Delay Tolerant Event Collection for Underground Coal Mine using Mobile Sinks

Ji Luo
Hong Kong University of
Science and Technology
luoji@cse.ust.hk

Qian Zhang
Hong Kong University of
Science and Technology
qianzh@cse.ust.hk

Dan Wang
The Hong Kong
Polytechnic University
csdwang@comp.polyu.edu.hk

***Abstract*—There is a growing interest in using wireless sensor networks for security monitoring in the underground coal mines. In such applications, the sensor nodes are deployed to detect interested events, e.g., the density of certain gas at some locations is higher than the predefined threshold. These events are then reported to the base station outside. Using conventional multi-hop routing for data reporting, however, will result in imbalance of energy consumption among the sensors. Even worse, the unfriendly communication condition underground makes the multi-hop data transmission challenging, if not impossible. In this paper, we thus propose to leverage tramcars as mobile sinks to assist event collection and delivery.**

We further observe that the sensor readings have spatial and temporal correlation. More precisely, the same event may be observed by multiple neighboring sensor nodes and/or at different time. Obviously, it can be more energy-efficient if the data are selectively reported. As such, we first provide a general, yet realistic definition on the events. We then transform the event collection problem into a set coverage problem; and our objective is to maximize the system lifetime with the coverage rate of events guaranteed. We show that the problem is NP-hard even when all the events are known in advance. We present an online scheme which leverages the spatial-temporal correlation of the events to balance the communication energy of the static sensor nodes. We prove that the expected event coverage rate can be guaranteed in theory. Through extensive simulation, we demonstrate that our scheme can significantly extend system lifetime, as compared to a stochastic collection scheme.

I. Introduction

Wireless sensor networks today have been applied to a wide range of applications for data collection. Recently, they are used to monitoring the conditions of underground coal mines [10]; see Fig. 1 for a concrete example. In such applications, static sensor nodes are deployed in the underground tunnels of the coal mines to collect data (e.g., temperature, humidity and gas density). These data can further indicate some special *events*; for example, high temperature implies an accident of explosion while exceptional gas density may indicate a gas leak. Such events should be reported to the sink outside for further analysis.

To design an efficient event collection system for underground coal mines, we face some unique challenges. First, as can be seen from Fig. 1, the underground coal mines are usually composed of long tunnels. Thus, conventional multi-hop data reporting can be extremely inefficient. More precisely, the burden of the nodes close to the tunnel exit will be much higher. Even worse, the communication underground is tough, with high ratio of packet loss [3]. This can result in a very low multi-hop data transmission rate. Second, different raw sensor readings from different sensors may indicate the same event; therefore, it may not be cost efficient to request all the sensors to report their readings.

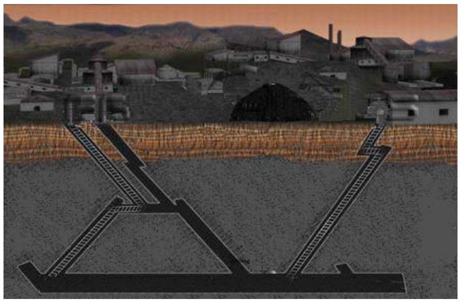


Fig. 1. A coal mine tunnel under ground.

To solve these challenges, we make two important observations. First, in every coal mine, there are some tramcars working in the tunnel. If we attach a sensor on each tramcar, they can serve as mobile sinks to communicate with the static sensors and collect events. Due to the velocity limitation of tramcars, it always takes time for a mobile sink to move back after collecting events. Thus, the mobile sinks do not focus on collecting delay sensitive events (e.g. emergency event as collapse) but delay tolerant events (e.g. event of interest as humidity variance). With these mobile sinks, there is no need for multi-hop transmission among the static sensors. Second, the sensor readings have spatial-temporal correlation: one event may last for a period of time and might be sensed by multiple static sensors. Thus, it is not necessary for a mobile sink to communicate with all the static sensors. Having these observations, the objectives of our system include 1) extending the system lifetime; and 2) guaranteeing small loss rate of delay tolerant events.

In this paper, we first present a general yet meaningful event model which quantifies the data correlation. We then transform the event collection problem in a coal mine to a coverage problem: every event could be represented as a spatial-temporal region of interest. Thus, the target is to cover these event regions and optimize the system lifetime. We prove that this problem is NP-hard even when all event regions are known in advance. In practice, the mobile sink can only know the past event information and has to decide whether to communicate with future static sensors. We propose an online scheme for such scenario. The basic idea of our approach is, we first use a classical Gaussian Process to predict the sensor readings; and then exploit the spatial-temporal correlation to formulate possible event regions. Finally we decide which static sensors should be selected to collect the data with the consideration of load balancing of all static sensors. We show that the event coverage rate is guaranteed in expectation. Extensive simulations on real data traces demonstrate that our scheme significantly extends the system lifetime.

In summary, the contributions of our paper are: 1) to the best of our knowledge, we are the first to investigate the event collection problem in tough communication environment (e.g. underground coal mine) with the help of mobile sinks; 2) we prove that the problem is NP-hard even with the global knowledge of the event regions in advance; 3) we leverage the spatial-temporal correlation to reduce the number of event collection and the expected event coverage rate is guaranteed in theory; 4) extensive simulations on real data trace show that our scheme achieves good performance compared with a stochastic scheme.

The rest part of the paper is organized as follows. Section II presents the related work. An overview of our problem is introduced in Section III. Global optimization of the problem and its NP-hardness will be given in Section IV while Section V is devoted to the design of the scheme for online problem and the theoretical analysis of event coverage rate. We then evaluate the performance of the system in Section VI; and finally Section VII concludes the paper.

II. Related Work

Event collection is an important issue in many sensor applications. Existing works have done a lot of studies on this topic but we face two unique challenges: introducing the node mobility and leverage spatial-temporal correlation.

Some existing works [9][16] have also exploited mobile node to collect events. Jea [9] has considered using multiple mobile nodes (called data mule) for the purpose of event collection. Data mules can move close to the static sensors and collect data from them. To balance the communication workload, an algorithm is proposed to control the movement of data mules. In our problem, however, the mobile nodes (tramcars) have their pre-defined route and is uncontrollable. We can only make decision on when a mobile node pass by a static sensor.

A different scenario in DTF-MSN [16] considered sensor mobility, loose connectivity, fault tolerability, delay tolerability, and buffer limitation. DFT-MSN proposed two schemes "Replication-Based Efficient Data Delivery Scheme (RED)" and "Message Fault Tolerance-Based Adaptive Data Delivery Scheme (FAD)" to achieve the desired data delivery ratio with minimum overhead. Our differences are, first, DFT-MSN does not consider the spatial and temporal correlation among data; and second, the paths of the mobile nodes in DFT-MSN are random walk whereas ours is predetermined.

