Zeroing-In on Network Metric Minima for Sink Location Determination

Zhenhua Liu,   Wenyuan Xu *
Dept. of Computer Science & Engineering
University of South Carolina, Columbia, SC, USA
liuz,wyxu@cse.sc.edu

ABSTRACT

The locations of base stations are critically important to the viability of wireless sensor networks. In this paper, we examine the location privacy problem from both the attack and defense sides. We start by examining adversaries targeting at identifying the sink location using minimum amount of resources. In particular, they launch a Zeroing-In attack leveraging the fact that several network metrics are 2-dimensional functions in the plane of the network and their values minimize at the sink. Thus, determining the sink locations is equivalent to finding the minima of those functions. We have shown that by obtaining the hop counts or the arrival time of a broadcast packet at a few spots in the network, the adversaries are able to determine the sink location with the accuracy of one radio range, sufficient to disable the sink by launching jamming attacks. To cope with the Zeroing-In attacks, we have proposed a directed-walk-based scheme and validated that the defense strategy is effective in deceiving adversaries at little energy costs.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and protection*

General Terms

Security

Keywords

Sink location privacy, Sensor networks

1. INTRODUCTION

Wireless sensor networks (WSNs) typically consist of a large collection of low power and resource-constrained sensor nodes that monitor the underlying physical phenomena, and a small set of base stations, aka. sinks, that collect sensor data in a multihop fashion. Such a many-to-one communication pattern makes the small number of sinks the central points of failure. Adversaries can easily leverage the sink location information to launch a series of attacks interrupting the network communication. For instance, an adversary can physically approach the sinks and initiate jamming attacks [17], which can prevent sinks from receiving measurements sampled by sensors. Alternatively, an adversary can destroy the sinks physically by human intervention, such as hammering them. Without sinks' gleaning and relaying measurements to data analysis and actuation components, the sensor networks will become paralyzed. Thus, it is crucial to preserve the sink location information.

Several attacks have been proposed to determine the locations of sinks, including trace-back attacks [8] and traffic analysis attacks [5]. Most of them assume resource-intensive adversaries. Some require the adversary to equip with special radio devices that can measure the angle of arrival [8] so that she can identify the immediate source of a transmitter. Some require the adversary to have a global view [10, 18] of the network communication. It requires the adversary to deploy its own sensors throughout the entire network to capture the whole network communications.

In this paper, we are primarily interested in budget adversaries who do not have specialized radio devices and are unable to monitor the entire networks simultaneously. Instead, each adversary can only eavesdrop the network communication at a single spot. Together they can launch Zeroing-In attacks leveraging the following observations. Essentially, several network metrics are two dimensional functions of locations, and their values minimize at the sink. Thus, determining the sink location becomes a problem of finding the minima of those functions. Therefore, by sampling network metrics at a few spots, they can derive the location of the sinks collaboratively.

We present our assumptions and models in Section 2 and overview the Zeroing-In attacks in Section 3. In Section 4 and Section 5, we detail a few attacks that require decreasing amount of information from the network. We evaluate the effectiveness of attacks in Section 6 and present defense strategies in Section 7. We conclude the paper with related work in Section 8 and concluding remarks in Section 9.

2. SYSTEM MODEL

In this section, we define the sensor network and adversary models that are most relevant to the sink location privacy problem in this paper.

2.1 Network Model

We consider a sensor network utilizing the popular many-to-one data dissemination methods [7, 9], whereby the sink is connected to a large portion of the sensor nodes. Without loss of generality, we assume that there is only one sink in the network. We note the Zeroing-In attacks can be applied to a network with multiple sinks. The network maintains a forwarding tree [19] to route data to the sink. This forwarding tree is often built with the assistance of hop counts, e.g., the number of hops from the sink. Additionally, since broadcast is a fundamental method for WSNs to discover routes and to deliver information to a large portion of the node, we assume that the sink will flood the network with controlling commands or query requests from time to time. Finally, each node uses the same type of hardware platform and sends messages at the same transmission power level. As a result, they have similar radio range.

2.2 Adversary Model

The adversaries considered in this paper have the following characteristics:

- *Eavesdrop-Enabled.* We assume that the adversaries have the same radios as the network nodes and, thus, are able to eavesdrop the radio communication in the network but unable to decipher the packet.

