Run to Potential: Sweep Coverage in Wireless Sensor Networks

Min Xi*, Kui Wu†,Yong Qi*,Jizhong Zhao*, Yunhao Liu‡, Mo Li‡

*Department of Computer Science, Xi'an Jiaotong University, China

†Department of Computer Science, University of Victoria, British Columbia, Canada

‡ Department of Computer Science and Engineering, Hong Kong University of Science and Technology

Abstract—**Wireless sensor networks have become a promising technology in monitoring physical world. In many applications with wireless sensor networks, it is essential to understand how well an interested area is monitored (covered) by sensors. The traditional way of evaluating sensor coverage requires that every point in the field should be monitored and the sensor network should be connected to transmit messages to a processing center (sink). Such a requirement is too strong to be financially practical in many scenarios. In this study, we address another type of coverage problem, sweep coverage, when we utilize mobile nodes as supplementary in a sparse and probably disconnected sensor network. Different from previous coverage problem, we focus on retrieving data from *dynamic* Points of Interest (POIs), where a sensor network does not necessarily have fixed data rendezvous points as POIs. Instead, any sensor node within the network could become a POI. We first analyze the relationship among information access delay, information access probability, and the number of required mobile nodes. We then design a distributed algorithm based on a virtual 3D map of local gradient information to guide the movement of mobile nodes to achieve sweep coverage on dynamic POIs. Using the analytical results as the guideline for setting the system parameters, we examine the performance of our algorithm compared with existing approaches.**

I. INTRODUCTION

Recent progress in Micro-Electro-Mechanical Systems (MEMS) makes it possible to embed sensors, micro-processors, memory, wireless transceivers, and power supply within a sensor node of several cubic millimeters [1]. Co-operating together, a large number of tiny sensor nodes can form an autonomous networking system, called a wireless sensor network. Such a network can monitor specific physical phenomena [2] (e.g., temperature, humidity, audio/video of animals) and transfer data in real-time approach [3]. It has been widely used in numerous applications, ranging from environment surveillance, object tracking, to structure monitoring, scientific observation. An important factor that affects the effectiveness of such systems is evaluated by the coverage of the networks. The coverage ratio generally determines how well the interested area is monitored by the sensors.

Based on different application scenarios, the coverage problem can be divided into three categories: full coverage, barrier coverage, and sweep coverage. Full coverage requires that each point of the area is continuously monitored by one or more sensors. Such a requirement is strong and requires dense node deployment for a critical region, where we do not want to miss any specific event. Barrier coverage requires that the network captures any intruder crossing a pre-defined barrier area, for example, the border of two countries. In both full coverage and barrier coverage, the interested area should be covered by sensors continuously. In many applications, however, full coverage and barrier coverage incur prohibitive system cost. For instance, in a sensor system for farming monitoring and management, full coverage requires thousands of sensors to cover a large field. To save cost, we might want to deploy sensor nodes sparsely. The entire field may not be covered all the time and the network may not necessarily be connected. We can use a number of mobile nodes moving within the field to collect data and deliver the data to the processing center. We call it sweep coverage. Sweep coverage requires that the interesting event, once detected by some sensor node, be recorded locally on the node so that the information could be later retrieved by the mobile nodes within a delay bound.

The support of sweep coverage is challenging. First, when an event happens, the sensor node detecting it becomes a Points of Interest (POI) and records corresponding data, which should be collected as soon as possible. In many applications, the number and locations of POIs change dynamically, and it is usually difficult to predict when and where an interesting event happens. The dynamics of POIs requires an algorithm to coordinate the movement of mobile nodes frequently. Second, we should not always assume the connectivity of the wireless sensor network especially when the monitored field is large. In a sparse network, how to schedule the movement of mobile nodes is not trivial since it is difficult for the mobile nodes to obtain complete information for their movement coordination. Third, it is common that the mobile nodes have much more powerful radio transceivers with a much larger transmission range than that of stationary sensor nodes. Logically we have two radio networks: the network consisting of stationary sensors and the network consisting of mobile nodes. The former one is called the ***stationary sensor network***; it is static and may be sparse and disconnected. The latter one is called the ***dynamic mobile network***. It dynamically changes over time and may or may not be connected from time to time.

The benefits as well as the challenges of using mobile nodes

motivate us to design efficient algorithms for sweep coverage with dynamic POIs. In summary, we make the following contributions in this paper:

1) We design a novel sweep coverage algorithm that uses the ***potentials*** of POIs as the driving force to attract mobile nodes toward the POIs. The ***potential*** of a sensor node is a logical value periodically updated. The ***potential*** field over the entire network provides guidance information for the movement of mobile nodes. Our algorithm is built on the idea of ***run to higher potentials***: by strategically calculating the potentials of sensor nodes, our algorithm forces a mobile node to move along the direction toward a nearby emergency POI. The algorithm only requires a node to communicate with its local neighbors and thus is fully distributed.

