An Error Minimizing Framework for Localizing Jammers in Wireless Networks

Zhenhua Liu, Hongbo Liu, Wenyuan Xu, and Yingying Chen

Abstract—Jammers can severely disrupt the communications in wireless networks, and jammers' position information allows the defender to actively eliminate the jamming attacks. Thus, in this paper, we aim to design a framework that can localize one or multiple jammers with a high accuracy. Most of existing jammer-localization schemes utilize indirect measurements (e.g., hearing ranges) affected by jamming attacks, which makes it difficult to localize jammers accurately. Instead, we exploit a direct measurement–the strength of jamming signals (JSS). Estimating JSS is challenging as jamming signals may be embedded in other signals. As such, we devise an estimation scheme based on ambient noise floor and validate it with real-world experiments. To further reduce estimation errors, we define an evaluation feedback metric to quantify the estimation errors and formulate jammer localization as a non-linear optimization problem, whose global optimal solution is close to jammers' true positions. We explore several heuristic search algorithms for approaching the global optimal solution, and our simulation results show that our error-minimizing-based framework achieves better performance than the existing schemes. In addition, our error minimizing framework can utilize indirect measurements to obtain a better location estimation compared with prior work.

Index Terms—Jamming, Radio interference, Localization.

1 Introduction

The increasing pervasiveness of wireless technologies, combined with the limited number of unlicensed bands, will continue to make the radio environment crowded, leading to unintentional radio interference across devices with different communication technologies yet sharing the same spectrum, e.g., cordless phones, Wi-Fi network adapters, Bluetooth headsets, microwave ovens, and ZigBee-enabled appliances. Meanwhile, the emerging of software defined radios has enabled adversaries to build intentional jammers to disrupt network communication with little effort. Regardless whether it is unintentional interference or malicious jamming, one or multiple jammers/interferers may co-exist and have a detrimental impact on network performance – both can be referred as jamming. To ensure the successful deployment of pervasive wireless networks, it is crucial to localize jammers, since the locations of jammers allow a better physical arrangement of wireless devices that cause unintentional radio interference, or enable a wide range of defense strategies for combatting malicious jamming attackers.

In this work, we focus on localizing one or multiple stationary jammers. Our goal is to extensively improve the accuracy of jammer localization. Current jammer-localization approaches mostly rely on parameters derived from the affected network topology, such as packet delivery ratios [1], neighbor lists [2], and nodes' hearing ranges [3]. The use of these indirect measurements derived from jamming effects makes it difficult to accurately localize jammers' positions. Furthermore, they mainly localize one jammer and cannot cope with the cases that multiple jammers are located close to each other and their jamming effects overlap.

To address the limitation caused by indirect measurements of the jamming effect, we propose to use the direct measurement of the strength of jamming signal (JSS). Localizing jammers using JSS is appealing yet challenging. First, jamming signals are embedded in the regular network traffic. The commonly used received signal strength (RSS) measurement associated with a packet does not correspond to JSS. To overcome this challenge, we devise a scheme that can effectively estimate the JSS utilizing the measurement of the ambient noise floor, which is readily available from many commodity devices (e.g., MicaZ motes). Our experiments using MicaZ motes with multiple sender-receiver pairs confirm the feasibility of estimating JSS under various network traffic conditions.

With the ability to estimate JSS, it appears that one may leverage existing RSS-based localization algorithms designed for regular wireless devices to localize jammers. However, we consider jamming localization different for the following reasons. (1) Most jammers start to disturb network communication af-

- *Z. Liu and W. Xu are with the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29205.*
  E-mail: {liuz,wyxu}@cse.sc.edu
- *H. Liu and Y. Chen are with the Department of Electrical and Computer Science, Stevens Institute of Technology, Castle Point on Hudson, Hoboken, NJ 07030.*
  E-mail: {hliu3,yingying.chen}@stevens.edu
- *W. Xu is the corresponding author.*

- *The work was supported in part by National Science Foundation Grants CNS-0845671, GEO-1124657, CNS-0954020 and CCF-1018270.*

Digital Object Indentifier 10.1109/TPDS.2013.33

ter network deployment, which makes it infeasible to obtain a site survey of radio fingerprints around jammers beforehand, a commonly used method for localization in an indoor environment. (2) No detailed prior knowledge about the jammers' transmission power is available. (3) Multiple jammers with overlapped jamming areas may collude and disturb network communication together, while attempting to hide their true locations.

To overcome these challenges and increase the localization accuracy, we formulate the jammer localization problem as a non-linear optimization problem and define an evaluation metric as its objective function. The value of evaluation metric reflects how close the estimated jammers' locations are to their true locations, and thus we can search for the best estimations that minimize the evaluation metric. Because traditional gradient search methods may converge to a local minimum and may not necessarily yield the global minimum, we adopt several algorithms that involve stochastic processes to approach the global optimum. In particular, we examined three algorithms: a *genetic algorithm*, a *generalized pattern search algorithm*, and a *simulated annealing algorithm*. Our extensive simulation results show that our localization error minimizing framework not only can improve the estimation accuracy of localizing one jammer compared to prior work [3], but also can estimate the positions of multiple jammers simultaneously, making it especially useful for identifying unintentional radio interference caused by multiple wireless devices or a few malicious and collaborative jammers. We summarize our main contributions as follows:

- Estimating JSS is challenging because the jamming signals are embedded in the regular signals. To the best of our knowledge, our work is the first that directly utilizes the JSS to localize jammers. Our results using direct measurements (e.g., JSS) exhibit significant improvement compared with those using indirect measurements (e.g., hearing ranges).
- We exploited path-loss and shadowing phenomena in radio propagation and defined an evaluation metric that can quantify the accuracy of the estimated locations. Leveraging such an evaluation metric, we formulated the jammer localization problem as an error minimizing framework and studied several heuristic searching algorithms for finding the best solution.
- Our error-minimizing-based algorithms can localize multiple jammers simultaneously, even if their jamming areas overlap. Localizing in such a scenario is known to be challenging [4], [5].

We organize the remainder of the paper as follows. We introduce our threat model in Section 2. In Section 3, we overview our error-minimizing framework for localizing jammers and formulate the jammer localization problem as a non-linear optimization problem utilizing JSS. Then, we address the challenge of estimating the JSS and present our real-systems experiment validation in Section 4 and Section 2 of the supplementary file. In Section 5, we analyze several searching algorithms for solving the optimization problem. Next, we present the performance study of our error-minimizing-based localization approaches in Section 6 and in Section 3 of the supplementary file. Finally, we conclude in Section 7. The related work is discussed in Section 1 of the supplementary file.