- *Resource-Limited.* The adversaries are not equipped with specialized radio devices, such as super sensitive antenna arrays. Thus, they are unable to trace the immediate sender by measuring the packet's angle of arrival. Additionally, the adversaries cannot afford to deploy their own sensor network to monitor the entire network. Each adversary can only monitor local traffic.

- *Able to Collude.* Multiple adversaries can share their local views via some communication methods and collude with each other to infer the location of the sink. For instance, they can move close to each other and exchange information. Additionally, we assume they have a loosely synchronized clock.

- *Location-Aware.* With the increasing popularity of localization services and the price drop of GPS devices, we assume that each adversary knows its own location.

- *Protocol-Aware.* According to Kerckhoff's Principle [14], we assume that adversaries know the protocols used in the networks. Additionally, the adversaries are aware of the typical transmission power of nodes and the areas within which the sensor network is deployed.

3. ZEROING-IN ATTACK OVERVIEW

Although a global adversary or an adversary armed with special radio devices can identify the sink location with the help of rich resources, we show in this paper that it is feasible to localize the sink by a small number of adversaries with each containing significantly less resources. This finding is valuable, as it demands the design of defense strategies that will raise the bar of attacks. In this section, we overview the proposed "Zeroing-In" attacks. In Section 4 and Section 5, we present attacks that can determine the sink location with decreasing information from the networks.

Several metrics in a sensor network are functions of locations. Typically, moving towards the sink either increases the values of those network metrics or decreases them monotonically. For instance, a hop count is the smallest number
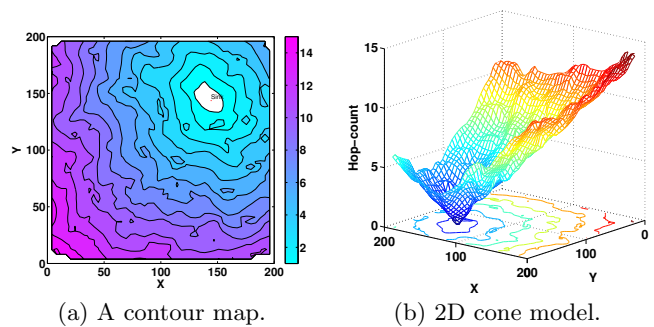

(a) A contour map.　　　(b) 2D cone model.

Figure 1: The hop count in a 400-node network with the sink located at $(144, 143)$.

of intermediate nodes a packet has to traverse in order to reach the sink. The hop count of a network node decreases as the node gets closer to the sink, and it becomes zero at the sink, as illustrated in Figure 1. The traffic rate increases as the distance to the sink decreases and reaches maxima at the sink. One can model those network metrics as two-dimensional (2D) parabolas, and they reach their extremum at the location of sink. Thus, if the 2D model is known, determining the location of sink becomes a problem of finding either the network metric minima or maxima. Without loss of generality, in this paper, we focus on finding the minima of network metrics. We note that it is trivial to convert the maximization problem into minimization one.

The goal of the "Zeroing-In" attack[1] is to identify the position of the sink leveraging the 2D model of the network metrics. Towards this goal, the Zeroing-In attacks consist of two steps:

Sampling Step. m adversaries place themselves in an area S, within which the network is deployed. Each adversary measures the network metric by eavesdropping her local communication. We denotes the coordinates of the ith adversary as $z_i = (x_i, y_i)$, and the i-th observation as a tuple (x_i, y_i, h_i), where h_i is a network metric. As we will show in the following sections, the metric h_i can either be the hop count or the arrival time of a packet at (x_i, y_i). For the simplicity of analysis, we assume that the adversary obtains the network metrics associated with the node that is closest to her.

