Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 648058, 16 pages
doi:10.1155/2012/648058

Research Article

Enhancing Sink-Location Privacy in Wireless Sensor Networks through *k*-Anonymity

Guofei Chai,[1] Miao Xu,[2] Wenyuan Xu,[2] and Zhiyun Lin[1]

[1] *College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China*
[2] *Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA*

Correspondence should be addressed to Wenyuan Xu, wyxu@cse.sc.edu

Received 23 May 2011; Revised 5 January 2012; Accepted 7 January 2012

Academic Editor: Yuhang Yang

Due to the shared nature of wireless communication media, a powerful adversary can eavesdrop on the entire radio communication in the network and obtain the contextual communication statistics, for example, traffic volumes, transmitter locations, and so forth. Such information can reveal the location of the sink around which the data traffic exhibits distinctive patterns. To protect the sink-location privacy from a powerful adversary with a global view, we propose to achieve *k-anonymity* in the network so that at least *k* entities in the network are indistinguishable to the nodes around the sink with regard to communication statistics. Arranging the location of *k* entities is complex as it affects two conflicting goals: the routing energy cost and the achievable privacy level, and both goals are determined by a nonanalytic function. We model such a positioning problem as a nonlinearly constrained nonlinear optimization problem. To tackle it, we design a generic-algorithm-based quasi-optimal (GAQO) method that obtains quasi-optimal solutions at quadratic time. The obtained solutions closely approximate the optima with increasing privacy requirements. Furthermore, to solve *k*-anonymity sink-location problems more efficiently, we develop an artificial potential-based quasi-optimal (APQO) method that is of linear time complexity. Our extensive simulation results show that both algorithms can effectively find solutions hiding the sink among a large number of network nodes.

1. Introduction

With the increasing advances of sensing devices and wireless technology, wireless sensor networks (WSNs) have been interwoven into the fabric of our daily life. In particular, WSNs have been deployed to monitor personal health, track targets, and sense pollutants. Those sensor networks typically consist of many resource-constrained sensor nodes and one sink. Each sensor node monitors the underlying physical phenomenon and reports the measurements to the sink in a multihop manner.

In spite of their popularity, the viability and success of those sensor networks hinge on a variety of security and privacy threats. One of the most challenging threats is location privacy, since it cannot be addressed by traditional cryptographic mechanisms [1]. Due to the shared nature of wireless communication media, an attacker can easily eavesdrop on the radio communication either by purchasing her own sensor devices or by leveraging other radio devices capable of monitoring message transmission. Thus, no matter whether messages are encrypted or not, an adversary is able to identify contextual information: where the communication has occurred and who has participated in communication, without accessing the content of messages. For example, an adversary can identify the sender of a message by analyzing the angle of arrival [2], or he can determine the receiver in the similar fashion when the receiver relays a message [3].

Since an adversary can locate both the origin and destination of messages (i.e., sinks) purely by observing the contextual information, the WSN location privacy problem can be divided into two categories: *source*-location privacy and *sink*-location privacy. The source-location privacy problem is concerned with preventing attackers from discovering the locations of message sources, which may reveal sensitive position information of assets being monitored, for example, endangered animals. Much effort has been devoted to preserve source-location privacy against a wide variety of attackers, ranging from resource-constrained attackers [2]

to powerful attackers that have a global view of network communications [1, 4].

In this study, we focus on preserving *sink*-location privacy against attackers with a *global* view. The sink node serves as the aggregating point for data collection and is crucial to assure the availability of a WSN. If the sink node is located and destroyed, the sensed data can no longer be relayed to a data center, rendering the entire WSN useless. Despite the great importance of sink node, the sink-location privacy problem has only been studied under the assumption of resource-constrained attackers [3, 5–8]. When a global adversary is involved, those strategies for resource-constrained attackers become inapplicable. Our work aims to fill in the absence in defending against powerful global adversaries.

To achieve the global view, an attacker can either deploy her own sensors [1, 4] or utilize powerful radio receivers with extremely sensitive antennas to pick up communications across the whole network [1]. As such, a global attacker can derive the location of sinks either by traffic-analysis attacks [5] or packet-tracing attacks [2, 3]. Traffic-analysis attacks utilize the fact that the closer a node is located towards the WSN sink, the higher the number of messages it needs to forward. Thus, moving towards a spot that exhibits a higher message volume can eventually lead the adversaries to find the sink. Packet-tracing attacks lead the adversary toward the travel direction of messages hop by hop till he reaches the sink.

Both traffic-analysis attacks and packet-tracing attacks require no access to the message content but message existence. Additionally, a global adversary can identify *every* node that has forwarded a message instantly, while most literature [4, 9] assumes that an adversary with a *local* view can only identify the sender when communication occurs within his observable range. We are unaware of any solutions that can defend against a global adversary, since it is virtually impossible to protect the network against a global eavesdropper [10]. Any local obfuscation created by fake messages cannot confuse a global adversary. For instance, fractional propagation [5] forks a fake message toward a random destination while the real message is forwarded towards the sink, which is likely to mislead an adversary with a local view. However, such an approach cannot deceive the adversary with a global view, since all real messages always arrive at the sink.

One naive defense strategy is to have each node send the same volume of messages as the sink (including both real and fake messages). However, such strategy imposes high energy consumption and is infeasible. To limit the energy conception while enhancing the privacy against a powerful adversary with a global view, we propose to achieve *k-anonymity* in the network so that at least k entities exhibit the same characteristics as the nodes located close to the sink. As such, they are indistinguishable even to the powerful attackers with regard to contextual communication information.

The concept of achieving k-anonymity [11] was originally proposed to protect personal identity while releasing person-specific data and has been studied extensively in the field of database and data mining. To our best knowledge, our work is the first attempt to apply this concept to preserving sink-location privacy in wireless sensor networks, and there are no other valid approaches dealing with the attacks of a global adversary. We summarize our contribution as follows.

(i) We identify the absence of defense strategies to enhance sink-location privacy against global adversaries.

(ii) To enhance sink-location privacy, we propose to achieve k-anonymity via an Euclidean minimum-spanning tree-based routing protocol, that is, create k designated nodes in the network.

(iii) We show that positioning k designated nodes is complex as it affects two conflicting goals: the routing energy cost and the achievable privacy level, and both goals are determined by a non-analytic function. To strike a balance between those two goals, we formulate the problem of k-anonymity routing protocols as a nonlinearly constrained optimization problem.

(iv) The nonlinearly constrained optimization problem is extremely challenging to solve. To tackle the problem, we design two quasi-optimal algorithms that can obtain the k-node locations closely approximating the optima, and our extensive simulations validate that both algorithms can effectively find solutions hiding the sink among a large number of network nodes.

The rest of the paper is organized as the following. In Section 2, we describe the network model, attack model, and formalize the problem of achieving k-anonymity as a nonlinear optimization problem. We present the routing algorithm for achieving k-anonymity in Section 3. In Section 4, we discuss two approximate algorithms that can obtain quasi-optimal solutions and show our validation effort. Finally, we discuss related work in Section 5 and provide concluding remarks in Section 6.

2. Problem Overview

A wide variety of WSNs have emerged as monitoring and controlling solutions for numerous applications. It is very hard, if even possible, to design a solution applicable to all types of WSNs and to address all attacks. In this section, we specify a popular type of WSNs, which were adopted by several work [12–16]. We formalize the problem below.

2.1. Network Model. We consider a network of wireless sensor nodes that is distributed throughout a bounded environment $Q \subset \mathbb{R}^2$ at positions $n_1, n_2, \ldots$, and we denote

$$\mathcal{N} = \{n_i \mid i \in I\}, \tag{1}$$

where n_i is indexed using an index set I. The network has the following features.

2.1.1. Periodic Data Reporting. WSNs can be classified as event-driven or periodic. In an event-driven sensor network, only those sensors that have observed events will generate

and deliver messages to sinks in a multihop manner while others remain silent. In a periodic network, each sensor will measure the underlying physical phenomena and will deliver its measurements periodically to sinks. We focus on periodic networks since in such networks, even aggregation cannot eliminate the data traffic accumulation towards the sink [9]. Further, we assume that no aggregation algorithms are applied to the networks.

2.1.2. Homogeneous Network with One Sink. We consider homogeneous sensor networks that consist of sinks and a large number of sensor nodes and are densely deployed in a square. Each sensor node is equipped with an omnidirectional antenna and transmits at the same transmission power level. Without loss of generality, we assume that one sink in the network collects data. We note that our scheme can be easily extended to a network with multiple sinks.

2.1.3. No ACK. We assume that the sensor networks do not rely on acknowledgement packets (ACKs) to achieve reliable communication, since the excessive number of ACKs transmitted by the sink will easily reveal its location. We assume that the sink only passively receives messages. Thus, the sink is hidden, and the adversary cannot pinpoint the location of the sink purely by relying on eavesdropping on ACKs.

2.1.4. End-to-End Data Encryption. We assume that messages are protected by an end-to-end encryption protocol using pairwise keys [17]. Due to the limitation of constrained resources, we do not consider the case where the messages are decrypted and re-encrypted at each hop. Therefore, a message exhibits the same cipher as it travels from the source to the sink.

2.2. Attack Model. We consider a powerful attacker who is able to eavesdrop on all communications across the whole network. The adversary does not actively interfere with regular communications in the network but passively eavesdrops on network communications. Her goal is to find the location of the sink and to compromise the sink via physical contacts. Additionally, according to Kerckhoffs' Principal [18], we assume the adversary is aware of all protocols being used but does not know the established keys of the network and is unable to decrypt messages.

To find the sink physically, the adversary will perform a two-phase search: (1) the *location-mining* phase and (2) the *visual searching* phase. In the location-mining phase, the adversary eavesdrops on the network traffic and identifies a set of nodes that appear to be close to the sink. Given the information on nearby nodes, the adversary will find the sink physically in the visual searching phase.

2.2.1. Phase I: Location Mining. Let m_p be the pth message in the network. When m_p is forwarded from its originator n_{p1} to the sink, the attacker will record a set of communication events represented by three tuples: $\{(m_p, n_{pq}, t_{pq}) \mid q = 1, \ldots, h_p\}$, where h_p is the number of hops that m_p has travelled and each three-tuple (m_p, n_{pq}, t_{pq}) maps to an event that the sensor node located at n_{pq} forwards m_p at time t_{pq}. Up to time t, the adversary will obtain the communication event set $M(t) = \{(m_p, n_{pq}, t_{pq}) \mid p \in P,\ q \in H_p,\ \max(t_{pq}) \le t\}$, where P and H_p are index sets of messages and the hop counts for the messages, respectively.

