Received November 24, 2017, accepted December 26, 2017, date of publication January 15, 2018, date of current version March 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2792678

On-Supporting Energy Balanced *k*-Barrier Coverage in Wireless Sensor Networks

CHIEN-I WENG[1], CHIH-YUNG CHANG[2], (Member, IEEE), CHIH-YAO HSIAO[2], CHAO-TSUN CHANG[3], (Member, IEEE), AND HAIBAO CHEN[4]

[1]Department of Information Management, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
[2]Department of Computer Science and Information Engineering, Tamkang University, Taipei 25137, Taiwan
[3]Department of Information Management, Hsiuping University of Science and Technology, Taichung 41280, Taiwan
[4]School of Computer and Information Engineering, Chuzhou University, Chuzhou 239000, China

Corresponding author: Chih-Yung Chang (cychang@mail.tku.edu.tw).

This work was supported in part by the Program for Science and Technology Innovative Research Team of Chuzhou University, in part by the Natural Science Foundation of Educational Government of Anhui under Grant KJ2015B07, in part by the Ministry of Science and Technology of Taiwan under Grant MOST 103-2221-E-032-020-MY3 and Grant MOST 106-2221-E-032-015-MY3, and in part by the Scientific and Technological Innovation Team (Key Technologies and Applications of Internet of Things) of Chuzhou University.

ABSTRACT Barrier coverage in the area of wireless sensor networks (WSNs) has received much attention in the past few years. To guarantee the competent monitoring quality of intruder detection, it is necessary that the barrier satisfies the *k*-barrier coverage. However, the efficient construction of the *k*-barrier whilst maximizing the network lifetime of a given WSN remains an issue. In this paper, an Efficient *k*-Barrier Construction Mechanism (*EBCM*) is proposed. Three phases are presented in *EBCM*. First, the Cover Adjacent Net (CA-Net) phase is developed, aiming to simplify the *k*-barrier coverage problem while reducing the complexity of computation. Second, based on the CA-Net, the construction of defense *k*-barrier phase is further introduced, aiming to construct a maximum number of distinct *k*-barriers. Each of the barriers is composed of a minimum number of sensors. Finally, a Barrier Energy Scheduling is proposed, aiming to schedule the sleep-wake time of all the constructed barriers for achieving its energy balance purpose. Simulation results show the effectiveness of the proposed mechanisms in terms of control overhead, number of barrier constructed and sensors required, sensor utilization, efficiency and success ratio of barrier construction, and network lifetime.

INDEX TERMS *k*-barrier coverage, energy balance, wireless sensor networks.

I. INTRODUCTION

Wireless sensor networks (WSNs) have a wide range of potential application such as measuring the magnitude of earthquakes, monitoring light or temperature in an environment and detecting enemies or chemical gas hazards in military detection application. Coverage is one of the most important issues to be addressed in WSNs. The coverage problems have been widely discussed in literatures, which can be classified into three categories: area coverage, target coverage, and barrier coverage. The area coverage and target coverage problems focus on determining a minimum set of sensors that can cover a given region and a set of specific targets, respectively.

Intrusion detection and border surveillance also constitute a major application category. These applications can be achieved by constructing a defense barrier based on a set of sensors. Barrier coverage is one of the most important issues which can be applied to applications including national border control, critical resource protection, and intruder detection. In a WSN, a defense barrier is formed by selecting a set of sensors whose sensing ranges are contiguous and span across the monitored field. Every object crossing the field from one side to another will be detected by at least one sensor on the defense barrier. A defense barrier composed of a set of sensors is said to satisfy the *k*-barrier coverage if any intrusion is detected by at least *k* distinct sensors [1]. This type of barrier lessens the possibility of missing intruders and enhances fortification and security. Hence, a barrier that satisfies the *k*-barrier coverage is ideal. This problem of barrier coverage has been a topic for discussion in many literatures and studies. These works can be further classified into four branches, which are *geometry-based* methods [1]–[5], *grid-based* method [6], *stripe-based* methods [7]–[9], and *graph-based* methods [10]–[23].

The *geometry-based* methods [1]–[5] analyze the factors that affect the construction of *k*-barrier, including the sensing range, sensing overlapping area, or the relative coordinates among sensors. Based on the results from the factor analyses, a set of sensors is selected for composing the k-barrier. For instance, study [1] proposed an algorithm which analyzed the boundary arcs of sensing overlapping area among sensors. Based on the projection length of boundary arcs, a set of sensors that have large projection length is picked for the construction of the k-barrier. However, analyzing the boundary arcs is time-consuming because of complex and cumbersome computations.

In *grid-based* method [6], the given monitoring area is divided into a number of equal size grids. A barrier that satisfies the k-barrier coverage is formed grid by grid from left to right boundaries. However, this type of method may lose a number of its barriers due to its simplification of the circle-shape sensing range of sensor as the grid-shaped sensing range.

The *stripe-based* methods [7]–[9] partition the monitoring area to the horizontal segments interleaved by the vertical strips. Each segment and strip individually constructs the local horizontal and vertical barrier, respectively. The global barrier of the whole given monitoring area can be achieved if the horizontal barriers in each segment are connected by vertical barriers in neighboring vertical strips. The problem with these methods is that these centralized algorithms have low scalability and high energy consumption rate due to the collection of information from all sensors.

The *graph-based* methods [10]–[23] adopt the graph theory, such as Minimum-Cost Flow, thus focusing on constructing the optimum number of k-barriers. For instance, the algorithm proposed in [14] finds k starting and end points at left and right boundaries, respectively. For each combinational pair of starting and end points, it constructs the shortest path which is disjoint with the other previously discovered paths. The number of combination pairs will be k^2. It then selects the k disjoint paths from the combination pairs as the solution of k-barrier. These centralized approaches, on the downside, consume remarkable amounts of energy and lack both flexibility and scalability.

In this paper, an *Efficient k-Barrier Construction Mechanism* (*EBCM*) considers the advantages of both the *grid-based* and *graph-based* methods. Two decentralized barrier construction algorithms are proposed, aiming to explore the maximal number of disjoint sets of sensors such that each set of sensors satisfies the k-barrier coverage. These two algorithms are *Best-fit Coverage Approach* (*BCA*) and *Top-down One-coverage Barrier Approach* (*TOBA*). Then, disjoint k sets of sensors can be organized as a k-barrier, and several k-barriers can be active by turns for reducing the energy consumption and achieving the purpose of load balance. Compared with both the *grid-based* and *graph-based* methods, the proposed *EBCM* improves not only the construction efficiency but the energy consumption and scalability. Part of the concept has been presented in [21], which focuses on the conceptual descriptions of the *BCA*.

The succeeding passages of this portion itemize the specific contributions of this paper.

1) ***Proposing a Near Optimal Solution for the k-Barrier Coverage Problem.*** By locally considering the coverage contribution of each sensor, the proposed decentralized mechanisms can construct a satisfactory number of distinct barriers. Each constructed barrier consists of a minimum number of sensors, but still guarantees k-barrier coverage. Compared with previous study [14], that is a centralized mechanism and can be treated as an optimal solution. Thus, the proposed *Mechanism EBCM* is a near optimal mechanism.
2) ***Overcoming the Obstacle.*** The *dead-end* problem might arise when constructing a defense k-barrier. The proposed *EBCM* adopts the backtracking policy to overcome this *dead-end* problem and also seeks more feasible solutions for the construction of defense k-barriers. Hence, the proposed *EBCM* can be applied to the obstacle environment.
3) ***Handling the Heterogeneous Problem.*** The proposed *EBCM* can also be applicable in heterogeneous wireless sensor networks. The reason is that each sensor only needs to calculate its coverage contribution for the construction of defense k-barrier based on its sensing range and coordinate.
4) ***Real-Time.*** The proposed *EBCM* adopts the decentralized strategy such that each sensor can locally make its own decisions. Therefore, an immediate reaction and adjustment can be further made when the constructed defense k-barrier encounters an unexpected interruption.
5) ***Problem Simplification.*** The proposed *EBCM* assimilates the advantages from the concepts of *grid-based* and *graph-based* approaches. Thus, the *EBCM* can simplify the problem and reduce the computational complexity.

The remaining parts of this paper is structured as follows. Section II briefly discusses the literatures of k-barrier coverage. Section III introduces the network environment and problem statement. Section IV gives the detailed descriptions on constructing the k-barrier. Section V shows the evaluation results while the paper is concluded in Section VI.

II. RELATED WORK

This section presents the recent studies of four categories: *geometry-based* [1]–[5], *grid-based* [6], *stripe-based* [7]–[9] and *graph-based* [10]–[23], on barrier coverage in WSNs.

A. GEOMETRY-BASED APPROACH [1]–[5]

The *geometry-based* approaches analyze the factors that impact the construction of the k-barrier. Based on the results of the analysis, a set that composes of sensors will be selected to construct the k-barrier. Huang and Tseng [1] investigate the arcs of sensing overlapping areas among sensors. Taking the projection length of arcs as the base, a set that is composed of sensors that have the large projection length will be selected to construct the k-barrier. A problem that arises, however, is that it is time-consuming to determine the arcs of sensing overlapping areas, which results in high complexity of computation. Yang and Qiao [2] analyze the sensing effect of each sensor on the construction of barrier. They also derive the projection length of sensing range of each sensor. A centralized scheme combining all the projection lengths is geometrically developed for constructing the barrier. Similarly, studies [3]–[5] partitioned the sensing ranges of sensors to a geometrical graph. Based on this graph, the geometrical approaches are proposed, thus aiming to minimize the number of sensors for the construction of barrier. There exists, however, several disadvantages. That is, the global information of a given sensor network is required to be collected at sink. This results in significant energy consumption for each sensor. These centralized approaches lead to low flexibility and scalability. As a result, compared to the other decentralized approaches, sensors easily exhaust energy, thus the need to redeploy additional sensors in order to guarantee the quality of required coverage.