2) We analyze the bounds on the delay of information retrieval, which is defined as the delay from the time a POI starts recording a detected event to the time the information is retrieved by a mobile node. Our analytical results disclose the relationship among the number of mobile nodes, their moving speed, the delay of information retrieval, and the information retrieval probability.

3) Using the analytical results as the guideline to set the system parameters, we conduct comprehensive simulations to evaluate this design. The results demonstrate the efficiency and effectiveness of our proposed sweeping algorithm.

The rest of this paper is organized as follow. Section II discusses the existing work. In Section III we present the system model and formulate the sweep coverage problem. We analyze relationship between delay bound, probability of sweep coverage, and number of mobile nodes in Section IV. We present a novel sweep coverage algorithm that builds up the potential of a POI and drives mobile nodes toward the POI in Section V. We conduct the experimental performance evaluation in Section VI. We finally conclude the paper in Section VII.

II. RELATED WORK

Sensor coverage can be considered as a measure of the quality of service of a wireless sensor network. It has been an active and important research topic, evidenced by many research contributions to this field in recent years. The coverage problem can be classified into three categories: full coverage, barrier coverage, and sweep coverage.

Full coverage: Full coverage means that every target in a certain area must be covered at any time. Based on the different types of targets, full coverage can be divided into area coverage and point coverage [4].

To achieve full coverage, people use deterministic or randomized methods to deploy wireless sensors. The deterministic deployment methods achieve the best coverage, but with too much overhead. The randomized methods are more flexible, but cannot guarantee 100% confidence. For deterministic sensor deployment, the studies mainly focus on how to decrease the number of sensors and increase the lifetime of the sensor network. For example, [5] proposed deployment patterns to achieve full coverage and three-connectivity, and full coverage and five-connectivity under different ratios of sensor communication range over sensing range.

Regarding randomized deployment methods, the research topics include the study of critical number of sensors for coverage, energy efficient coverage, mobile sensors assisted coverage, and so on. Kumar et al. [6] investigate the critical number of sensors to achieve coverage with random deployment. Li et al. [7] uses disjoint sets to decrease energy consumption. There is also work uses mobile sensors as well as stationary sensors to cover an area. For example, Chellappan et al. introduce an approach using mobile sensors to improve coverage [8], and Wang et al. [9] propose a distributed algorithm to let mobile sensors determine their moving direction only with local information.

Barrier coverage: The barrier coverage problem comes from boundary detection in some applications such as detecting intruders crossing a border of two countries. Kumar et al. [10] discuss the barrier coverage problem for the first time. They transform the barrier coverage problem into the connectivity problem and present the critical condition for k-barrier weak coverage. They also prove that this problem cannot be locally solved. If the assumptions and requirements are relaxd, localized algorithms can be obtained [11]. Balister et al. [12] study the relationship between sensor density and coverage in a thin strip.

Sweep coverage: The problem of sweep coverage comes from applications that do not require continuous sensor coverage while the system cost for full coverage is prohibitive. The main goal is to cover the targets in the area within a given time interval. In most cases, mobile nodes are introduced for sweep coverage.

The sweep coverage problem has appeared in many forms. The problem is considered as moving barriers to cover the whole region in [13]. In [14], Rekleitis et al. propose an approach to sweep all the destination zones. They assume that the mobile nodes can communicate with each other and know their positions. The area is divided into stripes, with each mobile node taking care of one stripe. Wong et al. [15] use topological mapping to sweep the destination area. They make cell decomposition and cover each cell by a zigzag pattern. Batalin and Sukhatme [16] propose a decentralized method and present the frequency coverage metric to evaluate the quality of sweep coverage. Although mobile nodes do not exchange information with each other, they need to communicate with the static sensors to avoid duplicate coverage. Muhammad et al. [17] provide a verification method for sweep coverage.

Some work [18], [19] uses information gradient to guide data retrieval. Information gradient can be formed using natural measures, such as temperature. Those methods, however, may encounter the problem of gradient plateau or single point failure, which makes local greedy algorithm invalid. Lin et al. [20] propose a gradient creation method using harmonic function to guarantee that greedy navigation never gets stuck.

Cheng et al. [21] define a new type of sweep coverage. The problem is to investigate how many sensors are needed to sweep pre-defined **fixed** Points of Interest (POI) rather than the entire area at a specified time interval. They prove that the problem of calculating the minimum number of sensors is NP-hard and propose a centralized algorithm together with a distributed sweep algorithm, called DSweep, to obtain an approximate solution.

We extend the sweep coverage problem in [21] by allowing POIs to be dynamic, i.e, the POIs are not pre-defined and can emerge at **any time and any position**. This assumption is more realistic because in many real applications, it is difficult, if not impossible, to accurately predict when and where an event might happen. Providing sweep coverage with dynamic POIs is much more challenging, making our work significant different from existing ones.

