

# On the Double Mobility Problem for Water Surface Coverage with Mobile Sensor Networks

Ji Luo, *Member, IEEE*, Dan Wang, *Member, IEEE*, and Qian Zhang, *Senior Member, IEEE*

**Abstract**—We are interested in the sensor networks for scientific applications to cover and measure statistics on the sea surface. Due to flows and waves, the sensor nodes may gradually lose their positions; leaving the points of interest uncovered. Manual readjustment is costly and cannot be performed in time. We argue that a network of mobile sensor nodes which can perform self-adjustment is the best candidate to maintain the coverage of the surface area.

In our application, we face a unique *double mobility* coverage problem. That is, there is an uncontrollable mobility, U-Mobility, by the flows which breaks the coverage of the sensor network. Moreover, there is also a controllable mobility, C-Mobility, by the mobile nodes which we can utilize to reinstall the coverage. Our objective is to build an energy efficient scheme for the sensor network coverage issue with this double mobility behavior.

A key observation of our scheme is that the motion of the flow is not only a curse but should also be considered as a fortune. The sensor nodes can be pushed by free to some locations that potentially help to improve the overall coverage. With that taken into consideration, more efficient movement decision can be made. To this end, we present a dominating set maintenance scheme to maximally exploit the U-Mobility and balance the energy consumption among all the sensor nodes. We prove that the coverage is guaranteed in our scheme. We further propose a fully distributed protocol that addresses a set of practical issues. Through extensive simulation, we demonstrate that the network lifetime can be significantly extended, compared to a straight forward back-to-original reposition scheme.

**Index Terms**—Coverage, mobile sensor networks, double mobility, dominating set.

## 1 INTRODUCTION

SENSOR networks today are penetrating into people's life in a stunning speed. The capabilities of the sensor nodes have gone broader beyond a static sensing device; and now they include mobility, communication, computation, etc. They provide valuable information that is originally difficult or impossible to obtain in all domains. To name a few, we have recently seen the design and deployment of applications such as the structural health monitoring for roads and bridges [18], monitoring of ecological systems (e.g., redwood trees), measurement and data collection of active volcanos [8].

In this work, we are interested in applications which need to collect data in a 2D area of interest on the sea surface. We take a scientific application, collecting salt level information at the mouth of a glacier, as an example scenario. The data are valuable since the measurement of salt level can serve a good indicator for the melt speed during different seasons for the research of ocean scientist. Clearly, we need to have the mouth of the glacier area monitored for such application. Different to underwater applications, in this work, sensor nodes are

deployed floating on the sea surface to form a network that covers the sea surface and reports the data.

The coverage issue for sea surface monitoring applications is challenging and in sharp contrast to static sensor coverage problems, as the sensor nodes face the continuous motion of the flows and waves. In some existing applications, drifters and moorings are used to build the sensor network [1] for water surface monitoring. If unattended, a drifter [2] cannot keep its position; thus not suitable for monitoring a fixed area. A mooring [4] is anchor at the sea bottom and is able to stay put. The deployment, however, is usually more complicated and requires professional technicians. In addition, it is restricted in shallow waters only. These solutions are thus not adequate to achieve an autonomous coverage for sea surface monitoring sensor networks.

We consider sensor nodes with movement capability to be the best candidate to form the sensor network for our applications. Some existing studies have used mobile sensor nodes to assist area coverage on land. We are different, however, as we face a *double mobility coverage problem*. On one hand, the sensor nodes will be affected by the motion of flows and waves. They will gradually lose their positions. This mobility is uncontrollable (we call it *U-Mobility* thereafter) and the coverage of the sensor network may be broken by the U-Mobility. On the other hand, the mobile sensor nodes have the movement capability to reinstall the network coverage. This mobility is controllable (we call it *C-Mobility* thereafter).

The sensor nodes are battery powered and the most stringent resource is energy. In our application, the sen-

- 
- J. Luo and Q. Zhang are with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong.  
E-mail: {luoji, qianzh}@cse.ust.hk.
  - D. Wang is with the Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.  
E-mail: csdwang@comp.polyu.edu.hk.

sensor network has to monitor the area of interest for a long period of time. If the battery of a sensor node is depleted, a manual battery replacement by sending a technician is expensive. Thus, the frequency of this operation should be minimized. For the mobile sensor nodes, the dominant factor of the energy consumption is the mechanical movement, i.e., the C-Mobility. Consequently, the objectives of our system include 1) the sensor nodes should collaboratively monitor and guarantee coverage of the area of interest; 2) the movement of the sensor nodes due to C-Mobility should be minimized and the power consumption should be balanced in the long run; and 3) the sensor network should be able to adapt to different U-Mobility patterns; as there is no universal U-Mobility model that can capture the behavior of all sea.

We achieve the aforementioned objectives by *considering the interaction between the U-Mobility and the C-Mobility*. Our key observation is that the U-Mobility should not only be considered as a curse that the C-Mobility should always counter-effect, but also a fortune. With U-Mobility, some sensor nodes may be pushed to positions that may improve the coverage. Thus, less C-Mobility is needed if taking this into consideration. We first discuss two general yet realistic U-Mobility models, namely, the meandering current model [11] and a random ring model [3] which may capture the mobility of different water bodies. Our C-Mobility is designed in a way such that it can fit for different U-Mobility models. In our C-Mobility scheme, each sensor node dominates (maintains) a set of points that it covers; and the sensor network dominates all the points of interest. We design a distributed event driven algorithm to maintain the coverage. When a point of interest is not covered, the algorithm finds a substitute by a joint optimization with consideration of the velocity constraints of the sensor nodes, the balance of power consumption and maximizing the advantage of U-Mobility. A low-overhead protocol is designed thereafter which addresses a set of practical difficulties such as collisions and local inconsistent views of the sensor nodes introduced by the distributed algorithm. We prove that the coverage is guaranteed based on our scheme. Extensive simulations have shown that our scheme significantly outperforms a straightforward back-to-original scheme for different U-Mobility models under various configurations.

As a summary, the contributions of the work are 1) We are the first to address the coverage issue for the double mobility scenario. We provide key observation that although U-Mobility brings significant challenge, we can also leverage it to solve the coverage issue in a more efficient way. 2) We formulate the double mobility problem, and propose a distributed dominating set maintenance approach to solve it. 3) We propose a practical distributed protocol and conduct comprehensive simulation which verifies the effectiveness of the proposed scheme.

The remaining part proceeds as follows. Section 2 presents the related work. An overview of the system

design and challenges is presented in Section 3. Section 4 is devoted to the design and optimization of the detailed sea surface coverage solutions. Protocol specifications are described in Section 5. We evaluate the performance of the system in Section 6; and finally Section 7 concludes the paper.

## 2 RELATED WORK

Sensor network today has been applied to many applications for data collection on land [19] [26]. Collecting data to better understand the sea, though also important, is far lagging behind. Currently some data of the sea are captured by the satellites [5]. These data, however, usually provide a macro view of a large geographic region during a long period of time; without the details that are suitable for applications that request for micro level data. Using sensor networks is considered to be a feasible and effective solution for these applications. Nevertheless, such difficulties as deployment, communication, etc., need to be addressed; see comprehensive research challenges in [7]. Providing effective coverage is among them before bringing these applications into reality.

Most of the existing works related to coverage issue in sensor networks focus on static sensor nodes [19]. A typical coverage problem is that the physical places are desired to be covered by at least  $K$  sensors so that the applications can guarantee the sensing veracity and perform sleep/wakeup scheduling to save energy. Focusing on the sensor deployment, Slijepcevic et al. [25] presented the definition of SET  $K$ -COVER problem and proved that is NP-complete. Abrams et al. [6] proposed an approximation algorithm which can be implemented in distributed way to solve that  $k$ -coverage problem and has a good performance close to the optimal solution. For the sea surface monitoring applications, we aim at 1-coverage of sea surface in this work and the significant reconsideration is needed as the motions of the flows and waves will push the sensor nodes away from their positions.

The motion of the sea flows and waves is affected by a number of factors [3], such as salt level, wind, temperature, geographic outlines and in water obstacles such as reefs [14]. Recent years has seen tremendous advances in study to model mobility; survey papers can be found in [9] [10]. In general, the sea flows can be approximated as a stochastic process. However, accurately modeling the water circulation is not an easy task. Especially, different sea or different regions of the sea may have different circulation patterns. Some current advances in oceanography can be found in [22]. There is a recent meandering current model for the underwater circulations [11] and captures the horizontal motion. In our paper, we will apply this model and propose a random ring model to capture the basic sea flow motion.

There are many studies that utilize mobile nodes with controllable mobility to assist or improve application

performance. For sensor coverage issues, they can be classified into two main categories. First, hybrid architecture with both mobile and static sensor nodes is proposed to assist area coverage [12] [27]. In these works, the mobile nodes move to fix the coverage holes caused by the uneven distribution of the static sensor nodes. None of them faces the effect of U-Mobility, however. Second, there are sensor networks with mobile nodes only. The nodes are self-controlled to move continuously to improve the overall coverage [20]. Howard [17] proposed a moving scheme based on potential field to maximize the coverage. However, this work does not guarantee full coverage; and again it does not face the U-Mobility.

In all these works, none of them faced the double mobility scenario as in this paper, where the interaction between the U-Mobility and the C-Mobility should be carefully considered. In our prior paper work [21], we propose a dominating set maintenance scheme to solve the double mobility problem. In that scheme, neighbor sensor node may inherit to cover the point of interest which loses the coverage from its original dominator. In this work, we improve the scheme so that the sensor node may not only inherit but also exchange the point of interest to cover. We also show that the improved scheme could achieve a near optimal performance with comprehensive simulations.