B. GRID-BASED APPROACH [6]

The *grid-based* approach transforms the given monitoring area into a grid-based region. The k-barrier is constructed if there is a set of grids that satisfy two requirements. First, each grid independently connects with two neighboring grids toward the left and right grids. Second, the right and left grids are also independently connected with the other two neighboring grids. A decentralized approach proposed in [6] aims to construct the k-barrier using a minimum number of sensors. The monitoring area is partitioned into a number of grids in which the size of grid proportional to the sensing range of sensor.

Ergo, the problem of selecting a set of sensors is transformed into that of selecting a set of grids that are fully covered by sensors. As a result, the k-barrier can be constructed in the manner of grid by grid from the left to right boundaries. However, the number of k-barriers that can be constructed is not maximized because of the simplification of sensing range of each sensor. This proposed approach may lose already existing barriers. The performance drops caused by adopting the *grid-based* method were not investigated. Energy differences among constructed k-barriers, which results in low network lifetime were also not considered.

C. STRIPE-BASED APPROACH [7]–[9]

The *stripe-based* approaches transform the construction of k-barrier into the line covering problem. Fan *et al.* [7] introduced a problem of covering a line interval, which is regarded as the l-barrier, by utilizing sensors. An algorithm is proposed for assigning an energy level to each sensor so that each sensor can fully cover the line interval, therefore minimizing the energy consumption of sensors. However, covering the line intervals is inefficient and consists of rather complex computation processes. Also, the k-barrier coverage problem was not considered. Liu *et al.* [8] partitions the monitoring area into horizontal rectangular segments interleaved by vertical thin strips. Each segment and strip individually constructs the local horizontal and vertical barrier, respectively. The complete barrier of the whole area can be accomplished if the horizontal barriers in each segment are connected by vertical barriers in neighboring vertical strips. However, these centralized approaches require collecting global information from all of the sensors in given monitoring area at sink side. This results in low scalability and significant energy consumptions at sensor side. Chen *et al.* [9] presented an algorithm that measures the quality of the k-barrier. When the quality of such barrier fell below the desired level, they further introduced a method in order to identify the weak zones in the given monitoring area. Then, a method was proposed which aimed to repair all the weak zones until the k-barrier achieved a desired level. However, finding the weak zones, similar to the aforementioned segment, is inefficient and yields complex computations. In addition, they did not consider the unbalanced energy problem among sensors, which results in low network lifetime.

D. GRAPH-BASED APPROACH [10]–[23]

The *graph-based* methods generally transform the given monitoring region into a network graph composed of vertices and edges. The vertices and edges denote the sensors and its communication links, respectively. Then, a centralized graph theory is applied to construct the k-barrier. Kumar *et al.* [10] firstly derived a coverage graph. Based on the coverage graph, they consequently investigated the necessary number for sensors from the probability point of view, such that the probability that any crossing path will be detected by at least k sensors is greater than 0.99. However, given a WSN with a dense deployment of sensors, this scheme does not determine the set of sensors that guarantees the surveillance quality of k-barrier coverage.

Lai *et al.* [11] transformed the given monitoring area into the auxiliary graph. Based on the graph, an algorithm adopting the Depth-First Search (*DFS*) was proposed to search all the possible k-barriers. Wang *et al.* [12] introduced a weighted graph for modeling the problem of barrier construction, and proved that the minimum number of sensors required for constructing the barrier is the shortest path on the weighted graph. Wang *et al.* [13] also adopted the previous algorithm, called Vertex-Disjoint Path, on the weighted graph for exploring all the possible 1-barriers.

In addition, many previous works [14]–[20] adopted the graph theory of Minimum-Cost Flow in order to construct a minimum number of k-barriers. The algorithm *MDP* proposed in [14] finds k starting and end points at left and right boundaries, respectively. For each combinational pair of starting and end points, it constructs the shortest path which is disjoint with the other previously discovered paths. The number of combination pairs will be k^2. It then selects the k disjoint paths from the combination pairs as the solution of k-barrier. This centralized approach, on the downside, consumes remarkable amounts of energy and lacks both flexibility and scalability.

A seeming issue is that the centralized schemes [10], [12], [14], [17]–[20] lack scalability and flexibility. Although some works [11], [13], [15] proposed greedy algorithms to improve the effectiveness, they generally utilized the concept of broadcast operations which lead to high time and message complexities. Moreover, since the computation of intersection sensing area between two sensors is complicated, their construction efficiency is considerably low.

III. NETWORK ENVIRONMENT AND PROBLEM STATEMENT

This section presents the network environment and assumptions of the considered WSNs. Further, the problem statements are also introduced.

A. NETWORK ENVIRONMENT

The monitoring area is considered to be a rectangle region $R = L \times W$, where notations L and W denote the length and width of R, respectively. Let L_{top}, L_{bottom}, W_{left}, and W_{right} denote the top, bottom, left, and right boundaries of R, respectively. A path is said to be a *crossing path* if it completely crosses from L_{bottom} to L_{top}. A set of n sensors $S = \{s_1, s_2, \ldots, s_n\}$ are randomly deployed in R. Each sensor s_i has a unique ID and is aware of its own location and the boundaries of R. Let r and $2r$ denote the sensing radius and communication radius of sensor, respectively. By exchanging the beacon messages with neighboring sensors, each s_i can collect its neighbors' IDs and positions.

B. PROBLEM FORMULATION

This section shows the objective function and several constraints of k-barriers coverage problem. Let m_A denote the number of 1-barriers constructed in R by applying a certain mechanism A. Let E_A denote the remaining energy of sensor that belongs to one of the constructed 1-barriers. Let T_A and e_A denote the network lifetime and energy consumption rate of sensor, respectively. Exp. (1) depicts the objective function of this paper, which aims to efficiently construct the m-barriers while maximizing the network lifetime.

$$\textbf{maximize}\ (T_A), \quad \text{where } T_A = \frac{(E_A \times m_A)}{(e_A \times k)} \tag{1}$$

Whereas, notations $(E_A \times m_A)$ and $(e_A \times k)$ denote the total network lifetime of the m_A constructed 1-barriers and the energy consumption rate of each k-barrier, respectively. Exp. (1) also invokes two objectives: 1) constructing maximal number (m_A) of 1-barriers, and 2) equalizing energy among the constructed barriers to maximize the network lifetime.

Let $DB_h^1 = \left(s_{\hat{1}}, s_{\hat{2}}, \ldots, s_{\hat{f}_h}\right)$ denote the h^{th} constructed 1-barrier composed of $\hat{f}_h$ ordered sensors, where notation $s_{\hat{\imath}}$ denote the i^{th} sensor participating in DB_h^1, when applying the certain mechanism. The definition of *defense k-barrier* is given below.

Definition 1 (Defense k-Barrier; DB^k): A defense k-barrier, denoted by DB^k, is a collection of k disjoint DB^1s. That is,

$$DB^k = \bigcup_{h=1}^{k} DB_h^1. \tag{2}$$

□

The construction of the DB^k can be achieved by discovering k *disjoint* DB^1s. Several constraints that should be satisfied during the construction of DB^k are introduced below;

Let $s_{\hat{\imath}}^{sen}$ denote the sensing range of the i^{th} sensor in DB_h^1. Constraint (3) expresses that there always exists an sensing overlapping area between any two neighboring sensors in DB_h^1.

Continuous Constraint:

$$s_{\hat{\imath}}^{sen} \cap s_{\hat{\imath}+1}^{sen} \neq \emptyset, \quad 1 \leq \hat{\imath} \leq \hat{f}_h - 1,\ \forall s_{\hat{\imath}},\ s_{\hat{f}_h} \in DB_h^1. \tag{3}$$

Let notations $s_{\hat{1}}$ and $s_{\hat{f}_h}$ denote the first sensor and last sensor in DB_h^1, respectively. Let notations $s_{\hat{1}}^{sen}$ and $s_{\hat{f}_h}^{sen}$ denote the sensing ranges of $s_{\hat{1}}$ and $s_{\hat{f}_h}$, respectively. *Boundary constraint* (4) states that the $s_{\hat{1}}^{sen}$ should be overlapped with the W_{left} and the $s_{\hat{f}_h}^{sen}$ should be overlapped with W_{right}.

Boundary Constraint:

$$s_{\hat{1}}^{sen} \cap W_{left} \neq \emptyset \wedge s_{\hat{f}_h}^{sen} \cap W_{right} \neq \emptyset, \quad \forall s_{\hat{1}},\ s_{\hat{f}_h} \in DB_h^1. \tag{4}$$

Let notation $x_{i,h}$ denote the Boolean value indicating whether the s_i participates in DB_h^1. That is,

$$x_{i,h} = \begin{cases} 1, & \text{if } s_i \in DB_h^1 \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Constraint (6) states that the number of sensors contributing to the k-barrier should be greater or equal to $\left\lceil \frac{L}{2r} \right\rceil$.

Number of 1-Barrier Sensors Constraint:

$$\sum_{\forall i \in S} x_{i,h} \geq \left\lceil \frac{L}{2r} \right\rceil, \quad \forall DB_h^1. \tag{6}$$

Constraint (7) expresses that each s_i should participate in at most one 1-barrier. That is, any two constructed 1-barriers should be disjoint.

Sensor-Disjoint Constraint:

$$\sum_{\forall DB_h^1} x_{i,h} \leq 1, \quad \forall i \in S. \tag{7}$$

The reason that the constructed DB^k should satisfy the sensor-disjoint constraint is given as follows. Consider DB^2, which comprises of two DB^1s, says DB_α^1 and DB_β^1. If there exists a common sensor contributing to both the DB_α^1 and DB_β^1, the crossing path might pass through such sensor. This triggers a single sensor that can detect an intruder. Consequently, the DB^2 that does not satisfy the (7) may only contribute 1-barrier coverage.