III. PROBLEM FORMULATION

We make the following assumptions on the system settings, which can be found true in many realistic applications:

- The field: Without loss of generality, we assume that the monitored field is a square area with the edge length of l meters.
- Sensors: There are n stationary wireless sensor nodes randomly (or strategically) deployed in the field. The network consisting of stationary sensors is called *stationary sensor network*, which may be sparse and disconnected. The radio transmission range of all the stationary sensors is the same and is equal to $r(r << l)$.
- Mobile nodes: There are m mobile nodes moving in the field to collect and process data. The network consisting of mobile nodes is called *dynamic mobile network*. Its topology changes over time. The radio transmission range of all the mobile nodes is the same and is equal to R. R is normally much larger than r, because mobile nodes can use a fuel-powered engine and thus energy concern of mobile nodes is of second-order importance.
- Moving speed: The moving speed of the mobile nodes is the same and is equal to $v(v << l)$ meters per second. Generally speaking, the radio transmission speed is much faster than the movement speed of mobile nodes.
- Moving direction: We assume that a mobile node knows the direction of a sensor node if they can *directly* communicate with each other. Location information, although important, may not be required to make this assumption feasible.

TABLE I
NOTATIONS

Symbol	Description
n	the number of sensor nodes
m	the number of mobile nodes
l	the edge length of the square field
r	radio range of sensor nodes
R	radio range of mobile nodes
v	moving speed of mobile nodes
T	delay bound on sweep time
$\gamma(.)$	potential update function at POIs
$\beta(.)$	potential update function at non-POI sensors
μ_p	potential update interval at POIs
μ_s	potential update interval at non-POI sensors

- Restriction on data retrieval: A mobile node can download the information from a stationary sensor node if and only if the mobile node is within the communication range of the sensor node. To save energy of sensor nodes, we do not assume multi-hop radio transmission for data download because the volume of data may be large. Nevertheless, we *do not put such a constraint on the control messages* of our protocol, which have a very small size compared to real data.
- Network connectivity: We assume that with links from both the stationary sensor network and the dynamic mobile network, all nodes (sensors and mobile nodes) are connected most of the time.

For ease of reference, the notation used in this paper is listed in Table I.

With the above settings, logically we have two wireless networks: the stationary sensor network and the dynamic mobile network. The stationary sensor network monitors the field. When an event happens, the sensor node which senses the event becomes a point of interest (POI). One of the mobile nodes in the mobile network should move to the POI and process the event. A POI is swept if its data is retrieved by a mobile node. For effective network design, we need to answer the following key questions:

1) At least how many mobile nodes are required so that any sensor node, once becoming a POI, can be swept within a time period of T with a high probability [1]?
2) How can the mobile nodes be guided to POIs without a centralized control?

We answer the first question and the second question in the next section and Section V, respectively.

IV. IDEAL SWEEP COVERAGE

In this section, we answer the first question by analyzing the sweep coverage in different scenarios. The analytical results disclose the relationship among the delay bound, the information access probability, and the required number of

---

[1]We do not assume a deterministic bound on time delay because it requires a perfect mobility coordination among mobile nodes, which is hard to guarantee without a centralized control in a system with dynamic POIs.

mobile nodes. They will serve as the guideline in parameter selection in our later experimental evaluation.

Lemma 1: **System with single POI:** With the system settings in Section III, for a network with single dynamic POI (i.e., any sensor could become a POI but there is only one POI within a time duration of T), if a sensor node, once becoming a POI, is swept within a time delay of T with a probability of p, the minimum number of mobile nodes required should be no smaller than

$$m = \begin{cases} 1 & \text{if } \pi * (T * v + r)^2 \geq l^2 \\ \left\lceil \frac{\ln(1-p)}{\ln(1-\pi*(T*v+r)^2/l^2)} \right\rceil & \text{otherwise} \end{cases}$$

Proof In the ideal case, the event that a sensor node becomes a POI is known to one or more mobile nodes, and with a "perfect" algorithm at least one mobile node moves directly to the POI to retrieve data. Due to the delay constraint, the distance between the mobile node and the POI must be no larger than $T * v + r$. Since the POI could be any sensor in the field and the mobile nodes may not be able to always communicate with each other for coordinated movement, the locations of mobile nodes at any time instant are *approximated* as random points in the field. The probability that the POI falls within the range of $T * v + r$ of at least one mobile node is

$$p = 1 - (1 - f)^{m_1}, \tag{1}$$

where $f = \min\{\frac{\pi*(T*v+r)^2}{l^2}, 1\}$ is the probability that the POI falls within the range of $T*v+r$ of a given mobile node, and m_1 is the number of mobile nodes that have known the event. From Equation (1), the minimum number of mobile nodes, m, should not be smaller than

$$m_1 = \begin{cases} 1 & \text{if } \pi * (T * v + r)^2 \geq l^2 \\ \left\lceil \frac{\ln(1-p)}{\ln(1-\pi*(t*v+r)^2/l^2)} \right\rceil & \text{otherwise} \end{cases} \qquad \blacksquare$$

