Mobile Barrier Coverage for Dynamic Objects in Wireless Sensor Networks

Linghe Kong[1], Yanmin Zhu[1], Min-You Wu[1], and Wei Shu[2]
[1]Shanghai Jiao Tong University, China
[2]University of New Mexico, USA
[1]{linghe.kong, yzhu, mwu}@sjtu.edu.cn, [2]shu@ece.unm.edu

***Abstract*—This paper studies mobile barrier coverage (MBC) surrounding dynamic objects. In the real world, several dynamic objects can benefit from MBC. For example, marching troop can detect any adversary intrusion without blind spot by MBC. However, conventional works only focused on barrier coverage for static objects, which fail when the objects start to move. Issues to address these dynamic-object scenarios, we propose the problem of mobile barrier coverage for dynamic objects. The most challenge is how to effectively maintain MBC when the motion of objects are unpredictable. We propose a fully distributed algorithm for mobile sensor nodes to cooperatively move and maintain the high-quality barrier coverage. The extensive simulations based on large-scale trace data demonstrate the efficiency and efficacy of the proposed algorithm.**

***Keywords*—Wireless sensor networks, mobile barrier coverage, dynamic objects**

I. Introduction

In wireless sensor networks (WSNs), barrier coverage has attracted wide and serious concern. A bulk of sensor nodes are organized as a sensing barrier for intrusion detection, so-called barrier coverage. This application can be set on the border line to discover the stowaways, or be deployed around the safe vault to sense thieves. Many efforts have been made for barrier coverage [5], [6], [14], [17] in the literature.

Nevertheless, all existing works, no matter the sensor nodes are mobile [1], [13], [15], [23] or not [4], [10], [22], [26], [27], only focus on static objects. *e.g.*, the border line, the safe vault. While in real world, many objects are dynamic (Their shape and position are time-varying). In several scenarios, they do need barrier coverage. When the objects begin to move, the-state-of-art solutions fail. This is the motivation behind our study of MBC for dynamic objects in WSNs.

Compared with the conventional barrier coverage, MBC inherits all properties such as intruder detection. Furthermore, a new feature of MBC is that MBC transforms according to the change of dynamic objects, namely, Transformation-On-Demand (TOD). For supporting TOD, the ability of BIdirectional Sensing (BIS) is indispensable. That is, MBC can sense not only the outside intruders, but also the boundary of the inside objects.

MBC is useful in many applications. We can divide them into two categories: (1) Internal protection. *e.g.*, in order to ensure the safety of a marching troop, mobile sensor nodes can keep moving surround the troop (due to TOD feature) to detect any adversary intrusion without blind spot. (2) External protection. *e.g.*, MBC can keep outlining the contaminated region of the diffusing poisonous gas (due to BIS feature), then sense and warn any entrant to leave away from this region.

There are three major challenges in maintaining MBC for dynamic objects. First, the object is dynamic and its time-varying shape is unpredictable. Second, it is desirable to maximize the detection capability of MBC, which is usually characterized by the number of barriers K [14]. Third, the energy of a sensor node is always limited. In short, the problem we are interested in is to provide MBC while maximizing its monitoring performance and reducing the cost of sensor movement. Moreover, since a WSN may extend to a very large scale, it is practical that the solution is distributed.

We study the problem of mobile barrier coverage for dynamic objects (MBC-DO) as follows:

First, we formulate the *K-D* MBC problem, which is to maximize K and then minimize the total distance D of all sensor nodes during the MBC period. Assume the change of dynamic object is known, we derive the theoretical maximum K when the number of sensor nodes n is given. After that, we summarize the optimal movement pattern of nodes for keeping the maximum K. Then we derive the minimum D while the optimal movement pattern is adopted.

Second, since the future change of object is unpredictable, the theoretical optimum cannot be achieved in practice. We propose a distributed and real-time Elastic Barrier Algorithm (EBA) for the motion plan of sensors. By imitating an elastic band stretched around an object, this algorithm coordinates the sensors to form a dynamic belt wrapping like the convex hull surrounding the object. Each sensor makes its movement decision locally by exchanging their real-time information with its neighbors. It is also a lightweight algorithm that any node's computation cost is O(1).

Finally, we do extensive simulation based on a real trace collected for toxic red tide populations in the Western Gulf of Maine. Performance results show that EBA retains MBC approaching the optimal K. And the total distance of all sensors is only 12% more than the theoretical minimum.

In summary, the contributions of this paper are as follows.

- To our best knowledge, this is the first work that address the mobile barrier coverage problem for dynamic objects in wireless sensor networks.
- We derive the theoretical optimal $K-D$ MBC, and prove the feasibility of a distributed approach to realize it.

- We develop a distributed algorithm that coordinates the sensor nodes to retain $K-D$ MBC for dynamic object. Through simulations, we demonstrate that the proposed algorithm performs well in a realistic setting.

The paper organization is as follows. The related work is presented in Section II. In Section III, we establish models, describe the metrics and formulate the problem. In Section IV, we show the derivation of the optimal K and D. In Section V, we prove the feasibility of a distributed algorithm and elaborate the design of EBA. The performance is simulated in Section VI. In Section VII, we conclude this paper.

II. Related Work

Barrier coverage [9] in WSN is valuable for intrusion detection and border surveillance. We classify the works of barrier coverage into three categories.

Stationary sensor nodes for static objects: Most works fall in this category, in which the barrier are organized by stationary nodes in order to protect static objects. Barrier coverage is firstly introduced [14] and an algorithm is proposed to determine whether a barrier is built or not. [14] also proves that no distributed algorithm can judge the existence of the global barrier coverage. Then, a distributed algorithm for judging the local barrier coverage [5] is studied. In recent years, a bulk of works study such barrier coverage in different directions such as [2] for reliable density estimation, [17] for strong barrier coverage, [6] for quality measurement, [22] for line-based deployment, and [15] for probabilistic sensors.

Mobile sensor nodes for static objects: With the development of robotics [8], [19], the mobile sensor node becomes practical for many real applications [7], [12], [16]. Several works fall in this category, in which mobile nodes form the barrier surrounding static objects. Such as [1] studies the optimal movement, [21] studies the mobile sensors with limited mobility, and [13] studies the barrier initialization.

Mobile sensor nodes for dynamic objects: In this category, the objects begin to move or transform, so the mobile sensor nodes are adopted in order to keep besieging the objects. Conventional methods for static objects cannot maintain the MBC. Research in this category is vacant. Thus, we study the problem of MBC for dynamic objects in this paper.

