

[Click here to view linked References](#)

Optimizing Detection Quality and Transmission Quality of Barrier Coverage in Heterogeneous Wireless Sensor Networks

Yung-Liang Lai^a, and Jehn-Ruey Jiang^b

^aTaoyuan Innovation Institute of Technology, Taoyuan, Taiwan

^bNational Central University, Taoyuan, Taiwan

Abstract—This paper proposes an algorithm, called the Optimized Barrier Coverage Algorithm (OBCA), to optimize the quality of the barrier coverage formed by a wireless sensor network (WSN). OBCA aims at optimizing the barrier coverage quality in terms of the detection degree, the detection quality, and the transmission quality (i.e., the expected transmission time). This paper also proposes a model to formulate the minimum detection probability between two WSN sensors with different sensing ranges. With the model, OBCA can be applied to heterogeneous WSNs whose sensors have various sensing ranges. OBCA's optimization is proved, and its time complexity is analyzed. The performance of OBCA is simulated and compared with those of two related algorithms.

Keywords — *wireless sensor network; Internet of Things; cloud system; barrier coverage; minimum cost maximum flow algorithm; expected transmission time; optimization*

1. INTRODUCTION

The development of the *wireless sensor network (WSN)* technology is instrumental in the emerging of Internet of Things (IoT) applications [1], ranging from smart cars to smart cities [2]. A WSN [3][4] consists of many *sensor nodes* (or simply *sensors*) and few *sink nodes* (or simply *sinks*). Sensors are small, inexpensive, battery-powered *microelectromechanical system (MEMS)* devices; they are deployed extensively in the *region of interest (ROI)* for sensing physical phenomena, such as temperature, humidity, illumination, and vibration, within their disk-shaped *sensing areas*. Sensors can process and transmit sensed data to sinks through multi-hop wireless communications. Sinks have more energy and are more powerful than sensors; they can send commands to and gather data from sensors, and can also connect to the backend system, such as a mainframe or a Cloud system [5][6], through wired or long-haul wireless communications (e.g., satellite links).

Numerous studies [7-23] investigate how to use WSNs to establish *barrier coverage* (or *virtual barriers*) for detecting *intruders* crossing a belt-shaped ROI, like a boundary, coastline, country border, and battlefield perimeter. Kumar et al. in [7] investigated how to construct barrier coverage of high *detection degrees* by deterministically deploying sensors. A WSN monitoring the ROI is said to form *k-barrier coverage* (or *barrier coverage of the detection degree k*) if any intruder crossing the ROI can be detected by at least *k* sensors. The authors in [7] also proposed an algorithm, called the *global determination algorithm (GDA)*, to determine whether a WSN of randomly deployed sensors can form *k-barrier coverage*. For doing barrier coverage research, some papers (e.g., [7]) adopted the *binary sensing model*, which assumes a sensor can (resp., cannot) detect an intruder if the distance between the sensor and the intruder is smaller (resp., larger) than the sensor sensing range. On the contrary, some papers (e.g., [8]) adopted the *probabilistic sensing model*, which assumes the probability that a sensor can detect an intruder is no longer 1 or 0, but is a value between 0 and 1 depending on the distance between the sensor and the intruder. Papers [9][10][11][12] discussed how to construct barrier coverage or its variants (e.g., local barrier coverage) for WSNs of different deployment patterns (e.g., line-based and curve-based deployment). Papers [13][14][15][16][17] investigated how to form barrier coverage or find/mend barrier gap for different types of WSNs, such as those with directional sensors, camera sensors, and mobile sensors. Papers [18][19][20] studied sleep-wakeup scheduling to maximize the lifetime of WSNs by using redundant sensors. Paper [21] estimated the desired sensor density that achieves barrier coverage with *s-t connectivity* in a thin strip (i.e., the belt-shaped ROI) of finite length, where *s-t connectivity* stands for the existence of a connected *route* between the two far ends of the thin strip.

To the best of our knowledge, only one paper [22] addressed barrier coverage optimization under the consideration of network connectivity. Lai et al. [22] proposed an algorithm, called the *Optimal Node Selection Algorithm (ONSA)*, to achieve *sink-connected barrier coverage* with the maximized detection degree by activating the minimal number of sensors of a WSN containing randomly deployed sensors and sinks. ONSA selects the minimum number of sensors as *detecting sensors* to turn on their sensing modules and communication modules to form barrier coverage with the maximized degree for detecting intruders (or intrusion events) and sending/forwarding *intrusion notifications* toward the sinks. Furthermore, ONSA selects the minimum number of extra sensors as *forwarding sensors* to turn on only their communication modules for forwarding intrusion notifications so that every detecting sensor has a route to send intrusion notifications to a sink. Fig. 1 shows a WSN of 8 detecting sensors, 3 forwarding sensors and 2 sinks to form sink-connected 2-barrier coverage for a rectangular ROI, where the sensor

communication range is a little larger than the sensing range. ONSA is optimal in the sense that it activates the minimum number of sensors to form sink-connected barrier coverage with the maximal detection degree. However, since ONSA aims at minimizing the number of detecting sensors and the number of forwarding sensors, the detection quality and the transmission quality of barrier coverage may not be good enough. This motivates us to find methods to optimize the detection quality and the transmission quality of barrier coverage.

