Hindawi Publishing Corporation
Journal of Computer Systems, Networks, and Communications
Volume 2010, Article ID 749298, 12 pages
doi:10.1155/2010/749298

Research Article

Hidden Anchor: A Lightweight Approach for Physical Layer Location Privacy

Rania El-Badry, Moustafa Youssef, and Ahmed Sultan

Wireless Intelligent Networks Center, Nile University, 12677 Cairo, Egypt

Correspondence should be addressed to Moustafa Youssef, mayoussef@nileu.edu.eg

Received 1 February 2010; Accepted 15 May 2010

Academic Editor: Christos Verikoukis

In hybrid wireless sensor networks, where trusted and un-trusted nodes coexist, it becomes important to allow trusted nodes to share information, especially, location information and prevent un-trusted nodes from gaining access to this information. We focus on anchor-based localization algorithms in WSNs, where a small set of specialized nodes, that is, anchor nodes, broadcast their location to the network and other nodes can use the broadcast information to estimate their own location. The main challenge is that both trusted and un-trusted nodes can measure the physical signal transmitted from anchor nodes and use it to estimate their locations. In this paper, we propose Hidden Anchor, an algorithm that provides anchor physical layer location privacy for different classes of localization algorithms. The Hidden Anchor algorithm exploits the inherently noisy wireless channel and uses identity cloning of neighboring trusted nodes to make anchors unobservable to un-trusted nodes while providing complete information to trusted nodes. Evaluation of the Hidden Anchor algorithm through analysis and simulation shows that it can hide the identity, and hence the location, of anchor nodes with very low overhead. In addition, the results show that by adding artificial noise, we can achieve significant improvement in anchor's location privacy.

1. Introduction

Location discovery has been an active area of research in wireless sensor networks (WSN) due to its critical need in many applications including location-based routing [1], coverage [2], node identification, and information tagging. Localization algorithms can be categorized as either anchor-based or anchor-free [3]. Anchor-based algorithms, for example, [4, 5], assume the existence of a small set of nodes with known locations, that is, anchor nodes, that broadcast their location information to the network in special *beacon* frames. A node with an unknown location estimates its distance to the anchor node, in a process known as ranging, and combines the estimated distance to at least three anchor nodes with the broadcast anchors' locations in beacon frames to estimate its location in 2D (Figure 1). On the other hand, anchor-free localization algorithms, for example, [6, 7], do not assume the existence of anchor nodes and estimate the relative topology of the network, in which the coordinate system is established by a reference group of nodes. This paper focuses on anchor-based localization algorithms using Received Signal Strength (RSS) for ranging.

In many hybrid wireless sensor networks' (HWSNs) applications, sensor nodes are deployed in hostile environments where trusted and un-trusted nodes co-exist. In such hybrid networks, it becomes important to allow trusted nodes to share information while, at the same time, prevent un-trusted nodes from gaining access to this information.

An anchor node may encrypt its beacon frames with a key shared only with trusted nodes. This will prevent un-trusted nodes from getting the information contained in the beacon frames. Although encryption can provide location information secrecy, it does not provide *physical layer* location privacy, where a group of un-trusted nodes can measure the received signal strength (RSS) of encrypted messages and cooperate to determine the anchor nodes' locations through trilateration. This paper proposes an algorithm, termed *Hidden Anchor*, that addresses the physical layer location privacy problem. In particular, the *Hidden Anchor* algorithm provides *anchor nodes unobservability*, where un-trusted nodes cannot detect (observe) the existence of anchor nodes.

In [8, 9], we proposed the *HyberLoc* algorithm for addressing the physical layer location privacy problem.

HyberLoc depends on the anchor nodes to dynamically change their transmission power and to include the used transmission power in the encrypted beacon frame. However, *HyberLoc*'s advantage is limited because of the current limitations of the sensor hardware as we elaborate in Section 5.1.

Our novel approach in the *Hidden Anchor* algorithm is to exploit the noisy characteristics of the wireless channel to hide the location of anchor nodes from un-trusted nodes, while providing complete information to trusted nodes. The idea is for anchor nodes to randomly use the identity of the *nearby*, that is, within a given distance, trusted nodes when broadcasting their beacon frames. As a result, un-trusted nodes will not be able to distinguish between anchor traffic and trusted node traffic. Shared information between the anchor and trusted nodes is used to give complete location information to trusted nodes. We evaluate the performance of the *Hidden Anchor* algorithm using analysis and simulation with metrics for both deterministic and probabilistic anchor-based location determination systems. The results show that the *Hidden Anchor* algorithm can hide the location and identity of anchor nodes while maintaining very low overhead.

Since the *Hidden Anchor* algorithm depends mainly on exploiting the noisy characteristics of the wireless channel, we further extend it by inducing artificial noise to the network in a way that does not affect the localization accuracy at trusted nodes. Analysis of the proposed noise induction technique shows significant improvement of the security of the proposed techniques. This is particularly important in handling the case when the un-trusted node uses a probabilistic location determination technique, which is known to give higher accuracy than deterministic location determination techniques [10, 11].

The rest of this paper is organized as follows. Section 2 discusses some related work. Section 3 highlights some background information. Section 4 presents the problem statement. Section 5 details the *Hidden Anchor* algorithm and analyzes its performance. Section 6 discusses our artificial noise extension to the basic *Hidden Anchor* algorithm. Section 7 evaluates the *Hidden Anchor* algorithm through simulation. Finally, Section 8 concludes the paper.

2. Related Work

In this section, we discuss different work related to the *Hidden Anchor* algorithm. The authors in [12, 13] proposed a technique that uses sophisticated PHY-layer measurements in a wireless network for location distinction. The proposed technique, temporal link signature, is a multipath-based location distinction technique. The main goal was to detect whether a node has moved from its location or not. Using link signature, this technique can determine that a node has changed its location, for example, when the anchor node clones the ID of a neighboring node in the *Hidden Anchor* algorithm. However, the technique depends on collecting training data, which makes it unsuitable for use with non-cooperating nodes, which is the case in hybrid WSNs.

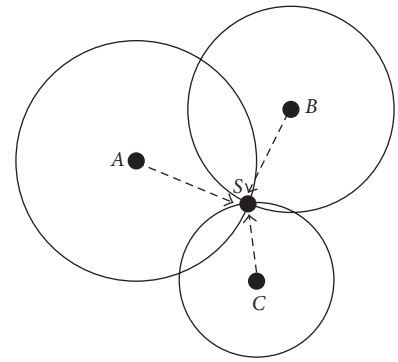

Figure 1: Node S can estimate its location in 2D using the location messages received from the three anchor nodes A, B, and C and the estimated range to them. Similarly, we can understand this figure as three un-trusted nodes A, B, and C cooperating to estimate the location of an anchor node S.

Therefore, it cannot be used as an attack on the *Hidden Anchor* algorithm.

The authors in [14] proposed *SlyFi*, an 802.11-like wireless link layer protocol that encrypts entire frames, including addresses, to remove explicit identifiers that could be used by third parties to link together frames from the same transmitter. It was proposed to improve wireless privacy. *SlyFi*, although suitable for a WiFi network, is based on a number of assumptions that may not fit the nature of sensor networks. First, *SlyFi* requires each node to keep a set of shared keys for every node that it may communicate with and a table of the possible incoming addresses for every time interval. Second, *SlyFi* requires all nodes in the network to be synchronized. These are considered a high overhead for sensor nodes which are limited in memory, processing power, and battery.