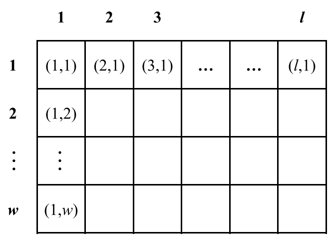


FIGURE 1. Grid-based partition.

Constraint (8) expresses that the number of sensors n should be larger than the total number of sensors contributing to the constructed k-barrier.

Constraint of value n:

$$n \geq \sum_{\forall DB_h^1 \in DB^k} \sum_{\forall i \in S} x_{i,h}. \tag{8}$$

The value of m_A has its lower and upper bounds as shown in (9), where $\lfloor 2r \times n/L \times k \rfloor$ reflects the maximum number of k-barriers that can be constructed.

Constraint of value m_A:

$$1 \leq m_A \leq \left\lfloor \frac{2r \times n}{L \times k} \right\rfloor. \tag{9}$$

For the next section, the proposed *EBCM* is presented which aims to achieve the objective given in (1) while satisfying all the constraints (3), (4), (6), (7), (8), (9).

IV. EFFICIENT *k*-BARRIER CONSTRUCTION MECHANISM

A novel graph model, named *Cover Adjacent Net* (CA-Net), is proposed in this section, aiming to simplify the k-barrier coverage problem while reducing the complexity of computation. Based on the developed CA-Net, the construction of defense k-barrier is further presented.

A. COVER ADJACENT NET (CA-Net)

Initially, the given R is partitioned into a number of equal-size grids, as shown in Fig. 1. Let the size of grid be $l \times l$. Each grid is assigned with a coordinates (x, y). The rules for assigning coordinates are as follows. The most top-left grid in R is assigned with $(1, 1)$. The x-coordinate and y-coordinate are increased by one if the location of a grid shifts one position toward right and down directions, respectively. The grid size is a tradeoff between computation complexity and construction preciseness. A larger grid size can reduce the computation complexity of each sensor but also drop the performance. This occurs because that the grid with big size can reduce the number of sensors that fully cover one grid, which reduce the number of sensors that can participate in the barrier construction task. However, if the grid size is larger than the sensing range of a sensor, none of sensors can cover any grid. In this case, no sensor has contribution to the coverage and hence no barrier can be constructed. Fig. 17 also investigates the impact of the grid size on performance in terms of success ratio for constructing the DB^3 by varying the grid size. A notation table is given below.

TABLE 1. Notation table

Notation	Description
DB^k	Defense k-Barrier.
$g_{x,y}$	A grid with coordinates (x, y).
$d_{x,y}$	The number of different sensors that can fully cover the $g_{x,y}$.
$S_c(g_{x,y})$	The *Cover Set* indicating which set of sensors can fully cover the $g_{x,y}$.
$G_c(\Phi)$	A *Cover Group* $G_c(\Phi)$ of a joinable grid set Φ.
$\lvert S_c(g_{x,y})\rvert$	The numbers of sensors in $S_c(g_{x,y})$.
$\lvert G_c(\Phi)\rvert$	The numbers of sensors in $G_c(\Phi)$.
$\xi = (V, E)$	The CA-Net $\xi = (V, E)$, where V and E denote the vertex set and edge set.
w_i^{vertex}	The *weight value* of a vertex v_i, where $w_i^{vertex} = \lvert G_c(\Phi)\rvert$.
$w_{i,j}^{edge}$	The *weight value* of an edge $e_{i,j} = (v_i, v_j)$, where $w_{i,j}^{edge} = min(w_i^{vertex}, w_j^{vertex})$.
(x_i, y_i)	The coordinates of v_i, where $v_i \in V$.
v_s	The *source vertex* located at (x_s, y_s).
v_d	The *destination vertex* located at (x_d, y_d).
w_p^{path}	The weight value of the path p, where $p = \{e_{s,s+1}, e_{s+1,s+2}, \ldots, e_{d-1,d}\}$ discovered on ξ.
ξ'	The *Decay CA-Net* updated from excluded the edges and vertices of the previous path.

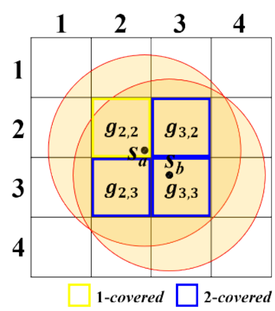


FIGURE 2. Fully covered grids of s_a and s_b.

Let $g_{x,y}$ denote a grid with coordinates (x, y). The *Coverage Degree* of $g_{x,y}$, denoted by $d_{x,y}$, represents the number of different sensors that can fully cover the $g_{x,y}$, where the $g_{x,y}$ can also be said to satisfy $d_{x,y}$-cover. For example, in given Fig. 2, the coverage degrees of $g_{2,2}$, $g_{2,3}$, $g_{3,2}$ and $g_{3,3}$ are 1, 2, 2, and 2, respectively. It also indicates that the grids $g_{2,2}$, $g_{2,3}$, $g_{3,2}$, and $g_{3,3}$ satisfy 1-cover, 2-cover, 2-cover, and 2-cover, respectively.

Let $S_c(g_{x,y})$ denote the *Cover Set* indicating which set of sensors can fully cover the $g_{x,y}$. A set of grids is said to be a *joinable grid set* if they have the same cover set. A *Cover Group* $G_c\ (\Phi)$ of a joinable grid set Φ is a collection

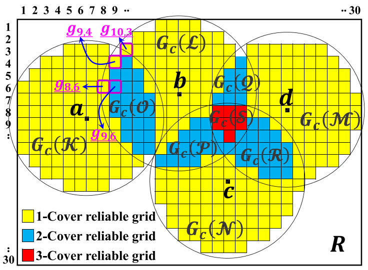


FIGURE 3. An example of $G_c\left(\Phi\right)$.

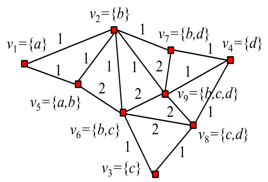


FIGURE 4. The CA-Net transformed from the cover groups depicted in Fig. 3.

of sensors that satisfy

$$G_c\left(\Phi\right) = \left\{i \,|\, s_i \in S_c\left(g_{x,y}\right), g_{x,y} \in \Phi\right\}. \tag{10}$$

Fig. 3 gives an example of $G_c\left(\Phi\right)$. In Fig. 3, there are 4 sensors which are s_a, s_b, s_c and s_d, deployed in R. The grids that are simultaneously and fully covered by sensors s_b, s_c, and s_d can be gathered as a joinable grid set, denoted by $G_c\left(\mathcal{S}\right) = \{b, c, d\}$, where $\mathcal{S}$ denotes the set of grids that is only covered by sensors s_b, s_c, and s_d.

Let $\left|S_c\left(g_{x,y}\right)\right|$ and $\left|G_c\left(\Phi\right)\right|$ denote the numbers of sensors in $S_c\left(g_{x,y}\right)$ and $G_c\left(\Phi\right)$, respectively. The *coverage degree* of $G_c\left(\Phi\right)$ is equivalent to the number of sensors that fully cover all of the grids in Φ. An example of coverage degree of $G_c\left(\Phi\right)$ is given in Fig. 3. In Fig. 3, the number of sensors in cover group $G_c\left(\mathcal{S}\right)$ is 3. Consequently, the coverage degree of $G_c\left(\mathcal{S}\right)$ is 3.

Two cover groups $G_c\left(\mathcal{A}\right)$ and $G_c\left(\mathcal{B}\right)$ are said to be *neighboring* if there exists $g_{x,y} \in \mathcal{A}$ and $g_{\alpha,\beta} \in \mathcal{B}$ having at least one common point or side of grid. For instance, given in Fig. 3, since the grids $g_{9,4} \in G_c\left(\mathcal{K}\right)$ and $g_{10,3} \in G_c\left(\mathcal{L}\right)$ share one common point of grid, the cover groups $G_c\left(\mathcal{K}\right)$ and $G_c\left(\mathcal{L}\right)$ are neighboring. Similarly, $G_c\left(\mathcal{K}\right)$ and $G_c\left(\mathcal{O}\right)$ are also neighboring because there exists a common side of grid between grids $g_{8,6} \in G_c\left(\mathcal{K}\right)$ and $g_{9,6} \in G_c\left(\mathcal{O}\right)$.

Next, The cover groups will be transformed into the CA-Net $\xi = (V, E)$, where V and E denote the vertex set and edge set, respectively. Each vertex v_i in V represents a cover group i. There exists an edge $e = \left(v_i, v_j\right)$ in E between vertices v_i and v_j, $v_i, v_j \in V$, if they are *neighboring*. Let w_i^{vertex} denote the *weight value* of a vertex v_i, where $w_i^{vertex} = \left|G_c\left(\Phi\right)\right|$. The weight value of an edge $e_{i,j} = \left(v_i, v_j\right)$, denoted by $w_{i,j}^{edge}$, is set by $w_{i,j}^{edge} = min\left(w_i^{vertex}, w_j^{vertex}\right)$. Fig. 4 demonstrates the CA-Net that is transformed from the example given in Fig. 3. The cover group $G_c\left(\mathcal{S}\right) = \{b, c, d\}$ determined in Fig. 3 is transformed into vertex $v_9 = \{b, c, d\}$ as shown in Fig. 4. The w_9^{vertex} is 3 since the number of sensors in cover group $G_c\left(\mathcal{S}\right)$ is 3. Since the minimum weight value between w_9^{vertex} and w_4^{vertex} is 1, the $w_{9,4}^{edge}$ is thus also 1.