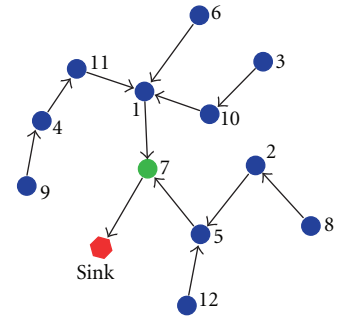


Figure 1: An example of a routing tree rooted at the sink. Each arrow points from a child to a parent.

Given $M(t)$, the adversary will perform statistical analysis on message transmission information. Formally, let $\mathcal{M}$ denote the space of all communication events. We describe the statistical analysis method as a composite function $\pi = \psi \circ \rho$: the function ρ maps the communication events to l traffic statistics associated with every node, and the function ψ selects the set of nodes who have unusual traffic statistics. That is,

$$\begin{aligned} \rho &: \mathcal{M} \longrightarrow \mathbb{R}^{|\mathcal{N}|l}, \\ \psi &: \mathbb{R}^{|\mathcal{N}|l} \longrightarrow 2^{\mathcal{N}}, \end{aligned} \tag{2}$$

where $|\cdot|$ is the cardinality of a set and $2^{\mathcal{N}}$ is the power set of $\mathcal{N}$.

We consider a powerful attacker who is able to perform traffic-analysis attacks and traffic-tracing attacks. Particularly, he is able to obtain two traffic statistics (e.g., $l = 2$): the traffic volume $\rho_v^{n_i}$ and the number of messages $\rho_e^{n_i}$ that end at a node n_i. Assume the attacker starts to record communication events at time $t = 0$, and he can obtain the following statistics at time t:

$$\rho_v^{n_i}(M(t)) = \frac{\left|\left\{\left(m_p, n_{pq}, t_{pq}\right) \mid n_{pq} = n_i, t_{pq} \le t\right\}\right|}{t}, \tag{3}$$

$$\rho_e^{n_i}(M(t)) = \frac{\left|\left\{\left(m_p, n_{ph_p}, t_{ph_p}\right) \mid n_{ph_p} = n_i, t_{ph_p} \le t\right\}\right|}{t}, \tag{4}$$

where h_p is the hop count for the message m_p. Given $\{\rho_v^{n_i}\}$ and $\{\rho_e^{n_i}\}$, the adversary can identify nodes that have either the maximum traffic volume or the maximum number of messages ending here:

$$\psi(\ldots, \rho_v^{n_i}, \rho_e^{n_i}, \ldots) = \left\{\arg\max_{n_i \in \mathcal{N}}(\rho_v^{n_i})\right\} \cup \left\{\arg\max_{n_i \in \mathcal{N}}(\rho_e^{n_i})\right\}. \tag{5}$$

Consider the example depicted in Figure 1, where a tree-based routing protocol is used and a routing tree is formed

with the sink node serving as the root of the tree. After one reporting period t, the adversary will conclude that $\pi(M(t)) = \{n_7\}$, since n_7 transmits 12 messages per period, one for each node.

2.2.2. Phase II: Visual Searching. Although $\pi(M(t))$ only identifies the nodes that are close to the sink and does not pinpoint the sink's location, it does help the adversary to refine the region $\mathcal{S}$ where the sink resides. To find the sink physically, the adversary needs to search $\mathcal{S}$ either visually or using equipment such as a metal detector. Assume the adversary is able to search an area of size v per second and the area of $\mathcal{S}$ is $A_{\mathcal{S}}$, then the amount of time required for the adversary to identify the sink physically is at most $A_{\mathcal{S}}/v$.

Continuing with the example depicted in Figure 1, $\pi(M(t))$ only contains a node n_7. The region $\mathcal{S}$ is the communication range of n_7 with a size A_c. The amount of time required for the adversary to find the sink is at most A_c/v.

2.3. k-Anonymity. Our goal is to design a routing strategy that can enhance sink-location privacy. Essentially, the risk of breaching the sink-location privacy is caused by the observable asymmetric traffic pattern of the sensor networks. The message traffic volume is the largest at the nodes close to the sink, and the travel paths of messages always end there as well. The basic idea of our approach is to change the traffic pattern such that at least k nodes located at $p_1, p_2, \ldots, p_k, \ldots$ may be far away from the sink but behave the same as the nodes around the sink; namely,

$$\pi(M(t)) = \{p_1, \ldots, p_k, \ldots\}. \tag{6}$$

In particular, we envision that each message is delivered to the sink prior to its last-hop transmission, and thus messages no longer end at the nodes around the sink. Further, a lot more nodes send high volumes of messages other than the ones around the sink. As a result, $|\pi(M(t))| \gg 1$.

The main design goal of the k-anonymity routing protocol is to enhance sink-location privacy, and it should also deliver messages without incurring high energy overhead. Therefore, we define a privacy measure and a network efficiency metric to evaluate a routing strategy.

(1) The safety period Φ is the *average* amount of time taken for a global attacker to find the sink physically. We use the safety period Φ to quantify the privacy level. A larger safety period maps to a higher level of sink-location privacy. The safety period Φ includes the amount of time needed for *location mining* and for *visual searching*. Because the duration for *location mining* is fixed and short, we consider the safety period equals the duration of *visual searching*. Since at least k nodes located at $p_1, \ldots, p_k, \ldots$ exhibit the same traffic statistics, the adversary has to visually search all the communication ranges of these nodes. Thus, the safety period is a function of $p_1, \ldots, p_k, \ldots$, denoted by $\Phi(p_1, \ldots, p_k, \ldots)$.

(2) The energy cost E is the average amount of energy consumed for transmitting one message from each sensor to the sink in one measurement period. Since for the routing strategy, the messages delivered to the sink are also transmitted to the nodes $p_1, \ldots, p_k, \ldots$, the energy cost also relates to the positions of these nodes, denoted by $E(p_1, \ldots, p_k, \ldots)$.

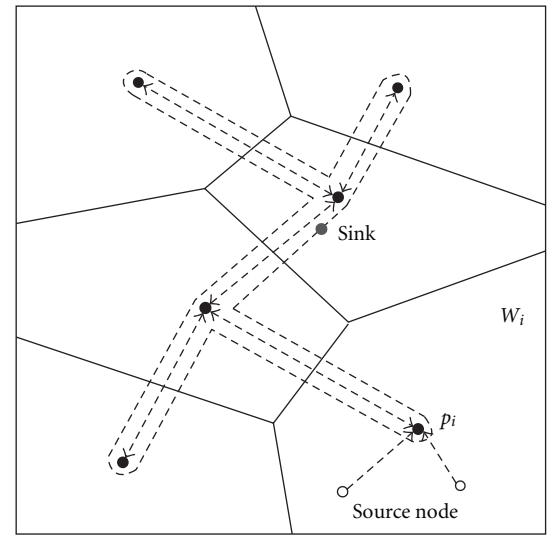


Figure 2: An illustration of the two-stage routing protocol: the intra-region routing and the inter-region routing.

An ideal routing protocol should provide a long safety period Φ at small energy cost E. However, typically a longer safety period requires the messages to be transmitted in a longer way to visit $p_1, \ldots, p_k, \ldots$ and thus imposes a larger energy cost. To find a balance between the safety period Φ and the energy cost E, we define the problem of designing the routing protocol as an optimization problem.

Problem 1.

$$\begin{aligned} \underset{p_1,\ldots,p_k,\ldots}{\text{minimize}} \quad & E(p_1, \ldots, p_k, \ldots) \\ \text{subject to} \quad & \Phi(p_1, \ldots, p_k, \ldots) \geq \Phi_o, \\ & p_i \in \mathcal{N}, \quad i = 1, \ldots, k, \ldots, \end{aligned} \tag{7}$$

where Φ_o is the required safety period.

3. Routing Algorithm Description

3.1. Algorithm Model. In order to achieve k-anonymity, we propose an Euclidean minimum-spanning tree-based (EMST-based) routing algorithm to create at least k nodes whose traffic volumes are equally high. Consider a network deployed in a square Q, as depicted in Figure 2. The EMST-based routing algorithm partitions the square into k non-overlapping sub-regions $W_1, \ldots, W_k$. Denote the partition by $W = \{W_1, \ldots, W_k\}$. In each subregion W_i, a node is chosen to be the *designated node*, which locates at p_i and collects all messages originating from the sub-region W_i.

Each message is forwarded in two stages, *intraregion forwarding* and *interregion forwarding*. During intra-region

forwarding, messages originating from W_i are routed to the designated node p_i through a routing tree rooted at p_i. Once the designated node p_i receives a message generated inside W_i, it starts the inter-region forwarding by sending the message to all other designated nodes through an EMST that connects those nodes. We envision that as the message travels through the EMST, it will reach the sink that is located at most one communication range r away from the EMST. Such an arrangement can be achieved by positioning the sink after the EMST is determined. We note that we adopt an EMST because, by definition, an EMST is a spanning tree with a weight less than or equal to the weight of all other spanning trees.

Interestingly, as a result of constructing an EMST connecting k designated nodes, the number of nodes that exhibit similar traffic statistics as these k designated nodes is larger than k; that is, $|\pi(M(t))| \geq k$. Typically, the distance between any pair of designated nodes p_i and p_j is larger than one communication range, and additional sensor nodes are needed to form a complete EMST for message relaying. As a result, additional nodes are added to $\pi(M(t))$ as a side effect of the proposed two-stage routing. To make the problem model simple yet representative, for the rest of the paper we denote k as the number of partitions, for example, the number of designated nodes, and denote K as the position vector of k designated nodes; that is,

$$K = \{p_1, \ldots, p_k\}, \tag{8}$$

even though the total degree of anonymity is larger than k. The selection of the partition number k is affected by many factors. For instance, a larger k suggests constructing larger number of routing trees rooted at p_j for each region and thus larger overhead, while a smaller k may not meet the requirement of the safe period, Φ_o. As a general rule, the value of k should be small so that it reduces the overhead of constructing multiple routing trees yet satisfies the constraint of Φ_o. We postpone the detailed discussion on the selection of k to Section 4.

3.2. Problem Elaboration. Before updating the problem definition according to two-stage routing, we define the length of the EMST as

$$\texttt{EMST}(K) = \sum_{(i,j)\in \texttt{EMST}} \left\| p_i - p_j \right\|, \tag{9}$$

where (i, j) is the edge that connects p_i and p_j and $\|\cdot\|$ is the Euclidean distance.