As will be discussed in Section 5, *HyberLoc* [8, 9] provides a weaker notion of location privacy, that is, location anonymity, than the unobservability provided by the *Hidden Anchor* algorithm.

3. Background

In this section, we provide background information on localization algorithms for WSNs. For more details, the reader is referred to [3].

3.1. Anchor-Based versus Anchor-Free Algorithms. Location discovery algorithms for sensor networks can be classified as anchor-based or anchor-free algorithms. Anchor-based algorithms, for example, [4, 5], assume that a small percentage of the nodes, that is, anchor nodes, are aware of their positions. Anchor nodes broadcast their location information to their neighbors which use this information to estimate their own location. In anchor-free algorithms, for example, [6, 7], no special anchor nodes exist in the network. In this case, the algorithm estimates relative positions, in which the coordinate system is established by a reference group of nodes. Relative positioning is suitable for

some applications, for example, location-aided routing [15]. However, the accuracy of anchor-free algorithms is typically less than anchor-based algorithms. This paper focuses on anchor-based algorithms.

3.2. Range Estimation Method. Ranging is the process of estimating node-to-node distances or angles. In order to determine its location in 2D, a sensor node needs to estimate its range to three or more anchor nodes. The most popular methods for estimating the range between two nodes are Time-based methods, for example, Time-of-Arrival (ToA), Angle-of-Arrival methods (AoA), and Received-Signal-Strength (RSS) methods. This paper focuses on RSS-based range estimation methods where the propagation loss can be calculated based on the difference in power between the transmitted and received signals. Theoretical and empirical models are used to translate this loss into a distance estimate. Combining the positioning information received from at least three anchor nodes with the estimated distances, a sensor node can estimate its location in 2D (Figure 1).

3.3. Wireless Channel-Based Security. The broadcast nature of wireless communications makes them vulnerable to many security attacks. Many researchers have used the characteristics of the wireless channel to solve this intrinsic security problem of wireless communications. For example, the authors in [16] exploit the multipath fading characteristic of wireless communication to facilitate secret key sharing between a sender and a receiver.

In this paper, we exploit the noisy characteristic of the wireless channel to provide physical layer location privacy for wireless devices. We start by exploiting the environment noise to secure the location of anchor nodes and then extend the idea by inducing artificial noise.

4. Problem Statement

This section outlines the network model and security and privacy requirements. We also delineate the different ways an un-trusted node can use to identify the existence of an anchor node.

4.1. Network Model. We assume a hybrid wireless sensor network where anchor, trusted, and un-trusted nodes co-exist. We also assume that nodes use an RSS anchor-based localization algorithm. Thus, anchor nodes continuously broadcast beacon frames containing their position information.

Any node in the network can observe any frame transmitted by other nodes within its range. Sensor nodes are randomly distributed in the area of interest. Trusted nodes can use standard encryption algorithms to hide the anchor nodes' position information where both anchor nodes and trusted nodes share the required common information, for example, cryptographic keys, prior to deployment.

Un-trusted nodes use the same radio hardware used by anchor nodes and trusted nodes. We further assume that there is no correlation between the frame information, such as size and content, and the frame type. Therefore, un-trusted nodes cannot differentiate between the frames from trusted nodes and those from anchor nodes. This can be achieved by encrypting the contents or padding the frames as needed.

We also assume that the goal of the un-trusted nodes is to estimate their range to anchor nodes based on the physical signal transmitted by anchor nodes.

4.2. Security and Privacy Requirements. By considering the network model discussed in the previous section, we have two main requirements that should be considered.

4.2.1. Location Information Secrecy. Anchor nodes should be able to broadcast their position information periodically and trusted nodes should be able to use this information to estimate their position. On the other hand, un-trusted nodes should not be able to use anchor nodes' beacon frames to gain information about anchor nodes' locations. This can be achieved, for example, by encrypting the anchor nodes' beacon frames.

4.2.2. Physical Layer Location Privacy. Un-trusted nodes should not be able to exploit the measured physical signal to estimate the location of anchor nodes. This paper focuses on this privacy requirement.

4.3. Identifying Anchor Nodes. An un-trusted node may exploit one or a combination of the following vulnerabilities to identify the existence of an anchor node. Note that detecting the existence of the anchor node is a necessary prelude to determine its location.

4.3.1. Separate ID-Space for Anchor Nodes. If the WSN is designed such that the ID-space for anchor nodes is separate from the ID-space of trusted nodes, an un-trusted node can identify the existence of an anchor node by its ID in the frame header.

4.3.2. Type of Transmission—Broadcast/Unicast. Beacon frames are broadcast in the network. If only anchor nodes broadcast messages in the network, their frames can be distinguished from the frames of other nodes by noting the broadcast destination address. However, regular sensor nodes use both broadcast and unicast to transmit their own messages, making this way less useful for the un-trusted node.

4.3.3. Periodicity of Frames. Even if other nodes send broadcast frames, the periodicity of the beacon frames make them easier to detect and hence expose the anchor node.

4.3.4. Frame Size. Beacon frames usually have a fixed size. This can make them easily distinguishable by the un-trusted node.

4.3.5. Type Field in Frame Header. If there is a field in the frame header to identify the different message types, this can be used to determine the anchor node by the type of messages it sends.

Except for the last two vulnerabilities, which can be easily mitigated by padding the frame with random data and encrypting the frame, respectively, the *Hidden Anchor* algorithm mitigates the remaining three vulnerabilities to achieve anchor node unobservability as discussed in the next section.

5. The Hidden Anchor Algorithm

In this section, we start by a possible technique that addresses the physical layer location privacy problem. We show that this technique has shortcomings, thus motivating the need for a the *Hidden Anchor* algorithm.

5.1. Possible First-Cut Solution. Anchor nodes can confuse un-trusted nodes using variable transmission power. For example in [8, 9], we proposed a light-weight algorithm that provides secure anchor-based localization in hybrid wireless sensor networks. The idea is for anchor nodes to continuously and randomly change the transmit power and to include the transmit power encrypted in the frame using the shared information between itself and trusted nodes. This change of transmit power will reduce the localization accuracy at the un-trusted nodes, achieving location anonymity (location anonymity refers to hiding the true location of an anchor node. This is a weaker notion than anchor node unobservability, where the anchor node existence is not detected at all). Trusted nodes can use the shared information to extract the transmit power from the frames and, therefore, their localization accuracy is not affected by the transmit power change.

Based on the current sensor network hardware, for example, [17], transmit power can be selected from a set of prespecified discrete power levels. This has the disadvantage that after receiving a sufficient number of frames, an un-trusted node will be able to distinguish between the different discrete received power levels, thus removing the ambiguity introduced by the random change of transmit power. As a result, un-trusted nodes will be able to localize anchor nodes accurately. In addition, location anonymity provided by this technique is a weaker privacy notion than the unobservability of anchor nodes provided by the *Hidden Anchor* algorithm.