The coordinates of v_i, denoted by (x_i, y_i) are set by the coordinates of the grid that are fully covered by all sensors in v_i and located closest to the right boundary W_{right}. Once there exists more than one grid that is located closest to W_{right}, the coordinates of the grid that are located closest to the boundary L_{top} becomes the coordinates of v_i. Recall that the notation l denotes the edge length of grid in R. Let v_s be the *source vertex* located at (x_s, y_s) where $x_s = 1$ and $1 \leq y_s \leq \lceil W/l \rceil$. Similarly, let v_d be the *destination vertex* located at (x_d, y_d) where $x_d = \lceil L/l \rceil$ and $1 \leq y_d \leq \lceil W/l \rceil$. Given a CA-Net ξ. Let notation $p = \left\{e_{s,s+1}, e_{s+1,s+2}, \ldots, e_{d-1,d}\right\}$ be a path discovered on ξ. The weight value of the path p, denoted by w_p^{path}, is expressed by

$$w_p^{path} = min\left(w_{s,s+1}^{edge}, w_{s+1,s+2}^{edge}, \ldots, w_{d-1,d}^{edge}\right). \tag{11}$$

The path p discovered on ξ indicates that the corresponding coverage area satisfies w_p^{path}-cover, where the path p is essentially a defense w_p^{path}-barrier. For instance, consider the example given in Fig. 4. There exists a path $p = \left\{e_{1,5}, e_{5,6}, e_{6,9}, e_{9,8}\right\}$. This path precisely represents the defense 1-barrier.

Theorem 1: The path p with weight value w_p^{path} discovered on *CA-Net* corresponds to the defense w_p^{path}-barrier in R.

Proof: Recall that an effective barrier should satisfy both the *Continuous constraint* (3) and *Boundary constraint* (4). Since the construction of path p starts from *source vertex* and ends at *destination vertex* on CA-Net, the (4) is held. Meanwhile, each edge, says $e_{i,j} = \left(v_i, v_j\right)$, in path p consists of two vertices, v_i and v_j, which are transformed from the *neighboring* cover groups i and j in R. There always exists a sensing range intersection between two sensors $s_a \in i$ and $s_b \in j$ if i and j are *neighboring* cover groups. Hence, the constraint (3) is also satisfied. Based on constraints (3) and (4), the path p discovered on CA-Net corresponds to an effective barrier in R. Furthermore, the path p with weight value w_p^{path} represents that all the edges belonged to the path p satisfy w_p^{path}-cover. Therefore, the path p with w_p^{path} discovered on CA-Net corresponds to a defense w_p^{path}-barrier in R. □

After accomplishing the path p, the edges and vertices that belonged to p should be excluded from original sets V and E, respectively, of CA-Net $\xi = (V, E)$, for satisfying the *Sensor-disjoint constraint* (7). The next path will be constructed by repeatedly executing the same processes mentioned above based on the *Decay CA-Net*, denoted by *DCA-Net*, which is updated from excluded the edges and vertices of the previous path.

Theorem 2: Let ξ and ξ' be a *CA-Net* and the *DCA-Net* of ξ, respectively. The sets of sensors, which construct the barrier discovered in the ξ and ξ' are disjoint. □

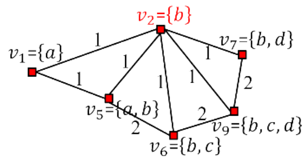


FIGURE 5. The partial CA-Net stored in the cache of s_b.

We emphasize that the example shown in Fig. 4 gives a global view of the CA-Net, aiming to help understand the proposed mechanism from the conceptual point of view. However, the proposed mechanism is a decentralized mechanism. That is, each sensor only needs to stores a partial CA-Net in its cache. Fig. 5 gives an example of the partial CA-Net which is actually stored in the cache of sensor s_b.

By locally exchanging with neighboring sensors the partial CA-Net, the defense k-barrier can be accomplished in decentralize method.

B. CONSTRUCTION OF DEFENSE k-BARRIER

The construction of the defense k-barrier in R can be simplified by exploring the path p with weight value $w_p^{path} = k$ on CA-Net. Let term k-path denote the path with a weight value k. To explore the k-path on CA-Net, in each vertex, a sensor with the highest residual energy will be selected to become a *Decision Maker* (*DM*). The *DM* is responsible for executing the operations of the developed *EBCM*. Once there exists more than one sensor that has the same highest residual energy, the sensor with the largest ID will become the *DM*. Two approaches for exploring the k-path based on CA-Net are given below, which are *Best-fit Coverage Approach* (*BCA*) and *Top-down Single Barrier Approach* (*TOBA*).

1) BEST-FIT COVERAGE APPROACH (BCA)

This section introduces the *Best-fit Coverage Approach* (*BCA*) for exploring the best-fit k-path on CA-Net. Let v_i^{pre}, v_i^{curr}, and v_i^{next} denote the initial, current, and forthcoming vertices considered v_i, respectively. To construct the k-path on CA-Net, the *source vertex* v_s with $w_s^{vertex} \geq k$ and is firstly treated as v_s^{curr}. Let $N(v_i^{curr})$ denote the *Candidate Set* of v_i^{curr}, which satisfies

$$N\left(v_i^{curr}\right) = \left\{v_j \mid \exists \left(v_i^{curr}, v_j\right) \in E, w_j^{vertex} \geq k\right\}. \quad (12)$$

Let $l_{i,j}^{max}$ denote the maximum defense length contributed by vertices v_i and v_j. The *DM* of v_s^{curr} will select an appropriate v_j from $N(v_s^{curr})$ to become the v_j^{next} if the $l_{s,j}^{max}$ has a maximal value. The value of $l_{i,j}^{max}$ can be evaluated by

$$l_{i,j}^{max} = \left(y_j - y_i\right) + \left(x_j - x_i\right) \cdot Max(W, L) \quad (13)$$

where the multiplication of $Max(W, L)$ and $(x_j - x_i)$ emphasize the defense length that increases in the x-axis direction because a straight barrier is the most appropriate when constructing the k-path from boundaries W_{left} to W_{right}.

Meanwhile, the edge (v_s^{curr}, v_j^{next}) has been added in the k-path. After accomplishing the first edge, the *DM* of v_s^{curr} will transfer the authority of next edge construction to the *DM* of v_j^{next}. The v_s^{curr} and v_j^{next} will become the v_s^{pre} and v_j^{curr}, respectively. The *DM* of v_j^{curr} will execute the same operations as those done by *DM* of v_s^{pre}. The edge construction will be executed repeatedly until a *destination vertex* is selected to be a part of k-path, which indicates that the k-path has been completed. Since the selection of each edge considers the corresponding length of each edge, the k-path is likely to be constructed with a minimum number of edges. That is, the number of sensors is minimized for constructing each DB^k and hence the number of DB^k that are constructed is ideally maximized.

Let notations DM_i^{pre}, DM_i^{curr}, and DM_i^{next} denote the initial, current, and forthcoming *DMs* of vertex v_i, respectively. The current *DM*, says DM_i^{curr}, will find the next vertex v^{next} to construct a barrier. However, the *Candidate Set* $N\left(v_i^{curr}\right)$ might be empty, which causes that the DM_i^{curr} to fail in finding the next *DM*. In this case, the authority of next edge construction will be returned from the current *DM* to the previous (initial) *DM*, says DM_i^{pre}. Let *Failure Candidate Set*, denoted by $F(v_i^{curr})$, be the set of all failure vertices of v_i^{curr}. That is,

$$F\left(v_i^{curr}\right) = \left\{v_j \mid \exists \left(v_i^{curr}, v_j\right) \in E, w_j^{vertex} \geq k, N\left(v_j^{curr}\right) = \emptyset\right\}. \quad (14)$$

As soon as the DM_i^{pre} obtains the construction authority from the DM_i^{curr}, it plays the role DM_i^{curr} and tries to find another v^{next} from the *Renewed Candidate Set*, denoted by $N'(v_i^{curr})$. That is,

$$N'\left(v_i^{curr}\right) = N\left(v_i^{curr}\right) - F\left(v_i^{curr}\right). \quad (15)$$

With this backtracking policy, the failed DM_i^{curr} may continue the process of constructing the k-path along other unvisited edges. The success probability for constructing the k-path on CA-Net may also be increased.

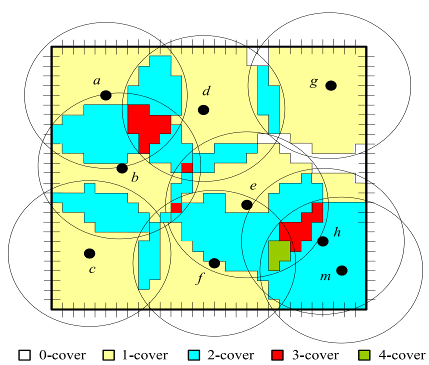


FIGURE 6. The construction of defense 2-barrier in the monitoring area R.