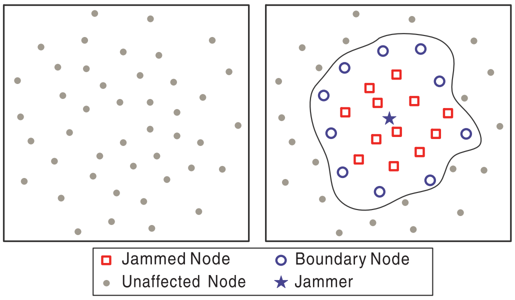


Fig. 1. Illustration of the network nodes classification due to jamming: [Left] prior to jamming and [right] after jamming.

2 Threat Model

There are many different attack strategies that jammers can perform in order to disrupt wireless communications. It is impractical to cover all the possible jamming attack models that might exist. Thus, we mainly focus on one common type of jammer – constant jammers. Constant jammers continually emit radio signals, regardless of whether the channel is idle or not. Such jammers can be unintentional radio interferers that are always active or malicious jammers that keep disturbing network communication.

Besides, we assume each jammer is equipped with an omnidirectional antenna. Thus, every jammer has a similar jamming range in all directions. Identification of jammers' positions will be performed after the jamming attack is detected, and we assume the network is able to identify jamming attacks and obtain the number of jammers, leveraging the existing jamming detection approaches [6], [7].

We classify the network nodes based on the level of disturbance caused by jammers, and identify the nodes that can participate in jammer localization, e.g., the ones that can measure and report the JSS. Essentially, the communication range changes caused by jamming are reflected by the changes of neighbors at the network topology level. Thus, the network nodes could be classified based on the changes of neighbors caused by jamming. We define that node B is a neighbor of node A if A can communicate with B prior to jamming. The network nodes can be classified into three categories according to the impact of jamming: *unaffected node*, *jammed node*, and *boundary node*:

Algorithm 1 Jammer Localization Framework.

```
p = MeasureJSS()
z = Initial positions  初始位置
while Terminating Condition True do  满足终止条件
    $e_z$ =EvaluateMetric($\mathbf{z}, \mathbf{p}$)  量化估计准确性
    if NotSatisfy($e_z$) then  不满足（ez）
        z = SearchForBetter()  搜索最佳估计
    end if
end while
```

- **Unaffected node.** A node is unaffected if it can communicate with all of its neighbors. This type of node is barely affected by jamming and may not yield accurate JSS measurements.
- **Jammed node.** A node is jammed if it cannot communicate with any of the unaffected nodes. We note that this type of node can measure JSS, but cannot always report their measurements.
- **Boundary node.** A boundary node can communicate with part of its neighbors but not from all of its neighbors. Boundary nodes can not only measure the JSS, but also report their measurements to a designated node for jamming localization.

Figure 1 illustrates an example of network topology changes caused by a jammer. Prior to jamming, all the nodes could communicate with their neighbors, shown as grey dots. Once the jammer became active (shown as a star), affected nodes lost their neighbors partially or completely. In the example depicted in Figure 1, the nodes marked as red squares lost all of their neighbors and became jammed nodes. The nodes depicted in blue circles are boundary nodes, since they lost part of their neighbors but still maintained communication capability to a few neighbors. Finally, the rest of the nodes that remained in grey dots are unaffected nodes, and they can still communicate with all their neighbors. Note that jammed nodes are usually those nodes located closest to the jammer, whereas the boundary nodes reside in between jammed nodes and unaffected nodes.

In this work, the boundary nodes play an important role, and our jammer localization algorithms rely on them for sampling and collecting JSS for jammer localization.

3 Localization Formulation

Essentially, our jammer localization approach works as follows. Given a set of JSS, for every estimated location, we are able to provide a quantitative evaluation feedback indicating the distance between the estimated locations of jammers and their true locations. For example, a small value of evaluation feedback indicates that estimated locations are close to the true ones, and vice versa. Although unable to adjust the estimation directly, it is possible, from a few candidate locations, to select the ones that are closest to the true locations with high probability, making searching for the best estimate feasible. Leveraging this idea, our jammer localization approach comprises two steps: (a) *JSS Collection.* Each boundary node locally obtains JSS. (b) *Best-Estimation Searching.* Based on the collected JSS, a designated node will obtain a rough estimation of the jammers' positions. Then, it refines the estimation by searching for positions that minimize the evaluation feedback metric. The details are described in Algorithm 1. The search-based jammer localization approaches have a few challenging subtasks:

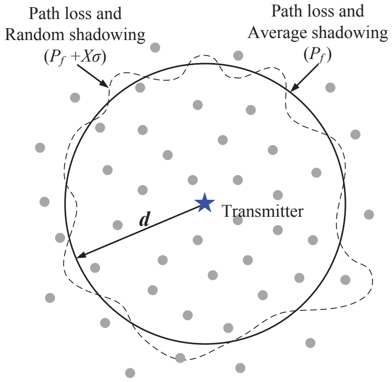


Fig. 2. The contour of RSS subject to path loss is a circle centered at the transmitter, and the contour of RSS attenuated 衰减 by both path loss and shadowing is an irregular loop 不规则环 fluctuating 波动 around the path-loss circle.

1) `EvaluateMetric()` has to define an appropriate metric to quantify the accuracy of estimated jammers' locations.
2) `MeasureJSS()` has to obtain JSS even if it may be embedded in regular transmission.
3) `SearchForBetter()` has to efficiently search for the best estimation.

In this section, we focus on formulating the evaluation feedback metric using collected JSS measurements. In particular, we model the jammer localization as an optimization problem. We delay the discussion of JSS measurement scheme `MeasureJSS()` to Section 4 and searching algorithms `SearchForBetter()` to Section 5.

3.1 Radio Propagation Basics

In wireless communication, the received signal strength attenuates with the increase of distance between the sender and receiver due to path loss and shadowing, as well as constructive and destructive addition of multipath signal components [8], [9]. Path loss can be considered as the *average* attenuation while shadowing is the *random* attenuation caused by obstacles through absorption, reflections, scattering, and diffraction [8], [9]. Figure 2 illustrates contours of a received signal strength and the relationship between shadowing and path loss. The attenuation caused by shadowing at any single location, d meters from the transmitter, may exhibit variation; the average

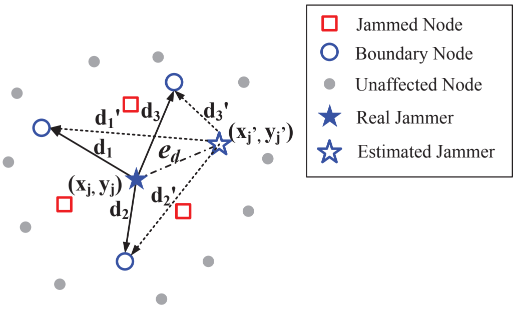


Fig. 3. Illustration of jammer localization basis. When the estimated jammer location is e_d meters from the true location, the estimated random attenuation is biased and the corresponding standard deviation is larger than the real one.