In the next section, we propose the *Hidden Anchor* algorithm as a light-weight algorithm that provides physical layer location privacy and allows trusted nodes to accurately estimate their positions at a low overhead.

5.2. Hidden Anchor Algorithm. The *Hidden Anchor* algorithm exploits the noisy wireless channel to hide the existence of anchor nodes. The idea is for anchor nodes to use the identity of the nearby trusted nodes when broadcasting their beacon frames. This way, un-trusted nodes cannot differentiate between anchor nodes and trusted nodes. On receiving a frame with ID a, an un-trusted node cannot determine whether this frame is from trusted node a or from an anchor node with the same ID. Note that since the anchor node chooses its ID from nearby nodes, its location is hidden within the noise of the wireless channel. The algorithm operates in two phases: the neighbor discovery phase and the location hiding phase.

5.2.1. Neighbor Discovery Phase. The purpose of this phase is for the anchor node to discover the IDs of the nearest neighbors so that the anchor node can select which IDs to use during the next phase.

The anchor node may broadcast an encrypted identity-request message using a random ID to all its neighbors within a certain radius. This can be controlled by a hop count parameter. All trusted nodes in the network that receive this message reply with an identify-reply message.

The anchor node waits for a certain time to collect the identify-reply messages along with their received signal strength. After that, the anchor node sorts the nodes in an ascending order with respect to their received signal strength and saves the identities of the k nearest trusted nodes in a set ($\mathbb{S}$).

The anchor node can also discover the IDs of the nearest neighbors by passively monitoring the network traffic for a sufficient time period.

5.2.2. Location Hiding Phase. In this phase, and when it is time to send a beacon frame, the anchor node chooses one ID from the set $\mathbb{S}$ randomly and uses it as its *unencrypted* identity. The true identity of the anchor node, along with the type of the message can be sent encrypted in the body of the message, if needed. Upon receipt of a broadcast beacon frame, a trusted node decrypts the frame and can determine the identity and location of an anchor node. On the other hand, an un-trusted node, not knowing the decryption key, cannot differentiate between the frames from the anchor nodes and trusted nodes, as they have the same identity and the difference in their signal strength is within the wireless transmission noise.

5.3. Discussion. For the vulnerabilities identified in Section 4.3, the *Hidden Anchor* algorithm eliminates any chance for un-trusted nodes to identify anchor nodes using their IDs as anchor nodes never use their real IDs in clear, which is equivalent to using only the trusted nodes ID-space. Also, the proposed algorithm removes the periodicity of beacon frames by using a different ID every time for the frames transmitted from anchor nodes. Finally, since anchor nodes clone the identity of trusted nodes, their traffic pattern, as seen by the un-trusted nodes, becomes indistinguishable.

Note also that, even if un-trusted nodes cooperate to determine the location of all trusted nodes, they cannot determine the location of anchor nodes, as anchor nodes are unobservable. Statistical analysis is not useful here too as anchor nodes use the IDs of trusted nodes for sending their own traffic.

Compared to *HyberLoc*, the *Hidden Anchor* algorithm does not depend on changing the power levels and therefore, it is not affected by the current sensor network hardware limitations. Also, while the *HyberLoc* algorithm provides anchor node location anonymity, the *Hidden Anchor* algorithm provides the stronger notion of anchor nodes unobservability.

5.4. Analysis. In this section, we derive expressions for the difference in received signal power, which is directly related to the average difference in estimated distance, and the statistical Kolmogorov-Smirnov (KS) test value at the un-trusted receiver both when the *Hidden Anchor* algorithm is used and without using it in the presence of noise.

5.4.1. Difference in Estimated Distance. The difference in estimated distance metric represents the error in estimated distance when the trusted node is sending by itself on one hand and when both the trusted node and the anchor node share the same ID, that is, using the *Hidden Anchor* algorithm. The distance estimate is based on the average received signal strength at the un-trusted node. The lower the value of this metric, the better in terms of privacy requirement as the un-trusted node will be unlikely to detect that both trusted node and anchor node are sharing the same ID.

5.4.2. KS Test Value. This metric represents the value of the Kolmogorov-Smirnov (KS) statistical test. The KS test value gives a measure of similarity between the probability distribution of the signal strength received at the un-trusted node with and without using the *Hidden Anchor* algorithm. It gives an insight about how a probabilistic location determination algorithm would perform with and without using the *Hidden Anchor* algorithm. A lower value of this metric represents better security as the un-trusted node will be less likely to differentiate between the signal strength distributions of the trusted node and anchor node.

5.4.3. Notation. We use the following notation.

(i) We consider a signal propagation model that has a dominant line-of-sight (LOS) component. In the presence of Additive White Gaussian Noise (AWGN), the probability density function (PDF) of the received signal power [18] is

$$f(P_r \mid h, x) = \frac{1}{2\sigma^2} \exp\left(-\frac{P_r + hx}{2\sigma^2}\right) I_0\left(\frac{\sqrt{P_r hx}}{\sigma^2}\right), \quad (1)$$

where P_r is the received signal power, h is the channel gain which is a function of distance, x is the transmission power, $2\sigma^2$ is the total noise variance, and $I_0(x)$ is the modified Bessel function of order zero.

Thus, the mean of the received signal power is

$$\mu = 2\sigma^2 + hx, \quad (2)$$

and the CDF of the received signal power follows a non-central chi-square distribution given by

$$\chi(P_r \mid h, x) = \sum_{j=0}^{\infty} \exp\left(\frac{-hx}{2\sigma^2}\right) \frac{(hx/2\sigma^2)^j}{j!} Q(P_r; 2 + 2j), \quad (3)$$

where $Q(m, n)$ is the CDF of a central chi-square distributed random variable with n degrees of freedom.

(ii) v_{HA}. A random variable representing the average received power from the same ID over n samples at the un-trusted receiver when the *Hidden Anchor* algorithm is used.

(iii) $v_{\overline{\mathrm{HA}}}$. A random variable representing the average received power from the same ID over n samples at the un-trusted receiver when the *Hidden Anchor* algorithm is not used.

(iv) P_r. A random variable representing the power received from the same ID at the un-trusted node, whether from the trusted node or the anchor node.

(v) P_{r_t}. A random variable representing the power received at the un-trusted node from the trusted node.

(vi) P_{r_a}. A random variable representing the power received at the un-trusted node from the anchor node.

(vii) r. Traffic ratio, that is, ratio of the traffic from the trusted node to the traffic from the anchor node.

(viii) α. In the KS test, the parameter α controls the probability of rejecting the null hypothesis that the samples are from the same distribution.