### 3 THE PROBLEM AND DESIGN CONSIDERATIONS

#### 3.1 Problem Statement

Assume the region is composed of  $m \times m$  cells. Let the set of points of interest be  $I$ , which is a subset of the cells of the region. There are  $N$  sensor nodes deployed in these cells to cover  $I$ . By convention, we use  $(px, py)$  to denote the cells in  $I$ ; and  $(ox_i^t, oy_i^t)$  to denote the position of the sensor node  $i$  at time  $t$ . All sensor nodes have the same coverage capability and we use a disk coverage model with a sensing range of  $R_s$ . We assume that the time is divided into slots. For the applications that we are interested in, the data sampled at the same time is more useful for data analysis. Consequently, the sensor network has to sample the data periodically every  $T$  time slots. Thus, our definition of coverage is:

*Definition 1:* Given a time  $T$ ,  $I$  should be covered every  $T$  time slots, i.e.,

$$\forall k > 0, \forall (px, py) \in I, \exists \text{ sensor node } i, \\ \text{s.t. } \sqrt{(ox_i^{kT} - px)^2 + (oy_i^{kT} - py)^2} \leq R_s$$

We assume that the sensor nodes have GPS or other facilities which can help them to obtain their positions and system time. The communication range of a sensor node is  $R_c$ .

In our application, the sensor nodes are initially deployed to provide full coverage of  $I$  by technicians. If the battery of a mobile sensor node is depleted, we will

have to send a technician for battery replacement as the mobile sensor nodes (with scientific apparatus) can be expensive and the loss of the sensor node by the flows may even cause environmental issues. Due to the cost of dispatching the technician, the batteries of all the nodes will be replaced. Therefore, we use the first depleted sensor as a measurement of the system lifetime, which is also widely used in current research as an indication of the end of a steady-state operation.

To illustrate the double mobility of a mobile sensor node, let the velocity constraint for the U-Mobility be  $V_u$ , i.e., at every time slot, a sensor node can only be pushed to another cell that is at most  $V_u$  steps away (in the stochastic process, the probability that the sensor node will be moved farther than  $V_u$  in one time slot is zero). We assume that in a grid(cell) divided region, the sensor node can only move to one of its four neighboring cell for each step. Thus, for C-Mobility, we make the constraint that the node could move at most  $V_c$  steps away in one time slot. Notice that we assume  $V_c \geq V_u$ . Otherwise, it is impossible to guarantee coverage.

$$\begin{aligned} |U_x(ox^t, oy^t) - ox| + |U_y(ox^t, oy^t) - oy| &\leq V_u \\ |C_x(ox^t, oy^t) - ox| + |C_y(ox^t, oy^t) - oy| &\leq V_c \end{aligned} \quad (1)$$

Let  $(U_x(x, y), U_y(x, y))$ ,  $(C_x(x, y), C_y(x, y))$  denote the next step locations of a node initialed at  $(x, y)$ , respectively after U-Mobility and C-Mobility. The location transformation of a node from  $t$  to  $t + 1$  can be represented as:

$$\begin{cases} ox^{t+1} = C_x(U_x(ox^t, oy^t), U_y(ox^t, oy^t)) \\ oy^{t+1} = C_y(U_x(ox^t, oy^t), U_y(ox^t, oy^t)) \end{cases} \quad (2)$$

Let the energy reserve for each node be  $e^0 = \text{max\_energy}$  at time slot 0. At every time slot, C-Mobility consumes energy and is represented by the total steps the node travels:

$$\begin{aligned} e^{t+1} &= e^t - |C_x(U_x(ox^t, oy^t), U_y(ox^t, oy^t)) - U_x(ox^t, oy^t)| \\ &\quad - |C_y(U_x(ox^t, oy^t), U_y(ox^t, oy^t)) - U_y(ox^t, oy^t)| \end{aligned}$$

Notice that we ignore the energy cost of sensing and communication on the sea surface. As shown in [23], a typical mobile sensor node consumes 27.96 Joule per meter in moving whereas the energy consumption of transmission is in the order of  $100 \times 10^{-9}$  Joule per bit and sensing is even less [16]. Generally speaking, it is widely accepted that mechanical movement is much more energy expensive than electronic communications.

The sensor nodes could cover all cells of  $I$  at time slot 0. With U-Mobility, the objectives of C-Mobility are thus:

- 1) The coverage is guaranteed according to Definition 1,
- 2) Maximize the network lifetime  $L$ :

$$\text{maximize } L, \text{ s.t. } \forall i \in N, e_i^L > 0$$

#### 3.2 Observations and Challenges

To have an energy efficient design for sea surface coverage, a key observation is that the U-Mobility not only

drives the sensor nodes away and breaks the coverage of the sensor network; it also sends nodes to positions that may improve the coverage for free. The sensor nodes should take the advantage of this U-Mobility as much as possible.

Nevertheless, we faced many challenges in the design. First, the U-Mobility is a stochastic process whereas our application is looking for a deterministic full coverage every  $T$  time slots. Notice that even each step of the U-Mobility is known in advance, it is still a max-min problem that is NP-hard. Second, though we may neglect the time for algorithm computation and communication at sensor nodes, the velocity of the nodes has a limit. It is unrealistic to design a scheme that the sensor network conducts a re-optimization at the last time slot in each period of  $T$  time slots, and assume the nodes are able to hurry to the respective locations to cover  $I$ . Third, though we may explore the benefit of the U-Mobility, the coverage scheme should not depend on a specific U-Mobility model.

In this paper, we take a fully distributed approach that every sensor node performs computation based on local information. Every node adjusts its status and may perform C-Mobility at every time slot, according to its estimation/optimization of the system parameters. We focus on period  $[0, T]$ ; different periods in our approach are statistically identical.

A straight forward algorithm is a *back-to-original* scheme. Sensor node calculates its position in every single time slot within the period  $[0, T]$ . The node starts to move back to its original position at time  $\Delta \in [0, T]$  given that after this time slot, there may not be enough time to return to the original position in the worst case. Apparently, this algorithm is not optimized as we will demonstrate in Section 6 as U-Mobility has not been leveraged.

Before further exposition, we use an example to illustrate the benefit.

**An Example:** We consider a very simple scenario that three sensor nodes  $A(1, 6)$ ,  $B(4, 8)$  and  $C(3, 1)$  cover three points of interest  $P_1(1, 8)$ ,  $P_2(0, 6)$  and  $P_3(1, 2)$ . As shown in Fig.1, at time 0, node  $A$  covers  $P_1$  and  $P_2$  while node  $C$  covers  $P_3$ . After U-Mobility in one time slot, node  $A$  moves from  $(1, 6)$  to  $(1, 5)$  and loses the coverage of  $P_1(1, 8)$ . Meanwhile, node  $C$  moves from  $(3, 1)$  to  $(5, 1)$  and loses the coverage of  $P_3(1, 2)$ .

U-Mobility drives node  $A$  out of the sensing range of  $P_1$ . However, U-Mobility also dispatches node  $B$  so that  $P_1$  is within the sensing range of  $B$ , who could take the responsibility of covering  $P_1$  from node  $A$ . Meanwhile, at time 1,  $P_3$  is out of the sensing range of node  $C$ . In back-to-original reposition scheme, node  $C$  has to move back in C-Mobility to maintain the coverage of  $P_3$ . This takes two steps of movement. However, with the advantage of U-Mobility, node  $A$  only needs to move for one step to cover  $P_3$  while maintaining the coverage of its current points. By exploiting the advantage of U-Mobility, we could thus come out a better scheme that request node

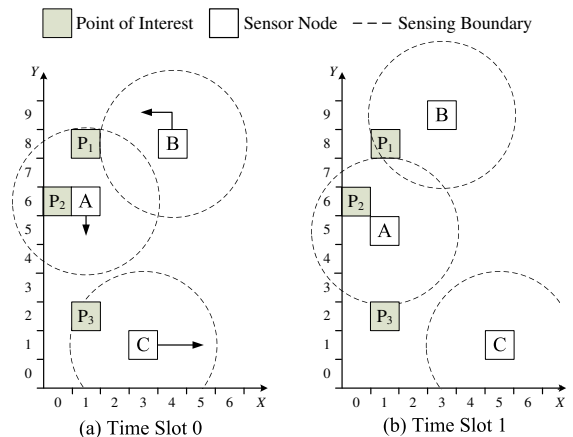


Fig. 1: A simple example of U-Mobility

$A$  to move from  $(1, 5)$  to  $(1, 4)$  to obtain the full coverage.

From this example, we make two simple yet useful observations: 1) A node can take advantage of U-Mobility as the U-Mobility may potentially move some nodes to cover the points of interest that it originally covered; (e.g. node  $B$  assists node  $A$  in Fig. 1); and 2) even a point of interest can not be covered and required some nodes to move in C-Mobility, the nodes could coordinate to explore the advantage of U-Mobility to save the energy. (e.g. to cover  $P_3$  in time 1, instead of asking node  $C$  that originally covered it move back, node  $A$  will move with short distance to cover  $P_3$ .)

## 4 SEA SURFACE COVERAGE: DESIGN AND OPTIMIZATION

### 4.1 Algorithm Outline

In our algorithm, each sensor node has to maintain two types of information. First, the set of cells that it needs to cover at the end of the period. Second, the location that it has to travel to so as to cover these cells. Formally, let  $D_i$  be the set of cells that node  $i$  need to cover. We name it the *dominating set*. By our assumption,  $I$  is fully covered by sensor nodes at time 0:

$$\bigcup_{1 \leq i \leq n} D_i^0 = I \quad (3)$$

For a dominating set  $D_i$ , a *target cell* is the location to move to at the end of a period, so that node  $i$  can cover  $D_i$ . Notice that this location must be feasible according to its velocity constraint and the flow speed. We propose the definition of a *feasible target cell* as follows:

*Definition 2:* Assume at time  $t$ , the dominating set that sensor node  $i$  (in location  $(ox_i^t, oy_i^t)$ ) maintains is  $D_i$ . A target cell  $(qx, qy)$  is called *feasible* if and only if the node can reach  $(qx, qy)$  before the end of period ( $T$  time slot in worst case) and cover  $D_i$  in  $(qx, qy)$ :

$$(V_c - V_u) \times (\lceil \frac{t}{T} \rceil T - t + 1) \geq |ox_i^t - qx| + |oy_i^t - qy| \\ \forall (px, py) \in D_i, \sqrt{(px - qx)^2 + (py - qy)^2} \leq R_s$$

Usually there are many feasible target cells for node  $i$ . The *closest* feasible target cell is the one that is reachable

with the least among of energy for node  $i$ . We temporarily delay the explanation of how the least energy is calculated (which will be detailed in next section). Intuitively, it is a shortest path with consideration of U-Mobility.