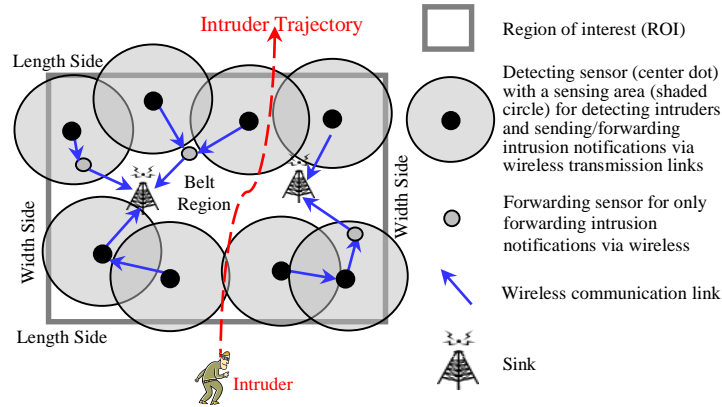


Figure 1. An example of sink-connected 2-barrier coverage of a WSN

This paper tries to optimize barrier coverage in terms of the detection quality and the transmission quality, in addition to the detection degree and number of sensors, under the consideration of sink-connectivity and heterogeneous sensing ranges. An algorithm, called the *Optimal Barrier Coverage Algorithm (OBCA)*, is proposed to use the minimum number of sensors to jointly optimize the detection degree, detection quality, and transmission quality of barrier coverage of a WSN consisting of randomly deployed sensors and sinks under the probabilistic sensing model. To optimize the detection quality, OBCA tries to maximize the minimum detection probability of detecting an intruder. A formula is proposed for OBCA to find the minimum detection probability between two sensors with different sensing ranges. With the formula, OBCA can be applied to heterogeneous WSNs whose sensors have various sensing ranges. To optimize the transmission quality, OBCA tries to minimize the average latency for transmitting data to sinks. The *Expected Transmission Time (ETT)* [24] is used to estimate the transmission latency of each direct wireless communication link between two sensors or between a sensor and a sink. The transmission latency to send an intrusion notification from a detecting sensor to a sink over a multi-link routes thus can be estimated by summing ETT values of all links on the route. By ETT values, OBCA derives a route for each detecting sensor such that all the derived routes have the minimized ETT summation average per route. In summary, in addition to the barrier coverage degree, OBCA has the following considerations never addressed before. First, it considers heterogeneous sensing ranges under the probabilistic sensing model for measuring the detection quality. Second, it considers the ETT estimation for measuring the transmission quality of intrusion notifications. Third, it tries to optimize the detection degree, detection quality, and transmission quality at a time.

The rest of this paper is organized as follows. Section 2 describes some related work. The network model and problem formulation are shown in Section 3. The correctness proofs and time-complexity analyses of OBCA are given in Section 4. OBCA's simulation results and comparisons with related algorithms, namely ONSA and GDA, are demonstrated in Section 5. Finally, Section 6 concludes the paper.

2. RELATED WORK

This section describes some relevant studies addressing barrier coverage problems. Kumar et al. in [7] applied the concept to WSNs and proposed a mechanism to deterministically deploy k -barrier coverage (or k virtual barriers) in WSNs consisting of the minimum number of sensors. They also suggested a global algorithm to make the decision about whether a WSN consisting of randomly deployed sensors can form k -barrier coverage or not, and proved that the decision can only be made globally but not locally. Yang et al. [8] proposed the concept of barrier information coverage to exploit collaborations and information fusion between neighboring sensors for reducing the number of active sensors needed to cover a belt-shaped ROI. They proposed a solution to identify the barrier information coverage set to information-cover the ROI with a small number of active sensors. Chen et al. in [9][10] introduced the notion of *L-local barrier coverage* guaranteeing the detection of intruders whose trajectory is confined to a narrow slice of length L . Therefore, it is possible for a sensor to locally decide whether L -local barrier coverage is achieved. Saipulla et al. in [11] studied the barrier coverage in WSNs of line-based sensor deployment in which sensors are distributed along a line with random offsets. They also derived a tight lower-bound for the existence of barrier

1
2 coverage in WSNs of line-based deployed sensors. He et al. in [12] considered curve-based sensor deployment. They identified
3 the characteristics for optimal curve-based deployment and proposed algorithms to achieve the optimal deployment.

4 Saipulla et al. [13][14] observed that barrier gaps may appear in WSNs if sensors are randomly deployed and if some sensors
5 fail or run out of energy. They proposed algorithms to find barrier gaps and relocate mobile sensors with limited mobility to
6 mend the gaps improve barrier coverage. Chen et al. in [15] and Tao et al. in [16] discussed barrier gap finding and mending
7 problem for WSNs consisting of directional sensors. Chen et al. proposed a gap-finding algorithm to find barrier gaps and
8 proposed two gap-mending algorithms to mend the gaps by rotating sensors. Tai et al. proposed algorithms to find barrier gaps
9 and to mend them with the minimal number of extra directional sensors. Deng et al. in [17] proposed barrier gap-mending
10 algorithms for WSNs consisting of static and mobile sensors with adjustable sensing ranges to maximize network lifetime or
11 to minimize the maximal energy consumption to move sensors.