According to the two-stage routing protocol, we elaborate the definition of the privacy and network efficiency metrics based on $\texttt{EMST}(K)$ and hop counts

3.2.1. Safety Period Φ Quantified by $\texttt{EMST}(K)$. In one reporting period, the number of messages transmitted by all nodes that are part of the EMST equals the total number of nodes in the network. Therefore, $\pi(M(t))$ contains all nodes belonging to the EMST. To further find the sink physically, the adversary has to search along the EMST. Assume that the adversary can travel at a very high speed when he is not performing visual search such that the time he spends traveling from one location to another can be ignored. Let v denote the adversary's searching speed, and let r be the node communication range. Then as Figure 2 illustrates, the searching time is approximately

$$\Phi(K) = \frac{\texttt{EMST}(K) \times r}{v}. \tag{10}$$

For the rest of the paper, we will use $\texttt{EMST}(K)$ as an indicator for the safety period to avoid possible confusion that might be caused by an inappropriately selected v.

3.2.2. Energy Cost E Quantified by Hop Counts. We define energy cost as the unit of hop counts. Assume the average hop size across the network is λ_h. Then, in a network consisting of uniformly distributed nodes, the average energy cost of routing a message from n_i to a designated node p_j can be approximated by the hop count [7]:

$$e_i \approx \frac{\|n_i - p_j\|}{\lambda_h}. \tag{11}$$

We note that this energy representation is sufficient to model energy spent both at the sending end and at the receiving end, since we can scale up e_i by multiplying by a coefficient α. The coefficient α can include the energy consumed both as the sender transmits the message and as its neighbors overhear and process the message.

The average total energy cost for each sensor node consists of intra-region communication E_a and inter-region communication E_e. Since every sensor node will generate one message per reporting period, the average intra-region energy cost per period per node is

$$E_a(K, W) \approx \frac{1}{\lambda_h |\mathcal{N}|} \sum_{j=1}^{k} \sum_{n_i \in W_j} \left\| n_i - p_j \right\|, \tag{12}$$

and the average inter-region energy cost per period per node is

$$E_e(K) \approx \frac{\texttt{EMST}(K)}{\lambda_h}. \tag{13}$$

Accordingly, the routing optimization problem defined as Problem 1 can be precisely formulated as follows.

Problem 2.

$$\begin{aligned} \underset{K,W}{\text{minimize}} \quad & E = E_a(K, W) + E_e(K) \\ \text{subject to} \quad & \texttt{EMST}(K) \geq \overline{\gamma}, \end{aligned} \tag{14}$$

where $\overline{\gamma} = v\Phi_0/r$ is the threshold value to satisfy the safety period requirement, Φ_0.

3.3. Problem Reduction. Problem 2 defines a non-linear optimization problem that contains two variables: the locations of k designated nodes, that is, K, and the partition W.

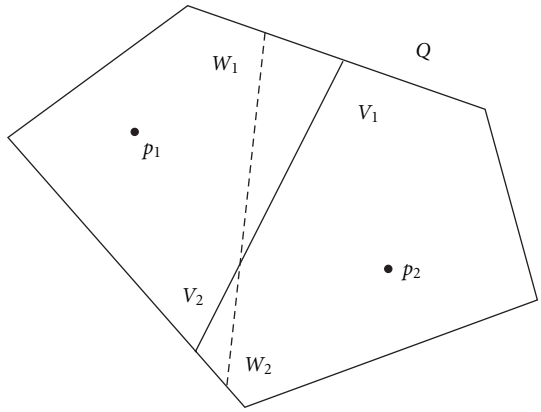


Figure 3: An illustration that the Voronoi partition minimizes E_a.

Solving such a nonlinear optimization problem is difficult. Thus, in this subsection we focus on reducing the problem to a simpler version.

We observe that the locations of k designated nodes will affect the inter-region communication energy cost E_e and the intra-region energy cost E_a while the partition W only affects E_a. Thus, we first examine the principle of the partition W that minimizes $E_a(K, W) + E_e(K)$. Intuitively, knowing the partitioning principle enables us to solve the problem defined in Problem 2 in two steps. (1) Finding the optimal locations of k designated nodes. (2) Applying the optimal partition W to further reduce $E_a(K, W)$.

Next, we present a result showing that, for given locations K, the Voronoi partition is the optimal partition for Problem 2.

Lemma 1. *If (K^*, W^*) is the global optimum that minimizes $E_a(K, W) + E_e(K)$, then W^* is the Voronoi partition $\mathcal{V}(K) = \{V_1, \ldots, V_k\}$, where*

$$V_i = \left\{ n_l \in \mathcal{N} \mid ||n_l - p_i|| \leq \left|\left| n_l - p_j \right|\right|, \forall\, j \neq i \right\}. \quad (15)$$

Proof. We prove the lemma by contradiction. Without loss of generality, we examine the case $k = 2$ as shown in Figure 3, and let p_1 and p_2 be the locations of the two designated nodes. The solid line located in the middle of the network region Q represents the Voronoi partition, and it perpendicularly bisects the line connecting p_1 and p_2. Let $W = \{W_1, W_2\}$ be the optimal partition that minimizes $E_a(K^*, W) + E_e(K^*)$, shown by the dashed line. Then,

$$E_a(K^*, \mathcal{V}(K^*)) > E_a(K^*, W); \quad (16)$$

that is, for $j = \{1, 2\}$,

$$\sum_{n_l \in V_1} \left|\left| n_l - p_j \right|\right| + \sum_{n_l \in V_2} \left|\left| n_l - p_j \right|\right| > \sum_{n_l \in W_1} \left|\left| n_l - p_j \right|\right| + \sum_{n_l \in W_2} \left|\left| n_l - p_j \right|\right|. \quad (17)$$

Let $\mathcal{X}_V^{n_l}$ denote the characteristic of n_l with regard to the set V; that is,

$$\mathcal{X}_V^{n_l} = \begin{cases} 1, & \text{if } n_l \in V, \\ 0, & \text{otherwise.} \end{cases} \quad (18)$$

Then, (17) is equivalent to

$$\sum_{n_l \in Q} \left|\left| n_l - p_j \right|\right| \mathcal{X}_{V_1}^{n_l} + \sum_{n_l \in Q} \left|\left| n_l - p_j \right|\right| \mathcal{X}_{V_2}^{n_l} > \sum_{n_l \in Q} \left|\left| n_l - p_j \right|\right| \mathcal{X}_{W_1}^{n_l} + \sum_{n_l \in Q} \left|\left| n_l - p_j \right|\right| \mathcal{X}_{W_2}^{n_l}. \quad (19)$$

For each $n_l \in Q$, it belongs to one of the following four cases. According to the definition of Voronoi partition, we have

(1) $n_l \in V_1$ and $n_l \in W_1$: $\|n_l - p_1\| \mathcal{X}_{V_1}^{n_l} = \|n_l - p_1\| \mathcal{X}_{W_1}^{n_l}$,
(2) $n_l \in V_1$ and $n_l \in W_2$: $\|n_l - p_1\| \mathcal{X}_{V_1}^{n_l} \leq \|n_l - p_2\| \mathcal{X}_{W_2}^{n_l}$,
(3) $n_l \in V_2$ and $n_l \in W_1$: $\|n_l - p_2\| \mathcal{X}_{V_2}^{n_l} \leq \|n_l - p_2\| \mathcal{X}_{W_1}^{n_l}$,
(4) $n_l \in V_2$ and $n_l \in W_2$: $\|n_l - p_2\| \mathcal{X}_{V_2}^{n_l} = \|n_l - p_2\| \mathcal{X}_{W_2}^{n_l}$.

Combining the above four cases, we have

$$||n_l - p_1|| \mathcal{X}_{V_1}^{n_l} + ||n_l - p_2|| \mathcal{X}_{V_2}^{n_l} \leq ||n_l - p_1|| \mathcal{X}_{W_1}^{n_l} + ||n_l - p_2|| \mathcal{X}_{W_2}^{n_l}, \quad (20)$$

which contradicts to (19). Thus, the optimal partition is the Voronoi partition. □

For the rest of the paper, we will use the following notation:

$$E_{a\mathcal{V}}(K) = E_a(K, \mathcal{V}(K)). \quad (21)$$

Additionally, to reflect the fact that E_e depends on $\texttt{EMST}(K)$, we reform Problem 2 to the following.

Problem 3.

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & E = E_{a\mathcal{V}}(K) + E_e(\texttt{EMST}(K)) \\ \text{subject to} \quad & \texttt{EMST}(K) \geq \overline{\gamma}. \end{aligned} \quad (22)$$

As a result, the sets of variables for the routing optimization problem have been reduced to K, the positions of k designated nodes.

4. Quasi-Optimal Solutions

Solving Problem 3 gives us the optimal solution of k-anonymity, that is, the positions of k designated nodes that minimize the total routing energy and guarantee the safety period requirement. However, solving Problem 3 is challenging. First, Problem 3 is related to the problem of finding a set of k points in a constrained planar region such that its Euclidean minimum spanning tree has the length of a given value. To the best of knowledge, such an problem has not been addressed in the literature so far, and it is unknown

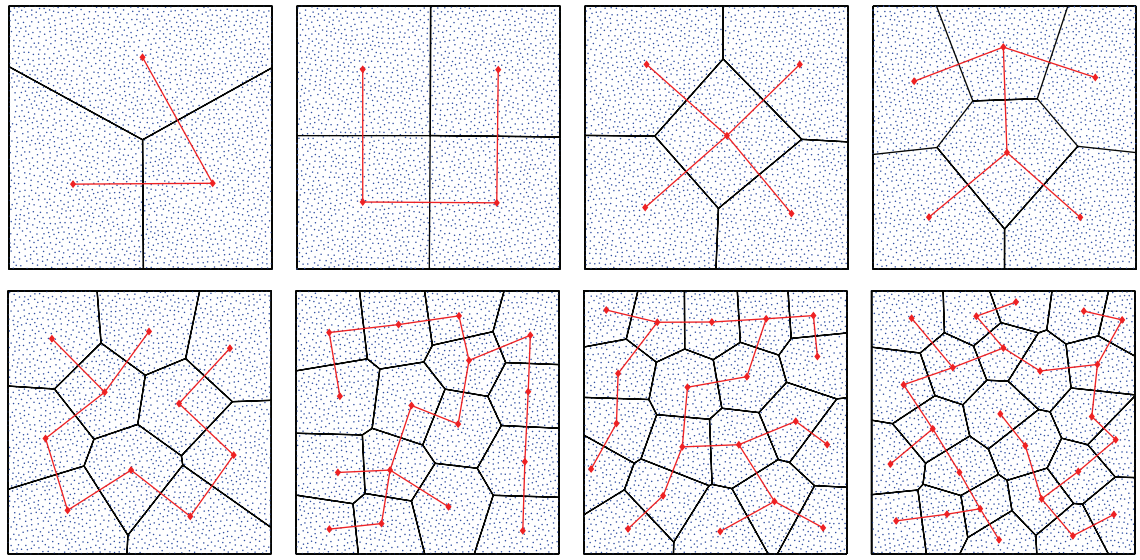


Figure 4: An illustration of the optimal locations K_l^* that minimize $E_{a\mathcal{V}}$, obtained by `GA4(k)`. The tiny dots denote sensor nodes, the black solid lines delimit the Voronoi partition, and the solid red lines denote the EMST for k designated nodes. From left to right, $k = \{3, 4, 5, 6, 10, 16, 20, 25\}$.