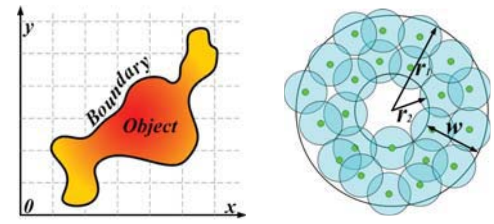
III. Problem Statement

A. Models

Our problem considers dynamic objects, intruders and mobile sensor nodes moving on a two-dimensional plane Γ during time T. The beginning of T is denoted $t=0$. The Euclidean distance between points a_1 and a_2 is noted by $d(a_1,a_2)$. If A_1 and A_2 are two point sets, $d(A_1,A_2)=min\{d(a_1,a_2)|a_1 \in A_1, a_2 \in A_2\}$.

Definition 3.1 [Dynamic object: O**]** A dynamic object O is an area enclosed by a continuous boundary, which deforms and moves on Γ.

By definition, a dynamic object has a closed, continuous boundary as shown in Fig. 1(a). Therefore it is single, which cannot split into several parts.



(a) (b)

Fig. 1. (a) A dynamic object on the plane Γ with boundary as a closed curve. (b) Sensors moving in a dynamic belt of dimension$(2\pi r_1, 2\pi r_2, r_1 - r_2)$.

For MBC, we are only interested in the boundary changing of O, denoted by $B(t)$. Let v_o denotes the maximum velocity of any point on the boundary. We assume when $t=0$, $B(t)$ is known. The change of $B(t)$ is unknown when $t>0$. There have been a few methods that can find the boundary of an object at $t=0$ (e.g., computing geometry [3] and boundary detection [24]) and then form the barrier coverage [13].

Definition 3.2 [Intruder: I**]** An intruder is an entity that is desired to be detected when it moves close to O.

The intruder I can be a point, a changeable area or several separate areas. Intruders are unknown to O.

Definition 3.3 [Warning distance: ε**]** It is the shortest allowed distance between an intruder and the object, by which MBC is required to detect the intruder. The value of ε is set according to diverse applications.

We assume that $d(O,I)>\varepsilon$ at $t=0$. At any $t>0$, when $d(O,I)\leqslant\varepsilon$, at least one sensor node should cover the intruder.

Definition 3.4 [Sensor nodes: $S=\{s_i\}$**]** In order to detect any unknown I, a closed barrier besieging O is necessary so that any I can be immediately detected when it goes across the barrier. A mobile sensor network is one of the best candidates providing such a barrier.

Totally n sensor nodes are utilized, denoted by $S=\{s_i, i=1,\cdots,n\}$. The maximum velocity of any s_i is v_s. Every node knows its own position. A node can sense both the boundary of O and the intruder I within its sensing range. We adopt the widely used *homogeneous disk model* [5]–[7], where every s_i has an equal sensing range r_s. The communication range r_c is assumed to be greater than $2r_s$, which is the common assumption in other coverage works [13], [14], [18].

Definition 3.5 [Dynamic belt] It is defined as a virtual closed belt region surrounding the O in T. Sensor nodes are moving in this virtual region. An intruder cannot reach O without crossing this belt.

Definition 3.6 [Dynamic belt of dimension $(\lambda_1,\lambda_2,w,t)$**]** A dynamic belt is defined as a time-vary region bounded by two closed curves λ_1 and λ_2, whose lengthes are denoted by l_1 and l_2. The width of this belt is $w=d(\lambda_1,\lambda_2)$.

Fig.1(b) illustrates illustrates a snapshot of a dynamic belt with dimension $(2\pi r_1, 2\pi r_2, r_1-r_2, t)$. *i.e.*, this belt is the region between two concentric circles at time t.

MBC demands that for $\forall t\in T$, $A(\lambda_1)\supseteq A(\lambda_2)\supseteq A(B(t))$ and $d(\lambda_2, B(t))\geqslant\varepsilon$, where $A(\lambda)$ denotes the area enclosed by a curve λ. Moreover, ε should be smaller than r_s; otherwise, a node cannot sense the change of boundary. If λ_1 and λ_2 are completed superposition, the belt region becomes a closed curve. In this case, nodes are moving along this curve.

B. Metrics

The goal of the sensor nodes is to maintain MBC in T. There are two performance metrics guiding our design.

Metric 3.1 [Number of barriers: K**]** A MBC has K barriers if and only if an intruder is detected by at least K nodes when it crosses the dynamic belt. A MBC with K barriers is referred to K-MBC. Increasing K can produce a higher overall reliability for intruder detection.

Metric 3.2 [Total travel distance: D**]** It is defined as the sum of travel distances of all nodes in a given time period from t_1 to t_2. $u_i(t_1, t_2)$ is the distance travelled by sensor s_i from t_1 to t_2. The total travel distance is given by

$$D(0,T) = \sum_{i=1}^{n} u_i(0,T). \quad (1)$$

Energy saving is crucial in the design of WSNs. The energy consumption is considerable [18] by the mechanical motion of a sensor node. For energy efficiency, it is important to reduce the total travel distance.

C. Problem Formulation

Definition 3.7 [*K-D* MBC] Given n sensors, how do the sensor nodes plan their movements with the objective to minimize the total travel distance under the constraint that the number of barriers is maximized at any time.

K-D MBC problem is formulated by

$$\begin{cases} \text{Objective:} & \min D(0,T), \\ \text{Subject to:} & \max(K) \text{ during } T. \end{cases} \quad (2)$$

IV. Theoretical Analysis

In this section, we analyze how to derive the maximum K when a dynamic object is given, and what is the optimal movement pattern for the mobile sensor nodes in order to achieve $\max(K)$ and $\min(D)$.

A. Maximum Number of Barriers K

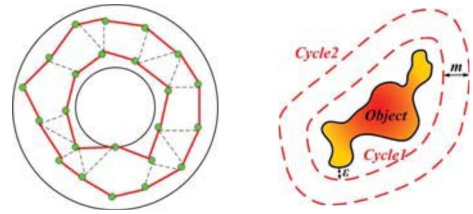
Since the dynamic object may be deforming over time and the number of sensor nodes is fixed, the maximum number of barriers K is changing in response to the changes of the dynamic object. Intuitively, K becomes smaller when the dynamic object becomes larger.

Theorem 4.1 In MBC, the optimum K is achieved when all sensors are evenly distributed on the ε-convex hull (ε-CH) of the dynamic object. Since the object is dynamic, K is time varying. The real-time optimum value is $\frac{2nr_s}{\beta(B(t))+2\pi\varepsilon}$, where $\beta()$ is a function to calculate the length of convex hull.