Fig. 6 gives an example of the construction of defense 2-barrier by applying the proposed *BCA*. As shown in Fig. 6, a set of sensors $\{s_a, s_b, s_c, s_d, s_e, s_f, s_g, s_h, s_m\}$ are deployed in the monitoring region. The intersection areas among sensing ranges of sensors are represented with 5 color areas,

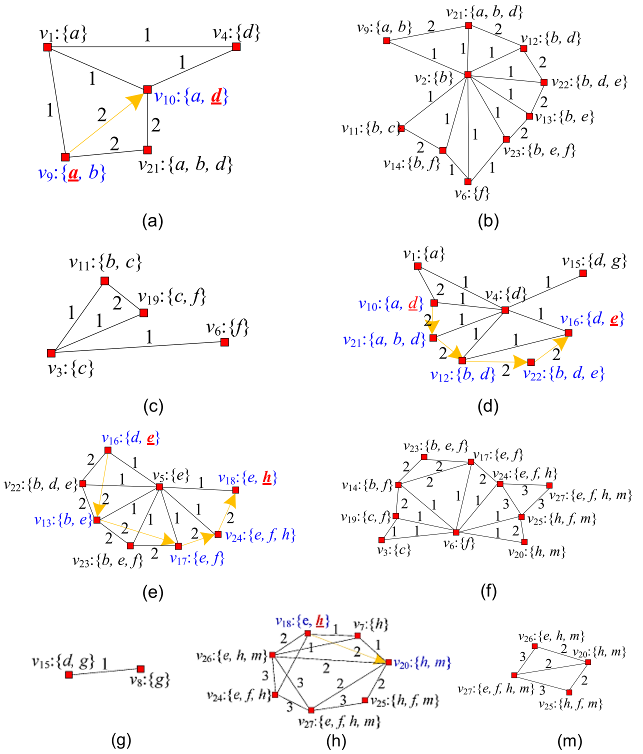


FIGURE 7. The partial CA-Net stored in cache of each sensor in Fig. 6. (a). Sensor s_a. (b). Sensor s_b. (c). Sensor s_c. (d). Sensor s_d. (e). Sensor s_e. (f). Sensor s_f. (g). Sensor s_g. (h). Sensor s_h. (m). Sensor s_m.

respectively representing the five different *coverage degrees*. For instance, the white, yellow, blue, red, and green areas represent the 0-cover, 1-cover, 2-cover, 3-cover, and 4-cover types of *cover groups*, respectively. The monitoring region shown in Fig. 6 will be transformed into the CA-Net. Fig. 7 shows the partial CA-Net stored in cache of each sensor. Since the proposed mechanism is a decentralized method, each sensor only maintains partial CA-Net.

Next, we will present how to construct the 2-path using CA-Net. Consider Fig. 7(a) which is the partial CA-Net maintained by sensor s_a. The *source vertex* v_9 will be treated as a *currently considered vertex* due to its *weight value*, $w_9^{vertex} = 2$, equals to the requested coverage degree. That is, the *currently considered DM* is $DM^{curr} = s_a$. The DM^{curr} will select the *next considered vertex* from its *Candidate Set* $N(v_9) = \{v_{10}, v_{21}\}$. Since the value of *maximum defense length* $l_{9,10}^{max}$ is larger than $l_{9,21}^{max}$, v_{10} will play the role of *next considered vertex* by applying Exp. (16). Therefore, an edge (v_9, v_{10}) will be added into the 2-path. After that the first edge is accomplished, the $DM^{curr} = s_a$ transfers the authority of the next edge construction to the *next considered DM* in v_{10}. That is, $DM^{next} = s_d$. Sensors a and d will then become DM^{pre} and DM^{curr}, respectively. To construct the next edge connected with the previous edge (v_9, v_{10}), as shown in Fig. 7(d), the $DM^{curr} = s_d$ will execute the same operations as those done by $DM^{pre} = s_a$. Therefore, the edges (v_{10}, v_{21}), (v_{21}, v_{12}), (v_{12}, v_{22}), and (v_{22}, v_{16}) will be

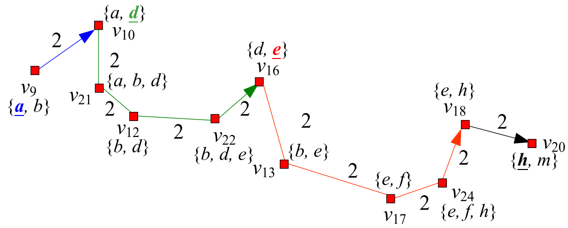


FIGURE 8. The result of 2-path construction by applying *BCA*.

successively constructed. After that, as shown in Figs. 7(e) and 7(h), the sensors s_e and s_h will become the consecutive decision maker. The edges (v_{16}, v_{13}), (v_{13}, v_{17}), (v_{17}, v_{24}), (v_{24}, v_{18}), and (v_{18}, v_{20}) will also be constructed accordingly. The construction of 2-path is accomplished when the *destination vertex* v_{20}, or edge (v_{18}, v_{20}) is selected to be a part of 2-path. Fig. 18 shows the result of 2-path construction by applying the proposed *BCA* mechanism.

The major advantage of *BCA* is simple and easy to implement. However, The *BCA* may to encounter possibilities of failure when constructing the k-path because the *BCA* only finds the *vertex* that satisfies the required *weight value* k. This indicates that all sensors in the vertex associated with a weight value smaller than k cannot be considered to participate in the k-path. To fully utilize all of the sensors and explore more potential k-paths, a *Top-down One-coverage Barrier Approach* (*TOBA*) is further proposed in the next section.

2) TOP-DOWN ONE-COVERAGE BARRIER APPROACH (TOBA)

The major idea of *Top-down One-coverage Barrier Approach* (*TOBA*) is to give more opportunities for those *vertices* that have smaller weight value than the required *weight value* k. According to Exp. (2), it is known that k-path can be decomposed into k disjoint 1-paths. That is, each sensor can be included in the k-path even if it falls in a vertex with a *weight value* $k = 1$.

The *BCA* and *TOBA* have the similar operations for constructing the k-path on CA-Net. The main difference between them is that *TOBA* only constructs one 1-path on CA-Net each time. Each time, the CA-Net is updated by removing those sensors that have joined the constructed 1-path in order to satisfy the *Sensor-disjoint constraint* as shown in constraint (7). The *TOBA* will repeat the same operation from L_{top} to L_{bottom} until there is no 1-path existed in CA-Net. As a result, the *TOBA* can find a larger number of 1-paths. Finally, these constructed 1-paths can be integrated to become multiple k-paths. Since the *TOBA* fully utilizes each sensor, the number of constructed k-paths by applying the *TOBA* is expected to be larger than that obtained by applying *BCA*.

Fig. 9 shows the result where three 1-paths are constructed by applying the *TOBA*. As shown in Fig. 9, the *source vertex* v_1 will be treated as a *currently considered vertex* since its weight value $w_1^{vertex} \geq 1$. Besides, vertex v_1 is closer to L_{top} than the other *source vertices* $v_2 = \{b\}$, $v_3 = \{c\}$,

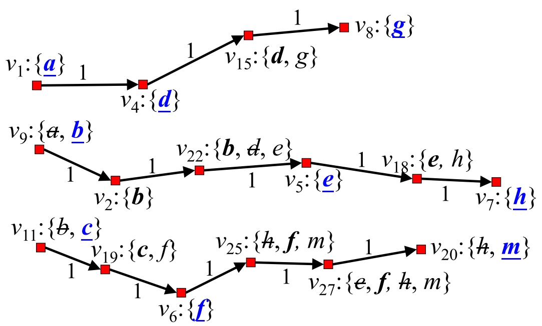


FIGURE 9. The result of 2-path construction by applying *TOBA*.

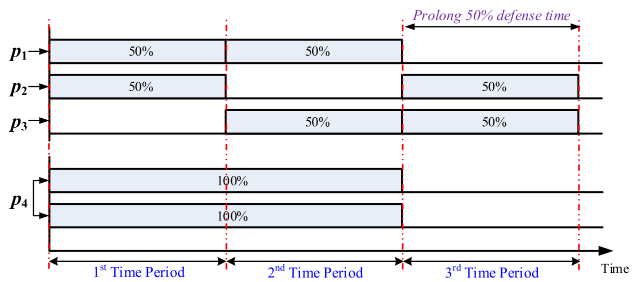


FIGURE 10. An example to describe the basic concept of *BES*.

$v_9 = \{a, b\}$, and $v_9 = \{b, c\}$. Similar to the execution process of *BCA*, the authority of edge construction will be transferred from sensors s_a to s_b and then s_d to s_h. As a result, these constructed edges (v_1, v_4), (v_4, v_{15}), and (v_{15}, v_8) will form a 1-path. After that, each sensor will update its partial CA-Net by removing the sensors that have joined the constructed 1-path. For example, as shown in Fig. 9, vertices $v_9 = \{a, b\}$, $v_{22} = \{b, d, e\}$, and $v_{18} = \{e, h\}$ will be changed to $v_9 = \{b\}$, $v_{22} = \{b, e\}$, and $v_{18} = \{e\}$, respectively, after the 1-path $\{v_1 - v_4 - v_{15} - v_8\}$ is formed. Furthermore, following the similar operations mentioned above, the 1-paths $\{v_9 - v_2 - v_{22} - v_5 - v_{18} - v_7\}$ and $\{v_{11} - v_{19} - v_6 - v_{25} - v_{27} - v_{20}\}$ can be successively constructed from L_{top} to L_{bottom}.

In comparison, giving the same example as shown in Fig. 6, the *TOBA* can construct three 1-paths while the *BCA* only constructs one 2-path (or two 1-paths). The three constructed 1-paths can work in turn for supporting 2-barrier coverage. In the next section, the *Energy Balanced Scheduling* (*EBS*) is proposed. By applying the *EBS*, all the constructed barriers can be scheduled and can work for supporting k-barrier coverage. The scheduled barriers can achieve the purpose of energy balance and hence maximize the network lifetime.

C. BARRIER ENERGY SCHEDULING PHASE

This section introduces the proposed *BES*, aiming to maximize the total lifetime for k-barrier coverage. Let notation m denote the total number of constructed 1-path in a given R, where $k \leq m$. Let notation x denote the maximal number of k-path in R. That is,

$$x = \left\lfloor \frac{m}{k} \right\rfloor. \tag{16}$$

If the value of m is equal to xk, then any k of m constructed 1-paths can be grouped for executing one-round of k-barrier coverage. In this case, the total lifetime duration can be counted by the time of executing x rounds of k-barrier coverage. On the other hand, let notation d denote the number of the remaining 1-path, where d is a value of $(m \bmod k)$. The $(k + d)$ of the m constructed 1-path will be scheduled by applying the *BES*, aiming to prolong the total lifetime of k-barrier coverage.