Zeroing-In Step. Identify the position of the sink: Adversaries determine the 2D network metric function $h = f(x, y)$ by analyzing m observations $\{(x_i, y_i, h_i)\}_{i=1...m}$ and find the sink location $z_s = (x_s, y_s)$ as the point where $f(x, y)$ reaches minimum

$$(\hat{x_s}, \hat{y_s}) = \underset{(x,y) \in S}{\arg \min} f(x, y).$$

4. HOP-COUNT-BASED ZEROING-IN ATTACKS

We start with the Zeroing-In attack utilizing hop counts, a metric that has been widely used in routing protocols [15]. In this section, we assume each adversary can acquire the hop count of any node in the network but is limited to acquiring the hop count at only one location in each routing

---

[1]We call such an attack "Zeroing-In" attack, since the hop count becomes zero at the sink. Determining the location of the sink is equivalent to find the location where the hop count equals to zero.
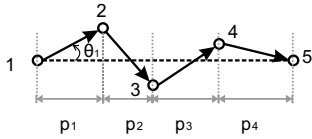
Figure 2: An illustration of calculating the hop size between node 1 and 5.

broadcast window. Additionally, we assume that the adversary can inject broadcast packets to the network. We note that these assumptions may not be true in some networks. We will show alternative strategies that do not depend on these assumptions later.

4.1 Model Hop Counts and Hop Sizes

The first step to launch the Zeroing-In attacks is to understand the underlying network metric model. In this section, we model hop counts as a function of locations, e.g., $h = f(x, y)$.

A node's hop count h depends on many factors, including its own location, the sink location, the positions of other nodes located towards the sink, the irregular radio range of each node, and etc. Building an accurate math model to represent hop counts requires significant complexity and is difficult, if possible. Thus, to understand the hop-count distribution across the entire network, we performed a numerical study by using the Castalia simulator [2] 2.1b, an open source simulator for wireless sensor network built on top of OMNeT++ [1]. We chose Castalia because it simulates an irregular channel that exhibits a "transitional region" [20], a typical phenomena in real systems caused by multi-path fading. Figure 1 depicts the values of hop counts across the network that uses the popular grid-based coverage model [16, 18]. The hop counts exhibit a cone surface, and we can model it as,

$$h = \frac{1}{\alpha}||z - z_s|| = \frac{1}{\alpha}\sqrt{(x - x_s)^2 + (y - y_s)^2}, \quad (1)$$

where α is the *hop size* that describes the relationship between hop counts and distances. In real systems, α is a variable in a 2D space, and if α has a small variance and can be estimated, then the sink location can be considered as the point that minimize the function f using the estimation $\hat{\alpha}$,

$$(\hat{x_s}, \hat{y_s}) = \arg\min_{(x,y)\in S}(\frac{1}{\hat{\alpha}}||z - z_s|| - h).$$

To examine the distribution of α, we consider the example illustrated in Figure 2, where $\alpha_{15} = ||z_1 - z_5||/h_{15}$. Let L_{ij} be the line segment that connects node i and node j, and let p_k be the projection of the kth link onto L_{ij}. α_{15} can be considered as the average value of the projection of the k-th link onto L_{15} for all intermediate nodes $k = 1, 2, 3, 4$. For a random pair of nodes i and j, let h_{ij} be the hop count between them, then

$$\alpha_{ij} = \frac{1}{h_{ij}}\sum_{k=1}^{h_{ij}} p_k = \frac{1}{h_{ij}}\sum_{k=1}^{h_{ij}} ||z_k - z_{k+1}||\cos\theta_k \quad (2)$$

where θ_k is the angle between the line L_{kk+1} and L_{ij} and $\theta_k \in [0, \pi/2]$.

Although it is complicated to get the distribution of the length of projections $\{p_k\}_{k=1...h_{ij}}$, we can predict that the distribution of α is a normal distribution according to the

central limit theorem (CLT), which says that the average of a sufficiently large number of independent variables with the same mean and variance follows a normal distribution approximately. Thus, the distribution of α_{ij} approaches a normal distribution as h_{ij} increases.

4.2 Determine the Sink Location

The hop-count-based Zeroing-In attack consists of the following two steps:

Sampling Step. Obtain $\{(x_i, y_i, h_i, \hat{\alpha_i})\}_{i=1...m}$, where h_i is the hop count and $\hat{\alpha_i}$ is the estimated hop size between the i-th adversary and the sink. To estimate $\hat{\alpha_i}$, each adversary floods a message to all other adversaries and obtains the hop counts between them. The $\hat{\alpha_i}$ is calculated by

$$\hat{\alpha_i} = \frac{\sum_{j=1,j\neq i}^{m} ||z_i - z_j||}{\sum_{j=1,j\neq i}^{m} h_{ij}}. \quad (3)$$

This method was also used in DV-hop localization [11]. We will introduce an alternative method that does not require the use of message flooding in the later section.