Proof: We prove Theorem 4.1 by introducing 4 lemmas.

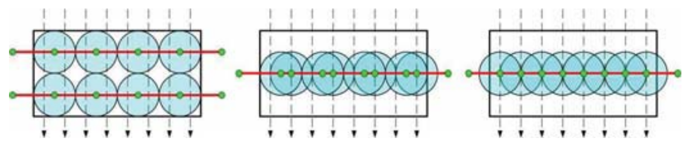
Lemma 4.1 A dynamic belt region provides K-MBC if and only if there are K vertex-disjoint cycles in its coverage graph.

The coverage graph $CG =< V, E >$ includes vertex V and edges E. V is the set of sensors' positions and E is the set of connected relationship when the distance between any two sensor nodes is less than $2r_s$. The example of CG for Fig.1(b) is shown in Fig.2(a). A cycle is defined as a closed sequence of edges and satisfies $A(cycle) \supseteq A(O)$. Cycles are disjoint, *i.e.*, they do not share any vertex. Lemma 4.1 can be proved in the same way as Theorem 4.2 in [14].



(a) (b)

Fig. 2. (a) Coverage graph of the sensor network represented by Fig. 1(b). There are two vertex-disjoint cycles in the closed belt region. (b) Two cycles barrier cover a dynamic object. The interval between chain and object is ε. The interval between two cycles is m.



(a) (b) (c)

Fig. 3. (a) Sensors distributing on two cycles with $K = 2$ in a segmented part of a dynamic belt. (b) Sensors distributing on one chain with $K = 2$. (c) Sensors evenly distributing on one chain with $K = 2$.

Given by Theorem 4.2 and 5.1 in [14], the value of K is the number of cycles in CG. Since n is given and any edge between two sensors is $\leq 2r_s$, the upper bound of total length of all cycles is $\leq 2nr_s$. Hence, the maximum K can be obtained when the length of every cycle is minimized.

Lemma 4.2 The shortest cycle is the ε-CH of a dynamic object.

As the analysis in [20], assume there is a finite point set Z, the smallest 2D polygon (denoted by Ω) contains Z is its convex hull. This convex hull has the smallest area and the smallest perimeter of all polygons that can contain Z.

ε-CH is a closed circle satisfying: (1) contain the convex hull (2) the distance between ε-CH and the convex hull is ε. *i.e.*, $A(\varepsilon\text{-CH}) \supseteq A(\text{CH})$ and $d(\varepsilon\text{-CH}, \text{CH}) = \varepsilon$. From [20], we obtain that the perimeter of an ε-CH is $\beta(B(t)) + 2\pi\varepsilon$ when the perimeter of a convex hull is $\beta(B(t))$.

At any certain time t, a dynamic object can be treated as a static area. Thus, when $\varepsilon = 0$ is set, the shortest cycle is the convex hull of O. When $\varepsilon > 0$, the shortest cycle is the ε-CH of the dynamic object. This derives Lemma 4.2.

Lemma 4.3 The maximum K is achieved when the dynamic belt is $(\beta(B(t)) + 2\pi\varepsilon, \beta(B(t)) + 2\pi\varepsilon, 0, t)$, and the value is $K = \frac{2nr_s}{\beta(B(t))+2\pi\varepsilon}$.

An obvious distribution pattern of the cycles is shown in Fig.2(b), where $d(cycle1, cycle2) = m(m < r)$. The perimeter of $cycle2$ is $\beta(B(t)) + 2\pi\varepsilon + 2\pi m$. Compared with the pattern that $cycle1$ and $cycle2$ are completed overlapped, this pattern (shown in Fig.2(b)) wastes $\pi\varepsilon(K-1)m/r$ nodes.

Hence, *K* is only determined by the number of cycles, but independent to the distribution of the cycles. When two cycles partially or completely overlap, *K* remains to be two, as illustrated in Fig.3(a) and Fig.3(b). Any intruder crossing the belt by any path is detected at least twice.

When all cycles take the shortest length, they overlap on the ε-CH. The dynamic belt region therefore becomes a dynamic chain whose $w = 0, l_1 = l_2 = \beta(B(t)) + 2\pi\varepsilon$. In this case, *K* is maximized, $K = \frac{2nr_s}{\beta(B(t))+2\pi\varepsilon}$. Now, it is apparent to find that a dynamic chain provides *K*-MBC if any point of the chain is covered by *K* nodes, *i.e.*, there are *K* sensors in any $2r_s$ distance of the ε-CH.

Lemma 4.4 It is a sufficient condition for maximizing the number of barriers *K* that the sensors are evenly distributed on the ε-CH of the dynamic object.

As shown in Fig. 3(c), the sensors are evenly distributed on the ε-CH of the dynamic object. The distance separating any two closest neighbors is $\Delta = \frac{\beta(B(t))+2\pi\varepsilon}{n}$. Thus, there are $K = \frac{2r_s}{\Delta} = \frac{2nr_s}{\beta(B(t))+2\pi\varepsilon}$ sensors over a chain segment of length $2r_s$, which is equal to the maximum *K* in Lemma 4.3.

Then Theorem 4.1 follows. ■

Observation 4.1 Towards maximizing the number of barriers *K* continuously, the sensors should stay on the ε-CH and evenly distribute themselves.

The ε-CH changes when the dynamic object deforms, and therefore the chain should adapt to the change of ε-CH. The sensors must move along the ε-CH; otherwise, the formed chain would not be shortest and results in a non-maximal *K*.

B. Minimum Total Travelled Distance D

After having deriving the maximum *K*, we proceed to the next question about how to move the sensors in order to achieve a minimum total travel distance. However, answering this question is a great challenge. There are countless possible ways to move for the sensors. In addition, at any time instance, the sensors should collectively form a belt providing a *K*-MBC where *K* is maximized at that time instance.

To gain the insight into the question, we simplify the motion of the dynamic object as a multi-staged process. With this model, we essentially divide the time into small time slots. We then provide a way to computing the minimum travel distance when the motion track of the dynamic object from 0 to *T* is given.

Theorem 4.2 The *K-D* MBC problem can be reduced to a multiple-stage minimal weighted bipartite matching problem.