Fig. 10 gives an example to describe the basic concept of the proposed *BES*. As shown in Fig. 10, suppose that the required monitoring quality of the given monitoring region is 2-barrier coverage. Consider two solutions for this requirement: 1) Only one 2-path p_4 can be constructed in the region. 2) Three 1-path $\{p_1, p_2, p_3\}$ can be constructed in the region. Both the solutions can support the requirement of 2-barrier coverage. However, the second solution is better than the first in terms of lifetime guarantee. The reason is given below. In the second solution, the *BES* arranges those sensors that construct two 1-path $\{p_1, p_2\}$ to work simultaneously for supporting 2-barrier coverage in the first time period. The sensors that construct the 1-path p_3 will be arranged to sleep for saving energy. In the second time period, the $\{p_1, p_3\}$ will take over the 2-covered sensing task of $\{p_1, p_2\}$ and the sensors that construct p_2 will be changed to the energy-saving mode. After the end of the second period, the p_1 exhausts its energy and hence cannot work in the third period. Finally, the sensors that construct $\{p_2, p_3\}$ will be arranged to execute the 2-covered sensing task in the third period. Therefore, in the second solution, the three 1-path can support 2-barrier coverage for 3 time periods.

Compared to the second solution, the first solution constructs p_1 as a 2-path. It needs to continuously execute the sensing task from beginning to the end. As a result, the 2-path can support 2-barrier coverage for only 2 time periods. The following formally details the proposed *BES*.

In the *BES*, one of the $(x - 1)$ k-path will be arranged to work for supporting k-barrier coverage. When all sensors belonging to the k-path exhaust their energies, the other one of the $(x - 1)$ k-path will be arranged to play a role of successor for continually supporting k-barrier coverage. This process will be executed until all sensors of the $(x - 1)$ k-path exhaust their energies. After that, the remaining $(k+d)$ 1-path will start to work. The lifetime of each sensor of the $(k + d)$ 1-paths will be partitioned into several time periods. These time periods will be scheduled by using the *BES* in order to prolong the network lifetime while satisfying the requirement of k-barrier coverage.

Let $C(k+d, k)$ denote the combination number of picking k 1-path from $(k + d)$ 1-path for supporting k-barrier coverage. Let notation w denote a duty ratio of energy consumption of each sensor for each time period. The value of w should satisfy

$$w = \frac{k + d}{k \times C(k + d, k)}. \tag{17}$$

Let notations E and e_{cons} denote the remaining energy of sensor and the energy consumption rate of sensor per unit of time, respectively. Let notation t denote the maximal sensing time of sensor for executing the sensing task. That is,

$$t = \frac{E}{e_{cons}}. \tag{18}$$

Each sensor will support the time $(w \times t)$ in each time period for executing k-barrier coverage. That is, in each time period, k of the $(k+d)$ 1-path will be arranged to work for time $(w \times t)$ in turn until all the $(k + d)$ 1-path are energy exhausted. The following analyzes the lifetime improvement of *BES*.

Let notations D and T denote the total lifetime of m 1-path with and without applying the *BES*, respectively. The values of D and T can be estimated using

$$D = \left(x + \frac{k+d}{k}\right) t \text{ and } T = xt. \tag{19}$$

The following equation shows the lifetime improvement of *BES*:

$$\frac{D}{T} = \frac{\left(x + \frac{k+d}{k}\right) t}{xt} = \frac{x + (1 + d)}{x}. \tag{20}$$

V. SIMULATION

Recall that the proposed mechanism *EBCM*, including *BCA*, *TOBA*, and *BES* approaches, has been presented in Section 4. Let terms *BCA* and *BCA*(*BT*) denote the proposed *BCA* working with and without the backtracking policy, respectively. Let terms *TOBA* and *TOBA*(*BT*) denote the *TOBA* working with and without the backtracking policy, respectively. Let the term *TOBA*(*BT*+*BES*) denote that the *TOBA* adopts both the backtracking policy and *BES* approach. This section compares the performance of the proposed *BCA*, *BCA(BT)*, *TOBA*, *TOBA(BT)*, and *TOBA*(*BT*+*BES*) against the existing Maximum Disjoint Paths (*MDP*) algorithm. The *MDP* is a centralized algorithm [14] and can be treated as the optimal solution. The following shows the simulation model and then discusses the simulation results.

TABLE 2. Simulation parameters.

Monitoring area	$400m \times 400m$
Number of Sensors	$300 - 600$
Sensing Radius	$15m$–$30m$
Communication Radius	Twice of sensing radius
Deployment	Randomly
Grid Size	$1m$, $2m$, $4m$, $8m$

A. SIMULATION MODEL

The simulation parameters are given in TABLE 2, where all parameters refer to the typical Berkeley motes. In TABLE 2, a set of sensors ranging from 300 to 600 is randomly deployed in $400m \times 400m$ monitoring area. The communication radius is twice of the sensing radius ranging from $15m$ to $30m$. The grid size is set by $1m$, $2m$, $4m$, and $8m$.

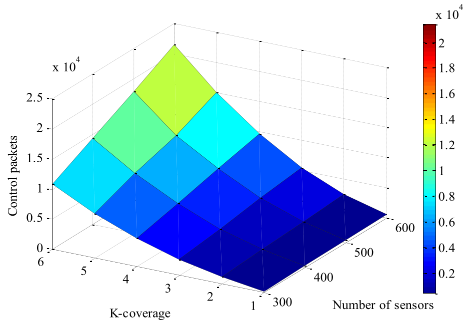


FIGURE 11. The control overhead of *MDP*.

B. SIMULATION RESULTS

The performance study investigates the *MDP* for both the complexity of computing and control overheads. First, the *MDP* individually searches for k starting points and k end points at the left boundary and right boundary, respectively. For each combinational pair of starting and end points, *MDP* explores the shortest path disjointing with previous explored paths. The number of combination pairs will be k^2. Then the *MDP* picks the explored k disjoint paths as the solution of DB^k. The time complexity is $O(k^2|V|)$, where notation $|V|$ denote the number of sensors deployed in given network. Beside the complexity of computing, the control overheads of *MDP* is investigated in Fig. 11. As shown in Fig.11, the control overhead of *MDP* is exponentially increased with value k. However, the control overhead of the *EBCM* is only $O(|V|)$ in worst cases, as compared with *MDP*. Therefore, the proposed mechanisms, including the *BCA*, *TOBA*, and *TOBA* with *BES*, have a better performance than *MDP* for both of the complexity of computing and communication overheads.

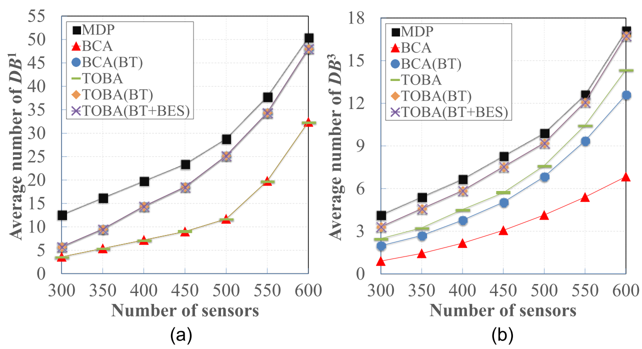


FIGURE 12. Performance comparisons of *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), *TOBA*(*BT*+*BES*), and *MDP* for the average number of DB^k by changing the number of sensors. (a) 1-Barrier coverage. (b) 3-Barrier coverage.

Fig. 12 investigates the average number of DB^k that are constructed by applying the *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), *TOBA*(*BT*+*BES*), and *MDP*. In Fig. 12(a), the optimal solution *MDP* outperforms the other comparisons for the average number of DB^1. This occurs since the *MDP* evaluates all of the possible solutions of DB^1, and then picks the one that is composed of the smallest number of sensors each time. However, as observed in Fig. 11, the *MDP* creates a considerable number of control overheads than the

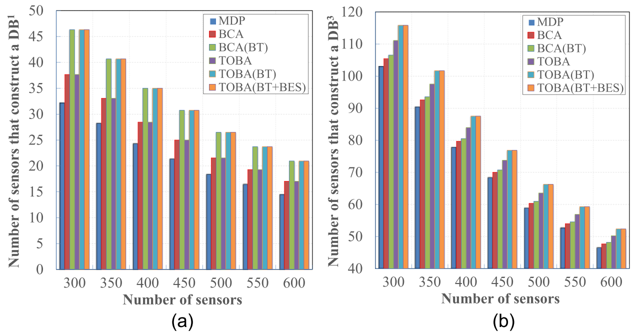


FIGURE 13. Performance comparison of ***BCA***, ***BCA(BT)***, ***TOBA***, ***TOBA(BT)***, ***TOBA(BT+BES)***, and ***MDP*** in terms of the number of sensors that constructs DB^k by varying the number of deployed sensors. (a) 1-Barrier coverage. (b) 3-Barrier coverage.

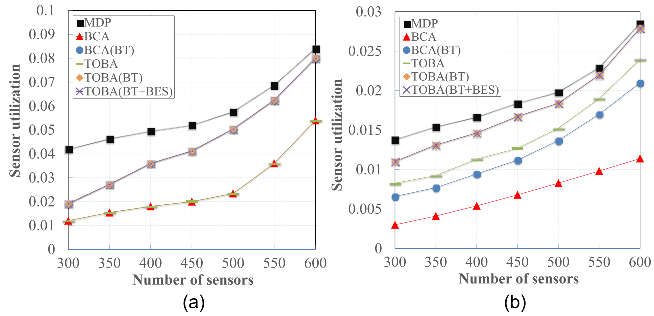


FIGURE 14. Performance comparison of ***BCA***, ***BCA(BT)***, ***TOBA***, ***TOBA(BT)***, ***TOBA(BT+BES)***, and ***MDP*** in terms of the sensor utilization by varying the number of deployed sensors. (a) 1-Barrier coverage. (b) 3-Barrier coverage.