5.5. Metric 1: Difference in Received Power. In this section, we derive an expression for the difference in received power with and without using the *Hidden Anchor* algorithm, which is directly related to the average in estimated distance. Using the above notation, the average received power when the *Hidden Anchor* algorithm is used, v_{HA}, over n samples is given by

$$\begin{aligned} v_{\mathrm{HA}} &= \frac{1}{n}\sum_{i=1}^{n} P_{r_i} \\ &= \frac{1}{n}\left(\sum_{i=1}^{nr/(1+r)} P_{r_{t_i}} + \sum_{i=1}^{n/(1+r)} P_{r_{a_i}}\right), \end{aligned} \quad (4)$$

where P_{r_i} represents the ith sample from the corresponding random variable. From (4),

$$\begin{aligned} E[v_{\mathrm{HA}}] &= \frac{1}{n}\left[\sum_{i=1}^{nr/(1+r)} E\left(P_{r_{t_i}}\right) + \sum_{i=1}^{n/(r+1)} E\left(P_{r_{a_i}}\right)\right] \\ &= \frac{1}{r+1}\left(rE[P_{r_t}] + E[P_{r_a}]\right) \\ &= \frac{r}{r+1}\left(2\sigma^2 + h_t x_t\right) + \frac{1}{r+1}\left(2\sigma^2 + h_a x_a\right) \\ &= 2\sigma^2 + \frac{r}{r+1}\left(h_t x_t\right) + \frac{1}{r+1}\left(h_a x_a\right). \end{aligned} \quad (5)$$

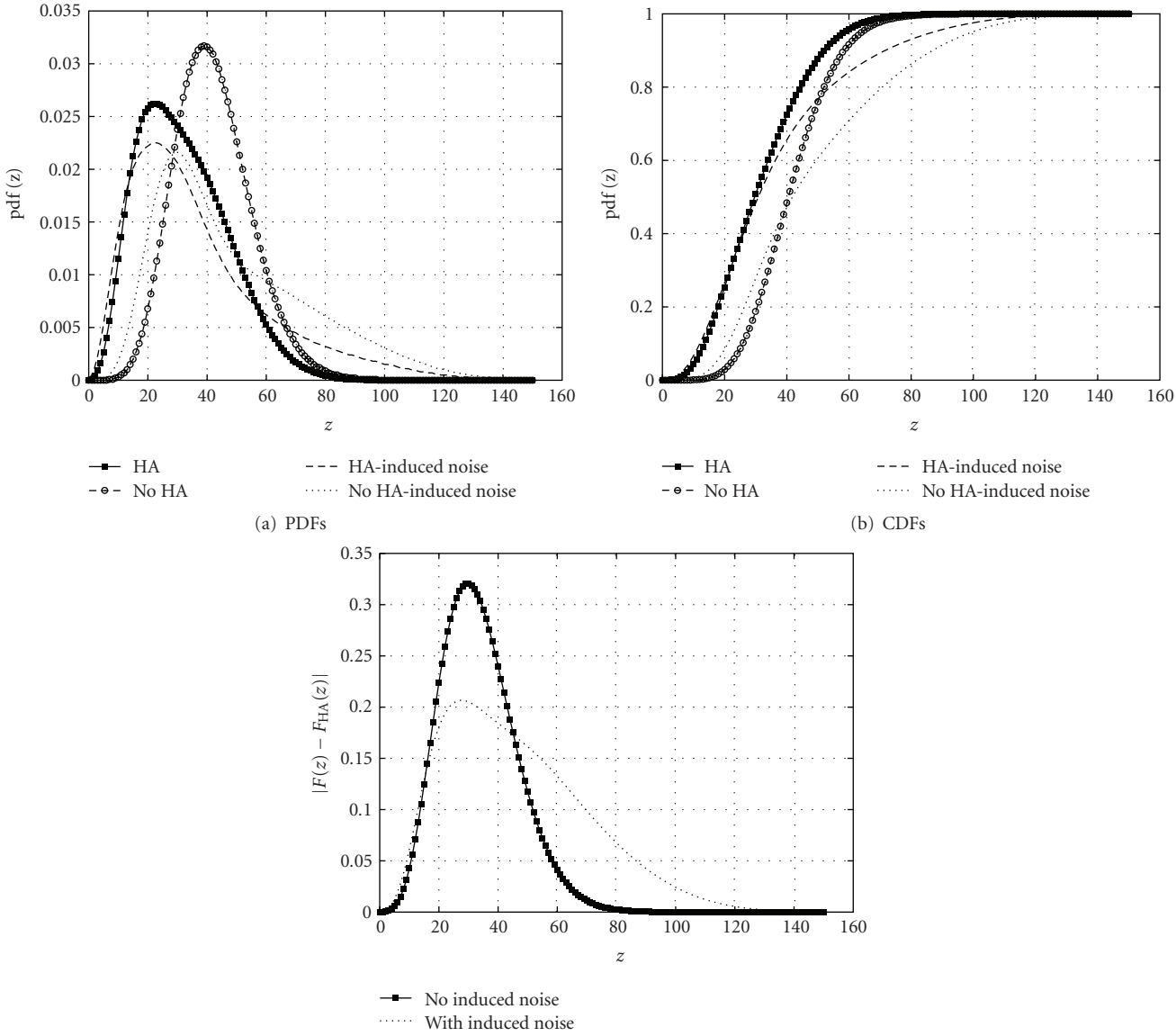

Figure 2: Performance of the *Hidden Anchor* algorithm with and without induced noise for low receiver noise.

Similarly, when the *Hidden Anchor* algorithm is not in use, the expected average received power, $E(v_{\overline{\text{HA}}})$, over n samples is given by

$$E[v_{\overline{\text{HA}}}] = 2\sigma^2 + h_t x_t. \tag{6}$$

By subtracting (5) from (6), the expected difference in received power between using the *Hidden Anchor* algorithm and not using it is given by

$$\begin{aligned} &E\,[\text{Difference in Power Received}] \\ &= \frac{1}{r+1}(h_t x_t - h_a x_a). \end{aligned} \tag{7}$$

5.6. Metric 2: Kolmogorov-Smirnov Test Value. The Kolmogorov-Smirnov statistic is used to test whether two underlying one-dimensional probability distributions differ. It is based on quantifying the distance between the empirical cumulative distribution functions, F_1 and F_2 of two sets and is defined as

$$\text{KS Test Value} = \sup|F_1(x) - F_2(x)|. \tag{8}$$

Using the above notation, the cumulative distribution function of the power received from the same ID when the *Hidden Anchor* algorithm is used, is given by

$$\begin{aligned} F_{\text{HA}}(P_r) &= \frac{1}{r+1}F_a(P_r) + \frac{r}{r+1}F_t(P_r) \\ &= \frac{1}{r+1}\chi_a(P_r \mid h_a, x_a) + \frac{r}{r+1}\chi_t(P_r \mid h_t, x_t). \end{aligned} \tag{9}$$

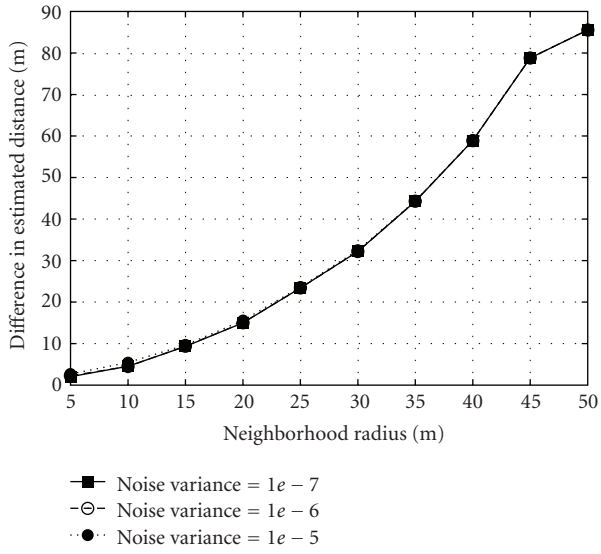

Figure 3: Effect of changing the noise level and the neighborhood radius on estimated distance at the un-trusted node.