Proof: We look at two arbitrary, consecutive stages t_j and t_{j+1}. Fig. 4(a) shows an example of the sensor movements from stage t_j to stage t_{j+1}. Since *K* should always be maximized, the sensors are evenly distributed on the ε-CH at both stage t_j and t_{j+1}. For the purpose of barrier coverage, the sensors can be considered as non-differentiable, i.e., exchanging the positions of any two sensors does not change the property of barrier coverage. Recall that the sensors are evenly distributed on the ε-CH. By shifting all the sensors along the ε-CH by a small distance, we can obtain another possible positioning of the sensors without losing MBC. Therefore, it is reasonable to assume that at each stage there are φ different ways of positioning of the non-differentiable sensors. Given a specific positioning of the sensors at stage t_{j+1}, the situation of sensor movements is shown in Fig. 4(b). It is intuitive that each sensor must fill in one of n candidate places at stage t_{j+1}. The movement problem of the sensors from stage t_j to t_{j+1} is essentially a weighted bipartite matching [25]. Each edge is associated with a weight representing the movement distance that a sensor moves from the position at stage t_j to the corresponding position at stage t_{j+1}.

Consider the whole movements of the sensors from stage 0 to stage T. It is a multi-staged, weighted bipartite matching process, as shown in Fig. 4(c). Each circle represents a specific positioning of all sensors. Each arrow represents a possible movement pattern of the sensors from a given positioning at the previous stage to another positioning at the next stage. From stage t_j to t_{j+1}, since there are φ different ways of positioning of the sensors, there are φ different edges from a positioning at stage t_j to stage t_{j+1}. Then Theorem 4.2 follows. ■

With Theorem 4.2, we have a brute force algorithm to compute minimum total travel distance *D* in the *K-D* MBC problem. For each edge in Fig. 4(c), there are $n!$ possible ways for the sensors to move from a positioning to the next positioning. Thus, the total search space is O($\varphi^T n!^T$). The following corollary gives a much faster algorithm to compute the minimum total travel distance.

Corollary 4.2 The minimum total travel distance *D* in the *K-D* MBC problem can be computed with complexity of O($\varphi^T n^{3T}$).

Proof: From stage 0 to stage T, there are in total φ^T combinations of possible positioning ways of the sensors. For a specific combination of positioning, we propose an efficient algorithm to compute the minimum travel distance.

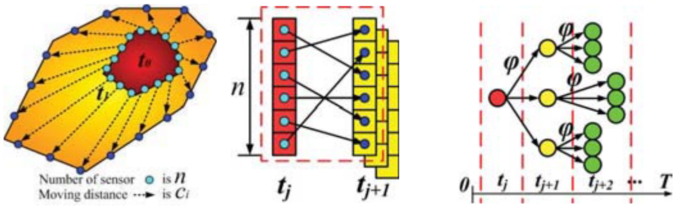
Finding an weighted bipartite matching is known as the assignment problem. It can be solved by using a modified shortest path search in the augmenting path algorithm. If the Bellman-Ford algorithm is used, the running time becomes $O(V^2E) = O(n^4)$, or the edge cost can be shifted with a potential $O(V^2log(V) + VE) = O(n^3)$ running time with the Dijkstra algorithm and Fibonacci heap. Such classical algorithms can solve the problem.

Thus, the total complexity is O($\varphi^T n^{3T}$). ■

Observation 4.2 Towards minimizing the total travel distance, the sensors plan their movements according to the following principles.

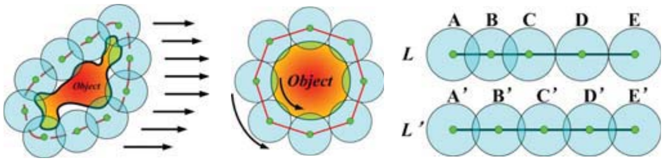
- When the object moves as translation of a rigid, the sensors keep the ε-CH shape and move together with the object.
- When the object grows or shrinks by scaling, the sensors scale the ε-CH shape with the object.
- When the object rotates, there exist other better strategies for the sensors than moving together with the boundary.

For simplification, the motion of the dynamic object is the combination of translation, scaling and rotation. For the translation movement, the sensors must move the same distance as



(a) (b) (c)

Fig. 4. (a) Sensors move to candidate positions on a convex hull from t_0 to t_1. (b) Two consecutive stages MBC problem converts to a weighted bipartite matching problem. (c) A *K-D* MBC problem reduces to a multiple-stage weighted bipartite matching problem.



(a) (b) (c)

Fig. 5. (a) Translation movement of the dynamic object. (b) Rotation movement of the dynamic object. (c) Sensors on chain *L* move to the positions on chain *L'* in order to satisfy the balance condition.

the dynamic object in order to maintain *K*-MBC. The same reason applies for the scaling situation. However, for rotation, the sensors can reduce the travel distance by forming a chain which does not rotate strictly with the dynamic object, with an analogy of motion without friction. Fig. 5(b) shows an ideal example in which the object is round and only rotates counterclockwise. We compare two movement strategies. In the first strategy, all sensors remain stationary, and in the second strategy, all sensors rotate together with the object. These two strategies provide *K*-MBC, but it is obvious that the first strategy has a zero travel distance. In summary, it is our observation that a good strategy for sensor movement is to move with the dynamic object, but without friction.

V. Elastic Barrier Algorithm

The analysis in the previous section provides us with a framework to understand what is the maximum number of barriers and what is the optimal motion pattern for the sensors. However, such theoretical analysis does not result in a practical solution for mobile sensor networks. First, the optimal movement pattern is derived when the motion track of the dynamic object is given. In practice, however, it is impossible to know the future motion of the dynamic object since it is very stochastic and may be influenced by many factors in the real world. Second, the algorithm assumes global information. To collect the global, dynamic information in the sensor network, the cost would be prohibitive. Third, even with the global information, the complexity of the algorithm is too high for a sensor to implement.

Thus, we propose a practical, distributed algorithm for the mobile sensor network to provide MBC continuously while reducing the travel distance of the sensors. Since the future motion of the dynamic object is unpredictable, the distributed algorithm in nature is heuristic.

To device a distributed algorithm, we face the first key question: is it feasible for the sensors to determine whether there exists *K*-MBC?

Kumar [14] has proved that no distributed algorithm can answer the question about the existence of global barrier coverage in a stationary scenario. Different from their scenario, our setting is that the sensors are mobile.

A. Feasibility of Distributed Implementation

Theorem 5.1 It is feasible for the mobile sensors to determine the existence of K-MBC locally.

Proof: We begin the proof with two definitions:

Definition 5.1 [Global balance condition] When the global balance condition satisfies, the sensors evenly distribute on the ε-CH.