12
13 Papers [18][19][20] studied sleep-wakeup schedules used to maximize the lifetime of WSNs. Kumar et al. [18] proposed an
14 optimal sleep-wakeup scheduling algorithm for k -barrier coverage. The basic concept is to derive sensors that form m -barrier
15 coverage ($m \gg k$) of a given WSN, and then divide the sensors into disjoint m/k sets such that sensors in every set are
16 alternatively scheduled to be activated to form k -barrier coverage to prolong the network lifetime by an optimal factor of m/k .
17 Luo et al. [19] proposed schemes to construct imperfect barrier coverage for prolonging the network lifetime achieved by the
18 above-mentioned scheduling algorithm [18]. DeWitt et al. [20] addressed the sleep-wakeup scheduling problem for WSNs with
19 energy harvesting sensors to prolong the network lifetime.

20
21 Some research discussed connectivity properties of barrier coverage. Balister et al. [21] estimated the desired sensor density
22 that achieves barrier coverage with s - t connectivity in a thin strip of finite length, where s - t connectivity stands for the existence
23 of a connected route between the two far ends of the thin strip. They developed a definition of break (a disruption in connectivity)
24 to do the estimation. Lai et al. in [22] proposed an algorithm, called the Optimal Node Selection Algorithm (ONSA), to construct
25 barrier coverage with the maximized detection degree by activating the minimal number of sensors of a WSN consisting of
26 randomly deployed sensors. Furthermore, the constructed barrier coverage has the sink connectivity property, which means
27 that every sensor on virtual barriers has a route leading to a sink for sending intrusion notifications. Since ONSA is the algorithm
28 most related to our work, we describe its details in the following paragraph.

29
30 The algorithm ONSA has two stages. The first stage is to construct the barrier coverage of the maximum degree with the
31 minimum number of sensors (i.e., detecting sensors). The second stage is to construct routes with the minimum number of
32 sensors (i.e., forwarding sensors) per route for detecting sensors to send messages to sinks. In the first stage, ONSA constructs
33 the coverage graph according to the sensing area relationship of sensors. There are edges (or arcs) going from sensor node n_i
34 to sensor node n_j and from n_j to n_i if the sensing areas of n_i and n_j overlap. The coverage graph has a virtual source node s and
35 a virtual target node t . A sensor node whose sensing area covers either of the ROI width sides has an edge either incident from
36 s or incident to t . It then performs node disjoint transformation on the graph. The transformation changes a node X with multiple
37 inbound edges and multiple outbound edges into a pair of virtual nodes X' and X'' which have an edge going from X' to X''
38 associated with Capacity=1 and Cost=0. The edges incident to the target node are also associated with Capacity=1 and Cost=0,
39 while all other edges are associated with Capacity=1 and Cost=1. ONSA then runs the minimum-cost maximum-flow algorithm
40 to return the maximum flow passing through the minimum number of nodes, which corresponds to the maximum degree of
41 barrier coverage constructed out of the minimum number of nodes. In the second stage, ONSA constructs the transmission
42 graph according to the transmission relationship of nodes. There is an edge (or arc) going from node n_i to node n_j if n_j can send
43 messages to n_i successfully. The coverage graph has a virtual source node s and a virtual target node t . Every detecting node
44 has an edge (or arc) incident from s ; and every sink node has an edge (or arc) incident to t . ONSA then performs the node-deg
45 transformation on the graph. Except for the sink node, every node X is changed into a pair of virtual nodes X' and X'' which
46 have an edge going from X' to X'' associated with Capacity= ∞ and Cost=1. The edges incident from s are also associated with
47 Capacity=1 and Cost=0, while all other edges are associated with Capacity= ∞ and Cost=0. ONSA then runs the minimum-cost
48 maximum-flow algorithm to return the flow plan which can be used to establish a route towards a sink node for every detecting
49 node. The flow plan incurs the minimum cost, which implies the number of nodes per route (the hop counts per route) is
50 minimized.

51 52 53 54 55 56 **3. NETWORK MODEL AND PROBLEM FORMULATION**

57 58 **3.1 Network Model**

59 We consider a WSN consisting of many sensors and few sinks, where sensors can detect intruders and send intrusion
60 notifications toward one of the sinks. The sensors and sinks are assumed to be randomly deployed and modeled as graph *nodes*
61
62
63
64
65

or vertices. Below, we define the coverage graph CG and the transmission graph TG to represent the sensing area (or sensing coverage) overlap relationships and the transmission relationships of nodes, respectively.

3.1.1 Coverage Graph

A coverage graph $CG=(N\cup\{u,v\}, CE)$ is a directed graph, in which N is the sensor set (or node set), CE is the coverage edge set, and u and v are two virtual nodes. The edge set CE represents the sensing area coverage overlap relationships of sensors. For two nodes n_i and n_j in N , there exist edges (n_i, n_j) and (n_j, n_i) in CE if n_i and n_j have overlapping sensing coverage. Fig. 2 shows the coverage graph CG of a WSN deployed over a rectangular ROI. The graph CG has virtual nodes u and v associated with the width sides; an edge (n_i, u) or (n_i, v) exists in CE if n_i 's sensing area overlaps either width side, where $1 \leq i \leq 10$.