Lemma 1 represents the ideal scenario based on three assumptions: (1) There is a "perfect" algorithm that can guide a mobile node to move directly to the POI; (2) the event that a sensor node becomes a POI is known to one or more mobile nodes; and (3) the locations of mobile nodes could be approximated as random points in the field. The first assumption is the major challenge that our sweep algorithm will address. Based on the assumption on network connectivity, the second assumption is trivial with certain control signaling messages. Without the second assumption, there is actually no way to get a bound on delay. The last assumption is only for an approximation of the locations of mobile nodes. This approximation is reasonable because the locations of POIs are random and the mobile nodes are attracted to the POIs.

Lemma 1 is useful to estimate the best result that a network with single dynamic POI can achieve. It is necessary to note that Lemma 1 is also suitable for a system in which at any time period of T only one event happens, because multiple sensors that detect the same event should be spatially close and could be roughly considered as a single POI.

Lemma 2: **System with multiple POIs when the delay bound T is large:** With the system settings in Section III, for a network with $k(k << n)$ dynamic POIs, if any POI is swept within a time period of T, the minimum number of required mobile nodes in the network should be no smaller than k if $\pi * (T * v + r)^2 \geq l^2$.

Lemma 2 is trivial and obvious because a one-to-one mapping between POIs and mobile nodes can meet the requirement.

Lemma 3: **System with multiple POIs when the delay bound T is small:** With the system settings in Section III, for a network with $k(k << n)$ dynamic POIs that are spatially separated with a distance of at least $2(T*v+r)$, if the number of mobile nodes is $\frac{l^2(\ln k + c)}{\pi*(T*v+r)^2}$ where c is a positive constant, then all POIs can be swept within a time period of T with a probability no smaller than $e^{-e^{-\frac{cl^2}{k\pi*(T*v+r)^2}}}$.

Proof With a tight delay bound, the field could be divided into k non-intersecting circular areas, denoted as $A_1, \ldots, A_k$, respectively, plus an irregular area that is not covered by any $A_i, i = 1, \ldots, k$. The size of $A_i(i = 1, \ldots, k)$ is $\pi * (T * v + r)^2$. We add mobile nodes one by one randomly into the field until each $A_i(i = 1, \ldots, k)$ includes at least one mobile node, i.e., A_i is covered. The problem of calculating the expected number of mobile nodes is similar to the coupon collector's problem [22].

Denote the number of required mobile nodes as a random variable X. We now determine $E[X]$. If X_i is the number of required mobile nodes to cover a new circular area while we have exactly $i - 1$ different circular areas covered. Clearly, $X = \sum_{i=1}^{k} X_i$. Each $X_i(i = 1, \ldots, k)$ is a geometric random variable. When exactly $i - 1$ different circular areas are covered, the probability of covering a new circular area is

$$p_i = (k - i + 1) * g,$$

where $g = \frac{\pi*(T*v+r)^2}{l^2}$ is the probability that the added mobile node falls within the new circular area. Hence, $E[x_i] = \frac{1}{p_i}$. Using the linearity of expectation, we have that

$$E[X] = E[\sum_{i=1}^{k} X_i] \tag{2}$$

$$= \frac{1}{g} \sum_{i=1}^{k} \frac{1}{k - i + 1} \tag{3}$$

$$= \frac{l^2(\ln k + \Theta(1))}{\pi * (T * v + r)^2} \tag{4}$$

The above result indicates that with a high probability, using $\frac{l^2(\ln k + c)}{\pi*(T*v+r)^2}$ (c is a positive constant) mobile nodes should be able to sweep all POIs within the delay bound T. Using Chernoff bound and Poisson approximation [22], this probability is no less than

$$e^{-e^{-\frac{c}{gk}}} = e^{-e^{-\frac{cl^2}{k\pi*(T*v+r)^2}}}.$$
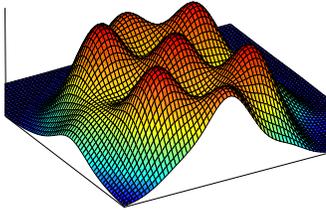
Fig. 1. A virtual map of potentials

The calculation is similar to that in Section 5.4.1 of [22] and is omitted to save space. ∎

It is extremely hard, if not impossible, to analyze a network of multiple dynamic POIs with a moderate delay bound on sweep coverage, because the k circular areas, $A_i, i = 1, \ldots, k$, are likely to intersect with each other. The analysis in this case relies on the locations of POIs and needs to consider all possible intersecting scenarios of different circular areas. Nevertheless, Lemma 2 and Lemma 3 provide the lower and upper bounds on the required number of mobile nodes for such a network, respectively.