Definition 5.2 [Local balance condition] When the local balance condition satisfies, a sensor has an equal distance to its two immediate neighbors.

Lemma 5.1 When every sensor satisfies the local balance condition, the global balance condition satisfies.

Since it is not difficult to understand this lemma, we omit the proof details.

We propose the following distributed adjustment procedure for the sensors, which eventually reaches the global balance condition. According to this adjustment procedure, a sensor that does not satisfy its local balance condition always moves to the point which equally separates its immediate neighbors. In Fig. 5(c), the adjustment process is illustrated. At the initial state of chain *L*, $d(A,B) = d(B,C)$ and $d(C,D) = d(D,E)$, but $d(B,C) < d(C,D)$, so *B* and *D* satisfy the local balance condition, but *C* does not. According to the adjustment procedure, *C* moves to the middle of *B* and *D*. After the *C*'s movement of *C*, the balances of *B* and *D* disappear. Then, *B* and *D* start to adjust, respectively. Chain *L'* in Fig. 5(c) shows the state where all sensors stopped. When the adjustment process terminates, all sensors reach the local balance condition.

This adjustment procedure requires no global information and can execute in a distributed fashion. The termination of the adjustment procedure implies that all sensors have reached the local balance condition. According to Lemma 5.1, the global balance condition also satisfies, which indicates that *K*-MBC is achieved. Hence, a sensor can locally determine the existence of *K*-MBC. ■

Corollary 5.1 The value of *K* can be locally computed by $K = 2r_s/\Delta$ when the global balance is satisfied, where Δ is the interval distance of the immediate neighbors.

From Lemma 4.4, we have $\beta(B(t)) + 2\pi\varepsilon = n\Delta$.

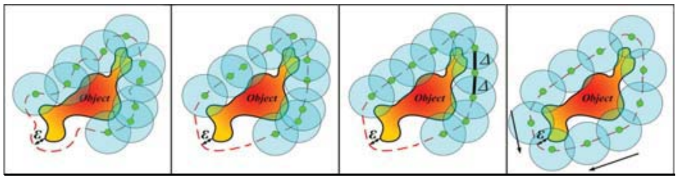
From Theorem 4.1, we have $\beta(B(t)) + 2\pi\varepsilon = 2nr_s/K$.

Combining these formulas together, we get $nK = 2nr_s/\Delta$.

Hence, $K = 2r_s/\Delta$. The result follows.

B. Algorithm Overview

In Section IV, we have two insightful observations for solving the *K-D* MBC problem. We find that an elastic band,



(a) (b) (c) (d)

Fig. 6. (a) Sensors distributing on the ε-boundary extension of the dynamic object. (b) Sensors move to the positions on the ε-CH. (c) Sensors move to the position of middle of two immediate neighbors. (d)Sensors move towards the direction where no immediate neighbor exists.

without friction, being stretched around the dynamic object provides a good strategy. First, an elastic band always wraps around the object and exactly forms the convex hull. Second, a friction-less elastic band can effectively reduce the movement distance by ignoring rotation motion and concave deformation of the dynamic object. Thus, we devise EBA, that mimics the principles of an elastic band.

EBA imitates an elastic band by introducing virtual force between sensors. The virtual force imitates the tension among elastic molecules in an elastic band. With such virtual force, the sensors can form a convex hull and evenly distribute on the convex hull. However, we ignore the friction effect that happens between a wrapping elastic band and the object. Y. Zou et al. [28] introduce the concept of virtual force produced by measuring the distance between sensors.

More specifically, we regulate movements of the sensors by introducing four rules.

Rule 5.1 The sensors should move close to the boundary of the dynamic object and stay on the ε-extension of the boundary.

By following Rule 5.1, the sensors can be distributed on the ε-boundary extension as shown in Fig. 6(a).

Rule 5.2 A sensor should maintain the internal angle being no more than $180°$. A sensor with its two immediate neighbors can form two angles. The internal angle is defined the angle is facing the boundary.

By following Rule 5.1 and 5.2, the sensors can be distributed on the ε-CH, as shown in Fig. 6(b).

Rule 5.3 A sensor should move to the point that has the same distance to its two immediate neighbors.

This rule forces each sensor to reach the local balance condition. In Fig. 6(c), it shows the result by following the previous three rules, where the distances between two immediate neighbors are equal, which is Δ.

Rule 5.4 A sensor should move towards the direction of zero neighbors.

Following this rule, a sensor with only one immediate neighbor moves towards the direction without any neighbor, and this produces the effect of pulling the other sensors to vacant sections on the ε-CH. In Fig. 6(d), it illustrates this effect and the resulting chain on which the sensors are evenly distributed.

Elastic Barrier Algorithm (Executed on sensor s_i)

Input: the sensing range r_s, the warning distance ε;
while True **do**
  Detect immediate neighbors s_{i-1} and s_{i+1};
  Exchange position information with s_{i-1} and s_{i+1};
  Detect the dynamic object and the distance $d(p_{s_i}, O)$;
  $p \leftarrow p_{s_i}$; //p is the next position s_i will to move
  if s_i has two immediate neighbors
    $p_1 \leftarrow$ the middle position of two immediate neighbors;
    if $d(p_{s_i}, O) < r_s$
      $p_2 \leftarrow q$: $d(O,q) = \varepsilon$ and $d(p_{s_{i+1}}, q) = d(p_{s_{i-1}}, q)$;
      $p \leftarrow d(O,p_2) > d(O,p_1) ? p_2 : p_1$;
    else $p \leftarrow p_1$;
  else if s_i has only one immediate neighbor
    $p \leftarrow$ q: $d(p_{s_{i\pm1}}, q) = 2r_s$ and $d(O,q) = \varepsilon$;
  if $p \neq p_{s_i}$
    Move to p;
end while

C. Algorithm Details

Combining the four rules gives EBA, the distributed algorithm to be executed on each sensor. Note that a sensor decides its movement after it combines all effects of the four rules. EBA is a light weight algorithm in which each sensor makes its decision only with local information. Due to the distributed nature of the algorithm, each sensor may undergo multiple adjustments of position before it stops, on the condition that the dynamic object remains stationary at some time instance.

The detail algorithm is given in the algorithm table. With this algorithm, each sensor repeats a series of operations.