相应的标准偏差

attenuation and average signal strength on the circle centered at the transmitter are roughly the same [8], [9]. This observation serves as the fundamental basis of our error minimizing framework.

To illustrate our jammer localization approach, we use the widely-used log-normal shadowing model [8], [9], which captures the essential of both path loss and shadowing. Let P_f be the received signal strength subject to path loss attenuation only, and let the power of a transmitted signal be P_t. The received signal power in dBm at a distance of d can be modeled as the sum of P_f and a variance (denoted by X_σ) caused by shadowing and other random attenuation,

方差

衰减

$$P_r = P_f + X_\sigma \quad (1)$$

$$P_f = P_t + K - 10\eta \log_{10}(d), \quad (2)$$

where X_σ is a Gaussian zero-mean random variable with standard deviation σ, K is a unitless constant which depends on the antenna characteristics and the average channel attenuation, and η is the Path Loss Exponent (PLE). In a free space, η is 2 and X_σ is always 0.

3.2 Localization Evaluation Metric

In this section, we discuss the definition of the evaluation metric e_z, and we show the property of e_z as well as its calculation. For the ease of reading, we summarize the frequently used notations in Table 1.

3.2.1 The property of e_z

The definition of e_z should have the following property: The larger the estimation errors of jammers' locations are, the larger e_z is. We define e_z as the estimated standard deviation of X_σ derived from the estimated jammers' locations. Considering the one-jammer case, when the estimated jammer's location equals the true value, e_z is the real standard deviation of X_σ, which is relatively small. When there is an estimation error (the estimated location is e_d distance away from the true location), e_z will be biased and will be larger than the real standard deviation of X_σ. The level of bias is affected by e_d: the larger e_d is, the bigger the estimated standard deviation of X_σ will likely be. The detailed relationship between e_z and e_d will be discussed in Section 5.1.

Here, we illustrate the property of e_z using the example depicted in Figure 3, where 3 boundary nodes are $\{d_1, d_2, d_3\}$ distance away from the jammer J. Let $\{X_{\sigma_1}, X_{\sigma_2}, X_{\sigma_3}\}$ be the true shadowing attenuation between the boundary nodes and J, then e_z is the true standard deviation of $\{X_{\sigma_1}, X_{\sigma_2}, X_{\sigma_3}\}$. If the estimated location of J is (x'_J, y'_J), the estimated distances between the three boundary nodes to J are $\{d'_1, d'_2, d'_3\}$. In this example, $d'_1 > d_1$, $d'_2 > d_2$, and $d'_3 < d_3$. When $d'_1 > d_1$, the estimated JSS contributed by pass loss only (P'_f) is smaller than the real one. Given the measured JSS, the estimated shadowing attenuation (X'_{σ_1}) has to be larger than the real one (X_{σ_1}) to make up for the under-estimated P'_f. Similarly, $X'_{\sigma_2} > X_{\sigma_2}$ and $X'_{\sigma_3} < X_{\sigma_3}$. Thus, the estimated shadowing attenuation $\{X'_{\sigma_1}, X'_{\sigma_2}, X'_{\sigma_3}\}$ exhibits a larger variance than the real one, and the estimated standard deviation (e'_z) corresponding to (x'_J, y'_J) is larger than the true standard deviation.

描述

We note that the relationship between e_z and e_d is independent to the distribution of X_σ. Thus, in cases where the log-normal shadowing model does not match with the real radio propagation, e_z can still provide quantitative feedback of e_d.

定量的

3.2.2 Calculation

Single Jammer. Assume a jammer J located at (x_J, y_J) starts to transmit at the power level of P_J, and m nodes located at $\{(x_i, y_i)\}_{i\in[1,m]}$ become boundary nodes. To calculate e_z, each boundary node will first measure JSS locally (the details will be discussed in Section 4), and we denote the JSS *measured* at boundary node i as P_{r_i}. Let the current estimation of the jammer J's location and the transmission power be

局部

$$\hat{\mathbf{z}} = [\hat{x}_J, \hat{y}_J, \hat{P}_J + \hat{K}].$$

	Description of variables
P_{r_i}	JSS at a boundary node i
P_{f_i}	Power component attenuated by path loss only
P_{J_j}	Transmission power of a jammer j
K	Unitless constant which depends on the antenna characteristics and the average channel attenuation
X_{σ_i}	Random attenuation at a boundary node i
$\mathbf{z}$	Unknown variable vector of jammers
$\mathbf{p}$	Vector of JSS at m boundary nodes
$\mathbf{s}$	Vector of n ANF measurements at a boundary node
(x_{J_j}, y_{J_j})	Coordinates of a jammer j
(x_i, y_i)	Coordinates of of a boundary node i
d_{ji}	Distance between a jammer j and a boundary node i
σ	Standard deviation of random attenuation
e_z	Evaluation feedback metric 度量标准，米制的
e_d	Localization error (distance between the estimated location and the true location)

TABLE 1
Frequently used notations.

Algorithm 2 Evaluation feedback metric calculation.

```
procedure EvaluateMetric($\hat{\mathbf{z}}, \mathbf{p}$)
    for all $i \in [1, m]$ do
        $\hat{X}_{\sigma_i} = P_{r_i} - P_{f_i}(\hat{\mathbf{z}})$
    end for
    $e_z = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{X}_{\sigma_i} - \hat{\bar{X}}_\sigma)^2}$
end procedure
```

Given $\hat{\mathbf{z}}$, we can estimate P_{f_i}, the JSS subject to path loss only at boundary node i as

$$
\begin{aligned}
P_{f_i}(\hat{d}_i) &= \hat{P}_J + \hat{K} - 10\eta\log_{10}(\hat{d}_i) \\
\hat{d}_i(\hat{\mathbf{z}}) &= \sqrt{(\hat{x}_J - x_i)^2 + (\hat{y}_J - y_i)^2}
\end{aligned}
\tag{3}
$$

The random attenuation (shadowing) between the jammer J and boundary node i can be estimated as

$$
\hat{X}_{\sigma_i} = P_{r_i} - P_{f_i}(\hat{d}_i). \tag{4}
$$

评价反馈指标

The evaluation feedback metric for the estimation $\hat{\mathbf{z}}$ is the standard deviation of estimated $\{\hat{X}_{\sigma_i}\}_{i\in[1,m]}$,

标准偏差估计

$$
e_z(\hat{\mathbf{z}}, \mathbf{p}) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{X}_{\sigma_i} - \hat{\bar{X}}_\sigma)^2}, \tag{5}
$$

where $\hat{\bar{X}}_\sigma$ is the mean of $\hat{X}_{\sigma_i}$. One of the biggest advantages of this definition is that by subtracting $\bar{X}_\sigma$, e_z is only affected by $(\hat{x}_J, \hat{y}_J)$ and is independent of the estimated jamming power $\hat{P}_J + \hat{K}$.