V. DRIVING MOBILE NODES WITH POTENTIALS

A. Basic Idea

Without a centralized control, each mobile node does not have a clear view of the global state information and as such we need to find a way to guide the movement of mobile nodes toward POIs with local information only. To achieve this, our basic idea is to build a virtual 3D map in the stationary sensor network, with POIs locating at the peaks as illustrated in Figure 1. Mobile nodes thus "climb" to these peaks with the help of local information, e.g., the height of the neighboring sensors in the virtual 3D map. Once the data of a POI is retrieved by a mobile node, the POI becomes a normal sensor node and it drops to the bottom in the 3D map, letting the mobile node move to another POI. The information at a sensor that is used to guide the movement of mobile nodes is called the *potential* of the sensor. In this paper, it is represented as a real number.

We need to answer two questions in the above potential-guided approach. First, how can we build up the potentials that can effectively guide the movement of mobile nodes toward POIs? Second, how can mobile nodes coordinate their movement to avoid collision (i.e., several mobile nodes move toward the same POI and thus result in futile movement)? In the following sections, we address the above problems and finally lead to a fully distributed sweep coverage algorithm.

B. Building Up Potentials

Clearly the potential of a sensor node will change dynamically from time to time. The changing process of a sensor's potential can be divided into three phases:

Initialization phase: When the system starts, all the sensors set their potentials to 0 as the initial value, meaning that the virtual 3D map is flat.

Potential build-up phase: When a sensor detects a phenomenon in interest, it becomes a POI and sets its potential to a value larger than that of any of its local neighbors, i.e., other sensor nodes that can be reached via one-hop radio. It then periodically increases its potential at an interval time of μ_p with a monotonically increasing function, $\gamma(t)$, where t is the time duration between current time and the time when the phenomenon was detected. In this paper, we adopt $\gamma(t) = \alpha t$, where α is a constant value. Note that other definition of γ function is also possible.

For a non-POI sensor, it updates its potential based on the potentials of its local neighbors at an interval time of μ_s (normally $\mu_s < \mu_p$). The potential of a non-POI sensor is calculated using a potential update function β, which is defined later in this section.

Regression: When a mobile node reaches a POI, the data at the POI can be retrieved and the potential of the POI is reset to 0.

The function γ decides how fast a POI raises its potential. It can be considered as the degree of emergency of the POI. If the POI has not been swept for a long time, its potential will become higher and higher, meaning that it is urgent to retrieve data from this POI. The function β decides how to calculate the potential at a non-POI sensor node. In order to build a virtual 3D map shown in Figure 1, this function needs to follow two rules:

- A POI should have the maximum potential among all its neighbors. This feature is called the *local maximum of POI*. The design of β function should not violate this feature.
- A non-POI sensor closer to a POI should has a higher potential than other non-POI sensors far away from the POI, that is, the potentials of a POI and its nearby neighbors should form a "hill" with the POI at the top of the "hill".

In this paper, we adopt a very simple way to calculate the function β: the potential of a non-POI node is calculated as the average value of its direct neighbors' potentials. It is very easy to see that this way of calculating β function meets the above requirements, based on the fact that the average of a set of values cannot be larger than the maximum value in the set.

Figure 2 illustrates an example for the potential update process. As illustrated in Figure 2, the potentials continuously change over time to reflect the dynamic changes of POIs, *without using a centralized control or any global information*. Guided with the potentials, a mobile node moving from a sensor of low potential toward a sensor of higher potential will finally reach a POI. When a tie exists, the mobile node randomly selects a direction where the sensor has a potential

no less than the current sensor. In this way, our approach *guarantees* that a mobile node moves toward a correct direction, that is, a direction including sensors that have potentials no smaller than the current sensor. A mobile node will not be stuck at a sensor unless this sensor is a POI. Once the POI's data is retrieved, its potential changes to 0, and the mobile node starts to move to another POI. In addition, no path loops could be formed because mobile nodes always "climb" higher in the virtual 3D map.
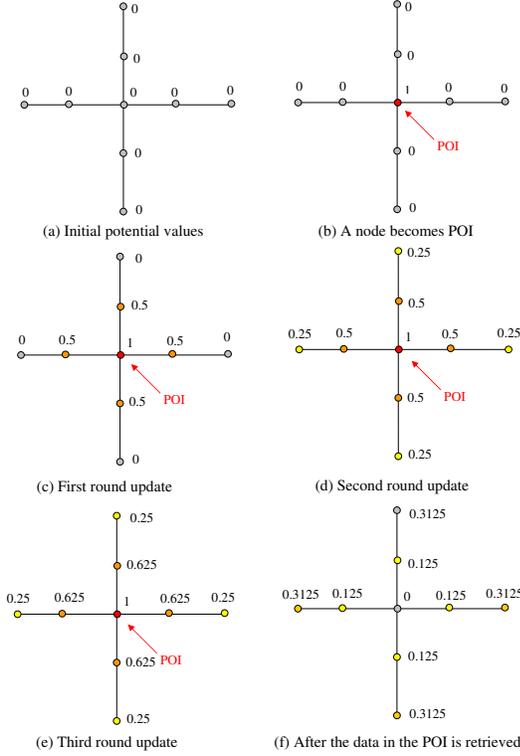
Fig. 2.   An example of potential update process