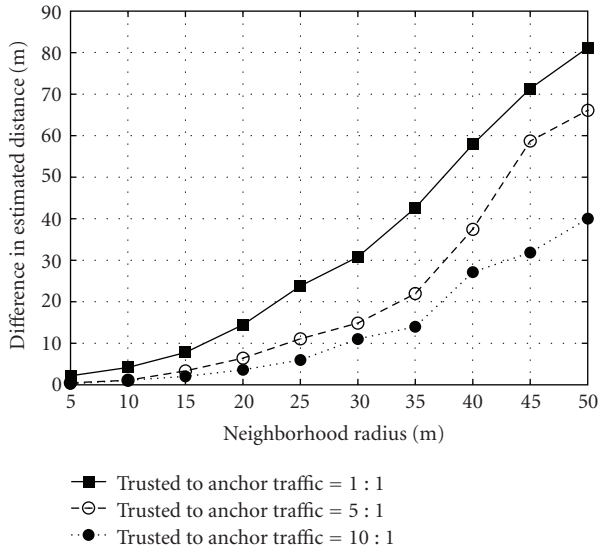

Figure 4: Effect of changing the traffic ratio and the neighborhood radius on estimated distance at the un-trusted node.

Similarly, the cumulative distribution function of the power received from the same ID when the *Hidden Anchor* algorithm is not used is given by

$$F_{\overline{\mathrm{HA}}}(P_r) = \chi_t(P_r \mid h_t, x_t) \tag{10}$$

From (9) and (10), the KS test value is given by

$$\text{KS Test Value} = \frac{1}{r+1} \sup \left| \chi_t(P_r \mid h_t, x_t) - \chi_a(P_r \mid h_a, x_a) \right|. \tag{11}$$

5.7. Discussion. The performance of the *Hidden Anchor* algorithm depends on many parameters including the noise level, the ratio between the traffic sent by the anchor and the traffic sent by the nearest trusted nodes, and the distance between the anchor node and the un-trusted node which affects the channel gain. For both metrics, the performance of the *Hidden Anchor* algorithm can be arbitrary enhanced by controlling the traffic ratio and the distance between the anchor node and trusted nodes (neighborhood radius parameter). For the deterministic metric, the performance is independent from the noise at the receiver. This is expected as averaging removes the effect of noise at the receiver. This is not true however for the probabilistic metric, where the entire signal strength distribution makes an effect on performance. Note that probabilistic location determination techniques are known to produce higher accuracy than deterministic techniques as they use more information [10, 11].

In Section 7, we validate our analysis and show the effect of each parameter on the performance of the algorithm in more detail.

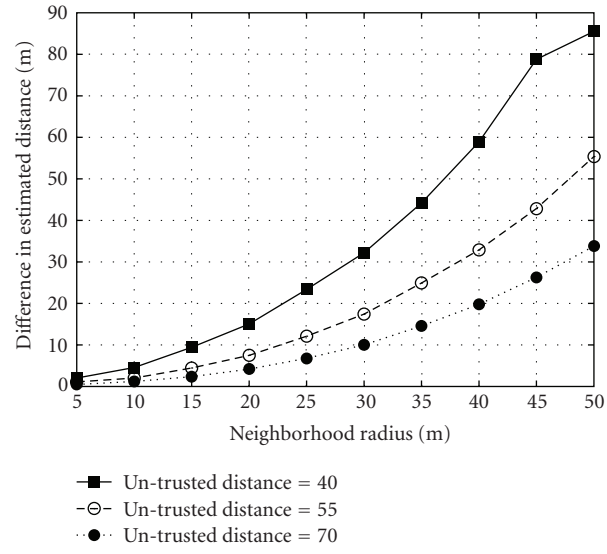

Figure 5: Effect of changing the un-trusted distance and the neighborhood radius on estimated distance at the un-trusted node.

6. Adding Artificial Noise

The KS test value metric evaluates the performance of the *Hidden Anchor* algorithm when un-trusted nodes use a probabilistic location determination technique. Intuitively, this metric should decrease, indicating better security, as the noise level increases (Figure 6). Thus, if we can increase the noise level artificially, we can achieve better anchor's location privacy. The challenge is to increase the ambiguity at the un-trusted nodes without affecting the localization accuracy at the trusted nodes. In this section, we present two techniques that can be used to achieve this goal.

6.1. Symbol-Level Noise. Trusted nodes and anchor nodes can intentionally add a noise vector to the transmitted signal in a way that guarantees statistically independent, identically

distributed, Gaussian random signal samples at the receiver. This induced noise can be picked out of a predetermined noise codebook. Knowing the noise vector used by an anchor node, any trusted node can easily subtract the added noise to accurately localize this anchor node. For the un-trusted nodes, the added noise vector is unknown. Thus, the noise level is higher than the actual level.

To clarify the idea, assume that the received signal (Z) can be represented as:

$$Z = Xh + n, \tag{12}$$

where X is the transmitted signal, h is the channel gain, and n is Additive White Gaussian noise.

Using the proposed technique, the transmitted signal is now composed of the actual transmitted signal and the added noise. Therefore, the received signal can be represented as

$$Z = (X + n_a)h + n, \tag{13}$$

where n_a is the noise added by the anchor node.

For trusted nodes, both X and n_a are known. Thus, trusted nodes can easily estimate the distance to anchor nodes by estimating h. For un-trusted nodes, not knowing the added noise vector, the received signal is given by

$$Z = Xh + n_a h + n, \tag{14}$$

$$Z = Xh + v. \tag{15}$$

As shown in (15), v is now the summation of two noise components. By applying this technique, the noise at the un-trusted receiver will be increased, leading to better privacy, without affecting the localization accuracy at trusted nodes.

6.2. Frame-Level Noise. Adding noise at the symbol-level requires changing the hardware. To use the same hardware, trusted nodes and anchor nodes can add a frame-level noise by randomly changing their transmission power. Using this technique, not knowing the transmission power used, un-trusted nodes can only estimate some noisy estimates of the distance to trusted node. If both trusted and anchor nodes randomize their transmission power at the same time, the un-trusted nodes will not be able to detect any changes in the received signal strength when both trusted and anchor nodes share the same ID, thus, increasing anchors' location privacy.

Note that this induced frame-level noise technique is different from the power randomization technique used in *HyberLoc* [8, 9] as in *HyberLoc*, only anchor nodes change their transmission power while in the induced frame-level noise techniques, both trusted and anchor nodes change their transmission power. The average transmission power of the distribution used can be constrained to a certain value to meet the energy efficiency requirements of the WSN.

6.3. Analysis. For the symbol-level induced noise, the analysis of the performance of the *Hidden Anchor* algorithm is identical to the analysis of Section 5.4 with increased noise. In this section, we analyze the performance of the *Hidden Anchor* algorithm with induced frame-level noise.