A barrier path of a coverage graph CG is defined to be a path starting from u , going along edges in E_c through nodes in S , and stopping at v . Note that all nodes in a barrier path form a virtual barrier. A coverage graph is similar to a flow network [23] and a barrier path is similar to a flow in this network. In the flowing context, the terms ‘‘barrier path’’ and ‘‘flow’’ will be used interchangeably. The coverage graph and its barrier paths are very useful for measuring the degree of barrier coverage. By the theorems developed in [7], a WSN forms k -barrier coverage if and only if there exist k node-disjoint barrier paths in the coverage graph associated with the WSN. For example, the WSN in Fig. 2 forms 2-barrier coverage, since there are two node-disjoint barrier paths $u-n_1-n_2-n_3-n_4-v$ and $u-n_5-n_6-n_7-n_8-v$ in the associated coverage graph.

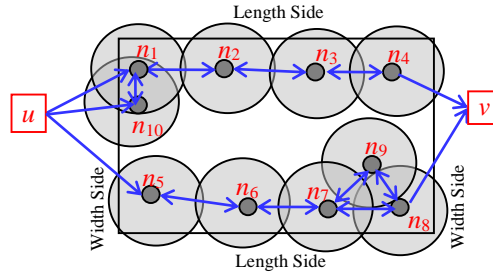


Figure 2. A WSN coverage graph with 2 node-disjoint barrier paths.

Below we present some definitions for measuring the detection quality of barrier coverage under the probabilistic sensing model.

Definition 1: Detection Probability

The detection probability of an intruder x sensed by a sensor s is defined as an exponential function $P(d, r, \alpha)$ shown in Eq. (1).

$$P(d, r, \alpha) = \begin{cases} e^{-\alpha d}, & d \leq r \\ 0, & d > r \end{cases}, \quad (1)$$

where d is the distance between s and x , as illustrated in Fig. 3(a), and $\alpha, 2 \leq \alpha \leq 5$, is a sensibility parameter related to the physical characteristics of the sensing module of s , and r is the sensing range (i.e., the radius of the circular sensing area) of the sensing module.

The function in Eq. (1) can reasonably characterize some sensing modules of different sensing ranges, such as infrared and ultrasound devices [24]. For example, a sensor node with the sensing range of 10 m is assumed to have the detection probability of $e^{-\alpha d}$ if the intruder is $d \times (10 \text{ m})$ away from the sensor, where $\alpha=2$ and $d \leq 1$. If the intruder is 10 m away from the sensor, the detection probability is $e^{-\alpha d} = 2.71828^{-2 \times 1} = 0.13533$. If the intruder is 5 m away from the sensor, the detection probability is $e^{-\alpha d} = 2.71828^{-2 \times 0.5} = 0.36787$. However, if the intruder is more than 10 m away from the sensor, the detection probability is assumed to be 0. On the contrary, a sensor node with the sensing range of 8 m is assumed to have detection probability of $e^{-\alpha d}$ if the intruder is $d \times (10 \text{ m})$ away from the sensor, where $\alpha=2.5$ and $d \leq 0.8$. For example, if the intruder is 8 m away from the sensor, the detection probability is $e^{-\alpha d} = 2.71828^{-2.5 \times 0.8} = 0.13533$. If the intruder is 5 m away from the sensor, the detection probability is $e^{-\alpha d} = 2.71828^{-2.5 \times 0.5} = 0.28650$. Similarly, if the intruder is more than 8 m away from the sensor, the detection probability is assumed to be 0.

Definition 2: Detection Quality of Edges

For the edge (n_i, n_j) between sensors n_i and n_j whose sensing areas overlap, we define the *detection quality* $Q(n_i, n_j, r_i, r_j, \alpha_i, \alpha_j)$ (or $Q(n_i, n_j)$ for short) of the edge (n_i, n_j) to be the minimum detection probability that an intruder can be detected by either n_i or n_j , where r_i and r_j are sensing ranges of n_i and n_j , and α_i and α_j are sensibility of n_i and n_j .

Let x be the point on the edge (line) between n_i and n_j that has the minimum detection probability. We can infer that $P(d, r_i, \alpha_i) = P(l-d, r_j, \alpha_j)$, where l is the length of the line segment $\overline{n_i n_j}$. As shown in Fig. 3(b), we have $\alpha_i d = \alpha_j (l-d)$, which in turn implies the following equation:

$$d = \frac{\alpha_j}{\alpha_i + \alpha_j} l \quad (2)$$

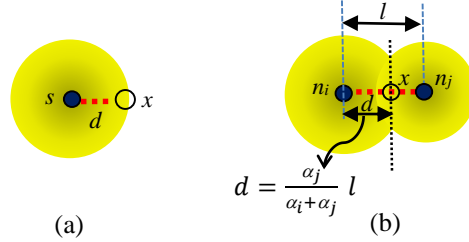


Figure 3. Illustration of the detection probability and the detection quality between sensors with various sensing ranges.