Table 1: A summary of notations.

Notations	Explanation	Problem
K_l^*	Global optimum minimizing $E_{a\mathcal{V}}$	4
K^*	Global optimum minimizing $E_{a\mathcal{V}} + E_e$ subject to $\mathtt{EMST}(K) \geq \overline{\gamma}$	3
K_q^*	Quasi-optimum minimizing $E_{a\mathcal{V}}$ subject to $\mathtt{EMST}(K) = \overline{\gamma}$ using GAQO algorithm	5
K_a^*	Quasi-optimum minimizing $E_{a\mathcal{V}}$ subject to $\mathtt{EMST}(K) = \overline{\gamma}$ using APQO algorithm	5

whether the problem is NP hard. Second, our Problem 3 seeks optimized locations for an energy cost function subject to an EMST constraint and thus creates more difficulties.

Popular methods for solving nonlinear optimization problems, such as the generalized reduced gradient [19], are inapplicable to solve Problem 3, because those methods leverage the first or second derivative of the objective function to search for the optimal solution and the derivative of $\mathtt{EMST}(K)$ is complicated to formulate. Searching for the optimal positions of designated nodes through every conceivable value is computationally infeasible. To tackle the problem, we first analyze Problem 3 by finding a K that minimizes $E_{a\mathcal{V}}(K)$ using genetic algorithms (GA) and then propose quasi-optimal algorithms to obtain a solution approximating the optimal one.

To facilitate discussion, we summarize the notation convention of optimal solutions to Problem 3 and its reduced subproblems in Table 1.

4.1. Minimizing $E_{a\mathcal{V}}(K)$. The objective function consists of two components: $E_{a\mathcal{V}}(K)$ and $E_e(K)$, and we start by searching for a K that minimizes the first component $E_{a\mathcal{V}}(K)$, namely, solving the following problem:

Problem 4.

$$\underset{K}{\text{minimize}}\, E_{a\mathcal{V}}(K) \tag{23}$$

Problem 4 is still a nonlinear optimization problem with an objective function whose derivative is difficult to calculate. We choose to exploit the widely adopted genetic algorithms (GAs) to find the optimal solution. GA mimics Darwin's theory about evolution. It iteratively generates a set of solutions known as a population and selects a subset of solutions to form a new population based on each solution's "fitness." The fitness level of a solution can be evaluated using the objective function of the optimization problem. "Fitter" solutions will be selected with higher probability while "weaker" solutions will still have chances to be selected. As a result, GA is likely to escape from local optima and evolves to the global optima with high probability. Thus, we call the solutions obtained by GA as optimal solutions.

We call our customized genetic algorithm that searches for optimal solutions of Problem 4 as `GA4(k)`, and we built our `GA4(k)` using Matlab toolbox GAtool and searched for optimal designated node locations in a 2500-node network that is deployed in a 1000 m $\times$ 1000 m square with a uniform density. The node communication range r was set to 40 m, which resulted in an average hop size λ_h of $(2/3) \times$ 40 m. We constructed the "chromosome" as K, that is, k coordinates of designated nodes and performed multiple runs of experiments while changing the value of k. For each k, we ran the experiments about 10 times, and we set the population size to approximately $k \times 100$, the crossover fraction to 0.8, and the maximum number of generations to 100. Figure 4 shows the typical patterns for optimal

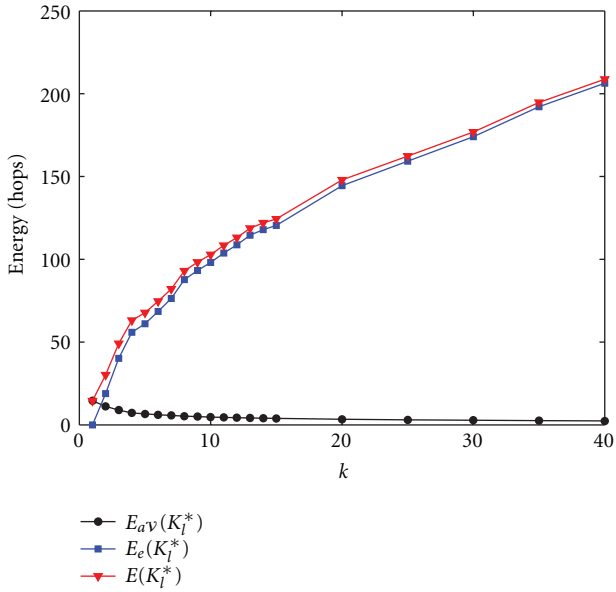


Figure 5: The routing efficiency measure: the intra-region energy $E_{a'v}(K_l^*)$, inter-region energy $E_e(K_l^*)$, and total energy $E(K_l^*)$ with regard to k. K_l^* is the optimal locations that minimize $E_{a'v}(K)$ obtained via `GA4(k)`, and $E(K_l^*) = E_{a'v}(K_l^*) + E_e(K_l^*)$. This plot shows that $E_e(K_l^*)$ dominates $E(K_l^*)$.

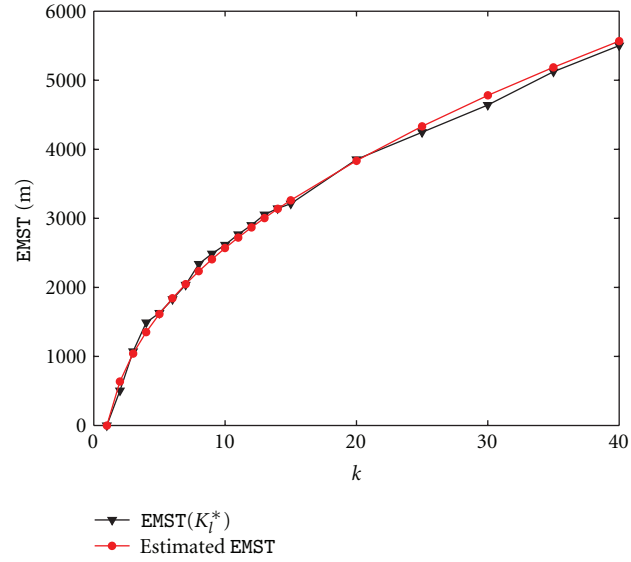


Figure 6: The privacy measure: a comparison of estimated EMST and $\mathtt{EMST}(K_l^*)$ with respect to k. The estimated EMST is calculated according to a regression formula (27), which turns out to be a close fit to the empirical one obtained using `GA4(k)`.

designated nodes' positions K that minimize $E_{a'v}(K)$ and the corresponding $\mathtt{EMST}(K)$, when $k = \{3, 4, 5, 6, 10, 16, 20, 25\}$.

Remark 2. From Figure 4, we observe that for each optimal layout the designated nodes are distributed almost uniformly across the network, and the network area Q is partitioned into regions with similar sizes. This observation can be intuitively explained by rewriting (12) as

$$E_{a'v}(K) = \frac{\overline{d}}{\lambda_h}, \tag{24}$$

where $\overline{d}$ is the average distance between every sensor node and its nearest designated node. To minimize $E_{a'v}(K)$ the designated nodes have to be deployed in such a way that $\overline{d}$ is minimized.

Remark 3. We depict $E_{a'v}(K_l^*)$, $E_e(K_l^*)$, and $E(K_l^*)$ in Figure 5 and $\mathtt{EMST}(K_l^*)$ in Figure 6, which show that both $E_e(K_l^*)$ and $\mathtt{EMST}(K_l^*)$ increase with k while $E_{a'v}(K_l^*)$ decreases with k. Intuitively, when the number of partitioned regions increases, the average distance between a sensor node and its nearest designated node $\overline{d}$ decreases and so does $E_{a'v}(K_l^*)$. However, the increase of k causes the designated nodes to further spread out and thus increases $\mathtt{EMST}(K_l^*)$. A slight change of $\mathtt{EMST}(K_l^*)$ will cause a larger level of ΔE_e than $\Delta E_{a'v}(K)$, because $\Delta\mathtt{EMST}(K)$ creates an equivalent level of ΔE_e while amortized among all nodes with regard to $E_{a'v}(K)$. Thus, we observe that as k increases, E_e grows quickly, and soon $E_e(K_l^*) \gg E_{a'v}(K_l^*)$.

To estimate the relationship between $\mathtt{EMST}(K_l^*)$ and k, we performed a regression analysis on the empirical results of $\mathtt{EMST}(K_l^*)$ and k. Rather than choosing a polynomial, we construct the regression function according to Remark 2; that is, the network area Q is very likely to be partitioned into regions of similar sizes and the distances between every two neighboring designated nodes (two designated nodes that are connected by an edge in the EMST) are roughly the same. Let $\overline{r}_w$ be the average distance between neighboring designated nodes. Then

$$\mathtt{EMST}\left(K_l^*\right) = (k-1)\overline{r}_w. \tag{25}$$

Additionally, we can use a disk with radius $\overline{r}_w/2$ to approximate the area of each region, and

$$k\pi\left(\frac{\overline{r}_w}{2}\right)^2 = A_Q \times \beta, \tag{26}$$

where A_Q is the area of the square Q and $0 < \beta < 1$ is a coefficient describing how close the disk approximates each region on average. Thus, the length of $\mathtt{EMST}(K_l^*)$ can be estimated by the following equation:

$$\mathtt{EMST}\left(K_l^*\right) = 2(k-1)\sqrt{\frac{\beta A_Q}{k\pi}}. \tag{27}$$

Our regression analysis showed that the fitting error is minimized when $\beta = 0.64$. As shown in Figure 6, the comparison between the estimated $\mathtt{EMST}(K_l^*)$ with $\beta = 0.64$ and the empirical one obtained by GA show that the regression line is a close fit.