Obviously, a node does not need to move immediately when it finds a feasible target cell given enough time slots left. We define the status of a node to be "IDLE" or "BUSY" to indicate whether the node has to start C-Mobility.

*Definition 3:* Assume at time  $t$ , sensor node  $i$  in location  $(ox_i^t, oy_i^t)$  finds a feasible target cell  $(qx, qy)$ . We call node  $i$  "BUSY" if and only if the following inequality holds, and "IDLE" otherwise:

$$(V_c - V_u) \times (\lceil \frac{t}{T} \rceil T - t) < |ox_i^t - qx| + |oy_i^t - qy| + V_u \quad (4)$$

Inequality (4) indicates that if node  $i$  does not move in C-Mobility at current time slot  $t$  there will not be enough time for it to move to the target cell. Thus, node  $i$  should be set as "BUSY" and start to move. Otherwise, the sensor node just take a greedy approach to stay "IDLE".

In every time slot, both the dominating set  $D_i$  of and the feasible target cell of node  $i$  will change; either because of the U-Mobility or with the collaboration of other sensor nodes. We would like to emphasize that these change does not necessarily result in C-Mobility. Intrinsicly, the sensor nodes monitor the effect of U-Mobility. They negotiate with other nodes and estimate the future U-Mobility given certain U-Mobility models so as to find a collective most efficient strategy to cover the points of interest at the end of  $T$ .

---

#### Algorithm 1 Operation of Sensor Node $i$

---

```

Initialize dominate set  $D_i^0$  and energy  $e_i^0$ 
 $t \leftarrow 0$ 
while  $e_i^t > 0$  do
  Update  $D_i^t$  by corporation with neighbor nodes;
  Find the closest feasible target cell for C-Mobility;
  Move in U-Mobility and C-Mobility;
   $t \leftarrow t + 1$ 
end while

```

---

The above distributed algorithm is executed at each sensor node. We can interpret it in the following way. At every time slot, a node first searches for the possible advantage of U-Mobility (updating its dominating set with neighboring nodes); it then computes the best way to take this advantage (finding a closest feasible target cell); and finally, the node predicts the advantage of U-Mobility in future and starts C-Mobility if necessary. The remaining task is to show the detailed designs of each step, such that the U-Mobility is better exploit and the energy consumption is balanced. We detail these designs in next section.

## 4.2 Updating Dominating Set

The objective of updating dominating set is to hold the coverage after the U-Mobility so that each point

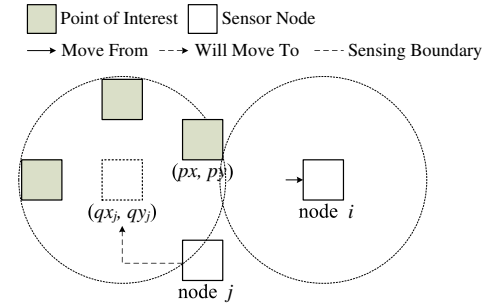


Fig. 2: Case I: the target cell of node  $j$  can cover  $(px, py)$

of interest will be dominated by a sensor node. The dominating set of a sensor node can change when this sensor is floating away by the U-Mobility. It can also change because that this sensor *inherits* the coverage responsibility from other nodes; or hands its coverage responsibility to other nodes. The sensor nodes update the coverage responsibility with neighboring nodes with concerns of residual energy.

Recall the simple example from Fig.1, node  $A$  lost the coverage of  $P_1$  in its dominating set  $D_A^1$  at time slot 1. If node  $A$  and  $B$  are within the communication range  $R_c$ , they can communicate with each other so node  $B$  could inherit the responsibility for covering  $P_1$  from node  $A$ . The dominating set of node  $A$  and  $B$  will be changed:

$$\begin{cases} D_A^1 &= D_A^0 \setminus \{P_1\} \\ D_B^1 &= D_B^0 \cup \{P_1\} \end{cases}$$

In this paper, we call two nodes to be neighbors if they are within each other's communication range  $R_c$ . Assume point  $(px, py) \in D_i^t$  is covered by node  $i$  at time slot  $t$ , there are three different cases to update  $D_i^t$ :

**CASE I:** Let node  $j$  be a neighbor of node  $i$ . If the status of node  $j$  is "BUSY" and its target cell  $(qx_j, qy_j)$  is within the sensing range of  $(px, py)$ , point  $(px, py)$  could be removed from  $D_i^t$  and added into  $D_j^t$ . See an example in Fig. 2).

We select the node  $j$  which best fit for load balance. Formally, let  $\mathcal{B}$  be the set of nodes that satisfies this condition,

$$\begin{aligned} & \text{maximize } e_j^t, j \in \mathcal{B} \\ & \text{s.t. (1) node } j \text{ is "BUSY"} \\ & \quad (2) \sqrt{(ox_j - ox_i)^2 + (oy_j - oy_i)^2} \leq R_c \\ & \quad (3) \sqrt{(qx_j - px)^2 + (qy_j - py)^2} \leq R_s \end{aligned} \quad (5)$$

This problem can be solved optimally by a greedy algorithm; and we choose the one with the maximum residual energy to inherit  $(px, py)$  from node  $i$  as shown in Equ. (5).

In this case, node  $j$  does not need to pay anything to inherit  $(px, py)$ . However, such neighbor node may not be available all the time. To deal with the case, a node has to change its target cell to inherit the coverage of  $(px, py)$  as shown in CASE II.

**CASE II:** Let node  $j$  be a neighbor of node  $i$ . Let the dominating set  $D_j^{t+}$  be:

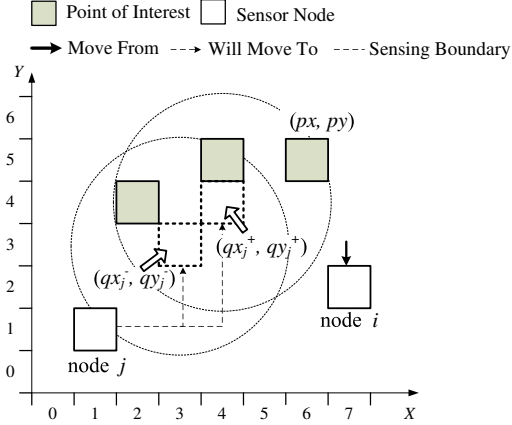


Fig. 3: Case II: node  $j$  can adjust its target cell to cover  $(px, py)$

$$D_j^{t+} = D_j^t \cup \{(px, py)\} \quad (6)$$

If the closest feasible target cell  $(qx_j^+, qy_j^+)$  for  $D_j^{t+}$  to  $(ox_j, oy_j)$  exists at the beginning of time slot  $t$ , node  $j$  is able to inherit the coverage of  $(px, py)$  from node  $i$ .

For a node  $j$  whose dominating set has  $(px, py)$ , define dominating set  $D_j^{t-}$  as:

$$D_j^{t-} = D_j^t \setminus \{(px, py)\} \quad (7)$$

Obviously a feasible target cell  $(qx_j^-, qy_j^-)$  for  $D_j^{t-}$  exists. The difference between selecting  $(qx_j^+, qy_j^+)$  or  $(qx_j^-, qy_j^-)$  is the extra cost that node  $j$  needs to pay if it decide to inherit the coverage of  $(px, py)$  from node  $i$ .

Figure. 3 illustrates CASE II by an example. Node  $i$  moves from  $(7, 3)$  to  $(7, 2)$  and loses the coverage of the point of interest  $(px, py) = (6, 5)$ . Node  $j$  has the dominating set  $D_j^{t-} = \{(2, 4), (4, 5)\}$  and the corresponding target cell is  $(qx_j^-, qy_j^-) = (3, 3)$  which can not cover  $(px, py) = (6, 5)$ . Node  $j$  can adjust its target cell to  $(qx_j^+, qy_j^+) = (4, 4)$  to cover the original points of interest from  $D_j$  and  $(px, py)$  at the same time. Thus, node  $j$  is a potential node that could inherit  $(px, py)$  from node  $i$  but it may need to take more steps to move to the new target cell.

There might be multiple potential nodes which could inherit the coverage of  $(px, py)$ ; and they have different costs. Let  $f_{ox, oy}(qx, qy)$  denote the energy cost for a node to move from  $(ox, oy)$  to target cell  $(qx, qy)$  (we will introduce how to calculate  $f$  in next section). For every potential node  $j$ , the extra energy cost to inherit coverage of  $(px, py)$  is:

$$\Delta_j = f_{ox_j, oy_j}(qx_j^+, qy_j^+) - f_{ox_j, oy_j}(qx_j^-, qy_j^-) \quad (8)$$

Again, we choose the node with the maximum residual energy that is deducted by  $\Delta$  for load balancing purpose:

$$\begin{aligned} & \text{maximize } (e_j^t - \Delta_j) \\ & \text{s.t. (1) } \sqrt{(ox_j - ox_i)^2 + (oy_j - oy_i)^2} \leq R_c \\ & \quad (2) \exists (qx_j^+, qy_j^+), (qx_j^-, qy_j^-) \text{ is the} \\ & \quad \text{closest feasible target cell for } D_j^{t+} \end{aligned} \quad (9)$$

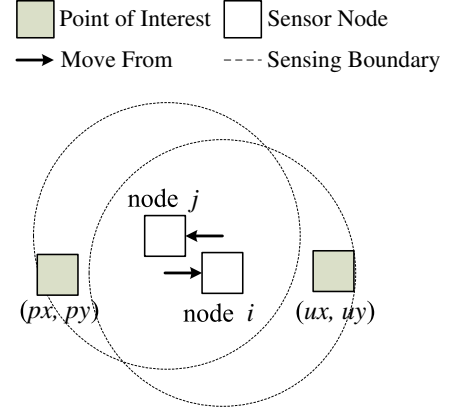


Fig. 4: Case III: node  $j$  can exchange  $(ux, uy)$  with node  $i$  to cover  $(px, py)$

In this case, node  $j$  needs to adjust its target cell to inherit  $(px, py)$  from node  $i$ . The extra movement cost should be paid based on the distance between original and modified target cell. However, simple "inherit" from one to another could not handle all the situations. In more general situations, a node has to exchange its dominating set with neighbor nodes.