Definition 3: Detection Quality of Edge Sets

The detection quality DQ of a given edge set E is defined as the minimum detection quality of all edges in E , as shown in Eq. (3).

$$DQ(E) = \min_{(n_i, n_j) \in E} Q(n_i, n_j) \quad (3)$$

3.1.2 Transmission Graph

A transmission graph $TG = (N \cup M, TE)$ is a directed graph, where N is the sensor node set, M is the sink node set, and TE is the *transmission edge set* to represent transmission relationships between nodes. For two nodes n_i and n_j in $N \cup M$, an edge (n_i, n_j) exists in TE if node n_i can successfully transmit data to node n_j over a direct wireless communication link. Based on the transmission graph TG of a WSN, we define the *sink-connectivity* and the *expected transmission time* as follows.

Definition 4: Sink-Connectivity

For a WSN associated with the transmission graph $TG = (N \cup M, TE)$, a set Z ($Z \subseteq N$) of sensor nodes is *sink-connected* if every sensor node in Z has a route that starts from the node, passes only nodes in Z , and reaches a sink node in M . For the WSN in Fig. 4, the node set $\{n_4, n_7, n_8, n_{13}\}$ satisfies the sink-connected property, but the node set $\{n_7, n_8\}$ does not satisfy this property.

Definition 5: Expected Transmission Time

We use the concept of the *expected transmission time (ETT)* proposed in [25] to evaluate the expected time for transmitting a data packet over a direct wireless communication link or an edge in TG . The expected transmission time ETT_e of an edge e is calculated on the basis of the forward transmission ratio and the reverse transmission ratio of the link. For an edge $e = (n_i, n_j)$ from node n_i to node n_j , the *forward transmission ratio* $FTR(n_i, n_j)$ is the measured probability that a data packet sent by n_i is successfully received by n_j . On the other hand, the *reverse transmission ratio* $RTR(n_j, n_i)$ is the measured probability that an acknowledge packet sent by n_j is successfully received by n_i . Note that we assume sensors' transmission modules are always on or some mechanisms are used to turn on the transmission modules when packets are ready to be transmitted.

For simplicity, we assume the forward packet (e.g., an intrusion notification message) and the reverse packet (e.g., an acknowledgement message) have the same length L . Similarly, we assume the forward packet and the reverse packet have the same data transmission rate B . The *expected transmission time* $ETT_e(n_i, n_j)$ of an edge $e = (n_i, n_j)$ for node n_i to successfully transmit a forward packet to node n_j and to receive the packet acknowledgement is formulated in Eq. (4).

$$ETT_e(n_i, n_j) = \frac{1}{FTR(n_i, n_j) \times RTR(n_j, n_i)} \times \frac{L}{B} \quad (4)$$

Based on Eq. (4), the total expected transmission time $ETT_r(R)$ for transmitting a data packet through a route R containing many edges is formulated according to Eq. (5).

$$ETT_r(R) = \sum_{(n_i, n_j) \in R} ETT_e(n_i, n_j) \quad (5)$$

Definition 6: Route Set

For a WSN with the transmission graph $TG=(N \cup M, TE)$, a *route set* RS associated with sensor sets X and Y ($X, Y \subseteq N$) is defined to be a minimal set of routes such that every sensor in X has exactly a route leading to a sink in M going through only sensors in $(X \cup Y)$.

Under the assumption that all sensors in X have the same probability to send equal-sized packets to sinks, the average expected transmission time $ETT(RS)$ of the route set RS is formulated according to Eq. (6).

$$ETT(RS) = AVG_{R \in RS} ETT_r(R). \quad (6)$$

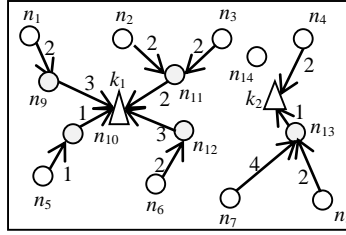


Figure 4. The partial transmission graph with edges associated with ETT values for a WSN with sensors n_1, \dots, n_{14} and sinks k_1 and k_2 .

3.2 Barrier Coverage Optimization Problem

This paper is to solve the (k, q, t) -barrier coverage optimization problem defined below. Given a WSN with the coverage graph CG and the transmission graph TG , the (k, q, t) -barrier coverage optimization problem is to find a *detecting (sensor) set* DS , a *forwarding (sensor) set* FS , and a *route set* RS associated with DS and FS for achieving the following two goals:

G1: DS is the set having the maximum number k of node-disjoint barrier paths in CG with the maximized detection quality q .

G2: RS is the set having the minimized average expected transmission time t , where $t = ETT(RS)$ in TG .

4. THE PROPOSED ALGORITHM

4.1 Algorithm Design

In this subsection, we describe the proposed algorithm, OBCA. Given the sensor node set N , sink node set M , sensing coverage relationship edge set CE , and transmission relationship edge set TE , OBCA can derive a detecting set DS , a forwarding set FS , and a *route set* RS associated with DS and FS to achieve the two goals G1 and G2 of the (k, q, t) -barrier coverage optimization problem.