Zeroing-In Step. Determine the position (x_s, y_s) of the sink by searching for $(\hat{x_s}, \hat{y_s})$ satisfying

$$(\hat{x_s}, \hat{y_s}) = \arg\min_{(x_s,y_s)}\sum_{i=1}^{m}[\sqrt{(x_i - x_s)^2 + (y_i - y_s)^2} - \hat{\alpha_i}h_i]^2 \quad (4)$$

Essentially, adversaries calculate their distances to the sink by $\hat{\alpha_i}h_i$ and find the position of the sink as the point that minimizes the overall estimation error.

We use least squares to solve Equation 4. To avoid the complicated nonlinear least squares calculation, we linearize the problem by eliminating the quadratic components. We start with m equations, $i = 1...m$:

$$(x_i - x_s)^2 + (y_i - y_s)^2 = (\alpha_i h_i)^2 \quad (5)$$

Assume that $h_m = \min\{h_i\}_{i=1...m}$, subtract the mth equation from both sides of the first $m - 1$ equations, we write the derived set of linear equations in the form $\mathbf{Az} = \mathbf{b}$ with

$$\mathbf{A} = \begin{pmatrix} x_1 - x_m & y_1 - y_m \\ \vdots & \vdots \\ x_{m-1} - x_m & y_{m-1} - y_m \end{pmatrix}$$

and

$$\mathbf{b} = \frac{1}{2}\begin{pmatrix} (x_1^2 - x_m^2) + (y_1^2 - y_m^2) \\ -(\alpha_1^2 h_1^2 - \alpha_m^2 h_m^2) \\ \vdots \\ (x_{m-1}^2 - x_m^2) + (y_{m-1}^2 - y_m^2) \\ -(\alpha_{m-1}^2 h_{m-1}^2 - \alpha_m^2 h_m^2) \end{pmatrix}.$$

The least squares solution of for Equations 5 can be calculated by

$$\hat{\mathbf{z}} = [\hat{x_s}, \hat{y_s}]^T = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\hat{\mathbf{b}}, \quad (6)$$

where $\hat{\mathbf{b}}$ is the estimation with errors.

5. TIME-OF-ARRIVAL BASED ZEROING-IN ATTACKS

We now discuss another Zeroing-In attack that utilizes the packets' Time-of-Arrival (ToA). The ToA-based Zeroing-In attack can be used when entire packets are encrypted and
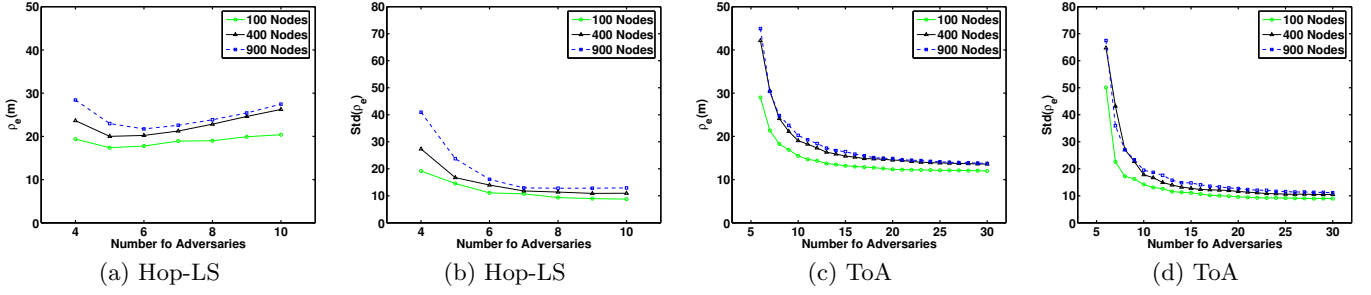
Figure 3: The impact of the network size and the number of adversaries to the attack performance.

the hop count is not accessible to the adversaries. Instead, adversaries can only distinguish whether two received packets are the same and witness the arrival time of packets. The attack starts with m adversaries placing themselves across the network. When the sink floods out a controlling message at t_0, the ith adversary records the packet arrival time as t_i. In total, m adversaries record m samples $\{(x_i, y_i, t_i)\}_{i=1...m}$ and collectively identify the sink location.