**CASE III:** Let node  $j$  be a neighbor of node  $i$ . Assume there is a point of interest  $(ux, uy) \in D_j^t$ . Define  $D_i^{t*}$  and  $D_j^{t*}$  as the dominating sets which have exchanged  $(px, py)$  and  $(ux, uy)$ :

$$\begin{aligned} D_i^{t*} &= D_i^t \cup \{(ux, uy)\} \setminus \{(px, py)\} \\ D_j^{t*} &= D_j^t \cup \{(px, py)\} \setminus \{(ux, uy)\} \end{aligned} \quad (10)$$

If the closest feasible target cell  $(qx_i^*, qy_i^*)$  for  $D_i^{t*}$  and  $(qx_j^*, qy_j^*)$  for  $D_j^{t*}$  exist, then node  $j$  is able to inherit the coverage of  $(px, py)$  by exchanging  $(ux, uy)$  with node  $i$ . As the example shown in Fig. 4, node  $j$  can not inherit  $(px, py)$  directly in Case II since there's no feasible target cell to cover  $(px, py)$  and  $(ux, uy)$  at the same time. However, by exchanging the points of interest, node  $i$  and  $j$  could still cover  $(ux, uy)$  and  $(px, py)$  while the movement energy cost is saved.

There might be multiple choices of node  $j$  and  $(ux, uy)$  to exchange. Let  $f_{ox_i, oy_i}(qx_i, qy_i)$  and  $f_{ox_j, oy_j}(qx_j, qy_j)$  be the cost for node  $i$  and  $j$  without exchanging  $(ux, uy)$  and  $(px, py)$ ; and  $f_{ox_i, oy_i}(qx_i^*, qy_i^*)$  and  $f_{ox_j, oy_j}(qx_j^*, qy_j^*)$  be the cost with exchanging  $(ux, uy)$  and  $(px, py)$ . The expect energy left with and without exchanging point of interest are:

$$\begin{cases} e_i^* &= e_i^t - f_{ox_i, oy_i}(qx_i^*, qy_i^*) \\ e_j^* &= e_j^t - f_{ox_j, oy_j}(qx_j^*, qy_j^*) \end{cases} \quad (11)$$

$$\begin{cases} e_i &= e_i^t - f_{ox_i, oy_i}(qx_i, qy_i) \\ e_j &= e_j^t - f_{ox_j, oy_j}(qx_j, qy_j) \end{cases} \quad (12)$$

We choose the one with the best energy balance and

extra energy cost.

$$\begin{aligned}
& \text{maximize } e_i^{t*} + e_j^{t*} - e_i^t - e_j^t \\
& \text{s.t. (1) } \sqrt{(ox_j - ox_i)^2 + (oy_j - oy_i)^2} \leq R_c \\
& \quad (2) \exists (qx_i^*, qy_i^*), (qx_i^*, qy_i^*) \text{ is the} \\
& \quad \quad \text{closest feasible target cell for } D_i^{t*} \\
& \quad (3) \exists (qx_j^*, qy_j^*), (qx_j^*, qy_j^*) \text{ is the} \\
& \quad \quad \text{closest feasible target cell for } D_j^{t*} \\
& \quad (4) e_i^t \leq e_i^{t*} \text{ and } e_j^t \leq e_j^{t*}
\end{aligned} \quad (13)$$

Obviously node  $j = i$  and  $(ux, uy) = (px, py)$  is one of the potential solution: exchanging point of interest with itself. Therefore, there always exists a solution in CASE III. It can be guaranteed that at every time slot the union of the dominate sets is  $I$ , as shown in Lemma 1.

*Lemma 1:* At every time slot  $t$ , the union of the dominating sets  $D_i^t$  from all sensor nodes equals to the set of points of interest  $I$ :

$$\forall t \geq 0, \quad \bigcup_{1 \leq i \leq n} D_i^t = I \quad (14)$$

*Proof:* For  $t = 0$ , it holds based on the problem assumption in Equ. (3). Assume the lemma holds for  $t = k$ :

$$\bigcup_{1 \leq i \leq n} D_i^k = I$$

For  $t = k + 1, \forall i, \forall p \in D_i^k, p$  will be either in  $D_i^{k+1}$  or  $D_j^{k+1}$  while node  $j$  is selected by the approach above according to the process of “updating dominating set”. Thus, we have:

$$\bigcup_{1 \leq i \leq n} D_i^{k+1} = \bigcup_{1 \leq i \leq n} D_i^k = I \quad (15)$$

The lemma holds by induction.  $\square$

### 4.3 Find the Closest Feasible Target Cell

Every time a sensor node updates its dominating set, or the current location of this sensor node is changed, it needs to recalculate the feasible target cells that it should move to for covering this dominating set. The objective is to find the feasible target cell that is optimized in terms of moving distance from its current location to this target cell.

We have on purposely delayed the description of the *closest* feasible target cell. If there is no U-Mobility, or the pattern of U-Mobility is entirely unknown, we use the shortest path from  $(ox, oy)$  to  $(qx, qy)$  as the estimated energy cost for the weights of the feasible target cells:

$$f_{ox,oy}(qx, qy) = |qx - ox| + |qy - oy| \quad (16)$$

However, different U-Mobility patterns may seriously affect the weight of the path and the selection of the target cell. Fig. 5 shows an example: assume the current location of the node is  $(ox, oy) = (3, 1)$ , the target cell of the node is  $(qx, qy) = (5, 5)$  and  $V_c = 2$ . The expected energy will be quite different in different pattern of U-Mobility.

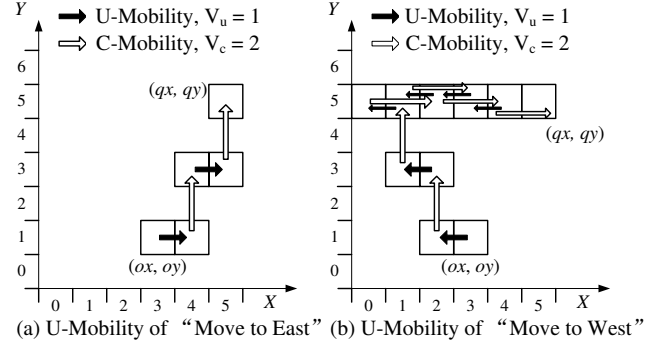


Fig. 5: Different U-Mobility Generates Different Shortest Path

Suppose two different movement patterns of U-Mobility: one is move to east by one step at every time slot and the other is move to west by one step at every time slot. In Fig. 5(a), under the U-Mobility of “Move to East”, the node only needs to take 2 time slots and move 4 steps in C-Mobility to reach the target cell. But in Fig. 5(b), the node has to spend 6 time slots and move at least 12 steps in C-Mobility because the direction of U-Mobility is opposite to the direction to the target cell. We have obtained two different expected energy costs  $f_{ox,oy}(qx, qy)_{east} = 4$  and  $f_{ox,oy}(qx, qy)_{west} = 12$  from this example based on different U-Mobility patterns. Therefore, U-Mobility should be taken into the computation of expected energy  $f_{ox,oy}(qx, qy)$ .

Let  $(\bar{U}_x(x, y), \bar{U}_y(x, y))$  denote the expected cell that the node will move from  $(x, y)$  in one time slot under a U-Mobility pattern. Given this  $\bar{U}$ , we construct a weighted direct graph  $G(V, E, W)$  such that:

$$\begin{aligned}
V &= \{v_i = (x_i, y_i) \mid \text{cell } (x_i, y_i) \text{ in the region}\} \\
E &= \{(v_i, v_j) \mid |x_j - \bar{U}_x(x_i, y_i)| + |y_j - \bar{U}_y(x_i, y_i)| \leq V_c\} \\
W &= \{w_{ij} = |x_j - \bar{U}_x(x_i, y_i)| + |y_j - \bar{U}_y(x_i, y_i)|\}
\end{aligned} \quad (17)$$

If the U-Mobility is known in advance and shows a long term behavior, the shortest path between any pair of two cells are calculated at the initial stage in graph  $G(V, E, W)$ .

Fig. 6(a) provides a partial graph in the U-Mobility of “Move to East” and the corresponding shortest path. Fig. 6(b) demonstrates a partial graph in pure random walk. In our simulation, we will see the effect of two different U-Mobility models discussed in Section 6 and Appendix.

### 4.4 Move in U-Mobility and C-Mobility

After finding the closest feasible target cell, a node may start to move. We would like to clarify that the closet feasible target cell is a macro target but it is usually not reachable in one time slot. After each time slot the status (the dominating set and the closest feasible target cell) of the sensor node can change. In local movement, the sensor node still can take advantage of the U-Mobility on the fly.

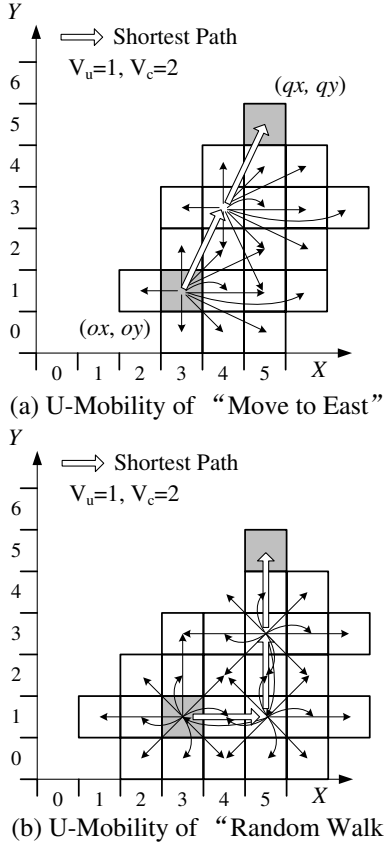


Fig. 6: Different Graph in Different U-Mobility

**CASE I:** If node  $i$  is "IDLE" at time  $t$ , it only moves in U-Mobility. The energy cost of node  $i$  for this time slot is zero:  $e_i^{t+1} = e_i^t$ .