The spatial-temporal correlation has been heavily studied in data mining community. Vuran et al. [15] have investigated several key elements to capture and exploit the correlation in the wireless sensor networks for the realization of advanced efficient communication protocols. A theoretical framework was developed to model the spatial and temporal correlations. Typical work to leverage spatial-temporal correlation in data aggregation is ISOLINE [14] and Contour maps [12]. In ISOLINE, contour maps are generated to represent the monitored condition. The resulting contour map consists of isolines that delimit regions within which the monitored condition is within a certain value range. High energy efficiency is achieved by taking advantage of the spatial and temporal correlation of the data field. However, ISOLINE focused on data collection in static sensor networks while we introduce mobile node to assist in collection. In addition to these, TiNA [13] exploited temporal correlation in sensor data while CAG [18] took advantage of spatial correlation.

Xing [17] used mobile nodes to exploit the spatial and temporal data correlation, and showed significant performance gain. To guarantee event detection quality, Xing [17] proposed a centralized scheme to design the movement of mobile nodes to minimize the total moving distance. However, in our scenario, the movement path of the mobile sinks is fixed and uncontrollable. In addition, we focus on the energy consumption of the static sensors; the mobility of the mobile sinks is free.

To the best of our knowledge, we are the first to investigate the event collection issue under tough communication environment with the help of mobile sink nodes. With the global knowledge of events given in advance, we prove it is a NP-hard problem. By leveraging the spatial-temporal correlation of events, we propose an online scheme for the mobile sink node to selectively communicate with static sensor nodes when it passes by. The expected event coverage rate can be guaranteed in theory.

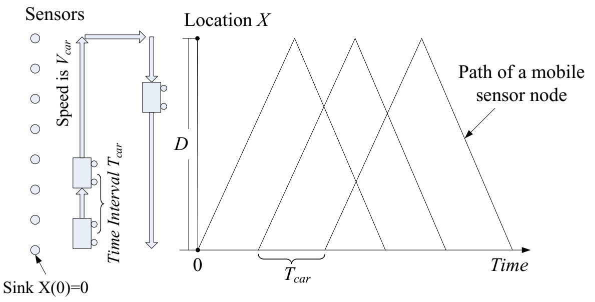
III. System Architecture and The Event Collection Problem

A. Preliminaries

Without loss of generality, we assume the coal mine tunnel is a single line with length D. N static sensors are deployed in the tunnel. Let x_i be the position of sensor i and we assume sensor 0 is the sink with $x_0 = 0$; see Fig. 2(a).

Mobile sinks (e.g. tramcars) move in and out in the tunnel with fixed velocity V_{car} and the time interval for dispatching two continuous mobile nodes is T_{car}. We model the location of mobile sinks in a 2D space with location and time. Figure 2(b) shows an illustration.

Due to the tough communication environment underground, a static sensor node very likely cannot directly communicate with other static nodes. We thus assume that the static sensors will sleep and could only be waken up by mobile sink passively and report the event. We assume that the mobile sink nodes could be recharged when they return to the base station outside. Moreover, they can exchange collected information efficiently. Therefore, we only focus on the energy efficiency of the static sensors. We divide the time into slots, in every time slot, the static sensor will sample readings from the nearby



(a) mobile nodes underground (b) movement path in spatial-temporal space

Fig. 2. A simple view of static and mobile sensor nodes under coal mine.

environment even its communication component is in sleep mode.

Definition 1. *Let $R(x,t)$ be the sensor reading at location x in time slot t. $R(x,t) = (r^1_{x,t}, r^2_{x,t}, ..., r^k_{x,t})$ is a vector containing k types of different sensing value (e.g. temperature, humidity, pressure, etc).*

In this paper, we are interested in some specific events happening underground. Most events could be represented by a predefined range of sensor reading, e.g., temperature higher than 60 degree or the humidity over 80 percent. We define an event as follows:

Definition 2. *Let $E(l_1, l_2, ..., l_k; u_1, u_2, ..., u_k)$ be the specification of an event. For a sensor reading $R(x,t)$, we say the event E happens at location x at time slot t if and only if:*

$$\forall i, 1 \le i \le k, l_i \le r^i_{x,t} \le u_i$$

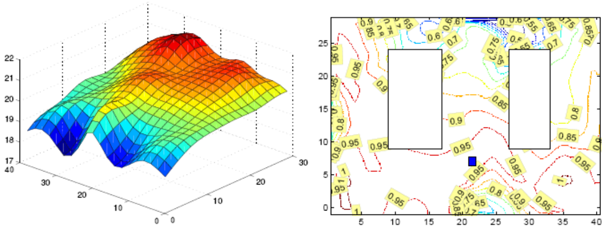
In Def. 2, we summarize an event as a sub-space in the whole k dimension space. An event happens only when the corresponding sensor reading falls into the space of that event definition. A static sensor will store every event it detected in its local buffer. Due to buffer limitation, the node will discard the earliest event when its buffer is full.

Notice that nearby sensors may record the same event in a period of time. To avoid duplicated event collection among these sensor nodes, an understanding of the spatial-temporal correlation of event plays an important role in developing a cost effective solution.

B. Spatial-temporal Correlation of The Events

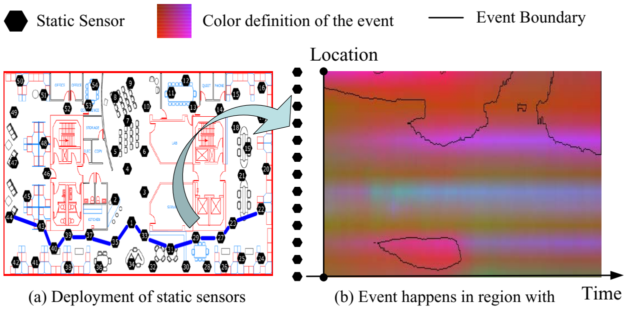
Sensor data are usually correlated. For example, a sensor network is deployed in Intel Berkeley Research lab [2]. It provides three types of sensing observations: temperature, humidity and light, in several days from 54 sensor nodes (the deployment is shown in Fig. 4(a)). Figure 3(a) shows the temperature in a 2D space and the temperature correlation is analyzed in Fig. 3(b) [8]. In the study, sensors are not deployed in two center regions of the lab. As such, temperature correlation in these regions is low.

Since the event is a subspace of the sensor readings, it also has spatial-temporal correlation. For simplicity, in this paper we focus on the correlation of the same type of event but not on the correlation between different types of event.



(a) Temperature Map (b) Temperature Correlations

Fig. 3. Example of the spatial correlation of temperature.



(a) Deployment of static sensors (b) Event happens in region with spatial-temporal correlation

Fig. 4. Example of the spatial-temporal correlation of event.