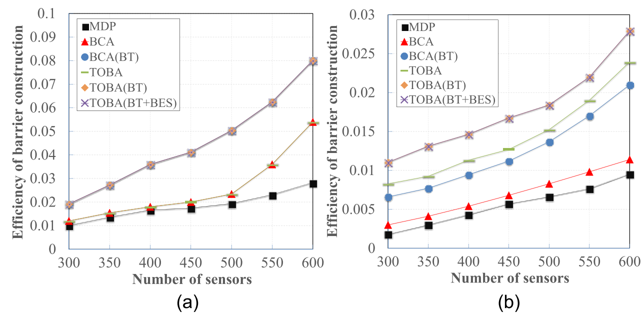


FIGURE 15. Performance comparison of ***BCA***, ***BCA(BT)***, ***TOBA***, ***TOBA(BT)***, ***TOBA(BT+BES)***, and ***MDP*** in terms of the efficiency of barrier construction by varying the number of deployed sensors. (a) 1-Barrier coverage. (b) 3-Barrier coverage.

proposed approaches. Fig. 12(a) also reveals that the performances of *BCA*(*BT*), *TOBA*(*BT*) and *TOBA*(*BT+BES*) are rather similar. In addition, three algorithms outperform the *BCA* and *TOBA* for every number of constructed DB^1. This occurs because the *BCA*(*BT*), *TOBA*(*BT*) and *TOBA*(*BT+BES*) can still adopt the detour policy when encountering a dead-end problem.

Fig. 12(b) further looks into the average number of DB^3 constructed by implementing the *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), *TOBA*(*BT+BES*), and *MDP*. The performances of *TOBA*(*BT*) and *TOBA*(*BT+BES*) are close to the optimal solution *MDP* and outperform the *BCA*, *BCA*(*BT*), and *TOBA*, especially when the sensor number is larger than 500. This occurs because both *TOBA*(*BT*) and *TOBA*(*BT+BES*) not only adopt the backtracking policy, but also visit more *vertices* with $w_1^{vertex} \geq 1$, when encountering the dead-end problem. Moreover, the results from Figs. 12(a) and 12(b) also indicate that the *TOBA*, *TOBA*(*BT*), and *TOBA*(*BT+BES*) are more effective when the value of coverage degree k is higher.

Fig. 13 studies the number of sensors required for constructing a DB^k by changing the number of sensors. As shown in Fig. 13(a), the *BCA*(*BT*), *TOBA*(*BT*), and *TOBA*(*BT+BES*) require more sensors than the other comparisons for constructing a DB^1. This occurs because that they adopt the detour policy when encountering the dead-end problem. However, as shown in Fig. 12(a), though adopting the detour policy can increase the number of sensors, it can construct more DB^1. Fig. 13(b) further looks into the number of sensors required for constructing a DB^3. Both the *TOBA*(*BT*) and *TOBA*(*BT+BES*) require more sensors than the other comparisons. As observed in Fig. 12(b), the number of DB^3 constructed by applying the two algorithms approximate the optimal solution *MDP* and the increased number of sensors is conversely decreased with the deployed number of sensors.

Let notations N_{barr} and N_{sens} denote the constructed number of DB^k and the deployed number of sensors, respectively. The sensor utilization, denoted by notation u, can be estimated using

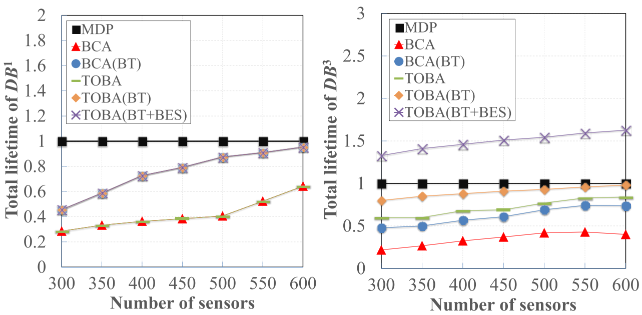
$$u = \frac{N_{barr}}{N_{sens}}. \tag{21}$$

Fig. 14 investigates the sensor utilization by varying the number of deployed sensors. As shown in Fig. 14(a), the approaches with backtracking strategy, including the *BCA*(*BT*), *TOBA*(*BT*), and *TOBA*(*BT+BES*), outperform both the *BCA* and *TOBA* when defense requirement is set at 1-barrier coverage. Granted, using the backtracking strategy may increase the average number of sensors needed for constructing the DB^1. It can explore more opportunities of DB^1 construction, therefore increasing the number of DB^1. Fig. 14(b) further investigates the sensor utilization when defense requirement is set by 3-barrier coverage. As a comparison with Fig. 14(a), the performance of *BCA*(*BT*) is no longer equivalent to those of *TOBA*(*BT*) and *TOBA*(*BT+BES*). This occurs because both the *TOBA*(*BT*) and *TOBA*(*BT+BES*) can not only adopt the detour policy when encountering the dead-end problem, but also construct extra DB^3 by exploring more DB^1.

Let N_{cont} denote the number of control packets. The efficiency of barrier construction, denoted by notation e, can be estimated using

$$e = \frac{N_{barr}}{N_{cont}}. \tag{22}$$

Fig. 15 investigates the efficiency of barrier construction. As shown in Fig. 15(a), all the proposed *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), and *TOBA*(*BT+BES*) outperform the optimal solution *MDP* when the defense requirement is 1-barrier coverage. This occurs because the *MDP* is a centralized algorithm and it has to collect the information and then broadcast the results between sink and all deployed sensors. On the contrary, the primary control overheads of the proposed approaches only occur between *Decision Maker* and its neighbors. Similar to Fig. 15(a), Fig. 15(b) further shows that the efficiencies of the proposed approaches are much better than the *MDP* especially in terms of the construction of 3-barrier coverage. This occurs since the proposed approaches have a better effectiveness when the value of coverage degree k is higher, as shown in Fig. 12(b).



(a) (b)

FIGURE 16. **Performance comparison of *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), *TOBA*(*BT*+*BES*), and *MDP* in terms of total lifetime by varying the number of sensors. (a) 1-Barrier coverage. (b) 3-Barrier coverage.**

Fig. 16 investigates the total lifetime of barriers by varying the number of sensors. The optimal solution *MDP* is treated as a baseline and its lifetime is considered as value 1 for clearly measuring the percentages of lifetime improvements of the proposed approaches. In Fig. 16(a), the performances of the proposed approaches are close to the *MDP* when the number of deployed sensors is increased. This occurs because the phenomenon of dead end occurrences can be reduced when the number of sensors increases. On the other hand, Fig. 16(b) shows that the proposed *TOBA*(*BT*+*BES*) outperforms the other comparisons, including the *MDP*, in terms of total lifetime of DB^3. Herein, we notice that the *MDP* is a centralized algorithm and is considered as the optimal mechanism since it can construct the largest number of DB^1. However, the *TOBA*(*BT*+*BES*) can fully utilize all the constructed DB^1. For example, the *MDP* can construct 8 DB^1 but only 6 DB^1 can be utilized for constructing $2DB^3$. On the contrary, the proposed *TOBA*(*BT*+*BES*) totally utilize 10 DB^1 to dynamically organize the DB^3. That's the primary reason the proposed *TOBA*(*BT*+*BES*) outperforms *MDP* in terms of the lifetime of DB^3.

Fig. 17 investigates the success ratio for constructing the DB^3 by varying the grid size, ranging from 1m to 8m. In general, the success ratio for constructing the DB^3 is decreased with the size of grid. A large sized grid may reduce the number of grid that can be fully covered by each sensor. For instance, as shown in Fig. 2, the grid $g_{3,3}$ cannot be fully covered by s_a if its size is increased. As a result, the number of fully covered grids when constructing DB^3 would be reduced and hence the success ratio for the construction of DB^3 is reduced accordingly. As shown in Fig. 16, although using the grid-based method may reduce the success ratio for constructing the barrier, the success ratio still keeps a very high value. Besides, the control overheads of the proposed *EBCM* is only $O(|V|)$ in the worst case, as compared with *MDP* observed in Fig. 11. The performances of the proposed *TOBA*(*BT*) and *TOBA*(*BT*+*BES*) are very close to the optimal solution *MDP*, especially when the grid size is set by 1m.

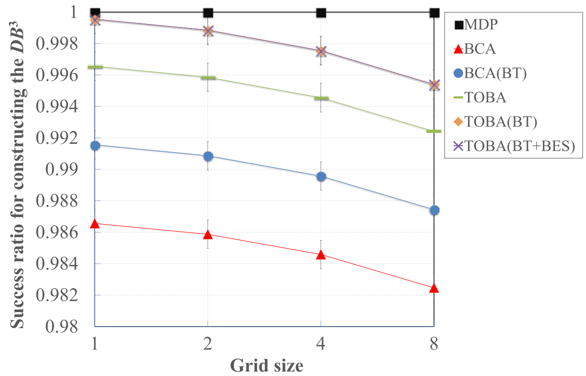


FIGURE 17. **Performance comparison of *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), *TOBA*(*BT*+*BES*), and *MDP* in terms of success ratio for constructing the DB^3 by varying the grid size, including 1m, 2m, 4m, and 8m.**

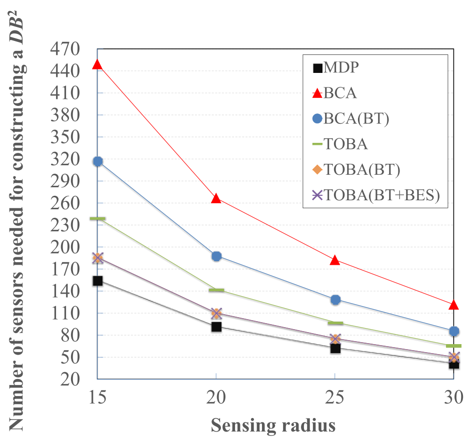


FIGURE 18. **Performance comparison of *BCA*, *BCA*(*BT*), *TOBA*, *TOBA*(*BT*), *TOBA*(*BT*+*BES*), and *MDP* for the number of sensors required for constructing a DB^2 when the sensing radius changes.**

Fig. 18 compares the number of sensors required for constructing a DB^2 when the sensing radius is changed, ranging from 15m to 30m. As shown in Fig. 18, the curves of six compared algorithms have a similar performance trend. That is, the number of sensors needed for constructing a DB^2 is decreased with the sensing radius. This occurs since the number of fully covered grids is increased with the sensing radius of sensor. Thus, constructing a DB^2 requires smaller number of sensors when each sensor has a larger sensing radius. Fig. 18 also shows that the performances of *TOBA*(*BT*) and *TOBA*(*BT*+*BES*) are close to the optimal solution (*MDP*) and outperform the others in terms of the number of sensors required for constructing a DB^2.