**CASE II:** If node  $i$  is "BUSY" at time  $t$ , it will move both in U-Mobility and C-Mobility.

Let  $tx_i^t$  and  $ty_i^t$  be the number of steps that node  $i$  will move along  $X$ -axis and  $Y$ -axis separately in C-Mobility during this time slot. After C-Mobility, at the end of time slot  $t$  (or beginning of time slot  $t+1$ ), we could obtain the new position of node  $i$  by:

$$\begin{cases} ox_i^{t+1} = U_x(ox_i^t, oy_i^t) + tx_i^t \\ oy_i^{t+1} = U_y(ox_i^t, oy_i^t) + ty_i^t \end{cases} \quad (18)$$

Intuitively, we prefer to select such a pair of  $(tx_i^t, ty_i^t)$  that could minimize the total cost of energy moving from  $(ox_i^t, oy_i^t)$  to  $(qx_i^t, qy_i^t)$ . Meanwhile,  $(tx_i^t, ty_i^t)$  should not exceed the maximum velocity of  $V_c$  and guarantee node  $i$  could reach the target cell in time after this C-Mobility:

$$\begin{aligned} & \text{minimize } (|tx_i^t| + |ty_i^t| + f_{ox_i^{t+1}, oy_i^{t+1}}(qx_i^t, qy_i^t)) \\ & \text{s.t. (1) } |tx_i^t| + |ty_i^t| \leq V_c \\ & \text{(2) } (qx_i^t, qy_i^t) \text{ is still a feasible target cell} \\ & \quad \text{at the beginning of time slot } t+1 \end{aligned} \quad (19)$$

Condition (1) in Equ. (19) indicates the velocity constraint of C-Mobility while condition (2) guarantees the node could arrive at the target cell in time with this C-Mobility. Once the shortest path on  $G(V, E, W)$  is found,  $(tx, ty)$  could be determined by the subsequent node on the shortest path. We also notice that CASE I is actually

a special case for CASE II while  $(tx, ty) = (0, 0)$  is a best solution.

**Lemma 2:** At the end of every time  $t$ , for every node  $i$ , there exists a feasible target cell  $(qx_i^t, qy_i^t)$  for  $D_i^t$ .

**Proof:** We just need to prove that there exists at least one pair of  $(tx_i^t, ty_i^t)$  which satisfies the conditions in Equ. (19) so that  $(qx_i^t, qy_i^t)$  is a feasible target cell at the beginning of time slot  $t+1$  (the end of time slot  $t$ ).

Let  $x' = U_x(ox_i^t, oy_i^t)$  and  $y' = U_y(ox_i^t, oy_i^t)$ , according to the maximum velocity of U-Mobility, we could obtain:

$$\begin{aligned} & |ox_i^t - x'| + |oy_i^t - y'| \leq V_u \\ \Rightarrow & -V_u \leq (|qx_i^t - ox_i^t| - |qx_i^t - x'|) + \\ & (|qy_i^t - oy_i^t| - |qy_i^t - y'|) \leq V_u \end{aligned} \quad (20)$$

We have two different cases to analyze:

1) If  $|x' - qx_i^t| + |y' - qy_i^t| \leq V_c$

Let  $tx_i^t = qx_i^t - x'$  and  $ty_i^t = qy_i^t - y'$  which could satisfy the velocity constraint of C-Mobility.

$$\begin{aligned} & |ox_i^{t+1} - qx_i^t| + |oy_i^{t+1} - qy_i^t| \\ = & |x' + tx_i^t - qx_i^t| + |y' + ty_i^t - qy_i^t| \\ = & |x' + (qx_i^t - x') - qx_i^t| + |y' + (qy_i^t - y') - qy_i^t| \\ = & 0 \end{aligned} \quad (21)$$

$$\begin{aligned} \text{thus } & (V_c - V_u) \times (\lceil \frac{t+1}{T} \rceil T - (t+1) + 1) \geq \\ & |ox_i^{t+1} - qx_i^t| + |oy_i^{t+1} - qy_i^t| = 0 \end{aligned}$$

In this case, we have a solution  $(tx_i^t, ty_i^t) = (qx_i^t - x', qy_i^t - y')$  to guarantee  $(qx_i^t, qy_i^t)$  is a feasible target cell at the beginning of time slot  $t+1$ .

2) If  $|x' - qx_i^t| + |y' - qy_i^t| > V_c$

In this case, there must exist a pair of value  $(x^*, y^*)$  so that:

$$\begin{aligned} & (a) \text{ minof}(x', qx_i^t) \leq x^* \leq \text{maxof}(x', qx_i^t) \\ & (b) \text{ minof}(y', qy_i^t) \leq y^* \leq \text{maxof}(y', qy_i^t) \\ & (c) |x^* - x'| + |y^* - y'| = V_c \end{aligned} \quad (22)$$

Let  $tx_i^t = x^* - x'$  and  $ty_i^t = y^* - y'$ , we could obtain:

$$\begin{aligned} & (|qx_i^t - ox_i^t| + |qy_i^t - oy_i^t|) - \\ & (|qx_i^t - ox_i^{t+1}| + |qy_i^t - oy_i^{t+1}|) \\ = & (|qx_i^t - ox_i^t| - |qx_i^t - x'| - |ox_i^{t+1} - x'|) + \\ & (|qy_i^t - oy_i^t| - |qy_i^t - y'| - |oy_i^{t+1} - y'|) \\ = & |qx_i^t - ox_i^t| - |qx_i^t - x'| + |qy_i^t - oy_i^t| - |qy_i^t - y'| + V_c \\ \geq & V_c - V_u \end{aligned} \quad \text{from (20)} \quad (23)$$

Since  $(qx_i^t, qy_i^t)$  is feasible at the beginning of time slot  $t$ , we have:

$$(V_c - V_u) \times (\lceil \frac{t}{T} \rceil T - t + 1) \geq |ox_i^t - qx_i^t| + |oy_i^t - qy_i^t| \quad (24)$$

Meanwhile, it is easy to get:

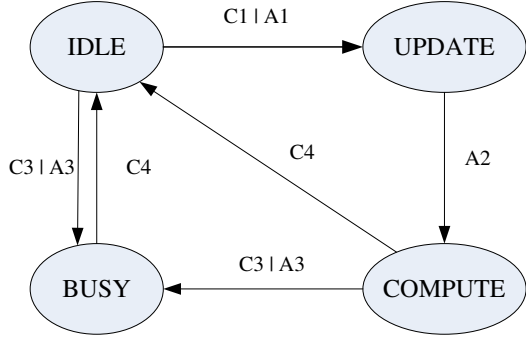
$$\begin{aligned} & (V_c - V_u) \times (\lceil \frac{t}{T} \rceil T - t + 1) - \\ & (V_c - V_u) \times (\lceil \frac{t+1}{T} \rceil T - (t+1) + 1) \leq V_c - V_u \end{aligned} \quad (25)$$

Combine three inequalities above, we could achieve:

$$(V_c - V_u) \times (\lceil \frac{t+1}{T} \rceil T - (t+1) + 1) \geq |qx_i^t - ox_i^{t+1}| + |qy_i^t - oy_i^{t+1}| \quad (26)$$

Therefore, in both two cases, we could obtain a solution pair of  $(tx_i^t, ty_i^t)$  which satisfies the constraints in





Conditions:

- C1: a cell in  $D_i$  is uncovered after U-Mobility  
 C3: need to move in C-Mobility immediately, otherwise it cannot reach the target cell in time  
 C4: node  $i$  could stay free at this time slot

Operations:

- A1: exchange the cell with neighboring nodes  
 A2: compute the feasible target cell with new  $D_i$   
 A3: move in C-Mobility on shortest path

Fig. 7: State transition of each sensor node.

Equ. (19). At the end of every time slot, for any node there exists a feasible target cell.  $\square$

In this section, we proposed a heuristic solution which takes three actions during every time slot: 1) learns the advantage of U-Mobility (updating the dominating set); 2) computes a best way to take the advantage of U-Mobility (finding the closest feasible target cell); 3) moves in C-Mobility while predicting future advantage of U-Mobility (designing a path to move in C-Mobility for current time slot). The detail state transition diagram of each sensor node is shown in Fig. 7. We will show in the next theorem that our solution guarantees the full coverage of  $I$  every  $T$  time slot.

As the sensor network is able to collectively hold the dominating sets and each sensor node is able to find feasible target cell to move to maintain the dominating set, the coverage is guaranteed, as proved by the following theorem.

**Theorem 3:** At the end of every  $T$  time slot, every point of set of points of interest  $I$  is covered by a sensor.

$$\forall k \geq 0, \forall (px, py) \in I, \exists u, 1 \leq u \leq N \quad (27)$$

$$s.t. \sqrt{(px - ox_u^{kT})^2 + (py - oy_u^{kT})^2} \leq R$$

*Proof:* At the end of every time slot  $kT$ , for any point of interest  $p \in I$ , we know that  $\exists u, p \in D_u^{kT}$  from Lemma 1. For the dominating set  $D_u^{kT}$ , according to Lemma 2, there exists a feasible target cell  $(qx_u, qy_u)$  which satisfies:

$$(i) \quad \sqrt{(qx_u - px)^2 + (qy_u - py)^2} \leq R_s$$

$$(ii) \quad (V_c - V_u) \times (\lceil \frac{kT+1}{T} \rceil T - kT)$$

$$\leq |ox_u^{kT} - qx_u| + |oy_u^{kT} - qy_u|$$

$$\Rightarrow (V_c - V_u) \times 0 \geq |ox_u^{kT} - qx_u| + |oy_u^{kT} - qy_u|$$

$$\Rightarrow |ox_u^{kT} - qx_u| + |oy_u^{kT} - qy_u| = 0$$

$$\Rightarrow ox_u^{kT} = qx_u, oy_u^{kT} = qy_u$$

(28)

Combine (i) and (ii), we can achieve that point of interest  $p$  is in the sensing range of node  $u$ .  $\square$

## 5 THE DISTRIBUTED PROTOCOL

In previous section, we have detailed the SSC algorithm. The algorithm outlines the computation for each node in each time slot. To fully realize this algorithm with a distributed protocol, two more components are needed: 1) the maintenance of local information at each node and 2) the strategy to update the dominating set with neighbor nodes.

### 5.1 Local Information Maintenance

In our distributed protocol, every sensor node maintains a local information table as shown in Table. 1:

TABLE 1: Local Information Table on Node

Name	Description
$D$	Dominating set
$(ox, oy)$	Current location
$(qx, qy)$	Target cell to reach at the end of next $T$ time slot
$(tx, ty)$	Temporary target in current time slot

The current location  $(ox, oy)$  of a sensor node is obtained by GPS or other mechanisms at the beginning of every time slot. For  $(qx, qy)$  and  $(tx, ty)$ , one may consider their relationship to be that  $(qx, qy)$  is the final destination at the end of the period  $T$  and  $(tx, ty)$  is the next hop. Notice, however, that  $(qx, qy)$  is also changing if  $D$  changes.