Fig. 4(a) [2] shows a concrete example by selecting 13 sensor nodes (out of 54 nodes) on a line in the trace data of Intel Berkeley Research lab. Fig. 4(b) maps three types of sensor readings (temperature, humidity and light) into a RGB map. For a given event definition "$25 < temperature < 32$ and $20 < humidity < 26.5$", the event boundary is shown in Fig. 4(b). Different static sensors in that bounded region may detect and record the same event. This example verifies the spatial-temporal correlation of event: an event may cover a region in spatial-temporal space. We call the events happening within the one event region are *identical*, formally,

Definition 3. *Let $E(x,t)$ indicate the event E happens at location x at time slot t. For two happening event $E(x_a,t_a)$ and $E(x_b,t_b)$, we call them "identical" if one of the following conditions hold:*

1) $|x_a - x_b| \le 1$ and $|t_a - t_b| \le 1$

2) There exists an event $E(x_c,t_c)$. $E(x_a,t_a)$ and $E(x_c,t_c)$ are identical while $E_{(}x_c,t_c)$ and $E_{(}x_b,t_b)$ are identical.

We call two events in the same event region if they are identical, otherwise we call them individual events.

In Def. 3, the first condition builds connection between two events if they're close in time and space while the second condition supports the event region as a set of identical events that any two of them could be connected via a sequence of identical events. To further describe the common property of spatial-temporal correlation of an event region, we name the event as "α−β event" if its corresponding event region contains a sub-rectangle region of $\alpha \times \beta$.

Definition 4. *For an event $E(x,t)$, we call it "$\alpha - \beta$ event" if and only if there exists an event $E(x_0,t_0)$ such that:*

$$\forall\ 0 \le i < \alpha, \forall\ 0\ \le j < \beta,$$
$$E(x_0 + i, t_0 + j) \text{ and } E(x,t) \text{ are identical.}$$

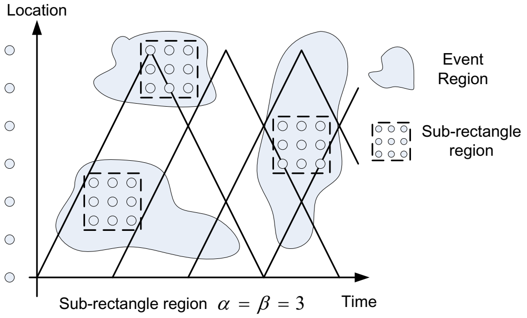


Fig. 5. An example of $\alpha - \beta$ event.

In Definition 4, a special case is that every event is "1 − 1 event". Figure. 5 provides an example that all the events are "3 − 3 event" since every event region has a sub-rectangle region of 3 × 3. According to this definition, as an event region consists of some identical events, the mobile sensor node only needs to collect one event for each event region. In other words, when a mobile sink conducts an event collection action on (x, t), it not only collects the events happened on (x, t) but the events that happened on a spatial-temporal space around (x, t). Intuitively, the communication can be saved and the system lifetime can be extended.

C. Problem Statement

One event collection on (x, t) may cover several different types of event regions (e.g. it may get a fire alarm and an accident of gas leak in one collection). Let $C(x, t)$ be the set of different types of individual events that the event collection action on (x, t) could get, we give the definition of event coverage as follows:

Definition 5. *Let $\Psi = \{(x, t)\}$ be a set of event collection actions and $C\{\Psi\} = \cup_{(x,t)\in\Psi} C(x, t)$ be the set of the individual events that covered by Ψ. We call Ψ provides a prob-Coverage of event with p if:*

$$p = \frac{|C\{\Psi\}|}{|\Omega|} \tag{1}$$

where Ω is the set of all individual events. Particularly, we call Ψ provides a full coverage when $p = 1$.

Once a mobile node collects an event on (x, t), we assume the energy consumed by the corresponding static sensor is $c(x, t)$. Since the energy consumption of idle/sleep mode for static sensor is quite small and can be ignored compared with event notifications[1], we only focus on $c(x, t)$. Let $e(x, t)$ be the residual energy of the static sensor on location x at time t:

$$e(x, t) = \begin{cases} e(x, t-1) - c(x, t) & (x, t) \in \Psi \\ e(x, t-1) & otherwise \end{cases} \tag{2}$$

We use the first depleted sensor as a measurement of the system lifetime. This is widely used in current research as an indication of the end of a stead-state operation. Clearly, the objective of our problem is to design a set of event collection actions Ψ on the movement path of mobile nodes to:

1) Ψ provides *prob*-Coverage of event with p
2) maximize the system lifetime L

$$L = maxof\{t \mid \forall\ 1 \le x \le D,\ e(x, t) \ge 0\}$$

IV. Optimization for Event Collection Problem

In this section, we first analyze the problem globally: all the events are known in advance. We prove that the global optimization is NP-hard. Furthermore, we consider the problem locally and propose a heuristic solution to solve it. Note that, we mainly discuss the optimization for the full coverage (e.g. p=1 in *prob*-Coverage) in this section.

A. Global Optimization: Min-max Cost Set-cover in Group (MCSG)

Given the information of all event regions and movement path of mobile nodes, some global knowledge could be known in advance: 1) the ground truth of Ω, the set of all individual events; 2) a group of family sets $\mathcal{F} = \{\mathcal{F}_x\}$, $\mathcal{F}_x$ is the set of possible event collection actions on location x which depends on the movement path of tramcar; 3) $C(x, t)$ for any possible event collection action (x, t), the cover set of individual events that the event collection action on (x, t) could collect.

For the purpose of selecting a set of event collection actions Ψ from the union of $\mathcal{F}_x$, we need to consider the load balance among all $\mathcal{F}_x$ to extend the system lifetime.

Global Optimization: we study the min-max optimization problem where our goal is to find a set of event collection actions Ψ where

$$\bigcup_{(x,t)\in\Psi} C(x, t) = \Omega$$

such that the minimum residual energy e_x over all static sensor nodes is maximized.

Note that our model is sufficiently general to allow any pattern of event regions and $c(x, t)$ can represent any meaningful communication energy cost function in general. We refer to this problem as the *min-max cost set-cover in group* (MCSG) problem. To facilitate our discussion, we here formally define the decision version of MCSG as follows:

INSTANCE Element set Ω and group of family set $\mathcal{F} = \bigcup \mathcal{F}_x$, a cost $c(x, t) \in R^+$ for each $(x, t) \in \mathcal{F}$ and a bound $B \in R^+$.