Multiple Jammers. Similar to single jammer, we assume n jammers located at $\{(x_{J_i}, y_{J_i})\}_{i\in[1,n]}$ start to transmit at the power level of $\{P_{J_i}\}_{i\in[1,n]}$ separately at the same time, and m nodes located at $\{(x_i, y_i)\}_{i\in[1,m]}$ become boundary nodes. To calculate e_z, each boundary node measures JSS locally and we denote the JSS *measured* at boundary node i as P_{r_i} which is a combined JSS from multiple jammers. We can include all the variables to be estimated, i.e., current estimation of the n jammers' locations and the transmission powers, in a form of matrix written as

$$
\mathbf{z} = \begin{pmatrix}
\hat{x}_{J_1} & \hat{y}_{J_1} & \hat{P}_{J_1} + \hat{K}_1 \\
\hat{x}_{J_2} & \hat{y}_{J_2} & \hat{P}_{J_2} + \hat{K}_2 \\
\vdots & \vdots & \vdots \\
\hat{x}_{J_n} & \hat{y}_{J_n} & \hat{P}_{J_n} + \hat{K}_n
\end{pmatrix} \tag{6}
$$

In the case of multiple jammers, p_{f_i} is the combined JSS from n jammers subject to path loss at a boundary node and can be calculated as

$$
\begin{aligned}
P_{f_i}(\hat{\mathbf{z}}) &= 10\log_{10}\Big(\sum_{j=1}^{n}\frac{10^{\frac{\hat{P}_{J_j} + \hat{K}_j}{10}}}{\hat{d}_{ji}^{\eta}}\Big) \\
\hat{d}_{ji} &= \sqrt{(\hat{x}_{J_j} - x_i)^2 + (\hat{y}_{J_j} - y_i)^2}
\end{aligned}
\tag{7}
$$

where $\hat{d}_{ji}$ is the estimated distance between jammer j and boundary node i. Note that $\hat{P}_{J_j}$, $\hat{K}$ and P_{f_i} are all in dBm.

Algorithm 3 Acquiring the Ambient Noise Floor (ANF). ANF approximates the strength of jamming signals.

```
procedure MeasureJSS
    $\mathbf{s} = \{s_1, s_2, ..., s_n\} = \text{MeasureRSS}()$
    if var($\mathbf{s}$) < varianceThresh then
        $\mathbf{s}_a = \mathbf{s}$                    方差翻滚？？？Σ(⊙▽⊙"a
    else
        $JssThresh = \min(\mathbf{s}) + \alpha[\max(\mathbf{s}) - \min(\mathbf{s})]$ ▷ $\alpha \in [0, 1]$
        $\mathbf{s}_a = \{s_i | s_i < JssThresh,\ s_i \in \mathbf{s}\}$
    end if
    return mean($\mathbf{s}_a$)
end procedure   程序
```

Then, the random attenuation between multiple jammers and the boundary node i can be estimated as

$$
X_{\sigma_i} = P_{r_i} - P_{f_i}(\hat{\mathbf{z}}), \tag{8}
$$

Thus, the evaluation feedback metric of $\hat{\mathbf{z}}$ is

$$
e_{\hat{z}}(\mathbf{z}, \mathbf{p}) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{X}_{\sigma_i} - \hat{\bar{X}}_\sigma)^2}. \tag{9}
$$

where $\hat{\bar{X}}_\sigma$ is the mean of $\hat{X}_{\sigma_i}$.

3.3 Problem Formulation

Given the definition of the feedback metric (e_z), we generalize jammer localization problem as one optimization problem,

Problem 1:

$$
\begin{aligned}
\underset{\mathbf{z}}{\text{minimize}} \quad & e_z(\mathbf{z}, \mathbf{p}) \\
\text{subject to} \quad & \mathbf{p} = \{P_{r_1}, \ldots, P_{r_m}\};
\end{aligned}
$$

where $\mathbf{z}$ are the unknown variable matrix of the jammer(s), e.g., $\mathbf{z}$ is defined in Eq. (6), and $\{P_{r_i}\}_{i\in[1,m]}$ are the JSS measured at the boundary nodes $\{1, \ldots, m\}$. As we will show in Section 5.1, the estimated location(s) of the jammer(s) at which e_z is minimized, matches the true location(s) of jammer(s) with small estimation error(s).

4 Measuring Jamming Signals

Received signal strength (RSS) is one of the most widely used measurements in localization. For instance, a WiFi device can estimate its most likely location by matching the measured RSS vector of a set of WiFi APs with pre-trained RF fingerprinting maps [10] or with predicted RSS maps constructed based on RF propagation models [11]. However, obtaining signal strength of jammers (JSS) is a challenging task mainly because jamming signals are embedded in signals transmitted by regular wireless devices. The situation is complicated because multiple wireless devices are likely to send packets at the same time, as jamming disturbs the regular operation of carrier sensing multiple access (CSMA). For the rest of this paper, we refer the regular nodes' concurrent packet transmissions that could not be decoded as a collision.

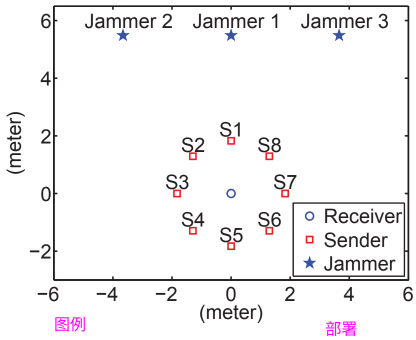


Fig. 4. An Illustration of Network Deployment.

While it is difficult, if ever possible, to extract signal components contributed by jammers or collision sources, we discover that it is feasible to derive the JSS based on periodic ambient noise measurement. In the following subsections, we first present basics of ambient noise with regard to jamming signals, and then introduce our scheme to estimate the JSS. Finally, we validate our estimation schemes via real-world experiments.

4.1 Basics of Ambient Noise Floor

In theory, *ambient noise* is the sum of all unwanted signals that are always *present*, and the ambient noise floor (ANF) is the measurement of the ambient noise. In the presence of constant jammers, the ambient noise includes thermal noise, atmospheric noise, and jamming signals. Thus, it is

$$P_N = P_J + P_W, \tag{10}$$

where P_J is the JSS, and P_W is the white noise comprising thermal noise, atmospheric noise, etc. Realizing that at each boundary node P_W is relatively small compared to P_J, the ambient noise floor can be roughly considered as JSS. Thus, estimating JSS is equivalent to deriving the ambient noise floor (ANF) at each boundary node. In this work, we consider the type of wireless devices that are able to sample ambient noise regardless of whether the communication channel is idle or busy, e.g., MicaZ sensor platforms; and derive the ANF based on ambient noise measurements.