C. Anti-Collision

The approach proposed above does not handle the collision of mobile nodes and is called the *naive sweep algorithm*. Collision happens when multiple mobile nodes move to the same POI, resulting in unnecessary moves for some mobile nodes. We solve the problem with an improved algorithm, called the *anti-collision sweep algorithm*.

The anti-collision sweep algorithm uses the same idea of the naive method. The only difference is that each mobile node uses an "agent" to mark its route ahead and follows the marks to POIs. The "agent" is actually a control message transmitted in the stationary sensor network. An agent uses the naive sweep algorithm to guide its movement. Once it reaches a POI, it resets the POI's potential to 0. Since an agent moves much faster than its corresponding mobile node, some mobile nodes will change their direction before they actually reach the POI because the POI's potential has been reset. In this

way, unnecessary moves of mobile nodes are greatly reduced. It is true that the agents might still collide. But this is not a problem at all because a mobile node always follows the newest marks made by its agent. The old marks that lead to the collision are obsolete and will not be used by the mobile nodes.

Since the agents use the naive sweep algorithm to guide their movement, they enjoy the same nice features of the naive method: They are guaranteed to move along correct directions; they are never stuck at a sensor node unless reaching a POI; and no path loops could be formed.

VI. Experimental Evaluation

A. Simulation Model

We perform simulation studies to evaluate our sweep coverage algorithm and compare it with the DSweep algorithm introduced in [21].

we fix the following parameters to make the simulation results concise. The simulated field is a square area with edge length of 3000 meters. We deploy 900 stationary sensors randomly in the field. The mobile nodes' moving speed is set to 10 m/s, and their initial locations are random in the field. The radio range of both mobile nodes and stationary sensor nodes is set to 130 meters. The potential update interval for POIs (μ_p) is set to 5 seconds.

Other system parameters, however, are changed in the simulation to capture important features of our algorithm in different scenarios. These parameters include the number of POIs, the number of mobile nodes, the constraint on sweep delay, and the potential update interval for non-POI sensors (μ_s). To reflect the flexibility of our algorithm, we assume that POIs are not known in advance, but instead their locations are determined by a set of randomly selected sensors and may change over time.

We test two important performance results: *average sweep delay* and *average moving distance*. Average sweep delay is defined as the total time required to sweep all POIs divided by the total number of POIs. The time required to sweep a POI is calculated as the time when a sensor becomes the POI to the time when a mobile node can communicate directly to the POI. The average moving distance is defined as the total moving distance of all mobile nodes to the total number of mobile nodes. For each simulation scenario, we run the simulation 50 times and take the average as the final results.

B. Performance Results

We test the performance of our sweep coverage algorithm with the anti-collision enhancement in different scenarios.

System with single POI: According to Lemma 1, if we require the sweep delay falls within 70 seconds with a probability higher than 98%, the number of mobile nodes should be larger than 15. Figure 3 shows the sweep delay with different mobile nodes. It can be seen that with 15
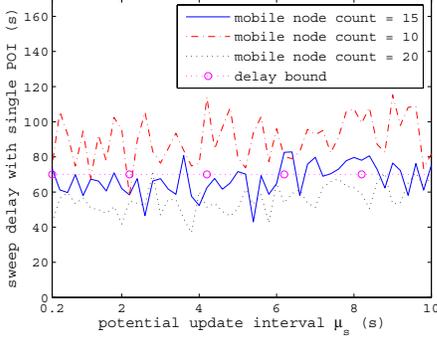
Fig. 3. Sweep delay with single POI

Fig. 4. Sweep delay with 10 POIs

mobile nodes, the sweep delay is mostly fall within the bound, meaning that our sweep coverage algorithm can actually achieve the performance close to the "ideal" system described in Section IV. For comparison purpose, we also test the sweep delay with 10 and 20 mobile nodes. Clearly, 10 mobile nodes cannot meet the requirement and 20 mobile nodes are overkill because the sweep delay mostly falls within 60 seconds.

System with multiple POIs: We select 10 POIs, which are randomly chosen from the sensors. When the requirement on sweep delay is loose such as 160 seconds, Lemma 2 indicates that a good sweep algorithm should require only 10 mobile nodes. When the requirement is tight such as 45 seconds, Lemma 3 shows that if the delay requirement is met with a probability higher than 99%, we should require about 26 mobile nodes, calculated with the constant c in Lemma 3 equal to 0.75.