QUESTION Is there a subfamily set $\Psi \in \mathcal{F}$ such that:

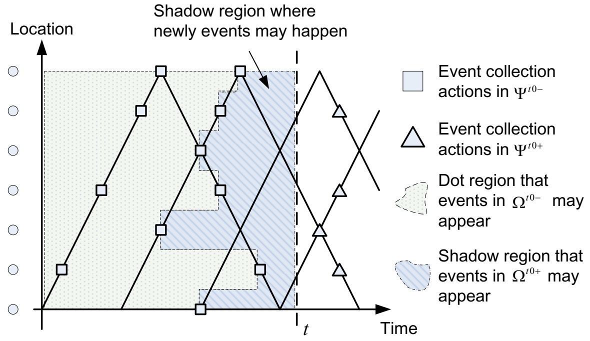
1) $\bigcup_{(x,t)\in\Psi} C(x, t) = \Omega$
2) $\forall x,\ e_x \ \ge\ B$

Claim 1. *MCSG is NP-hard.*

Proof: Consider a special case of MCSG where there is only one family set in $\mathcal{F}$. It turns to be a minimum weighted set-cover problem. If MCSG has a polynomial solution, then its subcase, SET-COVER problem which is well known to be NP-hard [7], would also have a polynomial solution which leads a contradiction. ■

B. Heuristics for Local Optimization

In previous section, we have shown that MCSG is NP-hard in global optimization. A scenario which better reflects practical situations is the one when the information about the events is not known in advance but revealed gradually. It leads us to make the decision to optimize the load balance with local knowledge in steps. In this section, we propose a heuristic solution which is adaptable by increasing the knowledge of events: computation is performed in steps; when a set of events



Shadow region where newly events may happen
Location
Time
t
Event collection actions in Ψ^{t0-}
Event collection actions in Ψ^{t0+}
Dot region that events in Ω^{t0-} may appear
Shadow region that events in Ω^{t0+} may appear

Fig. 6. Local knowledge at time slot t_0.

appear at time slot t_0, the algorithm has to assign them to a set of event collection actions not before t_0 and optimize the load balance. The event collection actions on t_0 will be processed irrevocably when the system move into next time slot.

Formally, let $\Psi = \Psi^{t_0-} + \Psi^{t_0+}$ be the set of existing event collection actions we selected:

$$\begin{aligned} \Psi^{t_0-} &= \{(x,t) | (x,t) \in \Psi, t < t_0\} \\ \Psi^{t_0+} &= \{(x,t) | (x,t) \in \Psi, t \geq t_0\} \end{aligned} \tag{3}$$

where Ψ^{t_0-} and Ψ^{t_0+} are the sets of event collection actions that have been processed and not processed yet respectively. When a set of events $\Omega(t_0)$ appears at time slot t_0, let Ω^{t_0-} and Ω^{t_0+} be the set of events that have been collected and not collected yet:

$$\begin{aligned} \Omega^{t_0-} &= (\Omega(t_0) \ \cup \ C(\Psi)) \ \cap \ C(\Psi^{t_0-}) \\ \Omega^{t_0+} &= (\Omega(t_0) \ \cup \ C(\Psi)) \ \setminus \ C(\Psi^{t_0-}) \end{aligned} \tag{4}$$

Let $(x, g^{t_0}(x))$ be the last event collection action happened on location x in Ψ^{t_0-}. Clearly, the events in Ω^{t_0+} may happen in the shadow region that between $g^{t_0}(x)$ and t_0 as shown in Fig. 6. Assume $\mathcal{F}^{t_0+}$ is the set of possible event collection actions not before t that could cover events in Ω^{t_0+}. For the purpose to cover the newly known events added in Ω^{t_0+}, the algorithm has to design a new set of event collection actions from $\mathcal{F}^{t_0+}$ to replace the old Ψ^{t_0+} and optimize the current communication load balance.

Local Optimization: at time slot t_0, we study the local min-max optimization problem where our goal is to find a set of event collection actions $\Psi^{t_0+} \subseteq \mathcal{F}^{t_0+}$ where

$$\bigcup_{(x,t)\in\Psi^{t_0+}} C(x,t) = \Omega^{t_0+}$$

such that the minimum residual energy $e(x, t_0)$ over all static sensor nodes is maximized.

Similar to global optimization, we could formally define the decision version of local optimization as follows:

INSTANCE Element set Ω^{t_0+} and group of family set $\mathcal{F}^{t_0+} = \bigcup_x \mathcal{F}_x^{t_0+}$, a cost $c(x,t) \in R^+$ for each $(x,t) \in \mathcal{F}^{t_0+}$, an irrevocable set Ψ^{t_0-} and a bound $B \in R^+$.

QUESTION Is there a subfamily set $\Psi^{t_0+} \in \bigcup \mathcal{F}_x$ such that:

1) $\bigcup_{(x,t)\in\Psi^{t_0+}} C(x,t) = \Omega^{t_0+}$
2) $\forall x, \ e(x,t_0) \ \geq \ B$

Algorithm 1 ***HeuristicCollect***(), design a set of event collection actions Ψ^{t_0+} to collect the events in Ω^{t_0+}.

```
Require: $\Psi^{t_0-}$, $\Omega^{t_0+}$, $\mathcal{F}^{t_0+}$, $C$
Ensure: $\Psi^{t_0+}$
$\Omega^* \Leftarrow \phi$, $\Psi^* \Leftarrow \phi$, $B^* \Leftarrow 0$;
while $\Omega^{t_0+} \subsetneq \Omega^*$ and $\Psi^* \neq \mathcal{F}^{t_0+}$ do
    for $(x,t)$ in $\mathcal{F}^{t_0+}$ do
        if $C(x,t) \cap \Omega^* \neq \phi$ then
            Compute $\triangle_e(x,t)$ and $w(x,t)$;
        else
            $\triangle_e(x,t) \Leftarrow \infty$, $w(x,t) \Leftarrow \infty$;
        end if
    end for
    Select $(x\prime, t\prime)$ with smallest $\triangle_e(x\prime, t\prime)$ (as primary key)
    and $w(x\prime, t\prime)$ (as secondary key);
    $\Psi^* = \Psi^* \cup \{(x\prime, t\prime)\}$;
    $\Omega^* = \Omega^* \cup C(x\prime, t\prime)$;
end while
$\Psi^{t_0+} \Leftarrow \Psi^*$;
return $\Psi^{t_0+}$;
```

By comparing the global and local decision problem, it can be easily to see that the local optimization is still MCSG which is the combination of load balancing and set cover problem. Existing works on load balancing [6] and set cover [4] have shown that greedy is a good approach to approximate the optimum solution. We propose a heuristic approach by taking both load balancing and cover-efficiency into the consideration. Let's consider a snapshot in choosing event collection actions from $\mathcal{F}^{t_0+}$: assume Ψ^* is the current set of event collection actions that have been chosen from $\mathcal{F}^{t_0+}$ and $\Omega^* = C(\Psi^*) \ \cap \ \Omega^{t_0+}$. Let $e^*(x)$ be the residual energy; and the current minimum residual energy is recognized as B^*:

$$\begin{aligned} e^*(x) &= e(x,t_0) - \textstyle\sum_{(x,t)\in\Psi^*} c(x,t) \\ B^* &= \min_{0\leq x\leq D}\{e^*(x)\} \end{aligned} \tag{5}$$