To derive the mathematical model between t_i and z_i, we define the message reaches at the ith adversary at the travel speed $s_i = ||z_s - z_i||/T_i$, where T_i is the end-to-end transmission time for a message to travel from the sink to the ith adversary. Then we have m equations:

$$(x_i - x_s)^2 + (y_i - y_s)^2 = s_i^2(t_i - t_0)^2 \qquad (7)$$

Since T_i is the summation of the transmission delay on each hop, we apply Central Limit Theory to s_i and consider the distribution of s_i approximately a normal distribution with the mean value $\bar{s}$. Eliminating the quadratic components, we can get $\mathbf{Az} = \mathbf{b}$ with:

$$\mathbf{A} = \begin{pmatrix} x_1 - x_m & y_1 - y_m & \frac{1}{2}(t_1^2 - t_m^2) & t_1 - t_m \\ \vdots & \vdots & \vdots & \vdots \\ x_{m-1} - x_m & y_{m-1} - y_m & \frac{1}{2}(t_{m-1}^2 - t_m^2) & t_{m-1} - t_m \end{pmatrix},$$

$$\mathbf{z} = [x_s, y_s, \bar{s}^2, \bar{s}t_0]^T,$$

and

$$\mathbf{b} = \frac{1}{2} \begin{pmatrix} (x_1^2 - x_m^2) + (y_1^2 - y_m^2) \\ \vdots \\ (x_{m-1}^2 - x_m^2) + (y_{m-1}^2 - y_m^2) \end{pmatrix}.$$

The location of the sink is

$$\hat{\mathbf{z}} = [\hat{x_s}, \hat{y_s}, \hat{s^2}, \hat{st_0}]^T = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\hat{\mathbf{b}}. \qquad (8)$$

6. EVALUATE THE EFFECTIVENESS OF ZEROING-IN ATTACKS

6.1 Evaluation Metrics

We define three metrics to evaluate the performance of Zeroing-In attacks.

–**Estimation Accuracy** indicates how well the adversaries estimate the sink location on average. We define estimation accuracy as the mean error, $\rho_e = E(||\hat{z_s} - z_s||)$, where z_s is the true location of the sink, and $\hat{z_s}$ is the estimated location.

–**Attack Stability** is the standard deviation of the estimation errors, $\sigma_e = std(\rho_e)$. A smaller σ_e indicates a higher

confidence of the estimation and, thus, maps to a more reliable attack.

–**Attack Cost** measures how much resource is used in terms of the number of adversaries.

6.2 Experiment Results

We evaluated the performance of Zeroing-In attacks using Castalia 2.1b, an OMNeT++-based simulator for Wireless Sensor Networks. We adopted the popular grid-based coverage model [16, 18]. Given that the average node transmission range is $18m$, we divided the square network area into $10 \times 10m$ grids and placed one node randomly inside each grid. Unless specified otherwise, the networks in experiments were deployed in a $200 \times 200m$ square and the sink was randomly placed anywhere inside the network region. To capture the average trend of each factor, we repeated our experiments in 1000 different network topologies for each experiment set. We studied two attack strategies: Hop-LS (the hop-count-based Zeroing-In attack) and ToA (the ToA-based Zeroing-In attack). We will use the short names for the rest of discussion.

The Network Size and the Number of Adversaries. This set of experiments aim at answering two important questions: (1) Does increasing the network size help to hide the location of the sink? (2) does increasing the number of adversaries always yield a better estimation of the sink location? To find the answers, we studied the attack performance in three network sizes where $\{100, 400, 900\}$ nodes were placed in a $\{100 \times 100, 200 \times 200, 300 \times 300\}$ square, respectively. In each network setup, we studied the attack performance by increasing the number of adversaries. Experiment results are depicted in Figure 3, which show that network size has little impact on both the mean error and the standard deviation of the attacks.