A naive approach of estimating the ANF could be sampling ambient noise when the wireless radio is idle (i.e., neither receiving nor transmitting packets). Such a method may not work in all network scenarios, since it may result in an overestimated ANF. For example, in a highly congested network, collision is likely to occur, and the collided signals may be treated as part of the ANF at the receiver, resulting in an inflated ANF. This is exactly the situation we want to avoid.

4.2 Estimating Strength of Jamming Signals

To derive the JSS, our scheme involves sampling ambient noise values regardless of whether the channel is idle or busy. In particular, each node will sample n measurements of ambient noise at a constant rate, and denote them as $\mathbf{s} = [s_1, s_2, \ldots, s_n]$. The measurement set $\mathbf{s}$ can be divided into two subsets ($\mathbf{s} = \mathbf{s}_a \cup \mathbf{s}_c$).

1) $\mathbf{s}_a = \{s_i | s_i = P_J\}$, the ambient noise floor set that contains the ambient noise measurements when only jammers are active, and
2) $\mathbf{s}_c = \{s_i | s_i = P_J + P_C\}$, the combined ambient noise set that contains ambient noise measurements when both jamming signals (P_J) and signals from one or more senders (P_C) are present.

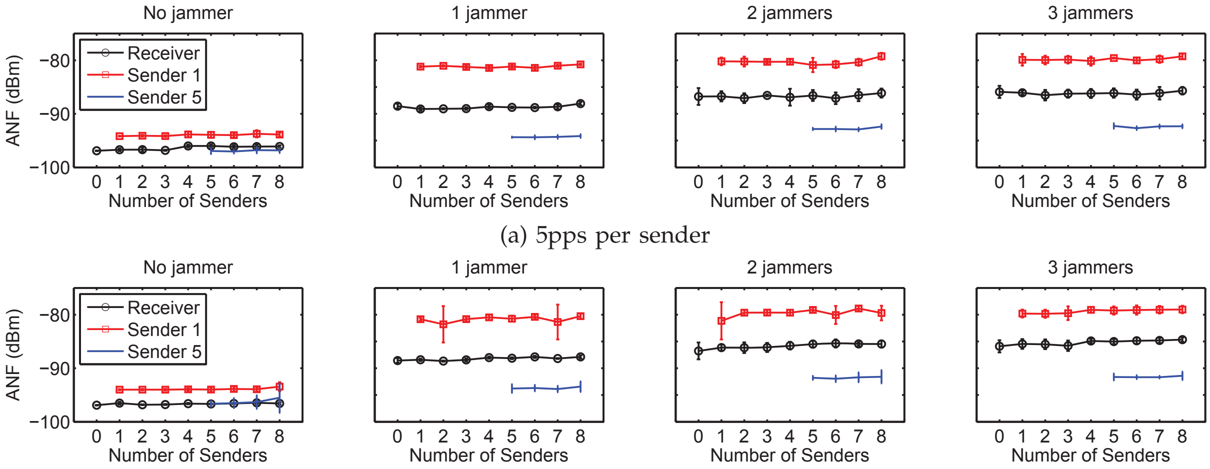
Calculating JSS is equivalent to obtaining the average of ANFs, i.e., mean($\mathbf{s}_a$). In most cases, $\mathbf{s}_c \neq \emptyset$ and $\mathbf{s}_a \subset \mathbf{s}$. In a special case where no sender has ever transmitted packets throughout the process of obtaining n measurements, $\mathbf{s}_c = \emptyset$ and $\mathbf{s}_a = \mathbf{s}$. The algorithm for calculating the ANF should be able to cope with both cases. As such, we designed an algorithm (referred as Algorithm 3) as follows: A regular node will take n measurements of the ambient noise measurements. It will consider the ANF as the average of all measurements if no sender has transmitted during the period of measuring; otherwise, the ANF is the average of $\mathbf{s}_a$, which can be obtained by filtering out $\mathbf{s}_c$ from $\mathbf{s}$. The intuition of differentiating those two cases is that if only jamming signals are present, then the variance of n measurements will be small; otherwise, the ambient noise measurements will vary as different senders happen to transmit.

The correctness of the algorithm is supported by the fact that $\mathbf{s}_a$ is not likely to be empty due to carrier sensing, and the JSS approximately equals to the average of $\mathbf{s}_a$. The key question is how to obtain $\mathbf{s}_a$. To do so, we set the upper bound (i.e., `JssThresh`) of $\mathbf{s}_c$ in Algorithm 3 as α percentage of the amplitude span of ambient noise measurements. We validate the feasibility of obtaining $\mathbf{s}_a$ using a filtering bound in the next experimental subsection.

4.3 Experiment Validation

To verify our algorithm that derives JSS, we conducted experiments involving one receiver and eight senders, which were implemented on MicaZ nodes. We deployed them on an outdoor playground as illustrated in Figure 4 and conducted a set of experiments to evaluate the performance of Algorithm 3.

To study how well Algorithm 3 estimates JSS with a various number of colliding sources and the network traffic, we increased the number of senders sequentially from 0 to 8, and summarized the estimated ANFs in all four scenarios (in Figure 5): no jammer, 1 jammer, 2 jammers, and 3 jammers. In general, the increase of the senders does not have much influence on the correctiveness of ambient noise floor estimation in all cases. Sender 1 transmitting at 20 packets per second did show a higher variance of estimation. That is caused by its low ambient noise sampling rate.



(a) 5pps per sender

(b) 20pps per sender

Fig. 5. An illustration of the estimated ambient noise floor with an increasing number of senders.

The detailed evaluation results are presented in the supplementary file.

5 Finding the Best Estimation

The jammer localization problem can be modeled as a non-linear optimization problem (defined in Problem 1), and finding a good estimation of jammers' locations is equivalent to seeking the solution that minimizes the evaluation feedback metric e_z. In this section, we illustrate the relationship between e_z and e_d (the distance between the true jammer's location and the estimated one), which shows that greedy algorithms that search for successively better solutions are unable to find the global optimal value. Instead, we use several heuristic search algorithms that rely on guided random processes to approach the global optimum without converging to a local minimum.