6.3.1. Metric 1: Difference in Received Power. The average received power when the *Hidden Anchor* algorithm with induced frame-level noise is used, v_{HA}, over n samples is given by

$$\begin{aligned} v_{\mathrm{HA}} &= \frac{1}{n}\sum_{i=1}^{n} P_{r_i} \\ &= \frac{1}{n}\left(\sum_{i=1}^{nr/(1+r)} P_{r_{t_i}} + \sum_{i=1}^{n/(1+r)} P_{r_{a_i}}\right), \end{aligned} \tag{16}$$

where P_{r_i} represents the ith sample from the corresponding random variable

$$\begin{aligned} E[v_{\mathrm{HA}}] &= \frac{1}{n}\left[\sum_{i=1}^{nr/(1+r)} E\left(P_{r_{t_i}}\right) + \sum_{i=1}^{n/(r+1)} E\left(P_{r_{a_i}}\right)\right] \\ &= \frac{1}{n}\left[\sum_{i=1}^{nr/(1+r)} \left(2\sigma^2 + h_t x_{t_i}\right) + \sum_{i=1}^{n/(r+1)} \left(2\sigma^2 + h_a x_{a_i}\right)\right] \\ &= 2\sigma^2 + \frac{1}{n}\left[h_t \sum_{i=1}^{nr/(1+r)} x_{t_i} + h_a \sum_{i=1}^{n/(r+1)} x_{a_i}\right]. \end{aligned} \tag{17}$$

Similarly, when the *Hidden Anchor* algorithm is not in use, the expected average received power, $E(v_{\overline{\mathrm{HA}}})$, over n samples is given by

$$E\left[v_{\overline{\mathrm{HA}}}\right] = 2\sigma^2 + \frac{h_t}{n}\sum_{i=1}^{n} x_{t_i}. \tag{18}$$

From (17) and (18), given an average power constraint, as $n \to \infty$, the difference in received signal converges to (7). Thus, for the deterministic case, the frame-level-induced noise gives no privacy advantage.

6.3.2. Metric 2: Kolmogorov-Smirnov Test Value. The cumulative distribution function of the power received when the *Hidden Anchor* algorithm with induced frame-level noise is used is given by

$$\begin{aligned} F_{\mathrm{HA}}(P_r) &= \frac{1}{r+1}\sum_{i=1}^{i_{\max}} P(x_{a_i}) F_a(P_r \mid x_{a_i}) \\ &\quad + \frac{r}{r+1}\sum_{i=1}^{i_{\max}} P(x_{t_i}) F_t(P_r \mid x_{t_i}) \\ &= \frac{1}{r+1}\sum_{i=1}^{i_{\max}} P(x_{a_i}) \chi_a(P_r \mid h_a, x_{a_i}) \\ &\quad + \frac{r}{r+1}\sum_{i=1}^{i_{\max}} P(x_{t_i}) \chi_t(P_r \mid h_t, x_{t_i}), \end{aligned} \tag{19}$$

where $i_{\max}$ is the number of power levels used.

Similarly, the cumulative distribution function of the power received when the *Hidden Anchor* is not used is given by

$$F_{\overline{\mathrm{HA}}}(P_r) = \sum_{i=1}^{i_{\max}} P(x_{t_i})\chi_t(P_r \mid h_t, x_{t_i}). \tag{20}$$

Thus, the KS test value is

$$\text{KS Test Value} = \frac{1}{r+1} \sup \left| \sum_{i=1}^{i_{\max}} P(x_{t_i})\chi_t(P_r \mid h_t, x_{t_i}) - \sum_{i=1}^{i_{\max}} P(x_{a_i})\chi_a(P_r \mid h_a, x_{a_i}) \right| \tag{21}$$

Note that (11) is a special case of (21) when $i_{\max} = 1$, that is, without randomization.

6.4. Numerical Evaluation. In this section, we numerically compare the performance of the proposed original *Hidden Anchor* algorithm (11) against the same algorithm with induced artificial noise (21). Figure 2 shows the PDF, CDF, and difference of CDFs (the maximum of this difference is proportional to the KS test value). The probability distribution used to induce the artificial noise is a discrete exponential distribution over three discrete power levels (i.e., $i_{\max} = 3$). The discrete exponential distribution is the entropy maximizing distribution under a given average power level constraint [9]. The figure shows that the proposed randomization technique reduces the KS test value, which is equivalent to better privacy. We will investigate the effect of different parameters on the performance of the *Hidden Anchor* algorithm in the next section.

7. Simulation Study

In this section we evaluate the performance of the *Hidden Anchor* algorithm and show the effect of changing different parameters on its performance. We also evaluate the performance of the *Hidden Anchor* algorithm with induced artificial noise.

7.1. Simulation Environment. The *Hidden Anchor* algorithm was implemented using Matlab. The sensor nodes were randomly distributed over a square of $100 \times 100\,\mathrm{m}^2$. Results represent the average over five different network topologies where every network was randomly generated with a different seed. Without loss of generality, we show the results for only one un-trusted node. We use the "difference in estimated distance" and "KS test value" metrics, described in Section 5.4.

7.2. Simulation Parameters. We evaluate the effect of different parameters on the difference in estimated distance metric and the KS test value. The parameters that we considered in this simulation are the following.

7.2.1. Noise Level. This parameter represents an additive white Gaussian noise with zero mean and total power of $2\sigma^2$.

7.2.2. Traffic Ratio. This parameter (r) represents the ratio of the traffic sent by the trusted node to the traffic sent by the anchor node.

7.2.3. Neighborhood Radius. This parameter represents the maximum distance between the anchor node and the trusted nodes whose IDs it clones. Note that the anchor nodes pick its identity randomly from this set of neighbors.

7.2.4. Un-Trusted Distance. This parameter represents the distance between the anchor node and the un-trusted node.

7.3. Results for the Hidden Anchor Algorithm

7.3.1. Effect of the Parameters on Metric 1: the Difference in Estimated Distance. Figure 3 verifies that the difference in estimated distance metric is not affected by the noise level at the receiver (7). Thus, we fix the noise level in this part to $1e-6$.

Figure 4 shows the effect of changing the neighborhood radius for different traffic ratios. The un-trusted distance is fixed to 70 m. The results show that as the traffic sent by the trusted node increases, relative to the traffic sent by the anchor node, the difference in the estimated distance decreases. Since typical anchor-based algorithms use low anchor traffic to trusted nodes traffic ratio, this shows the promise of the proposed *Hidden Anchor* algorithm.

Figure 5 shows the effect of changing the neighborhood radius with different un-trusted distance values. The traffic ratio is fixed to 1 : 1. The figure shows that as the distance between the anchor node and the un-trusted node increases, the difference in the estimated distance decreases. Consequently, the un-trusted node will not be able to differentiate between the physical signal transmitted by the anchor node and the physical signal transmitted by the trusted node. The reason is that the effect of the distance difference on signal strength between the anchor node and the neighboring trusted nodes diminishes as we go away from the anchor node.