### 5.2 The Protocol for Dominating Set Updates

There are many practical difficulties to address. Most importantly, as the algorithm requests distributed maintenance of the dominating sets, different nodes will have different views due to the latency of communication. More specifically, if node  $A$  finds that node  $C$  is able to inherit a point of interest  $P_A$  and node  $B$  also finds that  $C$  is able to inherit a point of interest  $P_B$ , there is a *collision* as node  $C$  may not inherit  $P_A, P_B$  simultaneously.

As an example shown in Fig. 8, we say there is a event of *cover-losing* for node  $i$ , if some point of interest  $(px, py)$  from  $D_i$  is uncovered at a time slot. In the SSC algorithm, node  $i$  will find a neighboring node (or maybe itself if

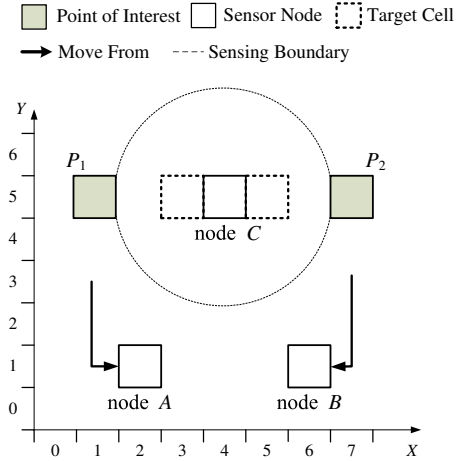


Fig. 8: Collision on Node C to Inherit  $P_1$  and  $P_2$  at the Same Time.

the cost is the smallest) to inherit the coverage. There may be collisions, however, as shown in Fig 8. In a time slot, node A(2, 1) lost the point of interest  $P_1(1, 5)$  from the sensing range. After computation, it found that node C(4, 5) is the best candidate to inherit  $P_1$  since C only needs to move 1 step to cover  $P_1$ . Meanwhile, similar thing happened to node B(6, 1) and his point of interest  $P_2(7, 5)$ . Unfortunately, node B also selected node C to inherit  $P_2$ . However, from the figure, it is obvious that node C can not inherit  $P_1$  and  $P_2$  at the same time. This generates a collision when node C is updating its dominating set by adding both  $P_1$  and  $P_2$ . We develop a distributed protocol to address such practical issues.

To avoid the collision in updating dominating set, we proposed a concept of “candidate” list so that node C could make the decision in some order (e.g. accept node A to inherit  $P_1$  and reject node B to inherit  $P_2$ ). The detail communication steps of each node is shown as follows:

1) Node  $i$  broadcasts a *Info* message to its one-hop neighbor nodes to introduce its dominating set.

$$i \Rightarrow neighbors \quad Info \mid i \mid D_i \quad (29)$$

2) Node  $i$  broadcasts a *Query* message to its one-hop neighbor nodes to claim that it is querying for some node to take the responsibility to cover  $(px, py)$ :

$$i \Rightarrow neighbors \quad Query \mid i \mid (px, py) \quad (30)$$

3) Node  $j$  receives this *Query* message and finds that it is possible to inherit or exchange  $(px, py)$  with  $(ux, uy)$  in its dominating set  $D_j$ . Then, node  $j$  sends a *Response* message back to node  $i$  with the status of node  $j$  and the cost to take  $(px, py)$ :

$$j \Rightarrow i \quad Response \mid j \mid (px, py) \mid (ux, uy) \mid status \mid evaluating \ value \quad (31)$$

Note that node  $j$  would response a special  $(ux, uy)$  (e.g., virtual point not in  $I$ ) to clarify the difference between inheriting and exchanging.

4) For all the *Response* node  $i$  receives, it ranks the candidates according to the cost metric defined in section

4. Node  $i$  will then send a *Request* message to request a node in the candidate list in order to inherit  $(px, py)$ :

$$i \Rightarrow j \quad Request \mid i \mid (px, py) \mid (ux, uy) \quad (32)$$

5) After receiving a *Request* message, node  $j$  will make a decision based on current dominating set to see whether it can accept this request (e.g., if node  $j$  has accepted a request from other node at this time, the dominating set maybe changed so the re-judgement is necessary) and send a feedback to node  $i$ :

$$j \Rightarrow i \quad Req\_Accept \mid j \mid (px, py) \mid (ux, uy) \quad (33)$$

$$j \Rightarrow i \quad Req\_Reject \mid j \mid (px, py) \mid (ux, uy)$$

If the request gets to be rejected, node  $i$  will send requests to the nodes in candidate list until a *Request\_Accept* is received or the end of the list is reached.

Note that the negotiating process (1)-(5) above is initialized by a “cover-losing” event. For multiple “cover-losing” events, there would be multiple negotiating processes to handle them separately and the evaluating values keep updating among these processes to ensure the fair and optimized decision. We will evaluate our protocol and show that it has a low average communication overhead by simulation.

## 6 PERFORMANCE EVALUATION

### 6.1 Simulation Settings

We evaluate our SSC in an event driven simulator. The default values of our simulation are as follows. We deploy 500 sensor nodes in a region of  $500m \times 500m$  for the random ring model, and  $1600m \times 400m$  for the meandering model. Each cell for computation is a square of  $5m \times 5m$ . In both models, 1000 points of interest are randomly and uniformly distributed in the region. For the mobile sensor nodes, the sensing range is  $R_s = 50m$  and the default communication range  $R_c = 100m$ . We adopt the movement parameters similar to Starbug AUV [13]. The maximum speed of Starbug is  $1.5m/s$  (3 knots), i.e.,  $V_c = 1.5m/s$ . The battery allows a continuous movement for a distance of 7500m. The maximum velocity of U-Mobility is  $V_u = 0.5m/s$  [11]. We set one time slot as 100 seconds (e.g.  $t = 100s$ ) and time interval  $T = 10t$ . 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 sensors at time  $t$ .

$$\sigma(t) = \sqrt{\frac{1}{N} \sum_{i=1}^N (e_i^t - \bar{e}^t)^2} \quad (34)$$

We evaluate both the meandering model and the random ring model for U-Mobility. The simulation results are the average of 10 randomly conducted experiments.

### 6.2 Network Lifetime and Balance of Energy

We first compare the SSC to the back-to-original reposition scheme that we presented in Section 3. We evaluate our SSC in two communication range settings, i.e.,  $100m$  and  $200m$  (notice that the sensing range does not change). In Fig. 9, Y-axis is the minimal residual energy

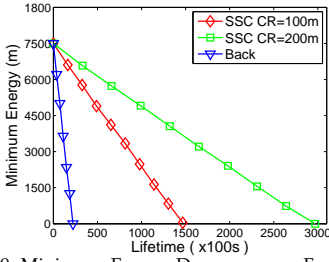


Fig. 9. Minimum Energy Decreases as a Function of Time (U-Mobility: Random Ring Model)

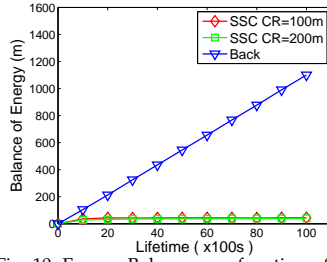


Fig. 10. Energy Balance as a function of Time (U-Mobility: Random Ring Model)

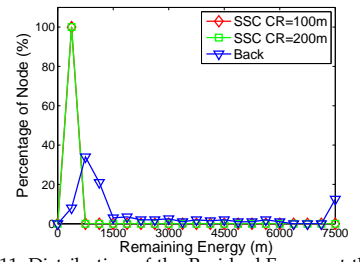


Fig. 11. Distribution of the Residual Energy at the End of System Lifetime (U-Mobility: Random Ring Model)

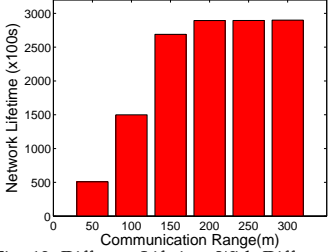


Fig. 12. Different Lifetime With Different  $R_c$  (U-Mobility: Random Ring Model)

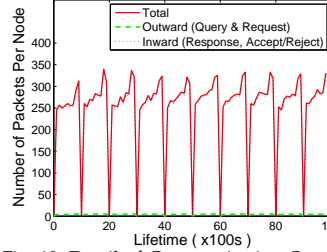


Fig. 13. Detail of Communication Overhead (U-Mobility: Random Ring Model)

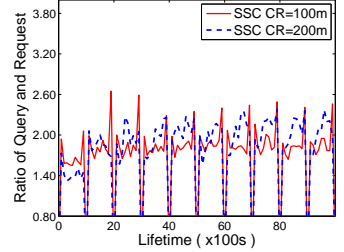


Fig. 14. Ratio of *Query* and *Request* (U-Mobility: Random Ring Model)

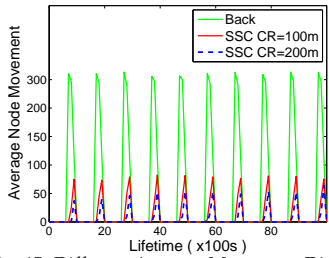


Fig. 15. Different Average Movement Distance (U-Mobility: Random Ring Model)

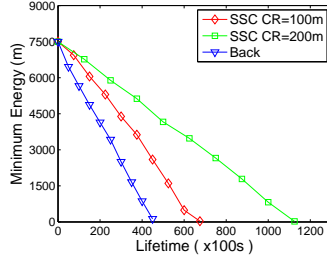


Fig. 16. Minimum Energy Decreases as a Function of Time (U-Mobility: Meandering Model)

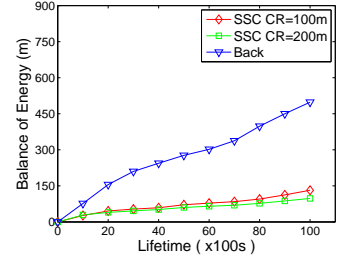


Fig. 17. Energy Balance as a function of Time. Y-axis represents the standard deviation (U-Mobility: Meandering Model)

of the sensor nodes in the network. It is normalized to represent the remaining distance a node can travel. Clearly, the minimum energy of the back-to-original scheme decreases much faster and the system lifetime is only around  $25 \times 10^3 s$ . Our SSC algorithm has a system lifetime of  $150 \times 10^3 s$ , a more than six-time improvement. If the communication range is  $200m$ , the lifetime is improved further to  $300 \times 10^3 s$ , i.e., around 83.3 hours.