VI. CONCLUSION

The barrier coverage problem addresses the issue of how to construct a defense barrier by organizing a set of sensors for detecting the intruders. Finding a minimum set of sensors for constructing the DB^k can extend the network lifetime because the maximal number of sensors can stay in sleep state, saving energy consumption. This paper has presented a novel graph model (CA-Net) that aims to simplify the problem of k-barrier coverage while reducing the complexity of computation. Based on the developed CA-Net, two decentralized approaches, called *BCA* and *TOBA*, are presented to address the k-barrier coverage problem. The *BCA* is simple and easy to implement for further exploring the best-fit DB^k on CA-Net. The *TOBA* can fully utilize all of the sensors and perform supplementary studies on DB^1. In addition, a backtracking (*BT*) policy is proposed to further improve the performances of both the *BCA* and *TOBA* when encountering the dead-end problem. The proposed *BES* scheduling approach aims toward full utilization of all the constructed DB^1.

As a result, the purposes of energy-balance and maximal lifetime can be achieved. The simulation study investigates the performances of the proposed approaches against the optimal solution *MDP* in terms of control overhead, number of DB^k constructed, number of sensors required, sensor utilization, efficiency of barrier construction as well as network lifetime. The simulation results show that the proposed decentralized *TOBA* approach with *BT* and *BES* policies have similar performances, as compared with the centralized *MDP* algorithm. In the future work, some sensors with mobility will be taken into consideration. The mobile sensors can move to proper position for participating in the construction of defense barrier. Thus, the number of constructed defense barriers can be significantly increased. However, the mobility of sensor also raises the problem of energy consumptions. Thus, how to efficiently take advantage of mobility to construct as many barriers as possible will be an important issue in our future work.

REFERENCES

[1] C.-F. Huang and Y.-C. Tseng, "The coverage problem in a wireless sensor network," in *Proc. WSNA*, 2003, pp. 115–121.

[2] G. Yang and D. Qiao, "Barrier information coverage with wireless sensors," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 918–926.

[3] G. Yang and D. Qiao, "Multi-round sensor deployment for guaranteed barrier coverage," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[4] Y.-T. Lin, K. K. Saluja, and P. Ramanathan, "Connected barrier coverage on a narrow band: Analysis and deployment," in *Proc. IEEE SECON*, Jun. 2010, pp. 1–9.

[5] M. Eftekhari, L. Narayanan, and J. Opatrny, "On multi-round sensor deployment for barrier coverage," in *Proc. IEEE MASS*, Oct. 2013, pp. 310–318.

[6] C.-Y. Chang, L.-L. Hung, Y.-C. Chen, and M.-H. Li, "On-supporting energy balanced k-barrier coverage in wireless sensor networks," in *Proc. ACM IWCMC*, 2009, pp. 274–278.

[7] H. Fan, M. Li, X. Sun, P.-J. Wan, and Y. Zhao, "Barrier coverage by sensors with adjustable ranges," *ACM Trans. Sens. Netw.*, vol. 11, no. 1, p. 14, Nov. 2014.

[8] B. Liu, O. Dousse, J. Wang, and A. Saipulla, "Strong barrier coverage of wireless sensor networks," in *Proc. ACM MobiHoc*, 2008, pp. 411–420.

[9] A. Chen, T. H. Lai, and D. Xuan, "Measuring and guaranteeing quality of barrier coverage for general belts with wireless sensors," *ACM Trans. Sens. Netw.*, vol. 6, no. 1, p. 2, Dec. 2009.

[10] S. Kumar, T. H. Lai, and A. Arora, "Barrier coverage with wireless sensors," in *Proc. ACM MobiCom*, 2005, pp. 284–298.

[11] C. Wang, B. Wang, H. Xu, and W. Liu, "Energy-efficient barrier coverage in WSNs with adjustable sensing ranges," in *Proc. IEEE VTC Spring*, May 2012, pp. 1–5.

[12] Z. Wang, H. Chen, Q. Cao, H. Qi, and Z. Wang, "Fault tolerant barrier coverage for wireless sensor networks," in *Proc. IEEE INFOCOM*, Apr. 2014, pp. 1869–1877.

[13] Z. Wang, J. Liao, Q. Cao, H. Qi, and Z. Wang, "Achieving k-barrier coverage in hybrid directional sensor networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 7, pp. 1443–1455, Jul. 2014.

[14] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*. Berlin, Germany: Springer, 2003.

[15] J. He and H. Shi, "Finding barriers with minimum number of sensors in wireless sensor networks," in *Proc. IEEE ICC*, May 2010, pp. 1–5.

[16] D. Kim, J. Kim, D. Li, S.-S. Kwon, and A. O. Tokuta, "On sleep-wakeup scheduling of non-penetrable barrier-coverage of wireless sensors," in *Proc. IEEE GLOBECOM*, Dec. 2012, pp. 321–327.

[17] Y.-L. Lai and J.-R. Jiang, "Barrier coverage with optimized quality for wireless sensor networks," in *Proc. WPMC*, Sep. 2012, pp. 192–196.

[18] D. Kim, W. Wang, J. Son, W. Wu, W. Lee, and A. O. Tokuta, "Maximum lifetime combined barrier-coverage of weak static sensors and strong mobile sensors," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1956–1966, Jul. 2017.

[19] X. Zhang, M. L. Wymore, and D. Qiao, "An iterative method for strong barrier coverage under practical constraints," in *Proc. IEEE ICC*, May 2016, pp. 1–7.

[20] L. Kong *et al.*, "Adaptive barrier coverage using software defined sensor networks," *IEEE Sensors J.*, vol. 16, no. 20, pp. 7364–7372, Oct. 2016.

[21] C.-Y. Chang, M.-Y. Su, C.-T. Chang, C.-Y. Hsiao, and J.-C. Chen, "An energy efcient k-barrier construction mechanism for wireless sensor networks," in *Proc. ICEECET*, 2016, pp. 1–7.

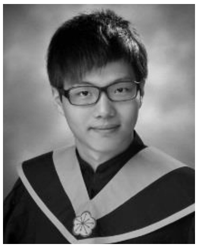
[22] H. Mostafaei, "Stochastic barrier coverage in wireless sensor networks based on distributed learning automata," *Comput. Commun.*, vol. 55, pp. 51–61, Jan. 2015.

[23] H. Mostafaei and M. R. Meybodi, "An energy efficient barrier coverage algorithm for wireless sensor networks," *Wireless Pers. Commun.*, vol. 77, no. 3, pp. 2099–2115, Aug. 2014.

CHIEN-I WENG is currently pursuing the Ph.D. degree with the Department of Information Management, National Taiwan University of Science and Technology. He is currently a Director with Taiwan Japan Industry, Collaboration Center of Institute for Information Industry. His research interests include managerial decision making, industrial policy planning, international business facilitation, management and market analysis, and innovative service generation.

CHIH-YUNG CHANG (M'05) received the Ph.D. degree in computer science and information engineering from National Central University, Zhongli, Taiwan, in 1995.

He is currently a Full Professor with the Department of Computer Science and Information Engineering, Tamkang University, New Taipei City, Taiwan. His current research interests include Internet of Things, wireless sensor networks, ad hoc wireless networks, and long term evolution broadband technologies. He has served as an Associate Guest Editor for several *SCI-indexed journals*, including the *Journal of Internet Technology* from 2004 to 2008, the *Journal of Information Science and Engineering* in 2008, *Telecommunication Systems* in 2010, *IET Communications* in 2011, the *International Journal of Ad Hoc and Ubiquitous Computing* from 2011 to 2016, and the *International Journal of Distributed Sensor Networks* from 2012 to 2014.



CHIH-YAO HSIAO received the B.S. degree from I-Shou University, Taiwan, in 2009, and the M.S. degree from Tamkang University, in 2011, where he is currently pursuing the Ph.D. degree, all in computer science and information engineering. His research interests include Internet of Things, wireless sensor networks, ad hoc wireless networks, and vehicular ad hoc networks. He received several scholarship grants in Taiwan and has participated in many projects related to wireless sensor networks, vehicular ad hoc networks, Internet of Things, and software defined networks.

CHAO-TSUN CHANG (M'12) received the Ph.D. degree in computer science and information engineering from National Central University, Taoyuan, Taiwan, in 2006. He is currently an Associate Professor with the Department of Information Management, Hsiuping University of Science and Technology, Taichung, Taiwan. In the past ten years, he has directed 12 research projects, including five National Science Council projects and seven information system developments. His current research interests include ad hoc wireless networks, wireless sensor networks, Internet of Things, and mobile computing. He is a member of the IEEE Computer and Communication Societies.

HAIBAO CHEN received the M.S. degree in computer application from Southwest University in 2007, and the Ph.D. degree in computer architecture from the Huazhong University of Science and Technology, China, in 2015. He is currently serving as the Director with the Department of Network and Communication Engineering, Chuzhou University, China. His is the Council Member of the Research Association of Computer Education with the Colleges and Universities of Anhui Province. His research interests include cloud computing, wireless sensor networks, Internet of Things, and big data.