5.1 Error Analysis

The evaluation feedback metric e_z is a nonlinear function of the estimated location of jammers and the measured JSS values. To understand e_z, we performed a numerical simulation and derived the numerical values of e_z on a grid of points in a 300-by-300 meter square, within which 200 nodes were randomly deployed with a transmission power of -45 dBm. Additionally, the jammer transmitted at a power level of -38 dBm, and affected about 20 boundary nodes. To examine the impact of an inaccurate estimation of P_J, we set the estimated jamming power $\hat{P}_j$ to -25 dBm, much larger than the true jamming power. To get enough resolution, we set the grid step to 0.5m and in total calculated $360,000$ data points for each network topology. We chose two representative network topologies and depicted their error contours in Figure 6, from which we drew the following observations:

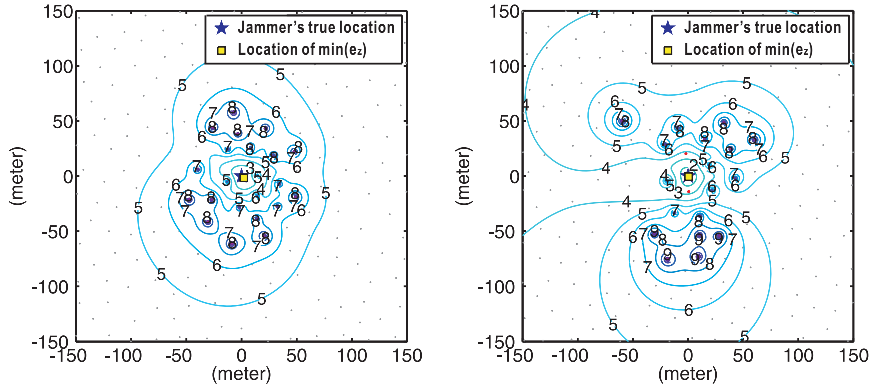
1) Despite the inaccurate estimation of jamming power, the global minimum of e_z is close to the true location of the jammer, suggesting that the estimated location that minimizes e_z is a relatively accurate estimation of a jammer's position, even if the estimated jamming power is inaccurate.
2) At each boundary node (marked by blue circles in Figure 6), e_z reaches its local maximum. This is because that at boundary node i, $\hat{d}_i$ is close to 0, which makes $\log(\hat{d}_i)$ approaches infinity and causes $\hat{X}_{\sigma_i}$ to be an outlier. As a result, the estimated standard deviation (e_z) of $\hat{X}_\sigma$ is large.
3) Interestingly, e_z is not strictly proportional to e_d. Although when the estimated location is in the close vicinity of the true value, the smaller e_d is, the smaller e_z becomes. When the estimated location increases to more than 100m, the larger e_d, the smaller e_z. This is because when the estimated jammers' locations are further away from the boundary nodes, their distances to all the boundary nodes become larger than the real ones. In turn, all the estimated random attenuation $\{\hat{X}_{\sigma_i}\}$ at each boundary node are consistently over-estimated and their standard deviation becomes smaller than the ones when part of $\{\hat{X}_{\sigma_i}\}$ are overestimated and part of $\{\hat{X}_{\sigma_i}\}$ are underestimated.

The combination of the 2nd and 3rd observations makes greedy algorithms impractical. For instance, the gradient descent, which moves towards the steepest decreasing direction of e_z, will not be able to climb the 'hill' of the global maximum at a boundary node, nor will it be guaranteed to search towards the global minimum solution. Thus, we examine several heuristic searching algorithms to find the global minimum. In this work, these algorithms take the measured JSS as inputs; however, they are not limited to it.

5.2 Algorithm Description.

5.2.1 A Genetic Algorithm

A genetic algorithms (GA) [12] searches for the global optimum by mimicking the process of natural selec-



(a) $\min(e_z) = 1.9\texttt{dB}$ (b) $\min(e_z) = 1.5\texttt{dB}$

Fig. 6. An illustration of error contours for e_z in a network of 200 nodes. e_z reaches its minimum at a location close to the jammer's true position. 轮廓

tion in biological evolution. A GA iteratively generates a set of solutions known as a population. At each iteration, a GA selects a subset of solutions to form a new population based on their "fitness" and also randomly generates a few new solutions. As a result, the "fitter" solutions will be inherited. At the same time, new solutions will be introduced to the population, which may turn out to be "fitter" than ever. As a result, over successive generations, a GA is likely to escape from local optima and "evolves" towards an optimal solution.

In the application of searching for the best estimation of jammers' locations, each individual (i.e., a solution) has a chromosome of $3n$ genes, comprising n jammers' coordinates and jamming power levels. We defined the fitness of each individual as e_z. The smaller e_z is, the better.

5.2.2 *A Generalized Pattern Search*

A generalized pattern search algorithm (GPS) [13] works similarly to the gradient descent algorithm. However, at each iteration, instead of making a step towards the steepest gradient, a GPS checks a set of solutions (called a mesh) around the current solution, looking for the one whose corresponding function value is smaller than the one at the current solution. If a GPS finds such a solution, the new solution becomes the current solution at the next step of the algorithm. By searching for a mesh of solutions, a GPS is likely to find a sequence of solutions that approach an optimal one without converging to a local minimum.

5.2.3 *A Simulated Annealing Search*

A simulated annealing algorithm (SA) [14] searches for the optimal solutions by modeling the physical process of heating a material and then controlled lowering the temperature to decrease defects. At each iteration, the simulated annealing algorithm compares the current solution with a randomly-generated new solution. The new solution is selected according to a probability distribution with a scale proportional to the temperature, and it will replace the current solution according to a probability governed by both the new object function value and temperature. By accepting 'worse' solutions occasionally, the algorithm avoids being trapped in local minima, and is able to explore solutions globally. As the temperature decreases, the annealing algorithm reduces its search scale so that it converges to a global minimum with high probability.

5.3 Reducing Searching Space

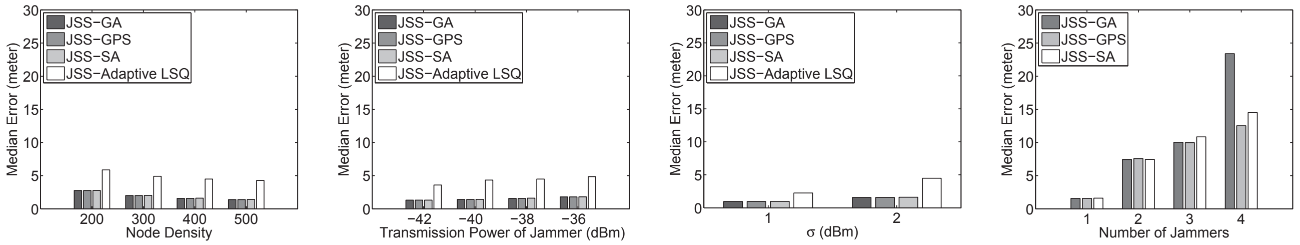
Aforementioned algorithms are all search-based, and their efficiency depends on the searching space. To improve the search efficiency, we first limited the range of each variable. For example, the coordinates of jammers (x_J, y_J) should reside inside the jammed area, which can be estimated by examining the positions of both jammed nodes and boundary nodes[1]. We also restricted a jammer's transmission power to the range of $[-50, 0]$ $\texttt{dBm}$. Note that this restriction is less important in terms of minimizing localization accuracy, since our objective function e_z does not depend on it. For the initial estimated position of jammers, we set the initial value to an estimation obtained by Adaptive LSQ methods proposed by Liu *et al.* [3] for one-jammer cases; and we randomly selected jammers' locations somewhere inside the jammed area for multi-jammer cases.