4.2. GA-Based Quasi-Optimal Algorithm. Analyzing Problem 4 utilizing GA provides important insights towards solving the original routing optimization problem defined in Problem 3. In this subsection, we introduce a GA-based quasi-optimal algorithm (GAQO) that can obtain an approximate

optimal solution for Problem 3. In particular, the GAQO algorithm provides the quasi-optimal solution K_q^* to the following problem:

Problem 5.

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & E_{a'v}(K) \\ \text{subject to} \quad & \texttt{EMST}(K) = \overline{\gamma}. \end{aligned} \tag{28}$$

We will show that the quasi-optimal solutions for Problem 5 closely approximate the solutions for Problem 3 empirically. Intuitively, according to Remark 3, a slight change of $\texttt{EMST}(K)$ will cause a larger level of increase of E_e than decrease of $E_{a'v}(K)$. Thus, our approach is to minimize $E_e(K)$ as much as possible. Note that $E_e(K)$ achieves its minimum when $\texttt{EMST}(K) = \overline{\gamma}$. Thus, ensuring that $\texttt{EMST}(K_q^*) = \overline{\gamma}$ will produce a solution approximating the optimal solution for Problem 3.

4.2.1. Approximation Evaluation Metric. To evaluate how close the solutions obtained by the GAQO algorithm approximates the optima, we define the approximation evaluation metric μ as the energy difference between $E(K_q^*)$ and $E(K^*)$;

$$\mu = E\left(K_q^*\right) - E(K^*). \tag{29}$$

We will show that μ is bounded by the difference between the intra-region energy $E_{a'v}$ of K_q^* and K_l^*:

$$\mu \le E_{a'v}\left(K_q^*\right) - E_{a'v}\left(K_l^*\right). \tag{30}$$

We now justify (39) by proving the following lemma.

Lemma 4. $E_{a'v}(K_l^*) + E_e(K_q^*) \le E_{a'v}(K^*) + E_e(K^*) \le E_{a'v}(K_q^*) + E_e(K_q^*)$.

Proof. (Second inequality.) By definition, for a given k, K^* is the global optimum which minimizes $E_{a'v}(K) + E_e(K)$, so $E_{a'v}(K^*) + E_e(K^*) \le E_{a'v}(K_q^*) + E_e(K_q^*)$.

(First inequality.) For a given k, K_l^* minimizes $E_{a'v}(K)$. Thus, $E_{a'v}(K_l^*) \le E_{a'v}(K^*)$. Additionally, by definition, $\texttt{EMST}(K_q^*) = \overline{\gamma}$ and $\texttt{EMST}(K^*) \ge \overline{\gamma}$. Thus,

$$E_e\left(K_q^*\right) \le E_e(K^*). \tag{31}$$

Combining both facts, we conclude that $E_{a'v}(K_l^*) + E_e(K_q^*) \le E_{a'v}(K^*) + E_e(K^*)$. Therefore, the lemma is proved. □

4.2.2. Algorithm Walk-Through. Searching optimum K_l^* for Problem 4 using GA has provided insights of K^*(We did not apply GA to solve Problem 5, because the constraint of $\texttt{EMST}(K) = \overline{\gamma}$ makes it prohibitively time consuming to obtain a feasible solution.). In particular, for a given k, if the required $\overline{\gamma}$ happens to equal $\texttt{EMST}(K_l^*)$, then K_l^* is the global optimum for Problem 3 that is, $K^* = K_l^*$. We take the hypothesis that optimal solutions for different threshold values $\overline{\gamma}$ are continuous and design our GA-based quasi-optimal (GAQO) algorithm with steps shown in Algorithm 1:

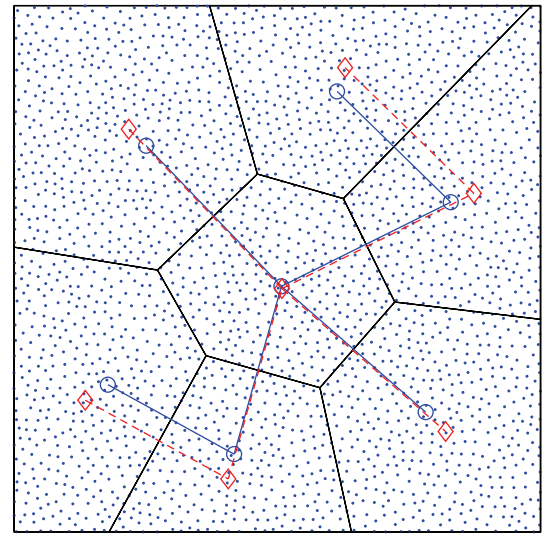


Figure 7: The red "◇" points are the optimal designated node locations that minimize $E_{a'v}(K_l^*)$ for $k = 7$, derived via `GA4(k)`, and the blue "◦" points are the quasi-optimal result derived with our GAQO algorithm.

Require:INPUT:
$\overline{\gamma}$;
OUTPUT:
K_q^*;
(1) **PROCEDURES**:
$k = \texttt{Closest\_EMST}(\overline{\gamma})$
(2) $K_l^* = \texttt{GA4}(k)$
(3) $\alpha = \overline{\gamma}/\texttt{EMST}(K_l^*)$
(4) $K_q^* = \alpha K_l^*$

Algorithm 1: GA-based quasi-optimal algorithm for the k-anonymity sink-location privacy problem.

Step 1. Call `Closest_EMST` to find k whose $\texttt{EMST}(K_l^*)$ is closest to the given $\overline{\gamma}$, according to (27).

Step 2. For the given k, find an optimal layout K_l^* for Problem 4 using genetic algorithm `GA4(k)`.

Step 3. Shrink or expand K_l^* with regard to the center of the network area Q until $\texttt{EMST}(K_q^*) = \overline{\gamma}$. Let the center of Q be the origin of the coordinate, and let $\alpha = \overline{\gamma}/\texttt{EMST}(K_l^*)$. Then $K_q^* = \alpha K_l^*$.

We note that the aforementioned GAQO algorithm, though not optimal, does approximate optimal solutions.

Example 5. Here, we illustrate how the GAQO algorithm achieves k-anonymity for a given safety period $\overline{\gamma}$ in Figure 7. We use the same parameters of the sensor network described in Section 4.1 and set the required safety period $\overline{\gamma} = 2000$ m. In the first step, based on (27), GAQO concluded that the closest $\texttt{EMST}(K_l^*) = 2035.76$ m when $k = 7$. Then, GAQO utilized the genetic algorithm `GA4(k)` to search for

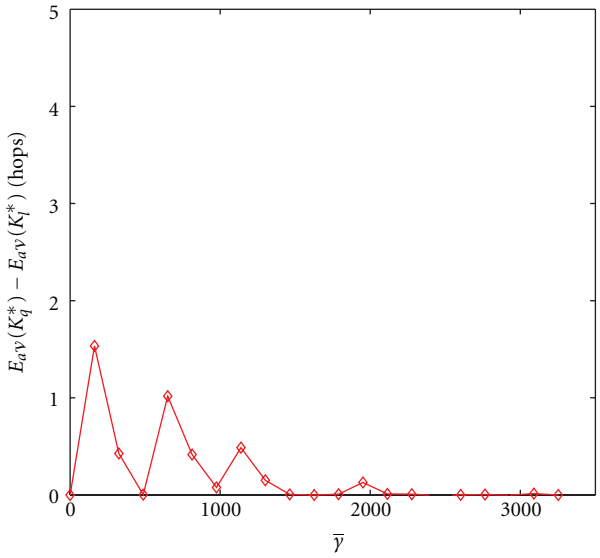


FIGURE 8: The algorithm approximation measure: the upper bound of the difference between the quasi-optimum (using the GAQO algorithm) and the global optimum (using `GA4(k)`), as $\overline{\gamma}$ varies.

the optimal positioning of 7 designated nodes. An example layout of K_l^* when $k = 7$ is denoted by the red "$\diamond$" points in Figure 7. Since $\texttt{EMST}(K_l^*) > \overline{\gamma}$, GAQO shrank K_l^* to the quasi-optimal layout of the designated nodes K_q^*, as marked by blue "$\circ$" points in Figure 7.

4.2.3. Evaluation. To evaluate how close the solutions obtained by the GAQO algorithm approximate the optimal solution, we performed an empirical study. In particular, we used the same network setup as before and searched for the quasi-optimal solutions in a 2500-node network deployed in the 1000×1000 m square. We changed the constraint of Problem 5 by varying the length of $\texttt{EMST}(K)$. To capture the statistical character of GAQO, for each $\texttt{EMST}(K)$ value, we ran the algorithm at least 10 times over randomly generated network topologies, and calculated the upper bound of the difference between the quasi-optimal solution and global optimal solution, that is, $E_{av}(K_q^*) - E_{av}(K_l^*)$. The plot in Figure 8 has confirmed that for the quasi-optimal solution obtained by the GAQO algorithm, K_q^* approaches K^* as $\overline{\gamma}$ increases.

4.3. Artificial Potential-Based Quasi-Optimal Algorithm. The GAQO algorithm can obtain quasi-optimal solutions of the k-anonymity sink-location problem. However, our simulation study shows that the run time of `GA4(k)`, that is, the algorithm that searches for K_l^* that minimizes $E_{av}(K)$ using genetic algorithms, increases quadratically as the constraint $\overline{\gamma}$ increases. To efficiently solve the k-anonymity sink-location problem, we design an artificial potential-based algorithm named `AP4(k)` to substitute `GA4(k)`, and we call the new quasi-optimal algorithm leveraging `AP4(k)` an APQO algorithm.

Artificial potential (AP) [20] (aka. artificial physics in some literature as opposed to natural physics) was originally developed for the purpose of obstacle avoidance. Later, it was used as a distributed control strategy to solve self-deployment problems of WSNs. The approach is simple enough to let each entity exert forces on other nearby entities and respond to forces from them; yet a uniform distribution will eventually emerge. Since the approach is largely independent of the number of entities, it scales well for large sets of entities. We take advantage of the linear time complexity of an AP-based method to solve the k-anonymity sink-location problem, since searching for optimal solutions of k designated nodes is equivalent to deploying nodes uniformly across the network (according to Remark 2).

We built our APQO algorithm on the AP-based self-deployment algorithm proposed by Ding et al. [21], whereby sensors are deployed into uniform lattices inside a bounded region. We start by assuming the k designated nodes can move to any position inside the network area and we denote $\mathbf{z} = [z_1^T, z_2^T, \ldots, z_k^T]^T$ the aggregate position vector of k mobile nodes. Once the AP-based algorithm converges and finds the final position $\mathbf{z}^*$, we select those sensor nodes that are closest to $\mathbf{z}^*$ to be the designated nodes.