Interestingly, for the Hop-LS attack, as the number of the adversaries increases, the mean error ρ_e decreases first and increases when more than 5 adversaries are involved. We believe this is caused by the way how the hop size is measured. Increasing the number of adversaries enhanced the mis-match between the estimated hop size and the true hop size and, therefore, led to bigger mean errors. Since 5 adversaries provided a good trade-off between mean error and attack cost, 5 adversaries shall be used to launch the Hop-LS attacks.

Compared with Hop-LS attack, as the number of adversary increases, the mean error and standard deviation of ToA attacks diminish monotonically, which suggests that increasing the number of adversaries improves the attack accuracy.
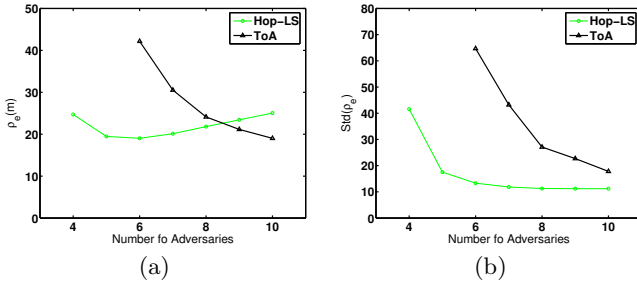
Figure 4: The comparison between various attacks. We do not show the 5 adversary case for ToA attacks to confine the display range for y axis.

The Attack Algorithms. When hop counts are not accessible to adversaries, ToA is the only option to determine the sink location. However, if adversaries are able to obtain hop counts and flood the network to estimate the hop size, which type of attacks should the adversaries launch? We provide a close-up comparison in Figure 4 to demonstrate the choices. Due to the extra variables in ToA, it takes extra measurements for ToA to decrease the uncertainty. Thus, when the hop count information is accessible to the adversaries and at most 8 adversaries are available, they shall determine the sink location via Hop-LS. Otherwise, ToA is preferred.

7. COPE WITH ZEROING-IN ATTACKS

We turn to the defense side and will focus on coping with the ToA-based Zeroing-In attack since the hop-count-based Zeroing-In attacks can be addressed by hiding the node's hop count in the routing update messages.

Random Buffering. To preserve the sink location privacy, one should make it difficult for the adversaries to infer the position of the sink leveraging the packet arrival time. One natural defense strategy is to have every node buffer a flooding message for a random amount of time before forwarding it to the next hop. However, random buffering does not change the statistical relationship between T_i and z_i, and it cannot hide the location of the sink. Our experiments confirmed this conclusion.

Directed Walk. To alter the relationship between T_i and z_i, the sink can unicast the message $\mathcal{M}_f$ to a designated node n_{ds} that is located many hops away. Once n_{ds} receives $\mathcal{M}_f$, it will start the flooding. The challenge of this method involves choosing n_{ds} and finding the route from the sink to n_{ds}. The routing protocols in sensor networks aim at facilitating sensors to report data to the sink in a multi-hop fashion and are not designed to send messages from the sink to any network nodes. We propose to leverage the routing tree and have $\mathcal{M}_f$ travel in the inverse direction of regular sensor data.

When the sink creates a flooding message, $\mathcal{M}_f$, it sets a counter h_{TTL} in the message header to indicate how many hops away n_{ds} is located. Then the sink unicasts $\mathcal{M}_f$ to one of its children n_i. After n_i receives $\mathcal{M}_f$, it first decreases the h_{TTL} by one. If the h_{TTL} equals 0 then n_i starts to flood the message $\mathcal{M}_f$. Otherwise, n_i sends $\mathcal{M}_f$ to one of its children. This process repeats until the h_{TTL} in $\mathcal{M}_f$ is reduced to zero. The advantage of using such directed walk is that $\mathcal{M}_f$ is forwarded away from the sink in the shortest path. Thus, $\mathcal{M}_f$ reaches one of the farthest nodes from the sink with respect to the amount of energy spent moving it.