OBCA is closely related to the maximum-flow algorithm (*MFA*) and the minimum-cost maximum-flow algorithm (*MCMFA*) for flow networks. A flow network is a directed graph where each edge has a capacity to receive a flow such that the amount of the flow cannot exceed the capacity. The graph has two special nodes, the source node u and target node v . The *max-flow problem* is to find a *flow plan (FP)* with the maximum flows going from u to v , where an FP is a function assigning the amount of flow on every edge under the capacity restriction. The Edmonds-Karp algorithm [23], which can be integrated into OBCA, is a famous algorithm to solve the max-flow problem. Furthermore, for a flow network where each edge has a capacity and a cost per unit of the flow going through the edge, the *min-cost max-flow* problem is to find an FP with the maximum flow going from u to v with the minimum cost. The Orin-Ahuja algorithm [26], which can also be integrated into OBCA, is a famous algorithm to solve the min-cost max-flow problem. Fig. 5 shows the overview flowchart of OBCA, and Fig. 6 shows the detailed pseudo code of OBCA.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

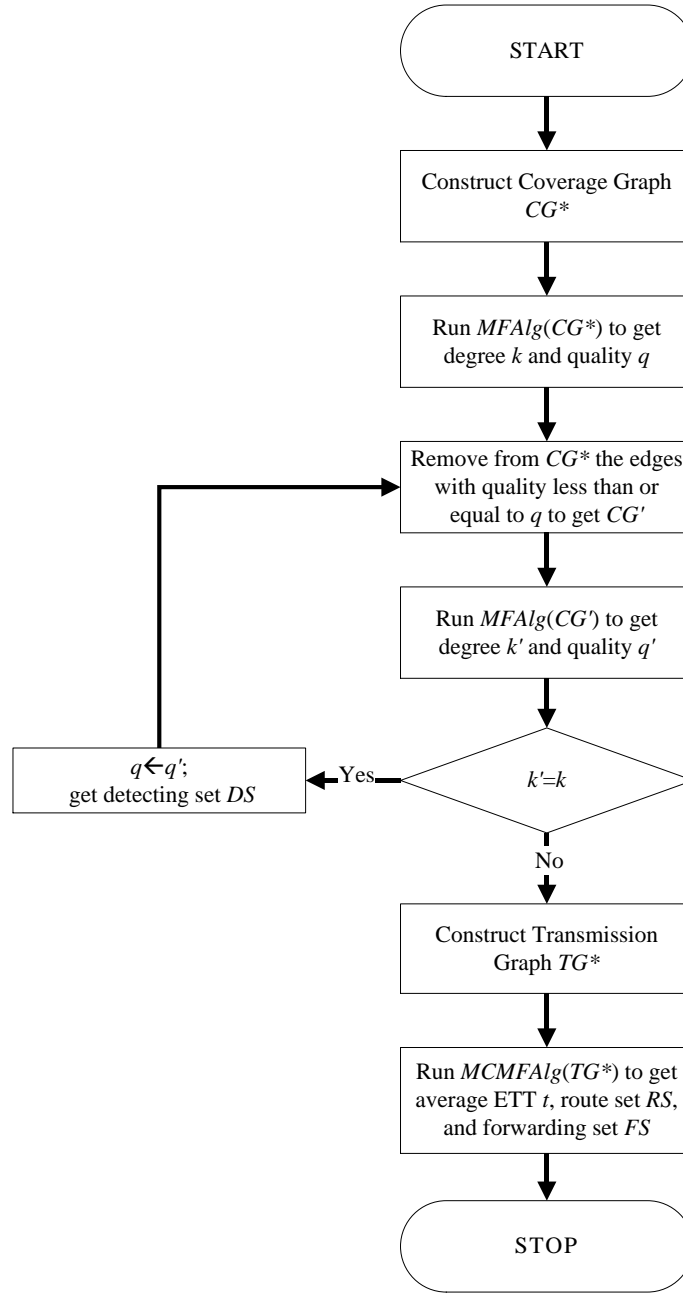


Figure 5. The flowchart of OBCA.

Optimal Barrier Coverage Algorithm (OBCA)

Input: N, M, CE, TE

Output: k, q, t, DS, FS, RS

1. $CG \leftarrow (N \cup \{u, v\}, CE)$, where u and v are virtual nodes, all edges adjacent to u and v are associated with Capacity=1 and Cost=0, and all other edges in CE are associated with Capacity=1 and Cost=detection quality formulated in Eq. (2)
 2. $CG^* \leftarrow \text{Node-Disjoint-Transform}(CG)$
 3. $FPC \leftarrow \text{MFAlg}(CG^*)$; $DS \leftarrow$ the detecting set associated with FPC ; $k \leftarrow$ the number of flows in FPC ; $q \leftarrow DQ(\text{edge set associated with } FPC)$ formulated in Eq. (3)
 4. repeat
 5. $CG' \leftarrow CG^* - \{(n_i, n_j) \mid Q(n_i, n_j) \leq q\}$
 6. $FPC' \leftarrow \text{MFAlg}(CG')$; $DS' \leftarrow$ the detecting set associated with FPC' ; $k' \leftarrow$ the number of flows in FPC' ; $q' \leftarrow DQ(\text{edge set associated with } FPC')$ formulated in Eq. (3)
-


```

1
2   7.   if ( $k=k'$ ) then
3       8.      $FPC \leftarrow FPC'$ ;  $DS \leftarrow$  the detecting set associated with  $FPC'$ ;  $q \leftarrow q'$ ;  $LoopStop \leftarrow$  false
4       9.   else
5       10.     $LoopStop \leftarrow$  true
6       11. until  $LoopStop$ 
7       12.  $TG \leftarrow (N \cup M, TE)$ , where each edge in  $TE$  is associated with Capacity= $\infty$  and Cost=ETT value formulated in Eq. (4)
8       13.  $TG^* \leftarrow TG\text{-FN-Transform}(TG)$ 
9       14.  $FPT \leftarrow MCMFAlg(TG^*)$ 
10      15.  $TS \leftarrow$  the transmission node set associated with  $FPT$ ;  $RS \leftarrow$  the route set associated with  $FPT$ ;  $FS \leftarrow TS - DS$ ;  $t \leftarrow$  (the
11          total cost associated with  $FPT$ ) /  $|RS|$ 
12
13      16. return  $k, q, t, DS, FS, RS$ 

```

Figure 6. The pseudo code of OBCA.