7.3.2. Effect of the Parameters on Metric 2: the KS Test Value. Figure 6 shows the effect of changing the neighborhood radius on the KS test value for different noise levels. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively. Other parameters were fixed at un-trusted distance = 70 m, and traffic ratio = 1 : 1. As the noise level increases, the KS test value decreases. This indicates that the difference between the distribution of the received signal strength using the *Hidden Anchor* algorithm and without using it decreases as the noise level increases.

Figure 7 shows the effect of changing the neighborhood radius for different traffic ratios. Other parameters were fixed at un-trusted distance = 70 m, and noise variance = $1e-6$. The results show that as the traffic sent by the trusted node increases, with respect to the traffic sent by the anchor node,

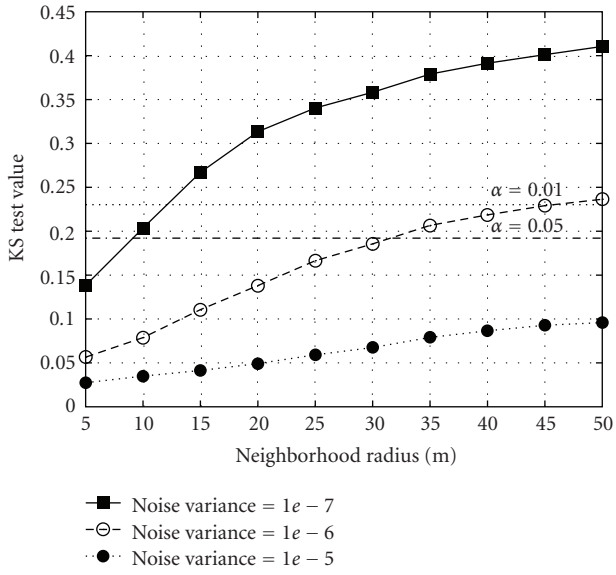

Figure 6: Effect of changing the noise level and the neighborhood radius on the KS test value. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively.

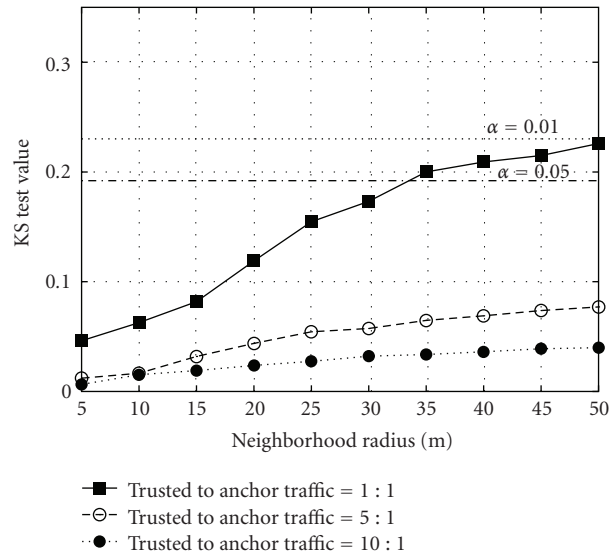

Figure 7: Effect of changing the traffic ratio and the neighborhood radius on the KS test value. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively.

the KS test value decreases. This means that we can improve the performance of the algorithm dramatically by reducing the traffic sent by the anchor, which is typical for anchor-based algorithms.

Figure 8 shows the effect of changing the neighborhood radius for different un-trusted distance values. Other parameters were fixed at traffic ratio = 1 : 1, and noise variance = $1e - 6$. The figure shows that as the distance between the anchor node and the un-trusted node increases, the difference in the estimated distance decreases. Consequently, the un-trusted node will not be able to distinguish between

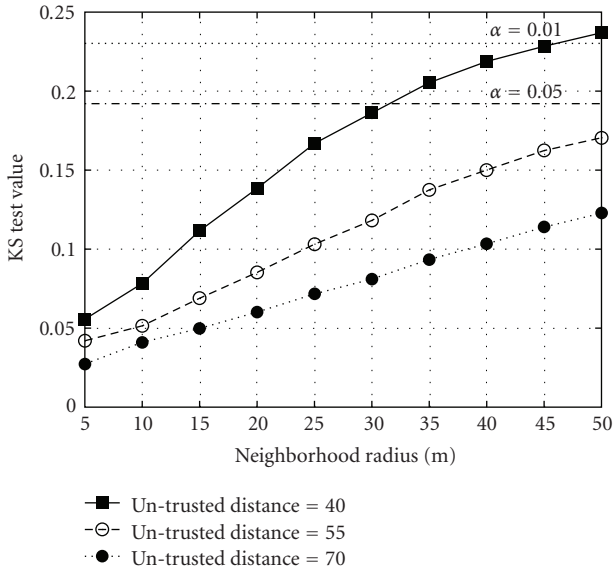

Figure 8: Effect of changing the neighborhood radius and the un-trusted distance on the KS test value. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively.

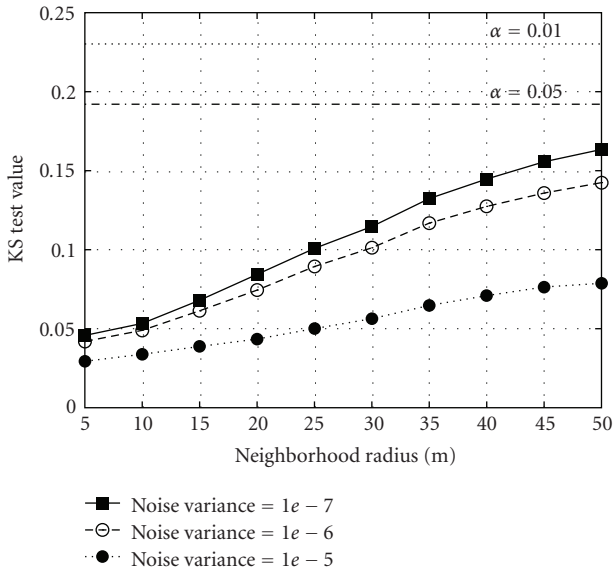

Figure 9: Effect of randomizing the transmission power on the KS test value for different noise levels. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively.

the physical signal transmitted by the anchor node and the physical signal transmitted by the trusted node.

7.4. Results for the Hidden Anchor Algorithm with Induced Artificial Noise. For Metric 1, as indicated by the analysis in Section 6, there is no advantage of randomizing the transmit power, since averaging over a large number of samples reduces the effect of randomization. The rest of this section

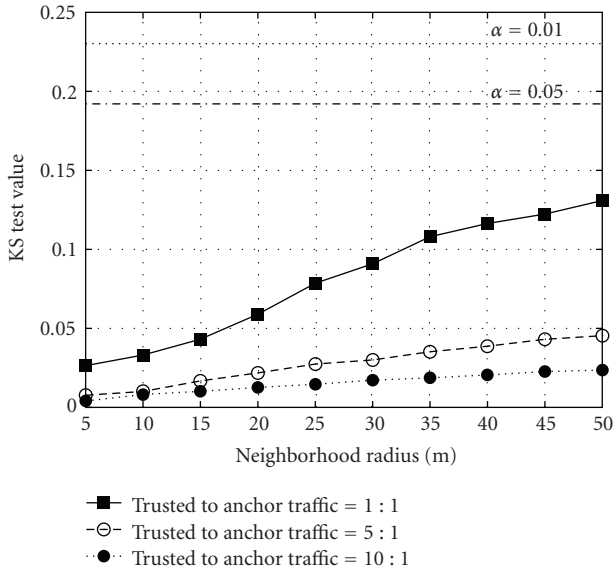

Figure 10: Effect of randomizing the transmission power on the KS test value for different traffic ratios. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively.