Fig. 10 illustrates that SSC better balances the residual energy of the sensor nodes. The standard deviation of the residual energy is very small compared with the back-to-original scheme. More interestingly, as time evolves, the increase of the standard deviation almost flattened. This shows that our SSC is able to scale better. This is further verified by Fig. 11, where we show the probability distribution of the residual energy at the end of the system lifetime. We can see that the residual energy of most sensor nodes (i.e., almost 100 percent in Fig. 11) can only support a traveling distance of less than 375 meters in SSC, i.e., less than 5% of the energy reserve. The exhausted status indicates that most sensor nodes are fully and balancing utilized in C-Mobility. As a comparison, in the back-to-original scheme, many sensor nodes have significant residual energy left which is a waste of energy.

In real applications, the energy cost of the movement

of sensor nodes is the dominating part compared with the communication cost. Thus, we also evaluate the average node movement as another metric in the simulation. The result in Fig. 15 shows that our SSC has a better strategy of movement design compared with back-to-original scheme. As the communication range increases, the nodes could improve the movement strategy even better.

### 6.3 Communication Range and Overhead

We see in Fig. 16, the system lifetime increases as the communication range increases. This is because that the sensor nodes are able to find more neighboring nodes and make better decisions. When the communication range is above  $200m$ , however, improvement is marginal. It is not surprising as the sensing range and velocity of a sensor node have a limit. Therefore, seeking help from sensors that are far away is not helpful. This locality property strongly support our distributed protocol design against a design of overall optimization.

Though we neglect the communication cost in the design of C-Mobility, we evaluate the communication overhead of our protocol as large overhead will affect the protocol latency and increase packet collisions and retransmission. There are five main types of packets in our protocol: *Query*, *Response*, *Request*, *Req\_Accept* and *Req\_Reject*. From Fig. 13, we can see that there is a

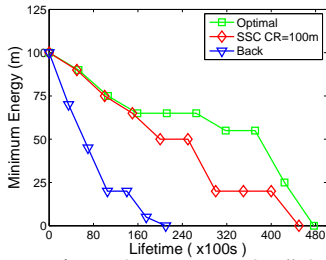


Fig. 18. Lifetime Comparison in Small Case with Optimal Solution (U-Mobility: Random Ring Model)

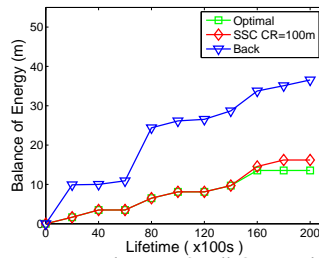


Fig. 19. Energy Balance in Small Case with Optimal Solution (U-Mobility: Random Ring Model)

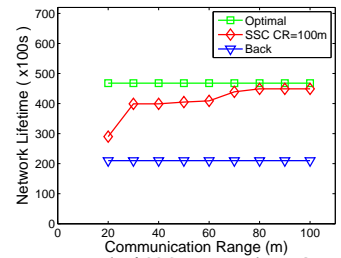


Fig. 20. Trend of SSC Approaching Optimum (U-Mobility: Random Ring Model)

bell shape for the number of packets generated during each time period  $T$ . During the initial period the sensor nodes have high probability to keep on covering its own dominating set. However, after a while when they are losing their positions, the nodes are more and more aggressively negotiating with the candidate nodes that can exchange the dominating set. Notice that every time  $T = 10t$ , the number of communication packets become zero which indicates no dominating set update happens. From another point of view, this verifies that our scheme guarantees the coverage. Fig. 13 shows that every node results less than 300 packets totally in one time slot. Since each time slot represents  $100s$ , this is an acceptable communication requirement.

Fig. 14 compares the ratio of *Request* and *Query*. This indicates the number of collisions in each query. As we can see, the ratio is less than 2.5; this means that most nodes can find a proper node to inherit that cell by the first two tries.

#### 6.4 Meandering Model

We then evaluate SSC for meandering current model. We see from Fig. 11 that though not as significant as the random ring model, SSC shows greatly increased (50% and 150% for two different CR respectively) lifetime than back-to-original scheme. From Fig. 12, we also see that SSC balance the energy consumption better. By looking into the details of the simulation data trace, we observe that the meandering model shows a much more deterministic jet behavior. All the sensor nodes are pushed more consistently in one direction. This makes the benefit of exploring the U-Mobility less significant. Considering an extreme case where all the sensor nodes are pushed straight towards one direction, then SSC will reduce to back-to-original as the sensor nodes have no choice other than going back.

From this case, we could imagine the worst situation that: nodes are moving out of the region of interest to be covered. In that way, C-Mobility has no other solution but only defeating the U-Mobility which is costly and the moorings are suggested to be introduced. Nevertheless, our simulation results have clearly shown that SSC exploits the benefit of the U-Mobility and beat Back-to-original even in the worst case.

#### 6.5 Near Optimality

Last but not the least, we try to claim the near optimality of our SSC. Since the problem is NP-hard, we evaluate our SSC in a small case: 8 sensor nodes are deployed

in  $200m \times 200m$  region to cover 12 points of interest with initial battery allows a continuous movement for  $100m$ . The optimal solution in such small case could be computed by brute force in a centralized way.

As shown in Fig. 18, our SSC achieves about 0.917-approximation ratio (i.e.,  $440/480 \approx 0.917$ ) of the optimal network lifetime. Meanwhile, Fig. 19 illustrates that SSC and optimal solution have nearly the same performance in energy balance. To further illustrate the near optimality of SSC, we increase the communicate range of SSC and see the trend of approaching to the optimal network lifetime as an upper bound.

#### 6.6 Impact of Network Size

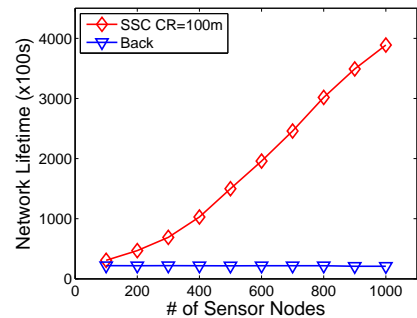


Fig. 21. Lifetime with Different Network Size (U-Mobility: Random Ring Model)

We also evaluate the effect of other network parameters, i.e., network size (number of nodes). In Fig. 21, with grown number of deployed nodes, the lifetime of SSC increases linearly while the back-to-original scheme does not leverage the collaboration between nodes.

### 7 CONCLUSION

In this paper, we for the first time study a double mobility coverage problem for sea surface monitoring. Our problem is sharply different from previous works as we face both an uncontrollable mobility by the sea flows and a controllable mobility of the sensor nodes. We made an observation that U-mobility not only breaks the coverage of the sensor network but also sends the sensor nodes to the locations that may improve the coverage. Thus, the key target of this paper is by leveraging U-Mobility, to minimize the movement distance in C-Mobility and balance the energy consumption among all the sensor nodes to provide a guaranteed coverage of the points of interest. We proposed a distributed sea surface coverage (SSC) algorithm which leverage the advantage of the U-Mobility. The algorithm naturally addressed a set of difficulties, such as the limitation of the velocity

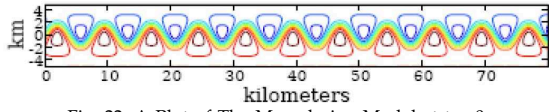
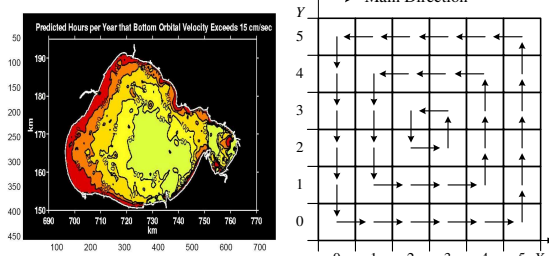


Fig. 22. A Plot of The Meandering Model at  $t = 0$ .



(a) The circulation of Lake Pontchartrain (b) An Example of Main Circulation  
Fig. 23. Main Circulation of Lake Pontchartrain

of the mobile sensor nodes, and the balance of energy consumption. We proved that the algorithm guarantees coverage. A distributed protocol was then developed that addressed a set of practical concerns, e.g., competition of transferring the coverage responsibility to other nodes. Our simulation results demonstrated that SSC can extend the system lifetime over a back-to-original scheme under various configurations significantly.

## APPENDIX A MODELING U-MOBILITY

Modeling the sea flows is not an easy task. Due to wind, salinity, reefs, temperature, different sea has different circulation characteristics. In general, the sea flow can be considered as a stochastic process with both local variety (caused by reefs, turbulence), and the main circulation effects (caused by wind, salinity) [15]. By no means this paper provides a comprehensive study on modeling sea flows. We refer interested readers to [22]. In this paper, we will use two concrete models as examples to discuss the interaction between U-Mobility and C-Mobility.

In [11], a simplified model (meandering current model) is presented for underwater circulation. The mobility is divided into a fast downstream motion (a jet) and a looping motion (a vortex). The trajectory of the move is the solution of the following equations [11]:

$$\dot{x} = -\partial_y \psi(x, y, t), \quad \dot{y} = \partial_x \psi(x, y, t)$$

where  $\psi(x, y, t) = -\tanh \left[ \frac{y-B(t)\sin(k(x-ct))}{\sqrt{1+k^2 B^2(t)\cos^2(k(x-ct))}} \right]$ .

Fig. 22 (Fig. 1 in [11]) shows the water motion of the meandering model. Notice that this model is deterministic. When the initial position of a sensor node is determined, its subsequent route is also determined.

In this paper, we further introduce a simpler ring-like model that consists of both local variety and main circulation, which captures some characteristics of a wider range of water bodies.

