i,j}^2}{2R^2}\Big(\ln\frac{R^2+h^2+(d_{i,j}/2)^2}{h^2+(d_{i,j}/2)^2}\\ &\quad -\frac{R^2}{R^2+h^2+(d_{i,j}/2)^2}\Big) \qquad (16)\end{aligned}$$

Let $x = \left(\frac{d_{i,j}}{2R}\right)^2$, (16) can be rewritten as

$$\begin{aligned}s_{i,j} \approx \delta\Big[Z+\frac{2\sigma_X^2}{\delta^2}+\frac{x}{2}\Big(\ln\frac{1+(h/R)^2+x}{(h/R)^2+x}\\ -\frac{1}{1+(h/R)^2+x}\Big]^{1/2} \qquad (17)\end{aligned}$$

Since $R$ is large enough, $x$ is smaller than 1, and is close to 0. Therefore, applying Taylor series expansion around $x=0$ to (17), similarity $s_{i,j}$ approximates

$$\begin{aligned}s_{i,j} \approx \delta\Big[&(Z+\frac{2\sigma_X^2}{\delta^2})^{1/2}+\frac{1}{4(Z+\frac{2\sigma_X^2}{\delta^2})^{1/2}}\\ &\times\Big(\ln\frac{1+(h/R)^2}{(h/R)^2}-\frac{1}{1+(h/R)^2}\Big)x+\mathcal{O}(x^2)\Big] \qquad (18)\end{aligned}$$

Consequently, similarity $s_{i,j}$ is a monotonically increasing polynomial function of distance $d_{i,j}$. It, therefore, can be used to estimate distance $d_{i,j}$.

## B. RANGING TECHNIQUE AND RANGING ERROR

The node separation distance can therefore be estimated via the inversion of Formula (1)

$$\hat{d} = f^{-1}(r) = 10^{\frac{P_0-r}{10\eta}} \qquad (19)$$

Assuming that $X \sim \mathcal{N}(0,\sigma_X^2)$, the root mean squared ranging error $\varepsilon_X$ due to $X$ can thus be calculated

$$\begin{aligned}\varepsilon_X &= (\mathbb{E}[(\hat{d}-d)^2])^{1/2} = \left(\mathbb{E}(d^2(10^{\frac{X}{10\eta}}-1)^2)\right)^{1/2}\\ &= d\Big[\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-X^2}{2\sigma^2}}(e^{\frac{X\ln 10}{10\eta}}-1)^2\mathrm{d}X\Big]^{1/2}\\ &= d\Big[\frac{1}{\sqrt{2\pi}\sigma}\Big(e^{(\frac{\sqrt{2}\sigma\ln 10}{10\eta})^2}\int_{-\infty}^{\infty}e^{-\frac{1}{2\sigma^2}(X-\frac{2\sigma^2\ln 10}{10\eta})^2}\mathrm{d}X\\ &\quad -2e^{(\frac{\sigma\ln 10}{10\sqrt{2}\eta})^2}\int_{-\infty}^{\infty}e^{-\frac{1}{2\sigma^2}(X-\frac{\sigma^2\ln 10}{10\eta})^2}+e^{-\frac{X^2}{2\sigma^2}}\mathrm{d}X\Big)\Big]^{1/2}\\ &= d\Big[e^{(\frac{\sqrt{2}\sigma\ln 10}{10\eta})^2}-2e^{(\frac{\sigma\ln 10}{10\sqrt{2}\eta})^2}+1\Big]^{1/2} \qquad (20)\end{aligned}$$

Thus ranging methods (19) are subject to errors that increase exponentially with the signal fluctuations. For instance, considering two nodes with a separation distance of 100m, path loss exponent $\eta = 2$, when $\sigma_X = 3$, $\sigma_X = 6$ and $\sigma_X = 12$ the expectation of the instantaneous distance estimation error is averagely 38 m, 102 m, and 642 m, respectively. In another environment, $\eta = 3$, and $\sigma_X = 10$, the average error is around 125 m.

## ACKNOWLEDGMENTS

The authors thank Dr. O. Georgiou, Dr. W. H. Thompson, and Dr. T. Hone for helpful advice, and also Dr. V. Suppakitpaisarn for the implementation of SDP localization method.

## REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
[2] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on 5G networks for the Internet of Things: Communication technologies and challenges," *IEEE Access*, vol. 6, pp. 3619–3647, 2018.
[3] A. Yassin *et al.*, "Recent advances in indoor localization: A survey on theoretical approaches and applications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1327–1346, 2nd Quart., 2016.

[4] C. Nguyen, O. Georgiou, and Y. Doi, "Maximum likelihood based multihop localization in wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 6663–6668.

[5] S. Wielandt and L. De Strycker, "Indoor multipath assisted angle of arrival localization," *Sensors*, vol. 17, no. 11, p. 2522, 2017.