6 Performance Validation

In this section, we evaluated the performance of our jammer localization approaches that utilize the error minimizing framework. Detailed evaluations are presented in the supplementary file.

We studied three heuristic search algorithms for finding the best estimation of jammers' position: a genetic algorithm (GA), a generalized pattern search (GPS) algorithm, and a simulated annealing (SA)

[1] Even if in rare cases that the jammer is outside the network deployed area, the layout of jammed nodes and boundary nodes (e.g. at the boundary of the network) will indicate the jamming regions.



(a) Node density (b) Jamming power (c) Propagation irregularity (d) Number of jammers

Fig. 7. The impact of various factors on median localization errors.

algorithm; and compared those three algorithms to the prior work by Liu *et al.* [3], i.e. the Adaptive LSQ algorithm. We developed a simulator in Matlab. We simulated the underlying radio propagation according to the log-normal shadowing model and used GA, GPS and SA functions provided in the Global Optimization toolbox in Matlab. To make a fair comparison, we set the parameters of the shadowing model to the same values as the ones used in the prior work by Liu *et al.* [3] (e.g., the path loss exponent $\eta = 2.11$).

We compared the algorithms in a variety of network configurations, including node densities, jammer's transmission power, the standard deviation of random attenuation, and the number of jammers. In addition, we examined our error minimizing framework when indirect measurements (i.e., hearing ranges [15]) are used. A hearing range is the area within which a node can successfully receive and decode packets, and it is affected by the jammers' locations and transmission power.

Impact of Node Density. We first investigated the impact of node density on the accuracy of localizing one jammer by deploying $\{200, 300, 400, 500\}$ nodes in our 300-by-300 meter network and fixing the jammer at the center $(0, 0)$.

We depicted the median localization errors for our heuristic searching algorithms and Adaptive LSQ algorithm in Figure 7(a). Firstly, we observed that GA, GPS and SA all achieved almost the same accuracy and consistently outperformed Adaptive LSQ algorithm in all the node densities and deployment setups. Secondly, as the network node density increases, the accuracy of all algorithms improves.

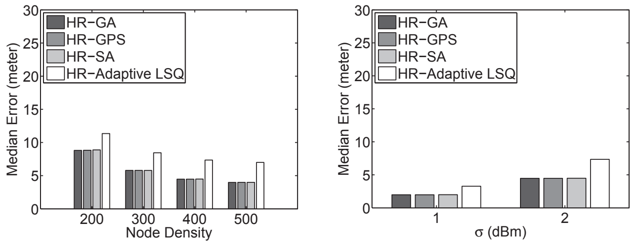
Impact of the Jamming Power. To study the effects of various transmission power of jammers to the localization performance, we examined networks with 400 nodes in a 300-by-300 meter field and set the jammer's transmission power to $\{-42, -40, -38, -36\}$ dBm, respectively. The results are plotted in Figure 7(b), which shows that GA, GPS and SA outperformed the Adaptive LSQ algorithm for all the jamming power levels.

Impact of Propagation Irregularity. To examine the impact of propagation irregularity on localization errors, we used standard deviation of random attenuation σ to quantify the propagation irregularity and compared algorithm performance in 400-node networks when the standard deviation σ was set to 1.0 and 2.0. From Figure 7(c), we observed that a larger standard deviation σ (i.e., 2.0) increases the median localization errors for all algorithms in both deployments. However, our error-minimizing searching algorithms can outperform the Adaptive LSQ algorithms more in an environment with a higher degree of irregularity.

Impact of the Number of Jammers. Then, we examined the impact of the number of jammers on the localization errors. We studied the cases when $\{1, 2, 3, 4\}$ jammers were emitting signals at -38 dBm, and the network was comprised of 1600 nodes in a 600-by-600 meter square, whose density is equivalent to 400 nodes in a 300-by-300 meter field. For multiple jammer cases, we placed the jammers in such a way that all of them had overlapped jamming regions, since such an arrangement is difficult to localize. In particular, we placed the jammers symmetrically to the center of the network $(0, 0)$ with pairs of jammers 60 meters away. The coordinates are as follows: (1) 2 jammers: $(-30, 0)$ and $(30, 0)$; (2) 3 jammers: $(0, 35)$, $(-30, -17)$ and $(30, -17)$; (3) 4 jammers: $(-30, 30)$, $(-30, -30)$, $(30, -30)$ and $(30, 30)$.

Figure 7(d) summarizes the median localization errors when one or multiple jammers were active. We observed that as the number of jammers increases, the performance of all algorithms decreases. Among three searching algorithms, the estimation errors of GPS didn't increase as much as the other two, since in each iteration, GPS involves searching for a better solution using a fixed pattern (e.g. adding a fixed dispersion metric to the current solution to generate new ones), while the other two randomly generate new solutions.

Impact of Using Indirect Measurements. Finally, we studied the performance of our error-minimizing framework using indirect measurements, e.g, hearing ranges. Since a hearing range is affected by JSS, we are able to calculate e_z according to the measured hearing ranges and find the estimated jammers' locations that minimize e_z. We depicted the performance of algorithms utilizing hearing ranges in various node densities in Figure 8(a) and in two setups of standard



(a) Node density (b) Propagation irregularity

Fig. 8. Performance of all algorithms that use indirect measurements (e.g., hearing ranges).

deviation in Figure 8(b). All the results show that GA, GPS and SA all constantly outperform the Adaptive LSQ algorithm in all setups. Similar to the results using JSS, the accuracy of all algorithms improves as the node density increases and as the standard deviation reduces. The results also show that using indirect measurements can not achieve as good localization accuracy as using direct measurements (e.g, JSS), due to that indirect measurements usually cannot capture jamming effects as precise as direct ones. We note that by using indirect measurements (e.g., hearing ranges), we can extend our error-minimizing framework to localize other types of jammers, such as reactive jammers. In summary, besides using direct measurements, our error minimizing framework can leverage other types of indirect measurements to obtain an improved estimation of jammers' locations than Adaptive LSQ.