The details of the OBCA pseudo code are described below:

- N, M, CE, TE are the input of OBCA. They are the sensor node set, the sink node set, the coverage edge set to represent the sensing area coverage overlap relationships of sensors, and the transmission edge set to represent transmission relationships between nodes, respectively.
- k, q, t, DS, FS, RS are the output of OBCA. They are the optimized degree, optimized detection quality, optimized transmission quality, detecting node set, forwarding node set and route set of barrier coverage.
- The first task of OBCA is to construct the coverage graph $CG=(N \cup \{u,v\}, CE)$ according to N, M , and CE (line 1). Note that two virtual nodes u and v are added into the graph, and all edges adjacent to u and v are associated with Capacity=1 and Cost=0, and all other edges in CE are associated with Capacity=1 and Cost=detection quality formulated in Eq. (2).
- OBCA then performs the *node-disjoint transformation* on CG to generate the transformed coverage graph CG^* (line 2). As shown in Fig. 7, the node-disjoint transformation changes a node x with multiple inbound flows and multiple outbound flows into a pair of virtual nodes x' and x'' which has an edge going from x' to x'' associated with Capacity=1 and Cost=0. The purpose of the transformation is to guarantee that the flows in the FP generated later are node-disjoint. Since there is only one edge with Capacity=1 between x' and x'' in the transformed CG^* , there is only one flow going through x in the original CG . Fig. 8 shows The example of the node-disjoint transformation on the coverage graph of Fig. 2.
- OBCA performs a max-flow algorithm (e.g., the Edmonds-Karp algorithm [23]) on CG^* to find an FP for coverage FPC with the maximum number k of flows. Since every flow goes from the virtual node u to the virtual node v through different intermediate nodes, so flows are equivalent to disjoint barrier paths and k is equivalent to the coverage degree. The node set DS and the maximum detection quality q associated with FPC are also derived (line 3).
- In the repeat-until loop (lines 4-11), OBCA tries to increase the detection quality q without decreasing the detection degree k . This is achieved by removing all the edges having the detection quality smaller than or equal to q (line 5), and then re-running the max-flow algorithm to derive new flow plan FPC' (line 6). If the new flow plan has the same detection degree as k (line 7), then the new flow plan FPC' has better detection quality than FPC but has the same detection degree as FPC , and thus FPC is replaced by FPC' and all associated variables are updated accordingly (line 8). In such a case, the variable $LoopStop$ is set as false (line 8); otherwise it is set as true (line 10) to stop the loop. Note that the loop continues until $LoopStop$ is true (line 11).
- After the best detection degree k and the best detection quality q is derived, OBCA tries to find the best transmission quality t . It first constructs the transmission graph $TG=(N \cup M, TE)$, where each edge in TE is associated with Capacity= ∞ and Cost=ETT value formulated in Eq. (4) (line 12).
- Afterwards, OBCA performs the *transmission graph to flow network transformation* (i.e., $TG\text{-FN-Transform}$) to transform the transmission graph TG into a flow network TG^* by adding a virtual source node u and a virtual target node v , and by adding an edge from node u to every node in DS with Capacity=1 and Cost=0, and adding an edge from every sink node in M to node v with Capacity= ∞ and Cost=0 (line 13). Fig. 9 shows the example of the transmission graph to flow network transformation on the transmission graph of Fig. 4.
- OBCA performs a min-cost max-flow algorithm (e.g., the Orlin-Ahuja algorithm [26]) on TG^* to decide the minimum cost flow plan for transmission FPT (line 14).

- OBCA sets the *transmission (node) set* associated with *FPT* as *TS*, sets the route set associated with *FPT* as *RS*, and sets the smallest average transmission time t as the ratio of the total cost associated with *FPT* over $|RS|$, the cardinality of *RS* (line 15). Note that a node in *DS* has exactly a route going from the node to a sink node through only nodes in $DS \cup FS$ due to the setting of the edge capacity.
- OBCA returns the largest detection degree k , the highest detection quality q , the highest transmission quality t , the detecting set *DS* of sensors, the forwarding set *FS* of sensors, and the route set *RS* of routes having the minimized average expected transmission time t (line 16).

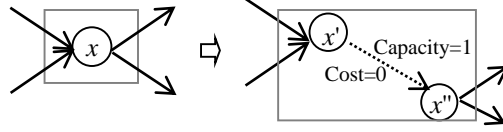


Figure 7. Illustration of the node-disjoint transformation.