For an event collection action (x,t) in $\mathcal{F}^{t_0+} \setminus \Psi^*$ which could cover the event not in Ω^*, if we select (x,t) into Ψ^* it could contribute an decrement $\triangle_e(x,t)$ to current B^*:

$$\triangle_e(x,t) = \min\{e^*(x) - c(x,t), \ B^*\} - B^* \tag{6}$$

Since the definition of system lifetime relies on the minimum residual energy B^*, we greedily choose the one with smallest $\triangle_e(x,t)$ to current Ψ^*. If there are multiple choices with same $\triangle_e(x,t)$, we choose the one with the highest the cover-efficiency $w(x,t)$:

$$w(x,t) = \frac{c(x,t)}{|(C(x,t) \cap \Omega^{t_0+}) \setminus \Omega^*|} + \frac{c(x,t)}{\lambda(t-t_0)} \tag{7}$$

where the first item in Equ. (7) indicates the average energy cost paid to cover a new event in Ω^{t_0+}. Note that (x,t) does not only cover the events happened before t_0 but might cover events that happened between t_0 and t. Though the number of events happened between t_0 and t in future is unknown at present, intuitively it is linear related to the time interval as demonstrated in the second item.

In Algorithm 1, the mobile node on (x, t_0) has to design a set of event collection actions Ψ^{t_0+} to cover Ω^{t_0+}. Ψ^{t_0+} is computed by iterations. In each iteration, the mobile node chooses the

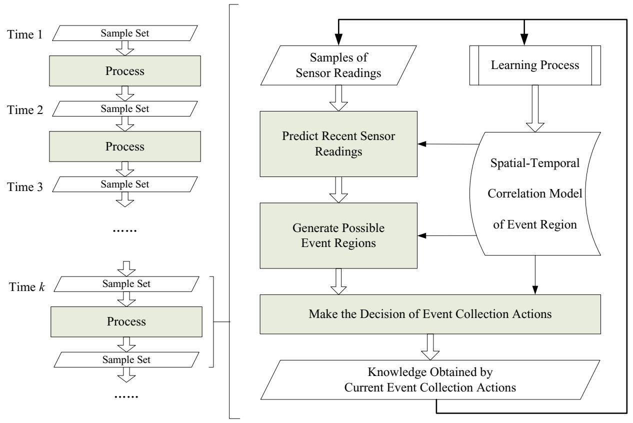

Time 1
Sample Set
Process
Time 2
Sample Set
Process
Time 3
Sample Set
......
Time k
Sample Set
Process
Sample Set
......
Samples of Sensor Readings
Learning Process
Predict Recent Sensor Readings
Spatial-Temporal Correlation Model of Event Region
Generate Possible Event Regions
Make the Decision of Event Collection Actions
Knowledge Obtained by Current Event Collection Actions

Fig. 7. The framework of online event collection scheme.

event collection action that contribute the smallest $\triangle_C(x,t)$ to current load balance. The coverage efficiency $w(x,t)$ is also considered.

As a summary of this section, we focus on the full coverage problem of event collection. For global optimization, even when the information of all the events are known in advance, we prove it is NP-hard. We also consider a more practical situation, the event information is known gradually, which leads a local optimization problem and the mobile node has to make the decision on whether to communicate with the static sensor when it passes by. We propose a heuristic algorithm which is adaptable to be implemented for the online scheme.

In this section, we make an assumption that the event information are known to us no matter it is given gradually or all in one time and we just need to design a load balancing scheme to collect them. However, it is not realistic since before querying we are not sure the event will happen in practical system. In other words, we could not get Ω^{t_0+} directly as the input of Algorithm 1. Fortunately, according to the existing event information known by processed event collection actions we could be a foreseer to predict the uncollected events as shown in the following section.

V. Online Event Collection Scheme

In this section, we mainly discuss how to get the knowledge of the events that happened recently (e.g. the knowledge of events happened in the shadow region of Fig. 6). We will first introduce the general framework of our online scheme; and then get into the details of predicting recent sensor readings and generating possible event regions.

A. General Framework

To be consistent with practical system, the event collection could be described as a circle-processing: predicting events from existing samples, making the decision of event collection actions, and collecting events as new samples. Our scheme is designed for such online case and the framework is shown in Fig. 7.

The process phase in one time slot in Fig. 7 consists of three major steps and is self-explanatory. We can also interpret it in the following way: at every time slot, a mobile node first predicts the recent sensor readings based on existing samples; it then computes the possible event regions that are necessary to be covered based on prediction; and finally, the mobile nodes make the decision of event collection actions to achieve load balance. The remaining task is to show the detailed design of each step in the following subsections.

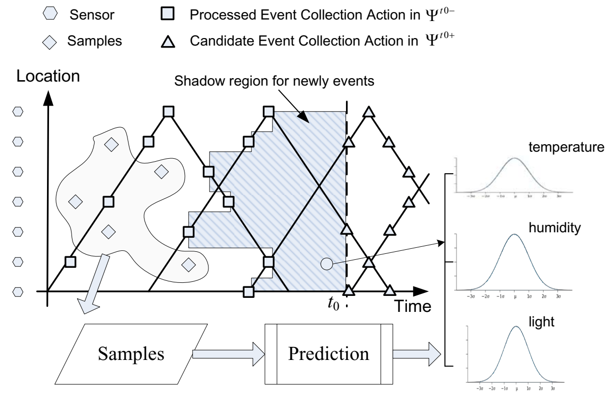

Sensor
Samples
Processed Event Collection Action in Ψ^{t0-}
Candidate Event Collection Action in Ψ^{t0+}
Location
Shadow region for newly events
temperature
humidity
light
t_0
Time
Samples
Prediction

Fig. 8. Illustration of predicting the sensor reading on (x,t) in S.

B. Prediction

Let's recall Fig. 6 in previous section. At time slot t_0, the shadow region S which is a gap between last collection time and current time slot could be represented as:

$$\begin{array}{lcl} g^{t_0}(x) & = & maxof\{t \mid (x,t) \in \Psi^{t_0-}\} \\ S & = & \{(x,t) \mid g^{t_0}(x) < t \leq t_0\} \end{array} \quad (8)$$

To get the knowledge about newly events after last collection, the mobile node has to forecast the sensor readings of all (x,t) in S according to the samples of past sensor readings that have been collected before.