[6] T. Van Haute, B. Verbeke, E. De Poorter, and I. Moerman, "Optimizing time-of-arrival localization solutions for challenging industrial environments," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1430–1439, Jul. 2017.

[7] Z. Han, C. S. Leung, H. C. So, and A. G. Constantinides, "Augmented Lagrange programming neural network for localization using time-difference-of-arrival measurements," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3879–3884, Aug. 2018.

[8] C. L. Nguyen, O. Georgiou, Y. Yonezawa, and Y. Doi, "The wireless localization matching problem," *IEEE Internet Thing*, vol. 4, no. 5, pp. 1312–1326, Oct. 2017.

[9] C. L. Nguyen and A. Khan, "WiLAD: Wireless localisation through anomaly detection," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–7.

[10] A. Bose and C. H. Foh, "A practical path loss model for indoor WiFi positioning enhancement," in *Proc. 6th Int. Conf. Inf., Commun. Signal Process.*, Dec. 2007, pp. 1–5.

[11] Y. Zhou, J. Li, and L. Lamont, "Multilateration localization in the presence of anchor location uncertainties," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2012, pp. 309–314.

[12] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Trans. Sens. Netw.*, vol. 2, no. 2, pp. 188–220, 2006.

[13] S. Čapkun, M. Hamdi, and J.-P. Hubaux, "GPS-free positioning in mobile ad hoc networks," *Cluster Comput.*, vol. 5, no. 2, pp. 157–167, 2002.

[14] 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.

[15] M. L. Sichitiu and V. Ramadurai, "Localization of wireless sensor networks with a mobile beacon," in *Proc. IEEE Int. Conf. Mobile Ad-hoc Sensor Syst.*, Oct. 2004, pp. 174–183.

[16] N. B. Priyantha, H. Balakrishnan, E. D. Demaine, and S. Teller, "Mobile-assisted localization in wireless sensor networks," in *Proc. 24th Annu. Joint Conf. IEEE Comput. Commun. Societies (INFOCOM)*, vol. 1, Mar. 2005, pp. 172–183.

[17] C. Y. Tazibt, M. Bekhti, T. Djamah, N. Achir, and K. Boussetta, "Wireless sensor network clustering for UAV-based data gathering," in *Proc. Wireless Days*, Mar. 2017, pp. 245–247.

[18] G. Han, J. Jiang, C. Zhang, T. Q. Duong, M. Guizani, and G. Karagiannidis, "A survey on mobile anchor node assisted localization in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2220–2243, 3rd Quart., 2016.

[19] J. Miranda *et al.*, "Path loss exponent analysis in wireless sensor networks: Experimental evaluation," in *Proc. 11th IEEE Int. Conf. Ind. Inform. (INDIN)*, Jul. 2013, pp. 54–58.

[20] S. Hamdoun, A. Rachedi, and A. Benslimane, "Comparative analysis of RSSI-based indoor localization when using multiple antennas in wireless sensor networks," in *Proc. Int. Conf. Sel. Topics Mobile Wireless Netw. (MoWNeT)*, Aug. 2013, pp. 146–151.

[21] L. Jayatilleke and N. Zhang, "Landmark-based localization for unmanned aerial vehicles," in *Proc. IEEE Int. Syst. Conf. (SysCon)*, Apr. 2013, pp. 448–451.

[22] J. D. Barton, "Fundamentals of small unmanned aircraft flight," *Johns Hopkins APL Tech. Dig.*, vol. 31, no. 2, pp. 132–149, 2012.

[23] E. Menegatti, A. Zanella, S. Zilli, F. Zorzi, and E. Pagello, "Range-only SLAM with a mobile robot and a wireless sensor networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2009, pp. 8–14.

[24] F. Caballero, L. Merino, P. Gil, I. Maza, and A. Ollero, "A probabilistic framework for entire WSN localization using a mobile robot," *Robot. Auto. Syst.*, vol. 56, no. 10, pp. 798–806, 2008.

[25] G.-L. Sun and W. Guo, "Comparison of distributed localization algorithms for sensor network with a mobile beacon," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, vol. 1, Mar. 2004, pp. 536–540.

[26] A. Galstyan, B. Krishnamachari, K. Lerman, and S. Pattem, "Distributed online localization in sensor networks using a moving target," in *Proc. 3rd Int. Symp. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2004, pp. 61–70.

[27] K.-F. Ssu, C.-H. Ou, and H. C. Jiau, "Localization with mobile anchor points in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 54, no. 3, pp. 1187–1197, May 2005.

[28] G. Teng, K. Zheng, and W. Dong, "Adapting mobile beacon-assisted localization in wireless sensor networks," *Sensors*, vol. 9, no. 4, pp. 2760–2779, 2009.