Figure 4 shows the sweep delay with different mobile nodes. It can be seen that when the delay requirement is loose, using only 10 mobile nodes our algorithm can meet the requirement most of the time. For a tight delay bound (e.g., 45 seconds), about 26 mobile nodes are required in theory. The sweep delay with our algorithm, however, is a bit larger than the bound if only 26 mobile nodes are used. This is mainly because a mobile node does not move to a POI directly, but instead follows a marked path toward the POI, which is not a strict line. We can also see that adding more mobile nodes (e.g., using 36 mobile nodes) does not reduce sweep delay significantly due to the same reason.

C. Benefit of Anti-Collision

To illustrate the benefit of the anti-collision mechanism, we compare the performance of the naive sweep algorithm with that of the anti-collision version. Figure 5 shows the results of a system with 10 mobile nodes and 50 dynamic POIs. Figure 5(b) shows that the average moving distance of the anti-collision method is much smaller than that of the naive method. The reason has been discussed in Section V-C. As a result, the sweep delay of the anti-collision method is smaller than that of the naive method, as demonstrated in Figure 5(a).
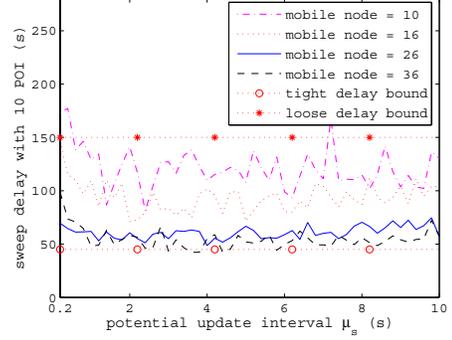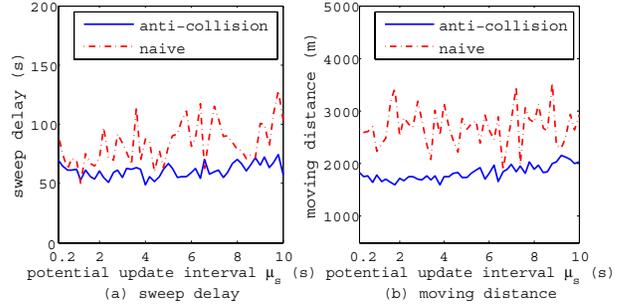
Fig. 5. Compare with anti-collision and naive method

D. Performance Comparison Between Our Algorithm and DSweep

The DSweep algorithm [21] is proposed for systems with static POIs. It assumes that all mobile nodes know their instant locations, and each POI has a globally unique ID. The locations and the sweep period of all POIs are pre-knowledge at each mobile node. With the above assumptions, DSweep adopts "store-carry-forward" method to make mobile nodes, once meeting each other, exchange their newest sweep information. A mobile node decides its movement toward the nearest and most urgent POI that it has known.

Although DSweep is the closest to our work, it is proposed for systems with static POIs, and as such it is impossible to compare DSweep directly to our algorithm. To make them comparable, we make two modifications on DSweep. In the first modification, also denoted as DSweep in Figure 6, we allow a mobile node to know any newly-formed POI in a range twice of its communication range. Second, whenever a new POI is formed, we allow every mobile node to know the POI's information immediately. This modification is denoted as *DSweep-global* in Figure 6.

Figure 6 shows the performance results with the same parameters for multiple POIs in Section VI-B. Because both DSweep and DSweep-global do not rely on the static sensor network, their performance is independent of potential update interval. It can be seen that the performance of our algorithm is better than that of DSweep. This is because DSweep only
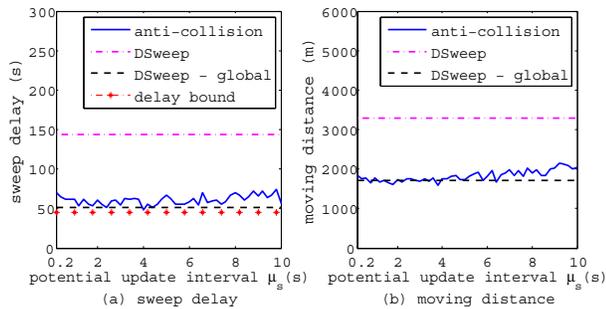
Fig. 6. Compare with anti-collision and DSweep

uses the dynamic mobile network to make moving decisions, and the dynamic mobile network is not always connected and thus newest information cannot be guaranteed to propagate to right mobile nodes. Nevertheless, our algorithm has worse performance than DSweep-global. This is because DSweep-global assumes that each mobile node knows the accurate global information to make better moving decisions. This assumption, however, is very hard to implement in reality.