4.3.1. AP Definition. Two types of artificial potential functions are defined for every node i: V_{ij}^1, which is the potential between node i and another node j ($j \neq i$), and V_{is}^2, which is the potential between node i and the boundary. The artificial potential has the following characteristics. When node i is located close to another node j or to the boundary, the potential is high and has a tendency to push node i away. When node i is very far away from another node or the boundary, the potential reduces to zero. V_{ij}^1 is defined as

$$V_{ij}^1 = \begin{cases} \left(l_{ij} - r_e\right)^2 + \dfrac{1}{l_{ij}^2}, & 0 < l_{ij} \leq r_e, \\ 0, & \text{else}, \end{cases} \tag{32}$$

where $l_{ij} = ||z_i - z_j||$ is the distance between these two mobile nodes and r_e is the effective radius of the potential.

We define V_{is}^2 as the potential between mobile node i and the nearest point on the boundary $q_s \in N_i$, where N_i is the set of all the nearest points, and $N_i = \{q \mid \arg\min_{q \in B} \|z_i - q\|\}$, B being the set of all points on the boundary. We note that N_i may not be a singleton. For example, when the z_i is on the diagonal of the square, there exist two nearest points with each on one edge of the square. V_{is}^2 is defined as

$$V_{is}^2 = \begin{cases} \left(l_{is} - r_e'\right)^2 + \dfrac{1}{l_{is}^2}, & 0 < l_{is} \leq r_e', \\ 0, & \text{else}, \end{cases} \tag{33}$$

where $l_{is} = \|z_i - q_s\|$ and r_e' is the effective radius of the boundary potential. Here we set $r_e' = r_e/2$.

The relationships between V_{ij}^1 and the distance of l_{ij} and between V_{is}^2 and l_{is} are depicted in Figure 9, which exhibit desired characteristics.

In addition, we define the total potential as

$$V(\mathbf{z}) = \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{k} V_{ij}^1 + \sum_{i=1}^{k}\sum_{q_s \in N_i} V_{is}^2. \tag{34}$$

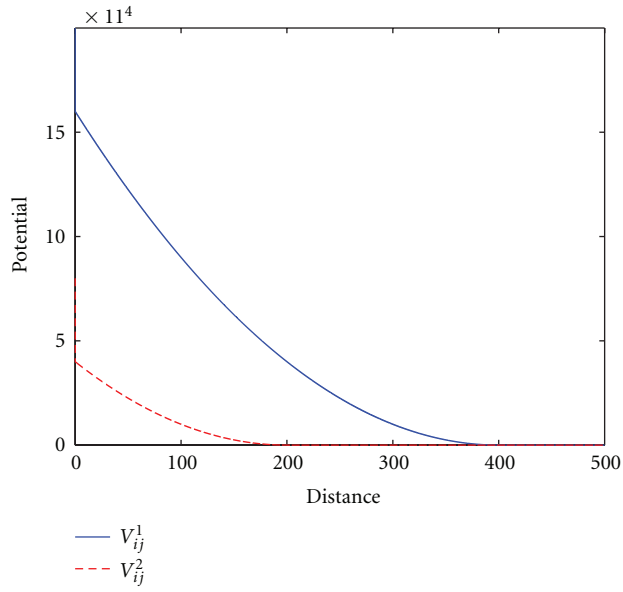


FIGURE 9: V_{ij}^1 and V_{is}^2 with regard to l_{ij} and l_{is} when $r_e = 400$.

To distribute k nodes approximately uniformly inside the network area is equivalent to finding $\mathbf{z}$ that minimize V:

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} V(\mathbf{z}). \tag{35}$$

We consider the gradient descent method to find the minimum for $V(\mathbf{z})$ and define the following position update scheme for mobile node i:

$$\dot{z}_i = -\frac{\partial V}{\partial z_i} = -\left(\sum_{j=1}^{k} \frac{\partial V_{ij}^1}{\partial z_i} + \sum_{q_s \in N_i} \frac{\partial V_{is}^2}{\partial z_i}\right), \tag{36}$$

that is, we let the mobile nodes move towards the negative of the gradient to minimize the total potential V.

4.3.2. Algorithm Walk-Through. Overall, the APQO follows the similar framework as shown in Algorithm 1. For a given $\overline{y}$, the function `Cloest_EMST()` returns k whose corresponding `EMST`(K_l^*) is closest to $\overline{y}$, according to the line fitting equation (27). Different from GAQO, APQO utilizes the AP-based function `AP4(k)` to find the quasi-optimal layout K_a that minimizes $E_{a\mathcal{V}}(K)$. Similar to GAQO, APQO also shrinks or expands K_a with regard to the center of Q until `EMST`$(K_a^*) = \overline{y}$; that is, $K_a^* = (\overline{y}/\texttt{EMST}(K_a))K_a$.

We listed the pseudocode of `AP4(k)` in Algorithm 2, which contains the following steps.

Step 1. Initialize the locations of the k nodes $\mathbf{z}$ to be around the center of the network square Q without overlapping.

Step 2. Obtain the gradients $\dot{\mathbf{z}}$, and update the location vector $\mathbf{z}$ according to the gradients $\dot{\mathbf{z}}$ and the step size Δ (a small constant we choose) iteratively until convergence. Denote the converged position as $\mathbf{z}^*$.

Require: INPUT:
k;
OUTPUT:
K_a;
(1) **PROCEDURES**:
$\mathbf{z}(0) =$ `Initialize_z(k)`;
(2) **repeat**
(3) $\mathbf{z}(n\Delta) = \mathbf{z}((n-1)\Delta) + \Delta \cdot \dot{\mathbf{z}}((n-1)\Delta)$;
(4) Error $= \|\mathbf{z}(n\Delta) - \mathbf{z}((n-1)\Delta)\|$;
(5) **Until** Error < Error_Threshold
(6) K_a = Closest_nodes($\mathbf{z}(n\Delta)$)

ALGORITHM 2: `AP4(k)`: AP-based method for solving Problem 4.

Step 3. Select the sensor nodes that are closest to $\mathbf{z}^*$ to be the designated nodes, and we call their positions as K_a.

We use the following lemma to show that the `AP4(k)` algorithm must converge.

Lemma 6. *The AP-based algorithm is convergent; that is, $z_i(t)$ asymptotically approaches the location where $\dot{z}_i = -\partial V/\partial z_i = 0$.*

Proof. Taking the derivative of V, we obtain

$$\begin{aligned} \dot{V} &= \left[\frac{\partial V}{\partial z_1}, \frac{\partial V}{\partial z_2}, \ldots, \frac{\partial V}{\partial z_k}\right]\dot{\mathbf{z}} \\ &= -\dot{\mathbf{z}}^T\dot{\mathbf{z}} = -\|\dot{\mathbf{z}}\|^2 \le 0. \end{aligned} \tag{37}$$

Therefore, $V(\mathbf{z}(t)) \le V(\mathbf{z}(0)) < \infty$ and $V(\mathbf{z}(t))$ is bounded for $t \ge 0$. Further, note from (33) that V tends to ∞ if l_{is} approaches 0. Thus, the boundedness of $V(\mathbf{z}(t))$ implies that l_{is} will never become 0 and $z_i(t)$ remains inside the network region Q all the time.

Let $\Omega = \{\mathbf{z} \in Q^k \mid V(\mathbf{z}(t)) \le V(\mathbf{z}(0))\}$. Then by LaSalle's invariance principle [22], the trajectory $\mathbf{z}(t)$ converges to the largest invariant set in $\mathcal{M} = \{\mathbf{z} \in \Omega \mid \dot{V} = -\|\dot{\mathbf{z}}\|^2 = 0\}$, which completes the proof. □

4.3.3. Evaluation. Similar to the GAQO algorithm, we have defined an approximation evaluation metric

$$\mu = E(K_a^*) - E(K^*), \tag{38}$$

and μ is bounded by the difference between the intra-region energy $E_{a\mathcal{V}}$ of K_a^* and K_l^*:

$$\mu \le E_{a\mathcal{V}}(K_a^*) - E_{a\mathcal{V}}\left(K_l^*\right). \tag{39}$$

To evaluate the APQO algorithm, we performed an empirical study using the same network setup as before: a 2500-node network deployed in the 1000 × 1000 m square. Figure 10 shows the result, and for the quasi-optimal solution obtained by the APQO algorithm, K_a^* approaches K^* as $\overline{y}$ increases. Additionally, the steady-state locations of the k designated nodes K_a, obtained by `AP4(k)`, are affected by the value of r_e.

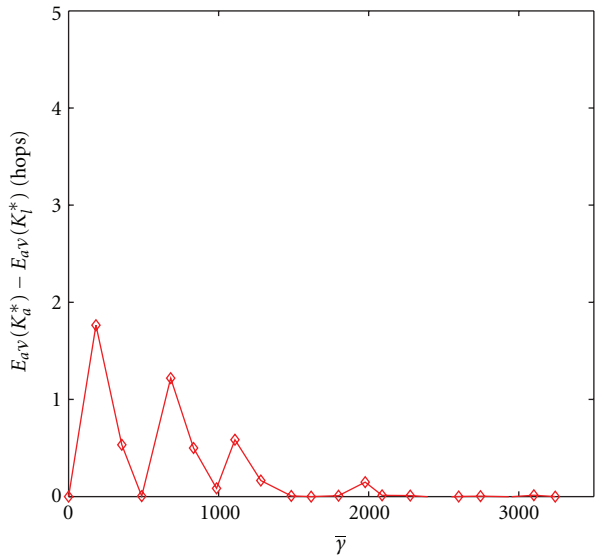


FIGURE 10: The algorithm approximation measure: the upper bound of the difference between the quasi optimum (using the APQO algorithm) and the global optimum (using GA4(k)), as $\overline{y}$ varies.

If r_e is small and k disks (with a radius of $r_e/2$) are not enough to fill the region Q, then in the steady state, each designated node is at least r_e away from its nearest designated node [23]. In comparison, if r_e is large and k disks are more than enough to fill the region, the distances from any pairs of nearest designated nodes in the steady state are less than r_e. For a given k, to ensure that the length of the EMST(K_a) obtained by AP4(k) is similar to the one obtained by GA4(k), we set r_e to $\overline{r}_w$, the average distance between neighboring designated nodes obtained by empirical equation (27). Additionally, we adopted the same setups as the one for the GAQO algorithm evaluation and used the same topologies to evaluate the APQO algorithm.