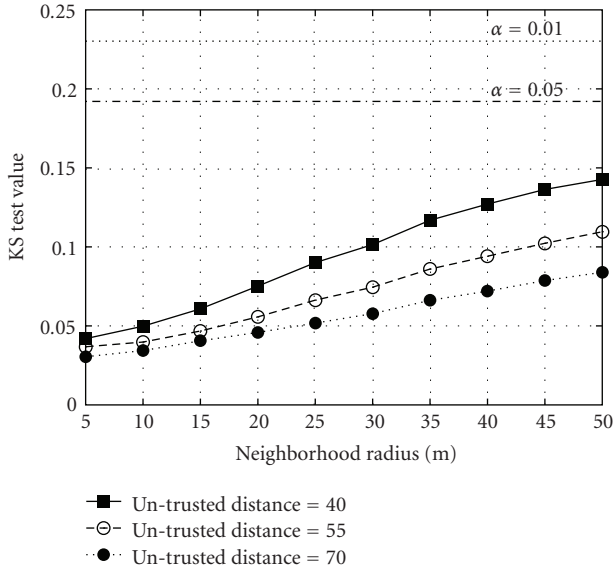

Figure 11: Effect of randomizing the transmission power on the KS test value for different un-trusted distances. The two horizontal lines represent the critical values for the test for $\alpha = 0.01$ and $\alpha = 0.05$, respectively.

evaluates the advantage of using induced artificial noise on Metric 2: the KS Test Value.

Figures 9, 10, and 11 evaluate the performance when the *Hidden Anchor* with induced artificial noise is used. The results show significant improvement in the KS test value when the anchor and trusted nodes randomize their power given an average power constraint. The simulation parameters used in Figures 9, 10, and 11 are the same parameters used in Figure 6, 7, and 8, respectively.

7.5. Summary. In this section, we have evaluated the performance of the *Hidden Anchor* algorithm for different parameters: the noise level, traffic ratio, the maximum distance between the anchor node and the neighboring trusted nodes, and the distance between the anchor node and the un-trusted node.

We have also evaluated the performance of the network when we added artificial noise to the *Hidden Anchor* algorithm by randomizing the transmission power. The results show significant improvement in anchors' location privacy when the un-trusted nodes use a statistical localization method like the KS test.

In all cases, the designer of the network can control the traffic ratio and the maximum distance between the anchor node and the neighboring trusted nodes. Reducing the anchor node to trusted nodes traffic ratio enhances security but may decrease the localization accuracy at trusted nodes. The results show that we can make the performance metric arbitrary small, indicating better privacy, by controlling these two parameters.

8. Conclusion

In this paper, we focused on the physical layer location privacy problem, where an anchor node wants to hide its physical signal information from un-trusted nodes, while at the same time allows trusted nodes to benefit from this information. We proposed the *Hidden Anchor* algorithm for solving the physical layer location privacy and evaluated its performance through analysis and simulation experiments. Our results show that the *Hidden Anchor* algorithm can effectively hide the location and identity of anchor nodes without limiting the localization accuracy for trusted nodes, thus providing anchor nodes' unobservability. The *Hidden Anchor* algorithm can underlie higher layer anchor-based localization algorithms that, when *Hidden Anchor* is employed, would not have to consider the physical layer threats to their operation. We also described a technique for improving the performance of the *Hidden Anchor* algorithm, when the un-trusted node employs a probabilistic location determination system, based on adding artificial noise to the network. The proposed technique significantly enhances the privacy of the network without affecting the localization accuracy at trusted nodes.

Acknowledgments

This research is supported in part by a grant from the Qatar National Research Fund (QNRF)—Grant number NPRP-1-7-7-3—and in part by the Egyptian National Telecommunication Regularity Authority (NTRA).

References

[1] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, August 2000.

[2] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '01)*, pp. 1380–1387, April 2001.

[3] A. Youssef and M. Youssef, "A taxonomy of localization schemes for wireless sensor networks," in *Proceedings of the International Conference on Wireless Networks*, 2007.

[4] P. Liu, X. Zhang, S. Tian, Z. Zhao, and P. Sun, "A novel virtual anchor node-based localization algorithm for wireless sensor networks," in *Proceedings of the 6th International Conference on Networking (ICN '07)*, April 2007.

[5] K. Langendoen and N. Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison," *Computer Networks*, vol. 43, no. 4, pp. 499–518, 2003.

[6] S. Capkun, M. Hamdi, and J.-P. Hubaux, "Gps-free positioning in mobile adhoc networks," in *Hawaii International Conference on System Sciences (HICSS-34)*, pp. 3481–3490, January 2001.

[7] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000.

[8] M. Adel, M. Ibrahim, K. Abulmakarem, M. Youssef, and M. Eltoweissy, "Hyberloc: demonstrating secure localization in hybrid sensor networks," in *Proceedings of the International Conference on Mobile Computing and Networking*, San Francisco, Calif, USA, September 2008.

[9] R. Elbadry, A. Sultan, and M. Youssef, "Hyberloc: providing physical layer location privacy in hybrid sensor networks," in *IEEE ICC '10—The Ad-Hoc, Sensor and Mesh Networking Symposium (ICC '10)*, 2010.

[10] M. Youssef and A. Agrawala, "The *Horus* location determination system," *Wireless Networks*, vol. 14, no. 3, pp. 357–374, 2008.

[11] M. A. Youssef and A. Agrawala, "Analysis of the optimal strategy for wlan location determination systems," *International Journal of Modelling and Simulation*, vol. 27, no. 1, pp. 53–59, 2007.

[12] N. Patwari and S. K. Kasera, "Robust location distinction using temporal link signatures," in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '07)*, pp. 111–122, September 2007.

[13] J. Zhang, M. H. Firooz, N. Patwari, and S. K. Kasera, "Advancing wireless link signatures for location distinction," in *Proceedings of the 14th Annual International Conference on Mobile Computing and Networking (MobiCom '08)*, pp. 26–37, September 2008.

[14] B. Greenstein, T. Kohno, D. McCoy, S. Seshan, J. Pang, and D. Wetherall, "Improving wireless privacy with an identifier-free link layer protocol," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 40–53, June 2008.

[15] I. Stojmenovic and X. Lin, "Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 12, no. 10, pp. 1023–1032, 2001.

[16] M. A. Latif, A. Sultan, and H. El Gamal, "ARQ-based secret key sharing," in *IEEE International Conference on Communications (ICC '09)*, June 2009.

[17] "Telosb mote platform," http://www.xbow.com/.

[18] J. Proakis, *Digital Communications*, McGraw-Hill Science, New York, NY, USA, 2000.