Experiment Evaluation. We conducted experiments to evaluate the defense strategies using the same simulation setup as the one in the attack evaluation experiments. We calculated the ρ_e and $Std(\rho_e)$ when ToA attacks were performed in a 400-node network deployed in the $200 \times 200m$ square. Figure 5 (a,b) shows the attack performance with the increasing number of adversaries in three setups: (1) a baseline case, whereby no defense strategy was used, (2) random buffering, whereby each node buffers the message for a duration that follows a uniform distribution, $U(0, 5s)$, (3) directed walk, whereby messages were designated to a node that is 8 hops away from the sink. The results show that the random buffering can only confuse the adversaries by 10 meters on average and does not provide sufficient protection to the sink location. Directed walk of 8 hops can consistently deceive the adversaries into thinking the sink is 100 meters away from its true location. Figure 5 (c) shows that the level of protection for the sink location grows linearly when the number of hops of the designated nodes increases, which proves that directed walk is an effective and energy-efficient method to cope with ToA-based Zeroing-In attacks.

8. RELATED WORK

Both source location privacy and sink location privacy have attracted attention from the research community. Source location privacy focuses on protecting the message source, as such information can reveal sensitive position information of the target that appears close to the message source. The source location privacy was first studied by Kamat et al. [8], where fake message injection and phantom routing are proposed to prevent a local eavesdropper from discovering the message source through hop-by-hop traces.

The problem of preserving source location privacy under a global eavesdropper has been studied extensively [10, 12, 13, 18]. Mehta *et al.* [10] have proposed periodic collection and source simulation techniques to prevent the leakage of message source location, and Yang *et al.* [18] have introduced dummy traffic to hide the real message source. Ouyang *et al.* [12] have devised a set of privacy-preserving algorithms involving sending periodical maintainable messages to address a laptop-class attacker who has longer radio range and can eavesdrop all communications in a sensor network. A notion of statistically strong source anonymity is proposed by Shao *et al.* [13], and a strategy called FitProbRate has been proposed to achieve statistically strong source anonymity with a reduced real event report latency.

Sink location privacy has been studied recently. Deng *et al.* [4] have shown that traffic analysis can reveal the location of sinks and proposed several anti-traffic analysis countermeasures to hide the direction of data flow and create fake sink locations with artificially high traffic rate. In the their follow-up work [5], multiple parent routing, controlled random walk, random fake paths, and combination of all three routing algorithms have been studied to generate randomness against traffic rate monitoring and traffic path direction attacks. Location Privacy Routing (LPR) [6] utilizes probabilistic routing and fake message injection to deceive an adversary from tracking the direction of traffic flow. Conner *et al.* [3] proposed the decoy sink protocol, whereby data are forwarded to a decoy sink for aggregation before they
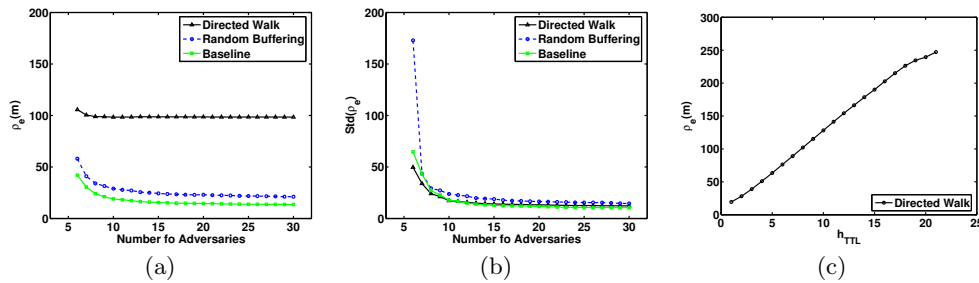
Figure 5: (a-b) The attack performance in the baseline case, random buffering case, and directed walk case. (c) The mean error of the attack with respect to various walk hops.

are relayed to the real sink. As a result, the traffic volume near the sink is reduced while decoy sinks exhibit high traffic volume, which makes traffic analysis attacks difficult.

In this paper, we focused on issues related to sink location privacy. Our work differs from prior work. Instead of examining powerful attackers, we studied budget adversaries and were interested in the minimum amount of resource required to make an attack feasible.