When position of immediate neighbors is updated or the change of object boundary is sensed, this sensor computes its new position to move. The next position p should meet $d(p, O) = \varepsilon$ on the perpendicular bisector of immediate neighbors. If this p constructs an internal angle larger than $180°$ against Rule 5.2, the next point must change to the middle position of immediate neighbors. If this sensor has only one immediate neighbor, it moves to the reverse direction of its current immediate neighbor and keeps the distance no more than $2r_s$. This decision of movement depends on only the local information by position messages exchanged with immediate neighbors, so the algorithm is distributed. This algorithm is also light weight since its computation complexity is O(1). The requirement of data storage is tiny as well, which only need to save and update two immediate neighbors' positions.

There are several constraints for the distributed algorithm.

- **Initial condition.** At initialization, the sensors can sense the boundary, or have constructed such a chain that each sensor has two immediate neighbors. In [1] [23], methods have been proposed for moving randomly deployed sensors to the boundary of an object. This makes sure that the initial sensor network deployment meets the initial condition.
- **Upper bound of required velocity.**The upper bound of sensor velocity, v_s, that is required by EBA is $v_o\sqrt{1 + (\frac{2\pi}{n})^2}$

Proof: Consider the worst case in which a sensor moves the longest distance in a unit time. The worst case takes place when the boundary moves outwards at the maximum velocity of v_o. In this case , a sensor has to move outward, and meanwhile adjusts its position on the ε-CH. In a unit time, the perimeter of the convex hull increases by no more than $2\pi v_o$. Since n sensors evenly distribute on the ε-CH, the average distance for a sensor to adjust its position over the ε-CH is no more than $2\pi v_o/n$. By combining the two kinds of distances, the aggregated travel distance of the sensor is $\sqrt{v_o^2 + (\frac{2\pi v_o}{n})^2}$. Thus, the upper bound $v_s = v_o\sqrt{1 + (\frac{2\pi}{n})^2}$. Note that, when *n* is large, the v_s is slightly greater than v_o. ■

- **Critical failure condition.** The critical failure condition reaches when the dynamic object grows too large for the sensors to provide 1-MBC. It is apparent that when the perimeter of the convex hull is larger than $2nr_s$, the critical condition reaches and the sensor network fails to provide 1-MBC.

D. Discussion

Throughout the paper, the dynamic object is assumed to be a continuum. In the real-world, a dynamic object can be a group of discrete entities. A marching troop is such an example. Our algorithm can also deal with such dynamic objects formed by a collection of discrete entities. The group of the discrete points in a plane has a unique convex hull. EBA relies only on its ε-CH and therefore is able to provide MBC for dynamic objects with discrete members.

A plane is assumed in our algorithm. However, in the real world, terrains are complex and may not be a plane. For instance, topographic relief and obstacles pose obstructs which influence sensor movements. If the sensors have the map information, EBA can be extended to deal with such situations. The sensor chain constructed by EBA can take in account the joint shape of both the object and terrain limitations.

VI. Performance Evaluation

A. Simulation Settings

To evaluate the algorithm under a realistic setting, we make use of a motion trace data of toxic red tide populations in the Western Gulf of Maine [29] during the spring run-off periods of 1993. Trace data are visualized as shown in Fig. 7(a). The conspicuous effects of red tides are the associated wildlife mortalities among marine and coastal species of fish, birds, marine mammals and other organisms. Thus, a mobile sensor network is deployed to monitor a red tide on the sea surface and keeps out unaware entities from the danger region.

The red tide region for which the sensor network provides barrier coverage is defined by a boundary on which the density of Alexandrium cells (one kind of red tide) are above 90 mg/L. The maximum area of this region is 1002400m$_2$ and the maximum perimeter of this region is 8220m from April 11 to May 22. During this period, this red tide region moves and deforms itself because of many factors, such as ocean current

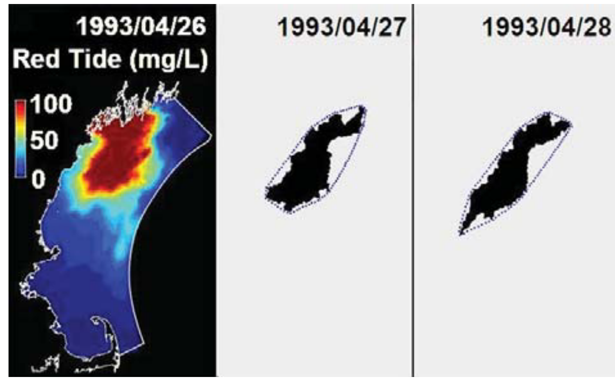


Fig. 7. (a) Red tide data on 1993/04/26. (b)on snapshot of mobile sensors 1993/04/27. (c) on snapshot of mobile sensors 1993/04/28.

and wind. The maximum velocity of the red tide region is 1.296 km per day (0.015m/s average).

In simulations, 100 sensors are deployed for monitoring the red tide. The sensing range is 50m, the communication range is 100m and the distance between the dynamic object and sensors is at least 30m. We adopt the movement parameters used by Starburg AUV [8]. The velocity of a sensor is at most 1.5m/s. The battery allows a continuous movement for a distance up to 7500m. We compare the proposed algorithm with the theoretical optimum, the boundary extension algorithm and the smallest ring algorithm.

We compare the performance of EBA with the theoretical optimum (Opt), the boundary extension algorithm (BE) and smallest ring algorithm (SR). The Opt is calculated according to the analysis in Section IV. With BE, the sensors always move to form a chain as an extended boundary of the dynamic object. With SR, the sensors always move to construct a smallest ring besieging the dynamic object.

B. Performance Results

In Fig. 7(b) and 7(c), the discrete points represent the positions of mobile sensors and the black area represents the red tide region. These figures show two snapshots of the region and the besieging sensors. We can see that the sensors form a chain as the extension convex hull of the region, and the intervals between immediate neighbors are almost equal.

For the shape of the region shown in Fig. 7(c), the perimeter of the 30-extension convex hull is 4965m, the 30 boundary extension is 7650m and the smallest ring is 7008m. We vary the number of sensors from 0 to 1000 and measure the number of barriers achieved by our algorithm. In Fig. 8 we can find

Algorithm	Maximum Velocity Requirement
EBA	0.015m/s
BE	0.015m/s
SR	0.028m/s

TABLE I
THE MAXIMUM VELOCITY REQUIREMENT OF DIFFERENT ALGORITHMS WHEN v_o=0.015M/S.