VII. CONCLUSION

Traditional approaches to addressing the problem of sensor coverage use full coverage or barrier coverage as the major evaluation metric. In many applications, however, we need a totally different coverage measure, called sweep coverage. Existing work studies sweep coverage only with static POIs. This model is not flexible and not suitable in applications where interesting events could happen anywhere and thus POIs are hard to predict. We solve the problem in this paper by presenting a novel sweep coverage algorithm for systems with dynamic POIs, relying on a virtual 3D map of local gradient information to guide the movement of mobile nodes. Our algorithm is fully distributed and can achieve performance close to the optimum (i.e., the "ideal" sweep coverage). We analyze the relationship among the number of mobile nodes, the bound on sweep delay, and the probability of sweep coverage. The analytical results provide a good guideline for system design and parameter selection.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] M. Wu, J. Xu, X. Tang, and W.-C. Lee, "Top-k monitoring in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 6, pp. 962–976, 2007.

[3] X. Liu, Q. Wang, L. Sha, and W. He, "Optimal qos sampling frequency assignment for real-time wireless sensor networks," 2003, in Proceedings of the 24th Real-Time Systems Symposium.

[4] M. Cardei and D. Z. Du, "Improving wireless sensor network lifetime through power aware organization," *ACM Wireless Networks*, no. 11, pp. 333–340, 2005.

[5] X. Bai, D. Xuan, Z. Yun, T. H. Lai, and W. Jia, "Complete optimal deployment patterns for full-coverage and k-connectivity (k≤6) wireless sensor networks." ACM New York, NY, USA, 2008, pp. 401–410, proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing.

[6] S. Kumar, T. H. Lai, and J. Balogh, "On k-coverage in a mostly sleeping sensor network," in *Proceedings of the Tenth Annual International Conference on Mobile Computing and Networking*, Philadelphia, Pennsylvania, USA, 2004.

[7] X. Y. Li and Y. Wang, "Simple approximation algorithms and ptass for various problems in wireless ad hoc networks," *Journal of Parallel and Distributed Computing*, vol. 66, no. 4, pp. 515–530, 2006.

[8] S. Chellappan, W. Gu, X. Bai, D. Xuan, B. Ma, and K. Zhang, "Deploying wireless sensor networks under limited mobility constraints," *IEEE Transactions on Mobile Computing*, vol. 6, no. 10, pp. 1142–1157, 2007.

[9] D.Wang, J. Liu, and Q. Zhang, "Field coverage using a hybrid network of static and mobile sensors," in *Proceedings of IEEE IWQoS*, 2007.

[10] S. Kumar, T. H. Lai, and A. Arora, "Barrier coverage with wireless sensors," *Wireless Networks*, vol. 13, no. 6, pp. 817–834, 2007.

[11] A. Chen, S. Kumar, and T. H. Lai, "Designing localized algorithms for barrier coverage," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, 2007, pp. 63–74.

[12] P. Balister, B. Bollobas, A. Sarkar, and S. Kumar, "Reliable density estimates for coverage and connectivity in thin strips of finite length," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, 2007, pp. 75–86.

[13] A. Howard and M. J. Mataric, "Cover me! a self-deployment algorithm for mobile sensor networks," in *2002 International Conference on Robotics and Automations, Washington DC, May*, 2002.

[14] I. M. Rekleitis, A. P. New, and H. Choset, "Distributed coverage of unknown/unstructured environments by mobile sensor networks," in *3rd International NRL Workshop on Multi-Robot Systems*, A. C. Schultz, L. E. Parker, and F. Schneider, Eds. Washington, D.C.: Kluwer, 2005, pp. 145–155.

[15] S. C. Wong and B. A. MacDonald, "A topological coverage algorithm for mobile robots," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, 2003.

[16] M. A. Batalin and G. S. Sukhatme, "Multi-robot dynamic coverage of a planar bounded environment," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002.

[17] A. Muhammad and A. Jadbabaie, "Dynamic coverage verification in mobile sensor networks via switched higher order laplacians," *Robotics: Science and Systems*, 2007.

[18] J. Faruque and A. Helmy, "Rugged: Routing on fingerprint gradients in sensor networks," in *IEEE International Conference on Pervasive Services*, 2004.

[19] J. Liu, F. Zhao, and D. Petrovic, "Information-directed routing in ad hoc sensor networks," in *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, 2003, pp. 88–97.

[20] H. Lin, M. Lu, S. Stonybrook, N. Milosavljevic, J. Gao, and L. J. Guibas, "Composable information gradients in wireless sensor networks," in *Proceedings of IEEE International Conference on Information Processing in Sensor Networks*, 2008, pp. 121–132.

[21] W. Cheng, M. Li, K. Liu, Y. Liu, X. Li, and X. Liao, "Sweep coverage with mobile sensors," in *Proceedings of IEEE International Symposium on Parallel and Distributed Processing*, Miami, FL, 2008, pp. 1–9.

[22] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.