For simplification, we first focus on predicting the sensor reading of only one type of measurement (e.g. temperature). The prediction has been well studied in [5]. Guestrin [8] provided a simple, yet often effective approach to predict in Gaussian process: the predicting value of sensor measurement is a joint Gaussian distribution of a set of random variables according to existing samples.

$$P(X=x) = \frac{1}{(2\pi)^{n/2}|\Sigma|} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \quad (9)$$

Let A be a set of sample sensor readings, we can predict the sensor reading at any spatial-temporal point $y=(x,t)$ conditioned on these samples, $P(R_y|R_A)$. Due to the spatial-temporal correlation of sensor readings, the distribution of R_y given these samples is a Gaussian whose conditional mean $\mu_{y|A}$ and variance $\sigma^2_{y|A}$ are given by:

$$\begin{array}{ll} \mu_{y|A} & = \mu_y + \Sigma_{yA}\Sigma^{-1}_{AA}(x_A - \mu_A) \\ \sigma^2_{y|A} & = \kappa(y,y) - \Sigma_{yA}\Sigma^{-1}_{AA}\Sigma_{Ay} \end{array} \quad (10)$$

Equation (10) could be extended to the sensor reading consists of multiple types of measurements. For the sensor reading on (x,t) of ith type measurement, we could obtain a probability density function $f^{(x,t)}(r_i)$ of Gaussian distribution $N(\mu_i^{(x,t)},\ \sigma_i^{(x,t)})$. Please refer [8] for the detail of prediction algorithm.

C. Initialization of Events to Cover

After prediction, for any point (x, t) in S we have a set of probability density functions $\{f^{(x,t)}(r_i) \mid 1 \le i \le k\}$ for k types of measurement. Considering the confidence interval of the Gaussian distribution for a single type of measurement, for the size of confidence interval η, the probability that the value of measurement are within the range of $[\mu_i^{(x,t)} - \eta\sigma_i^{(x,t)}, \mu_i^{(x,t)} + \eta\sigma_i^{(x,t)}]$ is $\theta = erf(\eta/\sqrt{2})$.

For the sensor reading $R = (r_1, ...r_k)$ on (x, t), let $\mathcal{K}_\eta^{(x,t)}(R)$ denote whether R is in the range of confidence:

$$\mathcal{K}_\eta^{(x,t)}(R) = \begin{cases} 1 & \forall 1 \le i \le k,\ -\eta\sigma_i^{(x,t)} \le r_i - \mu_i^{(x,t)} \le \eta\sigma_i^{(x,t)} \\ 0 & otherwise \end{cases} \tag{11}$$

meanwhile, we use $\mathcal{H}^{(x,t)}(R)$ to represent whether R indicates an event from Definition 2:

$$\mathcal{H}^{(x,t)}(R) = \begin{cases} 1 & \exists E, \forall 1 \le i \le k\ l_i \le r_i \le u_i \\ 0 & otherwise \end{cases} \tag{12}$$

By combining Equ. (11) and Equ. (12), the intersection $\mathcal{I}_\eta^{(x,t)}$ between confidence interval and event definition could be claimed as:

$$\mathcal{I}_\eta^{(x,t)} = \int \cdots \int \mathcal{K}_\eta^{(x,t)}(R)\mathcal{H}^{(x,t)}(R)\ dr_1...dr_k \tag{13}$$

To generate the possible event set Ω^{t_0+} which needs to be covered by future event collection actions, we put (x, t) in Ω^{t_0+} if $\mathcal{I}_\eta^{(x,t)} > 0$:

$$\Omega^{t_0+} = \{ (x, t) \mid (x, t) \in S,\ \mathcal{I}_\eta^{(x,t)} > 0\} \tag{14}$$

Lemma 1. *For any "$\alpha - \beta$ event", the expected probability that one identical event will be put into Ω^{t_0+} is at least:*

$$P_\Omega(\alpha,\beta,\eta) \ge 1 - (1 - erf(\eta/\sqrt{2})^k)^{\alpha\beta} \tag{15}$$

Please refer [11] for the detail proof of Lemma and Theorem in this paper. From Lemma 1, we could claim that if η satisfy the following condition,

$$\eta \ge \eta^* = \sqrt{2} \times erf^{-1}((1 - (0.001)^{\frac{1}{\alpha\beta}})^{\frac{1}{k}}) \tag{16}$$

then the identical event will be put into Ω^{t_0+} almost certainly with probability $P_\Omega(\alpha,\beta,\eta) \ge 0.999$. It implies that the contribution of η to the coverage rate will reach a bound when η grows larger than certain threshold η^*. We will verify it in the evaluation part.

D. Capability of Event Collection Action

Before selecting some event collection actions from further path $\mathcal{F}^{t_0+}$ to cover Ω^{t_0+}, the mobile node needs to know $C(x, t)$, the capability of every event collection action, which denotes the set of events that could be collected by the event collection action on (x, t). $C(x, t)$ relies on two factors: buffer limitation on static sensor and the spatial-temporal correlation of event.

The local buffer on static sensor enables the mobile node collect the event happened within a long time. Assume the static sensor node could store at most q recent events in local buffer. We use $P_{buf}(t - t^*, q)$ to represent the expected probability that the event happened in time t^* could be collected by event collection action in time t. P_{buf} decreases as $t - t^*$ increases or q decreases. It could be modeled in the learning process in Fig. 7. Due to the spatial-temporal correlation, the event collection action may not only collect the event happened on its location but also collect the event that happened on nearby locations.

Algorithm 2 Mobile node on (x, t_0) makes the decision to collect do event collection action.

```
Require: $\Psi^{t_0-}$, $\mathcal{F}^{t_0+}$
$\{f_{x,t}\}$ ⇐ Prediction in Gaussian process;
$\Omega^{t_0+}$ ⇐ Initialization of events;
$C$ ⇐ Generate capability of event collection actions;
$\Psi^{t_0+}$ ⇐ HeuristicCollection();
if $(x, t_0)$ in $\Psi^{t_0+}$ then
    Do event collection action;
end if
Move to next step;
```

Since different events may have different "$\alpha - \beta$" properties, we select a uniform α^* and β^* so that the probability for an event to be "$\alpha^* - \beta^*$ event" is $P_{event}(\alpha^*,\beta^*)$. By considering the "$\alpha^* - \beta^*$" property, we set $C(x, t)$ as:

$$\begin{aligned} C(x, t) \;=\; \{ (x\prime, t\prime) \mid \;& 0 \le t - t\prime < \beta^* + \gamma,\ |x - x\prime| \le \tfrac{(\alpha^*-1)}{2}\}, \\ & (x\prime, t\prime) \in \Omega^{t_0+}\} \end{aligned} \tag{17}$$

Lemma 2. *For "$\alpha^* - \beta^*$ event" in Ω^{t_0+}, the probability that at least one identical event could be collected by Ψ^{t_0+} computed in Algorithm 1 based on C in Equ. (17) is no less than $P_{buf}(\gamma, q)$.*

Theorem 3. *The system provides prob-Coverage of event with expected probability p:*

$$p \ge P_{event}(\alpha^*,\beta^*) \cdot P_\Omega(\alpha^*,\beta^*,\eta) \cdot P_{buf}(\gamma, q) \tag{18}$$

In this section, we introduce the online event collection problem: at every time slot t_0, with the knowledge of past collected events and further movement path, the mobile node has to make the decision whether to do event collection action when it passes by (x, t_0).