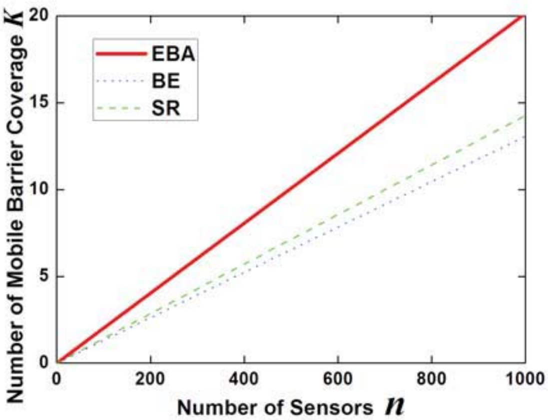


Fig. 8. The number of barriers K increases linearly with the number of sensors n varying from 0 to 1000.

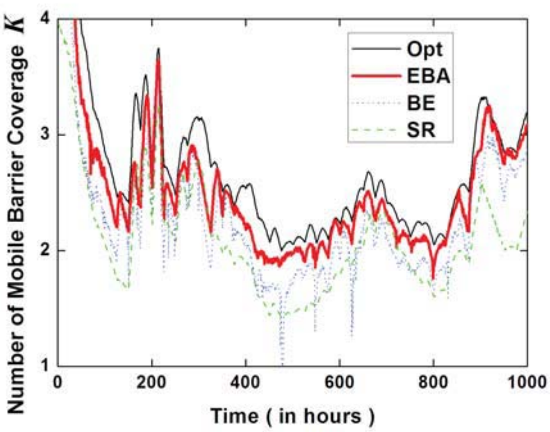


Fig. 9. The variation of the number of barriers K over time.

that the number of barriers increases as the number of sensors increases and yields a linear relation. The performance of EBA is better than BE and SR under any different number of sensors. Since the movement of the red tide is quite stochastic, its shape is usually irregular. EBA outperforms BE and SR in this situation. For instance, when n is equal to 1000, K is 20.14, 13.07 and 14.27 of EBA, BE and SR respectively. The $K$of SR is greater than that of BE, because there are many concavo-convex on the boundary as shown in Fig. 7(c). We can conclude that EBA can achieve the maximal K. Moreover, the more concave convex, the smaller K BE achieves. The less roundish the object is, the worse SR performs.

In Fig. 9 plots the variation of the number of barriers over time is shown. The number of barriers achieved changes under different algorithms, since the region is changing. It can be seen that the changing trends of K are similar among the four curves because they share the same goal of maximizing the number of barriers. The K of EBA is always the closest one to the theoretical optimum, which demonstrates that our algorithm achieves a greater number of barriers. In more detail, we observe the change trend of EBA lag that of the theoretical optimum. This phenomenon is due to the fact that a certain delay is required for the sensors to adjust positions. The fluctuations of BE is more significant than the other algorithms because the performance of BE is affected more by the smoothness degree of the boundary of the target region.

In Fig. 10, the accumulated travel distance over time is shown. As we know, a short traveled distance means lower energy consumption. The traveled distance in EBA is close to that of the optimum and is much smaller than those of BE and SR. At the end of simulation time, EBA, BE and SR move 12.18%, 47.19% and 104.69% distance more than theoretical optimum. At that time, the total traveled distance of EBA is 84122m and each sensor moves 841.22m in average. The maximal distance that a sensor moves in the simulation is 1139.45m. This is practical since a Starburg AUV can move up to 7500m.

Next, we study the convergence of our algorithm. A convergence is reached when all sensors achieve the global balance condition. We count the number of rounds before the convergence is reached. In Fig. 11, the number of rounds required by EBA to converge is shown. On one hand, since the region boundary changes with different velocities, the sensors need different convergence time. A great number of rounds of EBA can provide high probability of convergence. On another hand, sensors need to communicate with neighbors once per round. A small number of convergence rounds in EBA can reduce overheads in time of interest. Hence, it is a tradeoff between convergence probability and overheads to set the rounds of EBA running per hour. In this simulation, the number of rounds is set 60. Then the sensor network can converge in 89.3% of the time of interest.

In Table 1, we measure the maximum velocity requirement by different algorithms when the red tide region moves at a velocity of 0.015m/s. We find that it is sufficient for both EBA and BE that the sensors move as fast as the target region. However, SR needs a greater velocity since it has to maintain sensors as a ring. Since Starbug AUV can move at the speed of up to 1.5m/s, even if the red tide moves at 1.5m/s, by proportional calculation, the sensors are still able to maintain barrier coverage and successfully besiege the red tide region.

In conclusion, EBA outperforms the other two algorithms in terms of the number of barriers and the total travel distance, and is close to the optimum. The simulation demonstrates that EBA is practical in terms of convergence delay, overhead and velocity requirement for real-world applications.

VII. Conclusion

This paper studies the problem of providing barrier coverage for dynamic objects. To the best of our knowledge, this is the first work that raises the problem of mobile barrier coverage. We formulated this problem and provide theoretical analysis on the maximum number of barriers that can be achieved by a given number of sensors, and on the optimal movement pattern for the sensors moving to provide MBC. For

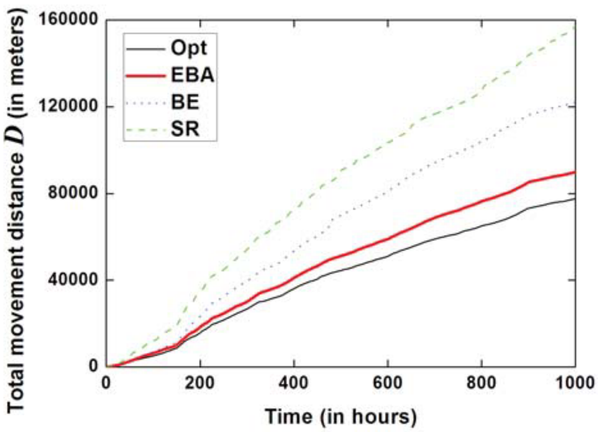


Fig. 10. The accumulated travel distance of all sensors with time.

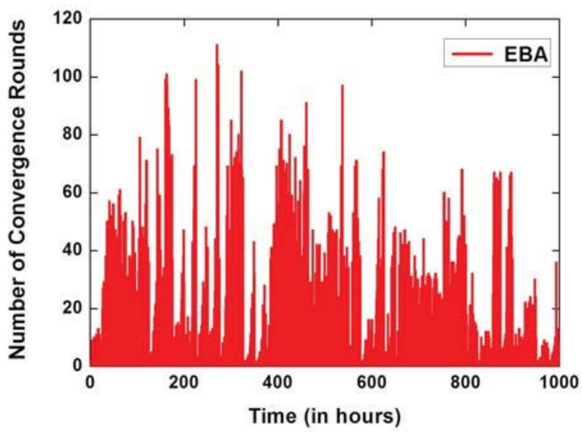


Fig. 11. The number of convergence rounds per hour required by EBA during time of interest.

practical implementation, we propose EBA, a fully distributed algorithm. By imitating an elastic band, EBA continuously retain MBC and achieves the maximum number of barriers. In addition, it effectively reduced the total travel distance of sensors. The algorithm of EBA is verified by trace driven simulations.