[29] X. Yang, Z. Gao, and Q. Niu, "Unmanned aerial vehicle–assisted node localization for wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 12, 2017, Art. no. 1550147717749818.

[30] L. A. Villas, D. L. Guidoni, and J. Ueyama, "3D localization in wireless sensor networks using unmanned aerial vehicle," in *Proc. 12th IEEE Int. Symp. Netw. Comput. Appl. (NCA)*, Aug. 2013, pp. 135–142,

[31] C. L. Nguyen and U. Raza, "Localization of WSNs using a location-unaware UAV," in *Proc. IEEE Int. Conf. Commun.*, May 2019.

[32] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, 2007.

[33] X. Ji and H. Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Societies (INFOCOM)*, vol. 4, Mar. 2004, pp. 2652–2661.

[34] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based localization in wireless sensor network," in *Proc. 30th Anniversary IEEE Conf. Local Comput. Netw.*, Nov. 2005, p. 514.

[35] O. Georgiou, K. Mimis, D. Halls, W. H. Thompson, and D. Gibbins, "How many Wi-Fi APs does it take to light a lightbulb?" *IEEE Access*, vol. 4, pp. 3732–3746, 2016.

[36] T. S. Rappaport, *Wireless Communications: Principles and Practice* (Prentice Hall Communications Engineering and Emerging Technologies Series), 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[37] A. Mosek, "The MOSEK python optimizer API manual version 7.1 (revision 62)," Tech. Rep., 2017. [Online]. Available: https://docs.mosek.com/7.1/pythonapi.pdf

[38] F. G. Costa, J. Ueyama, T. Braun, G. Pessin, F. S. Osório, and P. A. Vargas, "The use of unmanned aerial vehicles and wireless sensor network in agricultural applications," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2012, pp. 5045–5048.

[39] J. Polo, G. Hornero, C. Duijneveld, A. García, and O. Casas, "Design of a low-cost wireless sensor network with UAV mobile node for agricultural applications," *Comput. Electron. Agricult.*, vol. 119, pp. 19–32, Nov. 2015.

[40] G. Tuna, T. V. Mumcu, K. Gulez, V. C. Gungor, and H. Erturk, "Unmanned aerial vehicle-aided wireless sensor network deployment system for post-disaster monitoring," in *Proc. Int. Conf. Intell. Comput.* Berlin, Germany: Springer, 2012, pp. 298–305.

[41] I. Jawhar, N. Mohamed, and J. Al-Jaroodi, "UAV-based data communication in wireless sensor networks: Models and strategies," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2015, pp. 687–694.

[42] M. Dong, K. Ota, M. Lin, Z. Tang, S. Du, and H. Zhu, "UAV-assisted data gathering in wireless sensor networks," *J. Supercomput.*, vol. 70, no. 3, pp. 1142–1155, 2014.

[43] S. Rashed and M. Soyturk, "Analyzing the effects of UAV mobility patterns on data collection in wireless sensor networks," *Sensors*, vol. 17, no. 2, p. 413, 2017.

[44] H. Okcu and M. Soyturk, "Distributed clustering approach for UAV integrated wireless sensor networks," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 15, nos. 1–3, pp. 106–120, 2014.

[45] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni, "A survey on wireless indoor localization from the device perspective," *ACM Comput. Surv.*, vol. 49, no. 2, p. 25, 2016.

[46] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter level localization using WiFi," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 269–282, 2015.

**CAM LY NGUYEN** received the B.Sc. degree in computer science and engineering from Osaka University, Osaka, Japan, in 2010, and the M.Sc. degree in computer science and the Ph.D. degree in information and communications technologies from the University of Tokyo, Tokyo, Japan, in 2012 and 2019, respectively.

Since 2012, she has been with the Corporate Research & Development Center, Toshiba Corporation, Japan, as a Research Engineer, with a focus on wireless networks. She has filed over ten patents applications and has authored or co-authored seven papers as the first author in leading journals and conferences of the Internet and communications. Her current research interests include wireless localization, the Internet of Things, combinatorial optimization, and machine learning.

**USMAN RAZA** received the Ph.D. degree in information and communications technologies from the University of Trento, Italy. He is currently a Principal Research Engineer with Toshiba Research Europe Ltd., Bristol, U.K. During his multiple roles in both academia and industry, he has filed patents, published papers and book chapters, participated as an invited panelist, and co-chaired scientific events. His current research interests include cyber-physical systems, industrial wireless systems, and low power wide area networks. He was a recipient of the Endeavour Research Fellowship at The University of New South Wales, Australia, and an offer of the Fulbright Scholarship from the U.S. Department of State. He received two Best Paper Awards at the IEEE PerCom and the IEEE SenseApp, including the Mark Weiser Award. His *Wireless Communications letters* is recently nominated for the IEEE Communications Society Heinrich Hertz Award. He has successfully bid for very competitive and prestigious European Horizon 2020 Projects.