7 Concluding Remarks

In this work, we addressed the problem of localizing jammers in wireless networks, aiming to extensively reduce estimation errors. The jammers could be several wireless devices causing unintentional radio interference or malicious colluding jamming devices who co-exist and disturb the network together. Most of the existing schemes for localizing jammers rely on the indirect measurements of network parameters affected by jammers, e.g., nodes' hearing ranges, which makes it difficult to accurately localize jammers. In this work, we localized jammers by exploiting directly the jamming signal strength (JSS). Estimating JSS is considered challenging since they are usually embedded with other signals. Our estimation scheme smartly derives ambient noise floors as the JSS utilizing the available signal strength measuring capability in wireless devices. The scheme samples signal strength regardless whether the channel is busy or idle, and estimates the ambient noise floor by filtering out regular transmission (if any) to obtain the JSS. We implemented estimation scheme on MicaZ motes. Our experiment involving three jammers show that our estimation scheme can accurately derive the JSS from the measurements of ambient noise floor under various traffic scenarios.

To further improve the estimation accuracy, we designed an error-minimizing-based framework to localize jammers. In particular, we defined an evaluation feedback metric that quantifies the estimation errors of jammers' positions. We studied the relationship between the evaluation feedback metric and estimation errors, and showed that the locations that minimize the feedback metric approaches jammers' true locations and greedy algorithms may not find the global optimal solutions. Thus, we treated the evaluation feedback metric as the objective function for the error-minimizing purpose. We examined several heuristic search algorithms (GA, GPS and SA) under various network conditions: node densities, jammer's transmission power, the propagation irregularity, and number of jammers. Besides, we examined our error minimizing framework utilizing an indirect measurement–a hearing range. Our extensive simulation results show that our error-minimizing-based search algorithms utilizing both the direct and indirect measurements outperform the existing algorithms in all experiment configurations. In particular, among the three searching algorithms, we found that GPS can find the best estimation of multiple jammers' positions in the shortest duration.

References

[1] K. Pelechrinis, I. Koutsopoulos, I. Broustis, and S. V. Krishnamurthy, "Lightweight jammer localization in wireless networks: System design and implementation," in *Proceedings of IEEE GLOBECOM*, 2009.

[2] H. Liu, Z. Liu, Y. Chen, and W. Xu, "Determining the position of a jammer using a virtual-force iterative approach," *Wireless Networks (WiNet)*, vol. 17, pp. 531–547, 2010.

[3] Z. Liu, H. Liu, W. Xu, and Y. Chen, "Exploiting jamming-caused neighbor changes for jammer localization," *IEEE TPDS*, vol. 23, no. 3, 2011.

[4] H. Liu, Z. Liu, Y. Chen, and W. Xu, "Localizing multiple jamming attackers in wireless networks," in *Proceedings of ICDCS*, 2011.

[5] T. Cheng, P. Li, and S. Zhu, "Multi-jammer localization in wireless sensor networks," in *Proceedings of CIS*, 2011.

[6] A. Wood, J. Stankovic, and S. Son, "JAM: A jammed-area mapping service for sensor networks," in *Proceedings of RTSS*.

[7] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proceedings of MobiHoc*, 2005.

[8] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.

[9] T. Rappaport, *Wireless Communications- Principles and Practice*. Prentice Hall, 2001.

[10] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of INFOCOM*, 2000.

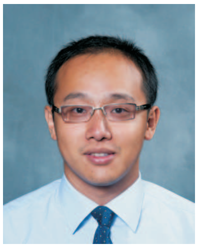
[11] J. Yang, Y. Chen, and J. Cheng, "Improving localization accuracy of rss-based lateration methods in indoor environments," *AHSWN*, vol. 11, no. 3-4, pp. 307–329, 2011.

[12] D. Goldberg, *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley, 1989.

[13] E. Polak, *Computational Methods in Optimization: a Unified Approach*. Academic Press, 1971.

[14] P. V. Laarhoven and E. Aarts, *Simulated Annealing: Theory and Applications*. Springer, 1987.

[15] Z. Liu, H. Liu, W. Xu, and Y. Chen, "Wireless jamming localization by exploiting nodes' hearing ranges," in *Proceedings of DCOSS*, 2010.



Zhenhua Liu received his Ph.D. degree in Computer Science and Engineering from University of South Carolina in 2012. His research interests include wireless PHY/MAC layer security, jamming/radio interference, wireless localization/tracking and mobile computing. He is currently working in the Arena for Research on Emerging Networks and Applications (ARENA) lab with Prof. Wenyuan Xu. He received his Bachelor degree of Electronics and Information Engineering from Department of Information Science and Technology in Central South University, Hunan, China, in 2006. He was awarded Outstanding Research Assistant at University of South Carolina in 2011.

Hongbo Liu is a Ph.D. candidate of the Electrical and Computer Engineering Department at Stevens Institute of Technology. His research interests include information security and privacy, mobile computing and wireless localization systems, wireless and sensor networks, and digital signal processing. He is currently working in the Data Analysis and Information SecuritY (DAISY) Lab with Prof. Yingying Chen. He received his Bachelor's and Masters degree in Communication Engineering from Department of Communication and Information Engineering at University of Electronic Science and Technology of China in 2005 and 2008 respectively. He was the recipient of the Best Paper Award from the ACM International Conference on Mobile Computing and Networking (MobiCom) 2011. He was also the recipient of the Outstanding Undergraduate Student Thesis Award 2005.

Wenyuan Xu received her Ph.D. degree in Electrical and Computer Engineering from Rutgers University in 2007. She is currently an Assistant Professor in the Department of Computer Science and Engineering, University of South Carolina. Her research interests include wireless network security and privacy, embedded device security, pervasive computing, and wireless sensor networks. Dr. Xu is a coauthor of the book Securing Emerging Wireless Systems: Lower-layer Approaches, Springer, 2009. She received the NSF Career Award in 2009, and has served on the technical programs for several IEEE/ACM conferences on wireless networking and security.

Yingying Chen is an Associate Professor in the Department of Electrical and Computer Engineering at Stevens Institute of Technology. Her research interests include cyber security and privacy, wireless and sensor networks, mobile social networks and pervasive computing. She received her Ph.D. degree in Computer Science from Rutgers University. Prior to joining Stevens Institute of Technology, she was with Alcatel-Lucent at Holmdel and Murray Hill, New Jersey. She has co-authored the book Securing Emerging Wireless Systems Springer 2009. She is the director of Data Analysis and Information Security (DAISY) Lab at Stevens. She is the recipient of the NSF CAREER Award 2010 and Google Research Award 2010. She received Stevens Board of Trustees Award for Scholarly Excellence in 2010. She is also the recipient of the Best Paper Awards from ACM International Conference on Mobile Computing and Networking (MobiCom) 2011 and International Conference on Wireless On-demand Network Systems and Services (WONS) 2009, as well as the Best Technological Innovation Award from the International TinyOS Technology Exchange 2006. She also received the IEEE Outstanding Contribution Award from IEEE New Jersey Coast Section each year 2005-2009.