Performance Comparison. The length of EMSTs obtained using GA4(k) and AP4(k) is presented in Figure 11(a), and the locations K_a derived by AP4(k) for various k are demonstrated in Figure 12. We note that the resulting EMSTs shown in Figure 12 appear slightly different from the ones that are obtained via GA4(k) (shown in Figure 4). This is because k designated nodes are scattered roughly evenly across the network and a slight variation of their locations will cause the EMST to go through edges connecting different pairs of nodes. However, the numerical results of EMST length show that the AP-based AP4(k) algorithm can acquire EMSTs of similar length as the ones derived by GA4(k). Further, as shown in Figure 11(b), for a given $\overline{y}$, the total energy levels obtained by the APQO algorithm fit closely with what the GAQO algorithm derives, which indicates that the APQO algorithm can also obtain quasi-optimal solutions for Problem 3.

Time Complexity Comparison. Since the majority of the run-time for the GAQO and APQO algorithms is contributed by executing GA4(k) and AP4(k), we measure the run-time of GA4(k) and GA4(k) only. We tested both GA4(k) and AP4(k) on a computer equipped with a 2.1 GHz AMD dual-core CPU and 3 GB RAM and depicted the run-time of these two algorithms when varying k in Figure 11(c). Figure 11(c) shows that the run-time of GA4(k) increases quickly as k increases while the run-time of AP4(k) remains short. This is because the time complexity for GA4(k) is $O(nk^2)$, where n is the total number of nodes in the network, and the time complexity of AP4(k) is $O(k)$.

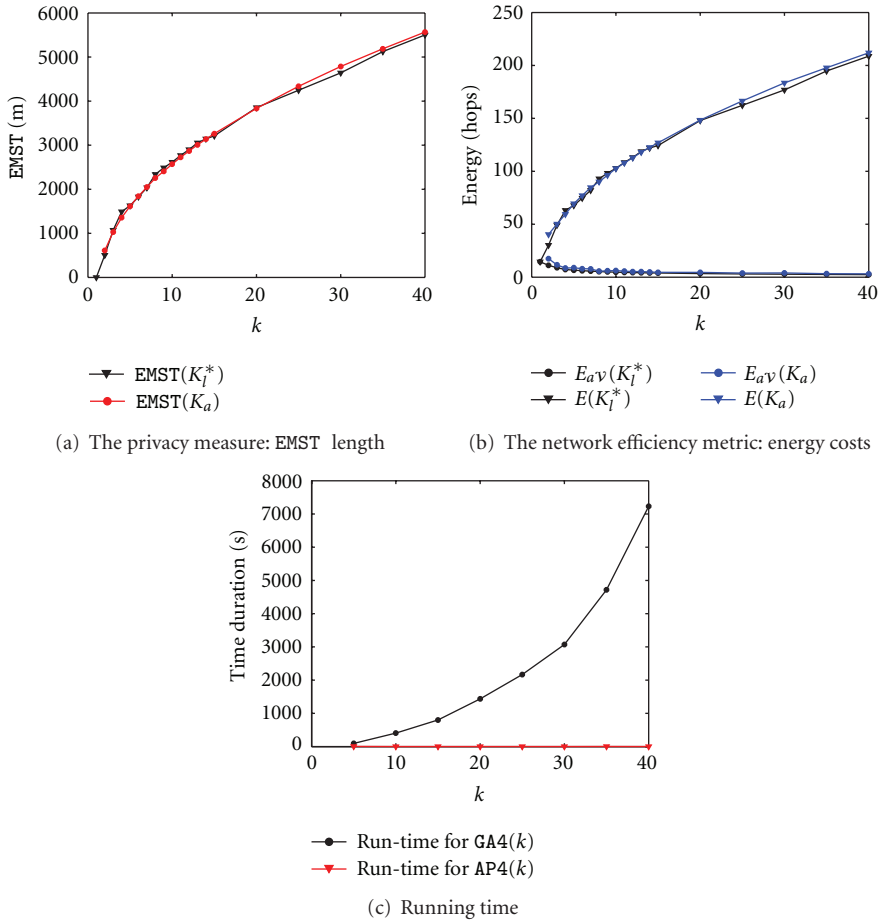
GA4(k) involves calculating multiple generations, and each generation has a population size of $k \times 100$. Computing the fitness function $E_{aV}(K)$ for each individual requires calculating the distance between k designated nodes and all n network nodes. Considering that the maximum number of generations is at most 1000 in our simulation, the time complexity of GA4(k) is $O(nk^2)$. In comparison, each iteration of AP4(k) only involves updating k locations z_i. Since the total number of iteration is independent of the number k, the time complexity of AP is $O(k)$. In our simulation, AP4(k) converged around 1s to 5s. Thus, APQO performs better than GAQO as the number of nodes in the network increases.

k-Anonymity Evaluation. We evaluated how effective the EMST-based routing protocol can change the traffic pattern around the sink. Let the node that is closest to the sink be n_{cs}. We are interested in the number of nodes exhibiting the same traffic statistics as n_{cs}. Denote N_{ρ_v} as the number of nodes whose traffic volumes $\rho_v^{n_i}$ (3) are the same as that of n_{cs}, and denote N_{ρ_e} as the number of nodes which has the same number of messages ended there $\rho_e^{n_i}$ (4) as n_{cs}. Figure 13 shows the trend of N_{ρ_v} and N_{ρ_e} when $\overline{y}$ and k increase. It indicates that the EMST-based two-stage routing algorithm can effectively hide the location of the sink. Almost all nodes in the network appear to have the same $\rho_e^{n_i}$ as that of n_{cs}, and a lot more network nodes other than k designated nodes forward the same amount of traffic as n_{cs}.

5. Related Work

Protecting the identity of traffic sources has been extensively studied in the context of general networks, where the usage of a series of intermediate mixes and onion routing [24] was proposed to cope with traffic analysis. The problems of tracking users' paths in wireless networks with location-oriented services were studied by Gruteser and Grunwald [25] and Hoh and Gruteser [26], and they proposed a path perturbation algorithm to increase source location anonymity. Since sensor networks have constrained resources, those methods are not applicable there.

In the context of wireless sensor networks, both source-location privacy and sink-location privacy have attracted attention from the research community. Source location privacy focuses on protecting the message source, since such information can reveal sensitive position information of the target that is close to the message source. Preserving source-location privacy against a local adversary was first studied by Kamat et al. [2], where fake message injection and phantom



(a) The privacy measure: EMST length

(b) The network efficiency metric: energy costs

(c) Running time

FIGURE 11: Comparison of GA4(k) and AP4(k). For both methods, we used the same network setup and searched for the quasi-optimal solutions in a 2500-node network deployed in the 1000 × 1000 m square.

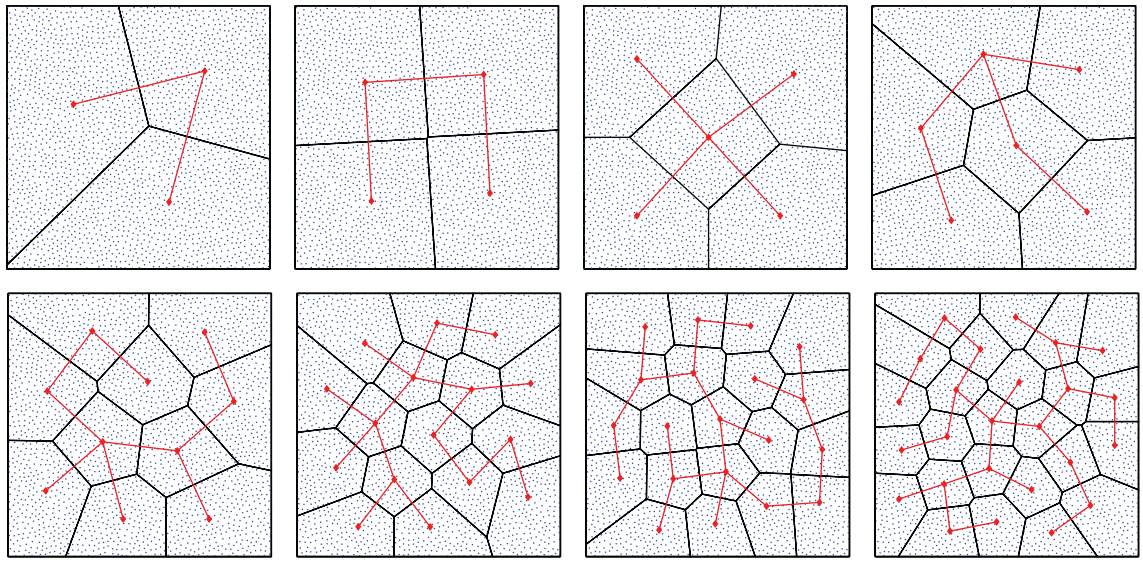
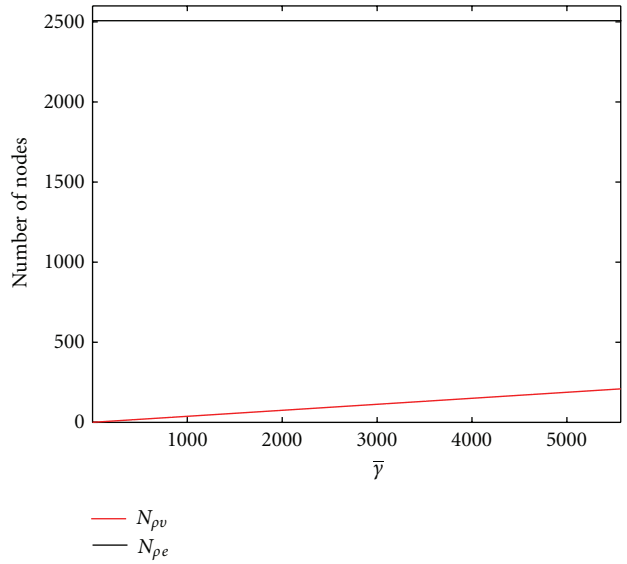
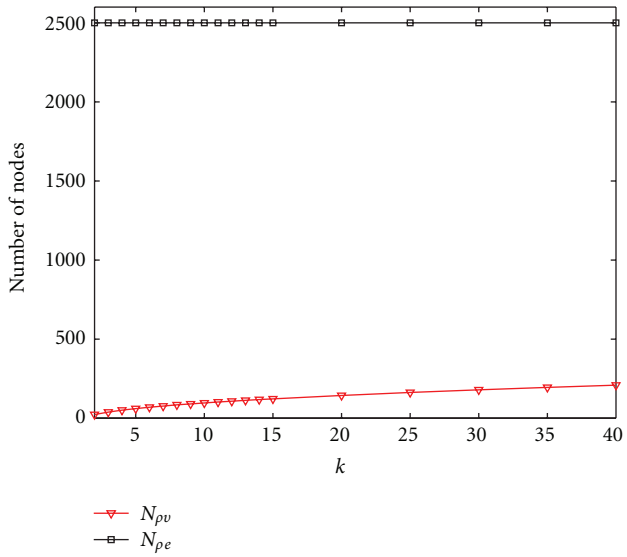


FIGURE 12: The quasi-optimal locations of k designated nodes which approximately minimize $E_{a'v}$, derived via AP4(k). From left to right and top to down $k = \{3, 4, 5, 6, 10, 16, 20, 25\}$.