A straight forward solution to this online problem is a *stochastic* collection scheme that mobile nodes do event collection action with probability P_s at every time slot:

$$P_s = N \cdot T_{car}/(\alpha^*(\beta^* + \gamma)(2N - 1)) \tag{19}$$

Instead of *stochastic* collection scheme, we propose a local *MCSG* heuristic solution of online scheme. As summarized in Algorithm 2, we first use Gaussian process to predict the unknown sensor readings; the possible events Ω^{t_0+} are then obtained and the capability of event collection action $C(x, t)$ is generated by "$\alpha^* - \beta^*$" property which reflects the spatial-temporal correlation of events; the set of event collection actions Ψ^{t_0+} to cover Ω^{t_0+} could be computed by Algorithm 1 with heuristic approach; finally, the mobile node could make the decision by checking whether (x, t_0) is in Ψ^{t_0+}.

VI. Performance Evaluation

A. Simulation Settings

The default values of our simulation are as follows. We deploy 50 static sensor nodes uniformly in the tunnel of 2000m length. The velocity of mobile node V_{car} is 4m/s. The

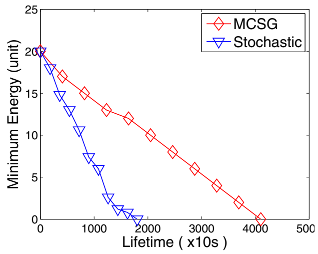


Fig. 9. Minimum Energy decreases as a function of Time.

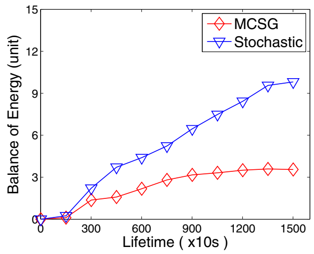


Fig. 10. Energy Balance as a function of Time.

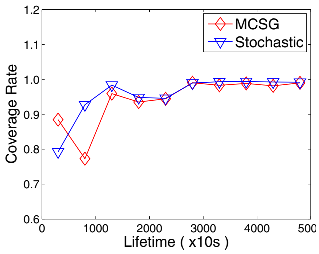


Fig. 11. Coverage Rate as a function of Time.

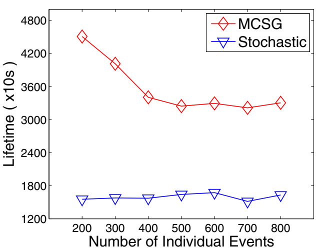


Fig. 12. Lifetime as a function of the number of events.

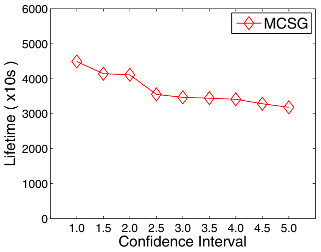


Fig. 13. Lifetime as a function of η.

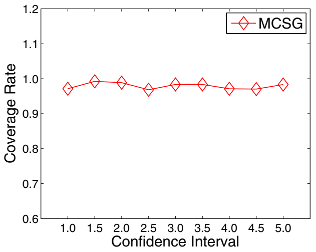


Fig. 14. Coverage Rate as a function of η.

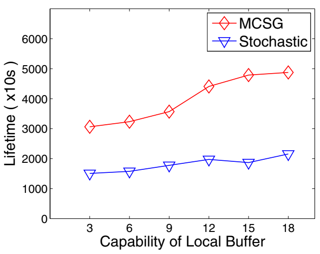


Fig. 15. Lifetime as a function of γ.

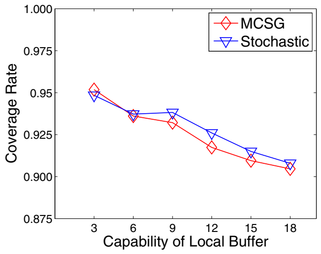


Fig. 16. Coverage Rate as a function of γ.

time interval for dispatching two continuous mobile nodes T_{car} is 100 seconds. The communication cost for each event collection action $c(x,t)$ is 1 unit and the initial energy capacity of every static sensor node $e(x,0)$ is 20 units. The system lifetime is the first sensor node that the energy is depleted. We use the standard deviation $\sigma(t)$ of the residual energy to represent the energy balance among static sensor nodes at time t.

$$\sigma(t) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(e(i,t) - \overline{e(t)})^2} \tag{20}$$

We set one time slot as 10 seconds and evaluate both the MCSG and stochastic scheme within 5000 time slots. Three types of sensor measurements (e.g. temperature, humidity and light) are pre-generated from partial real trace data in [2] by interpretation. For the event definitions, we define 50 different types of event by default. Based on these definitions, there're 308 individual events in total in the ground truth set. Every static sensor node can store at most $q = 10$ recent events in its local buffer.

In MCSG scheme, we apply $\eta = 2.0$ and $\gamma = 5$ by default. Both MCSG and stochastic schemes automatically adjust α^* and β^* to achieve the *prob*-Coverage of event with $p = 0.9$. We will also show the art of selecting η and γ to guarantee the desired p while extending the network lifetime in the following section.

B. Network Lifetime and Energy Balance

We first compare the general performance, network lifetime and energy balance, handled by MCSG and stochastic scheme. In Fig. 9, Y-axis stands for the minimal residual energy of the static sensor nodes in the network. Clearly, the minimum energy of the stochastic scheme decreases much faster and the system lifetime is only around $1.75 \times 10^4 s$. Our MCSG algorithm has a system lifetime of $4 \times 10^4 s$, a more than two-time improvement.

Fig. 10 illustrates that MCSG better balances the residual energy of the sensor nodes. The standard deviation of the residual energy is very small compared with the stochastic scheme. More interestingly, as time evolves, the increase of the standard deviation almost flattened. This shows that our MCSG is able to scale better. Fig. 11 further verifies the coverage rate. We provide sufficient initial energy of static sensor nodes (e.g. $e(x,0) = 100$) to have a long run of the system. After a startup phase of collecting sufficient samples, MCSG could adjust suitable parameters and provides a stable coverage rate of the event.

C. The Number of Individual Events

We see in Fig. 12 that the system lifetime decreases as the number of individual events increases. This is because when we add new event definition to generate more individual events, the mobile nodes have to do additional event collection actions so the system lifetime will be decreased.

In the worst case that the individual events happened on every (x,t) in the region, the mobile nodes would have a set of event collection actions. In other words, this set of event collection actions actually covers the whole spatial-temporal region which provides a bottom line of lifetime. Therefore, it is not surprising to see that the network lifetime of MCSG converges when the number of individual events is greater than 500 in Fig. 12. For the simple stochastic scheme, the probability that the mobile node decides to do event collection action does not rely on the number of individual events, so the network lifetime does not vary. From Fig. 12, the system lifetime of our MCSG is still twice of stochastic scheme even in the worst case.