For the local variety, we use a random walk model. Formally, let a sensor node be in the cell  $(ox, oy)$ . The probability that the node will be in cell  $(x, y)$  at next time slot is:

$$P_{local}((x, y)|(ox, oy)) = \begin{cases} \frac{1}{(2V_u+1)^2} & |x - ox| + |y - oy| \leq V_u \\ 0 & otherwise \end{cases} \quad (35)$$

For the main circulation effects, we consider the strength and speed of the flow. This depends on specific characteristics of different sea. We illustrate with an example of Lake Pontchartrain [24] (see Fig. 23(a)). Mainly caused by the wind in that region, its flow shows a ring-like pattern (see Fig. 23(b)). Thus, we model it as:

$$P_{main}((x, y) | (ox, oy)) = \begin{cases} 1 & x = ox + \Delta_x(ox, oy) \\ & \& y = oy + \Delta_y(ox, oy) \\ 0 & otherwise \end{cases} \quad (36)$$

The overall U-Mobility is a combination of these two models. We use factor  $\alpha$  and  $\beta$  to represent the weights of two walk model separately. Formally,

$$P((x, y) | (ox, oy)) = \alpha \times P_{main}((x, y)|(ox, oy)) + \beta \times P_{local}((x, y)|(ox, oy)) \quad (37)$$

A more accurate model ‘‘Meandering Current Mobility with Surface Effect’’ (MCM-SE) was proposed in [15]. This model better captures the behavior of major current (e.g., jet-like motion) and the stochastic local variance (e.g., random walk caused by local environment). We admit that using such MCM-SE model could improve the evaluation process of updating dominating set in our algorithm. However, it is not our interest: our SSC is not designed with requiring a precise model but focus on how to interact between U-Mobility and C-Mobility. The complex model may defeat a clear presentation of the main topic of this work.

Thus, we select U-Mobility in the simulation as meandering model and ring-like model because 1) we believe that the meandering model presents the features of movement in a current (e.g. narrow sea channel) while the ring-like model captures some major characteristics of the wave flow in a closed or half-closed region (e.g. sea bay); 2) the ring-like model is a randomized model in contrast to the deterministic meandering model. This makes the design and analysis of the C-Mobility of the mobile sensors more comprehensive.

In this paper, our C-Mobility model is designed such that it can integrate different U-Mobility models; and we will test both these two U-Mobility models.

## ACKNOWLEDGMENTS

The research was supported in part by grants from RGC under N HKUST609/07, by grant from HKUST under MRA05/06.EG03, by grant from National Basic Research Program of China (973 Program) under No. 2006CB303000, the NSFC oversea Young Investigator grant under No. 60629203, National 863 Program of China under No. 2006AA01Z228; by grant from HKPU under ICRG G-YG78, A-PB0R, and RGC under GRF PolyU 5305/08E.

## REFERENCES

- [1] Autonomous Ocean Sampling Network, <http://www.mbari.org/aosn/>.
- [2] MBARI drifters, <http://www.mbari.org/muse/platforms/drifters.htm>.
- [3] Lake Illawarra Authority - Lake Facts: Tides and Currents. [http://www.lia.nsw.gov.au/facts/tides\\_currents.html](http://www.lia.nsw.gov.au/facts/tides_currents.html).

- [4] UAM Moorings, <http://www.mbari.org/muse/platforms/UAM-mooring.html>.
- [5] E. Smith, et al., Satellite-Derived Sea Surface Temperature Data Available From the NOAA/NASA Pathfinder Program, [http://www.agu.org/eos/\\_elec/95274e.html](http://www.agu.org/eos/_elec/95274e.html).
- [6] Z. Abrams, A. Goel, and S. Plotkin, "Set K-Cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks", in *Proc. of IPSN'04*, April. 2004.
- [7] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater Acoustic Sensor Networks: Research Challenges", in *Ad Hoc Networks*, vol. 3, no. 3, pp. 257-279, Mar. 2005.
- [8] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh, "Fidelity and Yield in a Volcano Monitoring Sensor Network", in *Proc. USENIX OSDI'06*, Seattle, WA, Nov. 2006.
- [9] F. Bai and A. Helmy "A Survey of Mobility Models", <http://nile.usc.edu/important/chapter1.pdf>.
- [10] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research", in *Wireless Communication and Mobile Computing (WCMC): Special Issue on Mobile Ad Hoc Networking: Research, Trends, and Applications* vol. 2., no. 5., pp. 483-502, 2002.
- [11] A. Caruso, F. Paparella, L. Vieira, M. Erol, and M. Gerla, "The Meandering Current Mobility Model and Its Impact on Underwater Mobile Sensor Networks", in *Proc. IEEE INFOCOM'08*, Phoenix, AZ, Apr. 2008.
- [12] X. Du and F. Lin, "Improving Sensor Network performance by Deploying Mobile Sensors", in *Proc. the 24th IEEE International Performance, Computing and Communications Conference*, Phoenix, AZ, Apr. 2005.
- [13] M. Dunbabin, J. Roberts, K. Usher, G. Winstanley, and P. Corke, "A Hybrid AUV Design for Shallow Water Reef Navigation", in *Proc. IEEE ICRA'05*, Barcelona, Spain, Apr. 2005.
- [14] C. Ebbesmeyer, C. Coomes, V. Kolluru, and E. Edinger, "Net Water Movement in Budd Inlet: Measurements and Conceptual Model", in *Proc. Puget Sound Research Conference*, Seattle, WA. Mar. 1998.
- [15] M. Erol, L. F. M. Vieira, A. Caruso, F. Paparella, M. Gerla, and S. Oktug, "Multi Stage Underwater Sensor Localization using Mobile Beacons", in *IEEE SensorComm'08*, France, 2008.
- [16] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. HICSS'00*, Maui, Hawaii, Jan., 2000.
- [17] A. Howard, M. Mataric and G. Sukhatme, "Mobile Sensor Network Deployment Using Potential Field: a distributed scalable solution to the area coverage problem", in *Proc. International Conference on Distributed Autonomous Robotic Systems'02*, June. 2002.
- [18] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks", in *Proc. ACM IPSN'07*, Cambridge, MA, Apr. 2007.
- [19] S. Kumar, T. Lai, J. Balogh, "On k-coverage in a mostly sleeping sensor network", in *Proc. ACM MOBICOM'04*, Philadelphia, PA, Sept. 2004.
- [20] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley, "Mobility improves coverage of sensor networks", in *Proc. MOBIHOC'05*, Urbana-Champaign, IL, 2005.
- [21] J. Luo, D. Wang and Q. Zhang, "Double Mobility: Coverage of the Sea Surface with Mobile Sensor Networks", in *Proc. INFOCOM'09*, Rio De Janerio, Brazil, 2009.
- [22] J. Pedlosky, *Ocean Circulation Theory*. Heidelberg: Springer-Verlag, 1996.
- [23] G. Sibley, M. Rahimi and G. Sukhatme, "Robomote: A Tiny Mobile Robot Platform for Large-Scale Sensor Networks," in *Proc. IEEE ICRA'02*, Washington D.C., May, 2002.
- [24] R. Signell and J. List, "Modeling Waves and Circulation in Lake Pontchartrain", *Gulf Coast Association of Geological Societies Transactions*, vol. 47, pp. 529-532, 1997.
- [25] S. Slijepcevic and M. Potkonjak, "Power Efficient Organization of Wireless Sensor Networks", in *Proc. Int. Conf. Commun.*, Helsinki, Finland, June 2001, pp. 472-476.
- [26] D. Wang, Q. Zhang, and J. Liu, "Partial Network Coding: Theory and Application for Continuous Sensor Data Collection", in *IEEE IWQoS'06*, New Haven, CT, 2006.
- [27] D. Wang, J. Liu, and Q. Zhang, "Field Coverage using a Hybrid Network of Static and Mobile Sensors", in *IEEE IWQoS'07*, Chicago, IL, 2007.



**Ji Luo** (S'07) received the BEng degree in computer science from Tsinghua University, Beijing, China in 2006. He is current a PhD candidate in the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. His current research interests include wireless sensor networks and mobile networks. He received the Best Paper Award in IEEE GlobeCom 2007. He is a student member of the IEEE.



a member of the IEEE.

**Dan Wang** (S'05-M'07) received the B.Sc degree from Peking University, Beijing, China, in 2000, the M.Sc degree from Case Western Reserve University, Cleveland, Ohio, USA, in 2004, and the Ph. D. degree from Simon Fraser University, Burnaby, B.C., Canada, in 2007; all in computer science. He is currently an assistant professor at the Department of Computing, The Hong Kong Polytechnic University. His research interests include wireless sensor networks, Internet routing, and peer-to-peer networks. He is



conferences in the areas of wireless/Internet multimedia networking, wireless communications and networking, and overlay networking. She is the inventor of about 30 pending patents. Her current research interests are in the areas of wireless communications and networking, IP networking, multimedia, P2P overlay, and wireless security. She has also participated many activities in the IETF ROHC (Robust Header Compression) WG group for TCP/IP header compression.

Dr. Zhang is the associate editor for the IEEE Transactions on Wireless Communications, IEEE Transactions on Multimedia, IEEE Transactions on Vehicular Technologies (2004-2008), Computer Networks and Computer Communications. She has also served as guest editor for the IEEE Wireless Communications, IEEE Journal on Selected Areas in Communications, IEEE Communication Magazines, ACM/Springer Journal of Mobile Networks and Applications (MONET), and Computer Networks. She has also involved in the organization committee for many IEEE and ACM conferences.

Dr. Zhang received TR 100 (MIT Technology Review) world's top young innovator award in 2004, the Best Asia Pacific (AP) Young Researcher Award elected by the IEEE Communication Society in 2004, and the Best Paper Award by the Multimedia Technical Committee (MMTC) of the IEEE Communications Society and Best Paper Award in QShine 2006, IEEE Globecom 2007, and IEEE ICDCS 2008. She received the Oversea Young Investigator Award from the National Natural Science Foundation of China (NSFC) in 2006. Dr. Zhang is Chair of the Multimedia Communication Technical Committee of the IEEE Communications Society. She is also a member of the Visual Signal Processing and Communication Technical Committee and the Multimedia System and Application Technical Committee of the IEEE Circuits and Systems Society.