9. CONCLUDING REMARKS

Due to the many-to-one communication paradigm in wireless sensor networks, the locations of sinks are of critical importance. In this paper, we have studied the sink location privacy problem from both the attack and the defense sides. We have shown that many network metrics can be modeled as a two dimensional function of locations, and their values are either minimized or maximized at the sink. We have presented the Zeroing-In attacks, whereby a few adversaries observe the network metrics by eavesdropping the local communication and collectively determine the sink location by solving the least squares problem over the observations. We have investigated Zeroing-In attacks that utilize hop counts and the packet time of arrival (ToA). Our experiment results show that both Zeroing-In attacks can localize the sink at the accuracy level of one radio range, which is accurate enough for the adversaries to perform a jamming attack against the sink. To deal with ToA-based attacks, we presented a directed-walk-based defense strategy, whereby the sink unicasts the message to a designated node n_{ds} and has n_{ds} initiate the flooding. Our experiment result have validated that directed walk is effective in protecting the sink location information from the adversaries.

10. REFERENCES

[1] OMNeT++ homepage. http://www.omnetpp.org/.

[2] A. Boulis. Castalia: revealing pitfalls in designing distributed algorithms in wsn. In *SenSys '07: Proceedings of the 5th international conference on Embedded networked sensor systems*, pages 407–408, New York, NY, USA, 2007. ACM.

[3] W. Conner, T. Abdelzaher, and K. Nahrstedt. Using data aggregation to prevent traffic analysis in wireless sensor networks. In *DCOSS '06: International Conference on Distributed Computing in sensor networks*, 2006.

[4] J. Deng, R. Han, and S. Mishra. Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks. In *DSN '04: Proceedings of the 2004 International Conference on Dependable Systems and Networks*, page 637, Washington, DC, USA, 2004. IEEE Computer Society.

[5] J. Deng, R. Han, and S. Mishra. Countermeasures against traffic analysis attacks in wireless sensor networks. In *SECURECOMM '05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, pages 113–126, Washington, DC, USA, 2005. IEEE Computer Society.

[6] Y. Jian, S. Chen, Z. Zhang, and L. Zhang. Protecting receiver-location privacy in wireless sensor networks. In *INFOCOM'07: 26th IEEE International Conference on Computer Communications*, pages 1955–1963, 2007.

[7] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein. Energy-Efficient Computing for Wildlife Tracking: Design and Tradeoffs and Early Experiences with Zebranet. In *Proceedings of the Tenth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 96–107, 2002.

[8] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk. Enhancing source-location privacy in sensor network routing. In *ICDCS '05: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, pages 599–608, Washington, DC, USA, 2005. IEEE Computer Society.

[9] S. Madden, M. Franklin, J. Hellerstein, and W. Hong. TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks. In *Proceedings of the Usenix Symposium on Operating Systems Design and Implementation*, 2002.

[10] K. Mehta, D. Liu, and M. Wright. Icnp'07: Location privacy in sensor networks against a global eavesdropper. In *Proceedings of the IEEE International Conference on Network Protocols*, pages 314–323, 2007.

[11] D. Nicelescu and B. Nath. DV based positioning in ad hoc networks. *Telecommunication Systems*, 22(1-4):267–280, 2003.

[12] Y. Ouyang, Z. Le, D. Liu, J. Ford, and F. Makedon. Source location privacy against laptop-class attacks in sensor networks. In *SecureComm '08: Proceedings of the 4th international conference on Security and privacy in communication netowrks*, pages 1–10, New York, NY, USA, 2008. ACM.

[13] M. Shao, Y. Yang, S. Zhu, and G. Cao. Towards statistically strong source anonymity for sensor networks. In *INFOCOM'08: 27th IEEE International Conference on Computer Communications*, pages 51–55, 2008.

[14] W. Trappe and L. Washington. *Introduction to Cryptography with Coding Theory*. Prentice Hall, 2002.

[15] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 14–27, 2003.

[16] G. Xing, X. Wang, Y. Zhang, C. L. R. Pless, and C. Gill. Integrated coverage and connectivity configuration for energy conservation in sensor networks. *ACM Trans. Sen. Netw.*, 1(1):36–72, 2005.

[17] W. Xu, W. Trappe, Y. Zhang, and T. Wood. The feasibility of launching and detecting jamming attacks in wireless networks. In *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pages 46–57, 2005.

[18] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao. Towards event source unobservability with minimum network traffic in sensor networks. In *WiSec '08: Proceedings of the first ACM conference on Wireless network security*, pages 77–88, New York, NY, USA, 2008. ACM.

[19] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 1–13, 2003.

[20] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *SECON'04: Proceedings of*, pages 517–526, 2004.