D. Confidence Interval of Prediction

In Section 5.3, we initialize the events from the prediction of sensor readings with a confidence interval $[\mu - \eta\sigma, \mu - \eta\sigma]$ with probability $\theta = erf(\eta/\sqrt{2})$. Fig. 13 demonstrates that the system lifetime decreases when η increases. The reason is that for larger η, the confidence interval is wider and more number of sensor readings would likely to be predicted as an event and put into the cover set Ω^{t_0+}. Therefore, more event collection actions would be required to cover the events predicted by larger confidence interval and the system lifetime decreases.

In the simulation case, the system has $k = 3$, $\alpha^* = 4$ and $\beta^* = 30$. From Equ. (16), it is easy to compute that when η grows greater than $\eta^* = 0.5$, the coverage rate would nearly not be changed. The simulation in Fig. 14 verifies this feature. In other words, with the knowledge of k, α^* and β^*, the system could manually set $\eta = \eta^*$ to extend the network lifetime without loss of coverage rate.

E. Capability of Local Buffer

We use γ to expect how long an event could be stored in local buffer. As demonstrated by Equ. (17) and Equ. (19)in Section 5.4, the capability of event collection action could be enhanced by increasing γ in both MCSG and simple stochastic scheme. That is to say, larger γ expects event collection actions to collect more events and the system lifetime could be extended.

We set 1000 individual events for the mobile nodes to collect and increase γ from 3 to 18. Fig. 15 verifies that the system lifetime of MCSG and stochastic scheme both increase. On the other hand, Theorem 3 illustrates that γ also plays an important role in the event coverage rate. From Equ. (18, we could conduct that the coverage rate will decrease as γ increases which is verified by Fig. 16. Since when the mobile nodes expect a larger γ which exceeds the real capability of local buffer, the events maybe lost due to buffer limitation. Therefore, γ actually reflects the tradeoff between system lifetime and coverage rate. We could increase γ with reasonable value(e.g. related to buffer limitation and event frequency) to prolong the system lifetime and provide desired coverage rate at the same time.

VII. Conclusion

In this paper, we for the first time investigate the event collection issue in tough communication environment such as the coal mine. The event collection underground is challenging as we face imbalance energy consumption from a linear network topology and traditional multi-hop transmission cannot efficiently report event due to the high packet loss rate. In addition, the events have spatial-temporal correlation. In this paper, we leverage the sensor nodes attached on tramcars working underground as mobile sinks to collect the events. The mobile nodes selectively communicate with static sensor nodes to collect the event while guarantee the event collecting rate and balance the energy consumption among all the static sensor nodes. We proved that even with global information of all the events known in advance, the problem is NP-hard. For practical situation that only past event information are known and the mobile nodes have to make the event collection action decision instantly when it passes by a static sensor node, we proposed an online algorithm for MCSG. The algorithm naturally addressed a set of difficulties, such as the uncertainty of recent event, buffer limitation on static sensor nodes, and the balance of energy consumption. We proved that the algorithm guarantees an expected coverage rate of event. Our simulation results demonstrated that MCSG can extend the system lifetime over a simple stochastic scheme significantly.

Acknowledgement

The research was support in part by grants from RGC under the contracts CERG 622407 and 622508, the HKUST Nansha Grant NRC08/10.EG01, the Huawei-HKUST joint lab, the grant from ITC HK ITP/023/08LP, the grant from National Basic Research Program of China (973 Program) under the contract No. 2006CB303100, the foundation of scientific and technological planning project of Nansha District, Guangzhou, China, HKPU/ICRG under Grant G-YG78, A-PB0R and RGC/GRF under Grant 5305/08E.

References

[1] MICAz Datasheet, Crossbow Technology Inc. http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf.

[2] Intel Berkeley Research Lab Trace Data. http://db.csail.mit.edu/labdata/labdata.html.

[3] I. F. Akyildiz and E. P. Stuntebeckm "Wireless underground sensor networks: Research challenges", Ad Hoc Networks (Elsevier), Vol. 4, No. 6, pp. 669-686, November 2006.

[4] G. Ausiello, A. Giannakos and VT. Paschos, "Greedy algorithms for on-line set-covering and related problems", In *Proc. of the 12th Computing: The Australasian Theroy Symposium*, 2006.

[5] P. Brockwell and R. Davis. "Introduction to Time Series and Forecasting". Springer, 1994.

[6] P. Crescenzi, G. Gambosi, G. Nicosia, P. Penna and W. Unger, "On-line load balancing made simple: Greedy strikes back", Journal of Discrete Algorithms, 2007.

[7] M. R. Garey and D. S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness", W.H. Freeman and Company, 1979.

[8] C. Guestrin, A. Krause, and A. Singh, "Near-optimal Sensor Placements in Gaussian Processes", In *Proc. of ICML*, 2005.

[9] D. Jea, A. A Somasundara and M. B Srivastava, "Multiple Controlled Mobile Elements (Data Mules) for Data Collection in Sensor Networks," In *Proc. of IEEE/ACM International Conference on Distributed Computing in Sensor Systems*, June 2005.

[10] M. Li and Y. Liu, "Underground Structure Monitoring with Wireless Sensor Networks", In *Proc. of IPSN*, Cambridge, USA, April 2007.

[11] J. Luo, Q. Zhang and D. Wang, "Delay Tolerant Event Collection for Underground Coal Mine using Mobile Sinks", Technique Report, CSE Department of HKUST, 2008.

[12] X. Meng, T. Nandagopal, L. Li and S. Lu, "Contour Maps: Monitoring and Diagnosis in Sensor Networks" Computer Networks, 2006.

[13] M. A. Sharaf, J. Beaver, A. Labrinidis and P. K. Chrysanthis, "TiNA: A Scheme for Temporal Coherency-Aware in- Network Aggregation, In *Proc. of MobiDe*, September 2003.

[14] I. Solis and K. Obraczka, "Isolines: efficient spatio-temporal data aggregation in sensor networks", Wireless Communications and Mobile Computing, 2007.

[15] M. C. Vuran, O. B. Akan and I. F. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks", Computer Networks Journal (Elsevier), Vol. 45, No. 3, pp. 245-261, June 2004.

[16] Y. Wang and H. Wu. "DFT-MSN: The Delay Fault Tolerant Mobile Sensor Network for Pervasive Information Gathering", In *Proc. of INFOCOM*, Barcelon, April 2006.

[17] G. Xing, J. Wang, K. Shen, Q. Huang, X. Jia, and H. C. So, "Mobility-assisted Spatiotemporal Detection in Wireless Sensor Networks", In *Proc. of ICDCS*, Beijing, China, 2008.

[18] S. Yoon and C. Shahabi, "Exploiting Spatial Correlation Towards an Energy Efficient Clustered Aggregation Technique (CAG)", In *Proc. of the International Conference on Communications*, 2005.