ACKNOWLEDGMENTS

This research was supported by NSF of China under grant No.60773091, No.61100210, No. 61170238, No. 60903190, No. 61027009, Doctoral Program Foundation of Institutions of Higher Education of China under grant No.20110073120021, Shanghai Pu Jiang Talents Program No. 10PJ1405800, Innovation Research of SJTU under grant No. Z-030-022, Key Program of National MIIT under Grants no. 2009ZX03006-001-01 and 2009ZX03006-004. We thank Yunhuai Liu, Dong Xuan, and Ming-Chen Zhao for their valuable comments.

REFERENCES

[1] B. Bhattacharya, B. Burmester, Y. Hu, E. Kranakis, Q. Shi, and A. Wiese, "Optimal Movement of Mobile Sensors for Barrier Coverage of a Planar Region", *COCOA*, Canada, 2008.
[2] P. Balister, B. Bollobas, A. Sarkar, and S. Kumar, "Reliable Density Estimates for Coverage and Connectivity in Thin Strips of Finite Length", *ACM MobiCom*, Canada, 2007.
[3] M. Berg, M. Kreveld, M. Overmars, and O. Schwarzkopf, "Computational Geometry: Algorithms and Applications", Springer, Berlin, 1997.
[4] A. Chen, Z. Li, T.-H. Lai, C. Liu, "One-Way Barrier Coverage with Wireless Sensors", *IEEE INFOCOM*, Shanghai, China, 2011.
[5] A. Chen, S. Kumar, and T. H. Lai, "Designing Localized Algorithms for Barrier Coverage", *ACM MobiCom*, Canada, 2007.
[6] A. Chen, T. H. Lai, and D. Xuan, "Measuring and Guaranteeing Quality of Barrier-coverage in Wireless Sensor Networks", *ACM MobiHoc,* 2008.
[7] W. Chen, J. Hou, and L. Sha, "Dynamic Clustering for Acoustic Target Tracking in Wireless Sensor Networks", *IEEE Transactions on Mobile Computing*, Volume 3, Issue 3, pp. 258- 271, 2004.
[8] M. Dunbabin, J. Roberts, K. Usher, G. Winstanley, and P. Corke, "A Hybrid AUV Design for Shallow Water Reef Navigation", *IEEE ICRA*, Spain, 2005.
[9] D. Gage, " Command Control for Many-robot Systems", *In AUVS-92*, 1992.
[10] S. He, J. Chen, X. Li, S. Shen, and Y. Sun, "Cost-Effective Barrier Coverage by Mobile Sensor Networks", *IEEE INFOCOM*, USA, 2012.
[11] A. Howard, M. Mataric, and G. Sukhatme, "Mobile Sensor Network Deployment Using Potential Field: A Distributed Scalable Solution to the Area Coverage Problem", *International Conference on Distributed Autonomous Robotic Systems*, 2002.
[12] L. Kong, D. Jiang, M.-Y. Wu, "Optimizing the Spatio-Temporal Distribution of Cyber-Physical Systems for Environment Abstraction", *IEEE ICDCS*, Genoa, Italy, June, 2010.
[13] L. Kong, X. Liu, Z. Li, and M.-Y. Wu, "Automatic Barrier Coverage Formation with Mobile Sensor Networks", *IEEE ICC*, Cape Town, South Africa, 2010.
[14] S. Kumar, T. H. Lai, and A. Arora, "Barrier Coverage With Wireless Sensors", *ACM MobiCom*, Germany, 2005.
[15] J. Li, J. Chen, and T.-H. Lai, "Energy-Efficient Intrusion Detection with a Barrier of Probabilistic Sensors", *IEEE INFOCOM*, USA, 2012.
[16] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley, "Mobility Improves Coverage of Sensor Networks", *ACM MobiHoc*, USA, 2005.
[17] B. Liu, O. Dousse, J. Wang, and A. Saipulla, "Strong Barrier Coverage of Wireless Sensor Networks," *ACM MobiHoc*, 2008.
[18] J. Luo, D. Wang, and Q. Zhang, "Double Mobility: Coverage of the Sea Surface with Mobile Sensor Networks", *IEEE INFOCOM*, Brazil, 2009.
[19] Y. Mei, Y. Lu, Y.C. Hu, and C.S.G. Lee, "A Case Study of Mobile Robot's Energy Consumption and Conservation Techniques", *IEEE ICRA*, Spain, 2005.
[20] R. Rockafellar, "Convex Analysis", Princeton University Press, 1997.
[21] A. Saipulla, B. Liu, G. Xing, X. Fu, and J. Wang, "Barrier Coverage with Sensors of Limited Mobility", *ACM Mobihoc*, 2010.
[22] A. Saipulla, C. Westphal, B. Liu and J. Wang, "Barrier Coverage of Line-Based Deployed Wireless Sensor Networks", *IEEE INFOCOM*, Brazil, 2009.
[23] C. Shen, W. Cheng, X. Liao and S. Peng, "Barrier Coverage with Mobile Sensors", *IEEE ISPAN*, Australia, 2008.
[24] Y. Wang, J. Gao, and J. Mitchell, "Boundary Recognition in Sensor Networks by Topological Methods", *ACM MobiCom*, USA, 2006.
[25] D. West, "Introduction to Graph Theory (2nd Edition)", Published by Prentice Hall, pp. 111, 2001.
[26] G. Yang, and D. Qiao, "Multi-Round Sensor Deployment for Guaranteed Barrier Coverage", *IEEE INFOCOM*, San Diego, CA, 2010.
[27] G. Yang, and D. Qiao, "Barrier Information Coverage with Wireless Sensors", *IEEE INFOCOM*, Rio de Janeiro, Brazil, 2009.
[28] Y. Zou, and K. Chakrabarty, "Sensor Deployment and Target Localization Based on Virtual Forces", *IEEE INFOCOM*, USA, 2003.
[29] Red Tide Data Trace
http://woodshole.er.usgs.gov/operations/modeling/wgulf/wgulf.html