(a) The safety period



(b) The number of designated nodes

FIGURE 13: The number of nodes that exhibit the same traffic statistics as the nodes around the sink. N_{p_v} is the number of nodes which has the same traffic volume, and N_{p_e} is the number of nodes which has the same number of messages ended there.

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 [1, 4, 27, 28]. Mehta et al. [4] have proposed periodic collection and source simulation techniques to prevent the leakage of message source location, and Yang et al. [1] have introduced dummy traffic to hide the real message source. Ouyang et al. [27] have devised a set of privacy-preserving algorithms involving sending periodic maintainable messages to address a laptop-class attacker who has longer radio range and can eavesdrop on all communications in a sensor network. A notion of statistically strong source anonymity is proposed by Shao et al. [28], and a strategy called FitProbRate has been proposed to achieve statistically strong source anonymity with a reduced real event report latency.

In the areas of enhancing sink-location privacy, Deng et al. [9] have shown that traffic analysis can reveal the location of sinks and proposed several antitraffic analysis countermeasures to hide the direction of data flow and create fake sink locations that exhibit artificially high traffic. In their follow-up work [5], multiple parent routing, controlled random walk, random fake paths, and combinations of all three routing algorithms have been studied to generate randomness against traffic rate monitoring and traffic path direction attacks. Location privacy routing (LPR) [3] utilizes probabilistic routing and fake message injection to deceive an adversary from tracking the direction of traffic flow. Conner et al. [29] proposed the decoy sink protocol, whereby data are forwarded to a decoy sink for aggregation before they 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. Liu and Xu [7] presented a zeroing-in attack that can be launched by resource constraint adversaries and proposed a random walk-based defense strategy. Gu et al. [6] proposed a privacy-preserving scheme which obfuscates the sink's location with dummy sink nodes and can help secure existing mobility control protocols against attacks. However, those strategies cannot cope with a global adversary.

To deal with global adversaries, Ngai [8] proposed randomized routing with hidden address (RRHA), whereby packets are routed from the source to the sink along a random path and the destination field is not included in the header of the packets. Such a routing protocol does provide sink anonymity, but the packet may not reach the sink at all. Additionally, Nezhad et al. [10] designed an anonymous routing protocol to preserve the sink-location privacy against a global adversary. However, their global adversaries are only capable of packet-tracing attacks not traffic-analysis attacks. In this paper, we focused on addressing the problem of enhancing sink-location privacy against a global adversary capable of both attacks, while assuring that messages will arrive at the sink.

Artificial potential was originally developed in Khatib [20] for the purpose of obstacle avoidance. Later, it was used as a distributed control strategy for a large number of entities to achieve certain geometric configurations, such as in coverage and connectivity problems of WSNs [21, 30] and formation and flocking problems of collective artificial agents [31]. Since the approach is largely independent of the size and number of entities, the results scale well to larger sets of entities. We take advantage of the linear time complexity of this method to solve a nonlinear optimization problem that defines the k-anonymity sink-location problem.

6. Concluding Remarks

Wireless sensor networks rely on the sink to collect the measurements across the entire network; thus it is essential to protect the location information of the sink. However, the traffic around the sink typically exhibits distinctive patterns,

and an adversary with a global view can identify the location of the sink by measuring the traffic statistics of the entire network. In this study, we addressed such a threat, and we proposed an EMST-based two-phase routing algorithm that can achieve k-anonymity of the sink. In particular, the network is partitioned into k regions with each containing one designated node. Messages are first delivered to one designated node and then forwarded onto the EMST that interconnects all other designated nodes. The two-phase routing algorithms can effectively create many entities that exhibit the same traffic pattern as the nodes located close to the sink.

The positioning of k designated nodes affects two conflicting goals: the routing energy cost and the privacy level of the sink's location, and thus we formulated it as a nonlinear optimization problem. To tackle this problem, we first utilized a genetic algorithm to search for quasi-optimal solutions and developed a genetic algorithm-based quasi-optimal (GAQO) algorithm that can obtain solutions which closely approximate global optimal solutions. Further motivated by the observation that the quasi-optimal solution partitions the network into areas with similar sizes, we designed an artificial potential-based quasi-optimal (APQO) algorithm that can also obtain a quasi-optimal positioning of k nodes but which requires significantly reduced run-time. Our simulation results validated that both algorithms can effectively derive the positions of k designated nodes which meet the requirement of privacy at the minimum routing energy cost.

Acknowledgments

Thw authors thank Dr. Jianjun Hu for his feedback on the genetic algorithms. This work is partially supported by the National Science Foundation Grant CNS-0845671.

References

[1] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, "Towards event source unobservability with minimum network traffic in sensor networks," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 77–88, ACM, 2008.

[2] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing sourcelocation privacy in sensor network routing," in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS '05)*, pp. 599–608, IEEE Computer Society, 2005.

[3] Y. Jian, S. Chen, Z. Zhang, and L. Zhang, "Protecting receiver-location privacy in wireless sensor networks," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM'07)*, pp. 1955–1963, 2007.

[4] 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*, pp. 314–323, 2007.

[5] J. Deng, R. Han, and S. Mishra, "Countermeasures against traffic analysis attacks in wireless sensor networks," in *Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM '05)*, pp. 113–126, IEEE Computer Society, 2005.

[6] Q. Gu, X. Chen, Z. Jiang, and J. Wu, "Sink-anonymity mobility control in wireless sensor network," in *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pp. 36–41, 2009.

[7] Z. Liu and W. Xu, "Zeroing-in on network metric minima for sink location determination," in *Proceedings of the 3rd ACM conference on Wireless network security (WiSec '10)*, pp. 99–104, ACM, 2010.

[8] E. C.-H. Ngai, "On providing sink anonymity for sensor networks," in *Proceedings of the International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 269–273, ACM, 2009.

[9] J. Deng, R. Han, and S. Mishra, "Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks," in *Proceedings of the International Conference on Dependable Systems and Networks (DSN '04)*, p. 637, IEEE Computer Society, 2004.

[10] A. A. Nezhad, A. Miri, and D. Makrakis, "Location privacy and anonymity preserving routing for wireless sensor networks," *Computer Networks*, vol. 52, no. 18, pp. 3433–3452, 2008.

[11] Samarati P. and Sweeney L., "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," Tech. Rep., 1998.

[12] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell, "The bikenet mobile sensing system for cyclist experience mapping," in *Proceedings of the 5th international conference on Embedded networked Sensor Systems (SenSys '07)*, pp. 87–101, ACM, New York, NY, USA, 2007.

[13] L. Krishnamurthy, R. Adler, P. Buonadonna et al., "Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the north sea," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. 64–75, ACM, New York, NY, USA, 2005.

[14] L. Selavo, A. Wood, Q. Cao et al., "Luster: wireless sensor network for environmental research," in *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pp. 103–116, ACM, New York, NY, USA, 2007.

[15] V. Singhvi, A. Krause, C. Guestrin, J. Garrett, and S. Matthews, "Intelligent light control using sensor networks," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. ACM–218, New York, NY, USA, 2005.

[16] N. Xu, S. Rangwala, K. K. Chintalapudi et al., "A wireless sensor network for structural monitoring," in *Proceedings of the Second International Conference on Embedded Networked Sensor Systems (SenSys'04)*, pp. 13–24, New York, NY, USA, November 2004.

[17] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *EEE Symposium on Security And Privacy (SP '03)*, pp. 197–213, IEEE Computer Society, May 2003.

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

[19] C. L. Hwang, J. L. Williams, and L. T. Fan, *Introduction to the Generalized Reduced Gradient Method*, Institute for Systems Design and Optimization, 1972.

[20] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.

[21] W. Ding, G. Yan, and Z. Lin, "Self-deployment and coverage of mobile sensors within a bounded region," in *Proceedings of*

the Chinese Control and Decision Conference, pp. 3683–3688, 2009.

[22] Rouche N., Habets P., and Laloy M., *Stability Theory by Lyapunov's Direct Methods*, Springer, 1977.

[23] D. Dimarogonas and K. Kyriakopoulos, "An inverse agreement control strategy with application to swarm dispersion," in *Proceedings of the 46th IEEE Conference on Decision and Control*, pp. 6148–6153, 2007.

[24] Mixmaster Remailer, http://mixmaster.sourceforge.net/ .

[25] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services (MobiSys '03)*, pp. 31–42, ACM, 2003.

[26] B. Hoh and M. Gruteser, "Protecting location privacy through path confusion," in *Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM '05)*, pp. 194–205, IEEE Computer Society, 2005.

[27] Ouyang Y., Le Z., Liu D., Ford J., and Makedon F., "Source location privacy against laptop-class attacks in sensor networks," in *Proceedings of the 4th international conference on Security and Privacy in Communication Netowrks (SecureComm '08)*, pp. 1–10, ACM, 2008.

[28] M. Shao, Y. Yang, S. Zhu, and G. Cao, "Towards statistically strong source anonymity for sensor networks," in *Proceedings of the 27th IEEE International Conference on Computer Communications (INFOCOM'08)*, pp. 51–55, 2008.

[29] W. Conner, T. Abdelzaher, and K. Nahrstedt, "Using data aggregation to prevent traffic analysis in wireless sensor networks," in *Proceedings of the International Conference on Distributed Computing in Sensor Networks (DCOSS '06)*, pp. 202–217, 2006.

[30] A. Howard, M. Mataric, and G. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem," *Distributed Autonomous Robotic Systems*, vol. 5, pp. 299–308, 2002.

[31] T. Balch and M. Hybinette, "Behavior-based coordination for large-scale robot formations," in *Proceedings of the 4th International Conference on Multiagent Systems*, pp. 363–364, 2000.