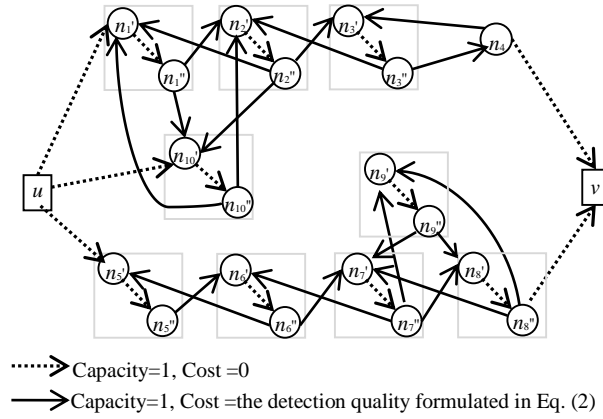


Figure 8. The example of the node-disjoint transformation on the coverage graph of Fig. 2.

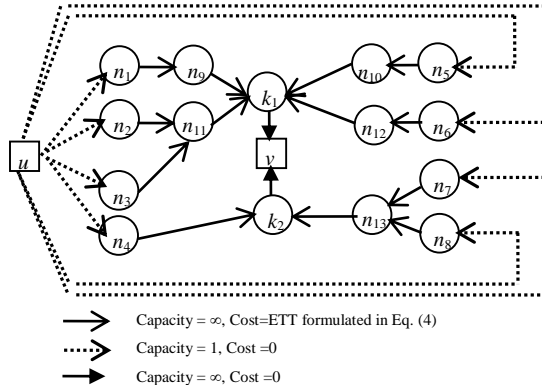


Figure 9. The example of the transmission graph to flow network transformation on the transmission graph of Fig. 4.

4.2 Correctness Proofs

In this subsection, we prove the OBCA algorithm can return a detecting set *DS*, a forwarding set *FS*, and a route set *RS* for achieving the two goals of the the (k,q,t) -barrier coverage optimization problem.

Theorem 1

Let CG be a coverage graph, CG^* be the graph transformed from CG by the node-disjoint transformation, and FPC be the max-flow FP on CG^* derived by OBCA. The detecting set DS associated with FPC forms the barrier coverage of the maximum detection degree and the maximum detection quality in CG .

Proof:

The barrier paths associated with FPC are node-disjoint in CG , as each node on CG with multiple inbound edges and multiple outbound edges is converted into two virtual nodes in CG^* between which exists only one virtual link of capacity 1. The number of barrier paths is maximum, as the paths are derived from FPC , which is a max-flow FP. The edges in the barrier paths have the highest detection quality, as OBCA repeatedly removes every edge of the quality less than q , which is the detection quality of the first max-flow FP, and then repeatedly derive new max-flow FPs with the number of flows being k , which is the number of flows of the first derived max-flow FP. \square

Theorem 2

Let TG be a transmission graph with the sink node set M , TG^* be the graph transformed from TG by the transmission graph to flow network transformation, FPT be the max-flow FP on TG^* derived by OBCA, and DS , FS and RS be the detecting set, the forwarding set, and route set returned by OBCA, respectively. RS has a route going from a node in DS towards a sink in M through only nodes in FS , and the average transmission time of routes in RS is minimum.

Proof:

In the transmission graph to flow network transformation, an edge with Capacity=1 and Cost=0 is added to connect the virtual source node u to every node in DS , and an edge with Capacity= ∞ and Cost=0 is added to connect a sink node in M to the virtual target node v , and all other edges are with Capacity= ∞ and Cost= ETT value formulated in Eq. (4). RS has a route going from a node in DS towards a sink in M through only nodes in FS , as RS is associated with FPT , the min-cost max-flow FP with the number of flows being $|RS|$, the number of routes in RS . The total cost C of FPT is minimum; thus $C/|RS|$ is minimum, implying the average transmission time of routes in RS is minimum. \square

4.3 Time Complexity Analysis

In this subsection, we analyze the time complexity of OBCA. Note that we below use X to represent the set X and its cardinality (i.e., $|X|$). The time complexity of OBCA is dominated by the repeat-until loop (lines 4-11) calling $MFAIlg$ (line 6) and the call of $MCMFAIlg$ (line 14). The loop has at most CE iterations, as OBCA eliminates one edge from CE at every iteration, where CE is the edge set of the flow network CG transformed from the coverage graph CG . $MFAIlg$ can be implemented by the Edmonds-Karp algorithm [23], which is of $O(V \cdot E^2)$ time complexity for vertex set (or node set) V and edge set E . Thus, the loop is with the time complexity $O(V_c \cdot E_c^3)$, where V_c and E_c are the node set and the edge set of the flow network CG^* . $MCMFAIlg$ can be realized by combining the Edmonds-Karp algorithm [23], which is of $O(V \cdot E^2)$ time complexity, and the min-cost flow Orlin-Ahuja algorithm [26], which is of $O(E \cdot \log V \cdot (E + V \cdot \log V))$ for a graph of vertex set V and edge set E . The time complexity of OBCA is thus $O(V_c \cdot E_c^3 + E_t \cdot \log V_t \cdot (E_t + V_t \cdot \log V_t))$, where V_c (resp., V_t) is the vertex set in CG^* (resp., TG^*) and E_c (resp., E_t) is the edge set in CG^* (resp., TG^*) and TG^* is the flow network transformed from the transmission graph TG .

5. SIMULATIONS AND EXPERIMENTAL RESULTS

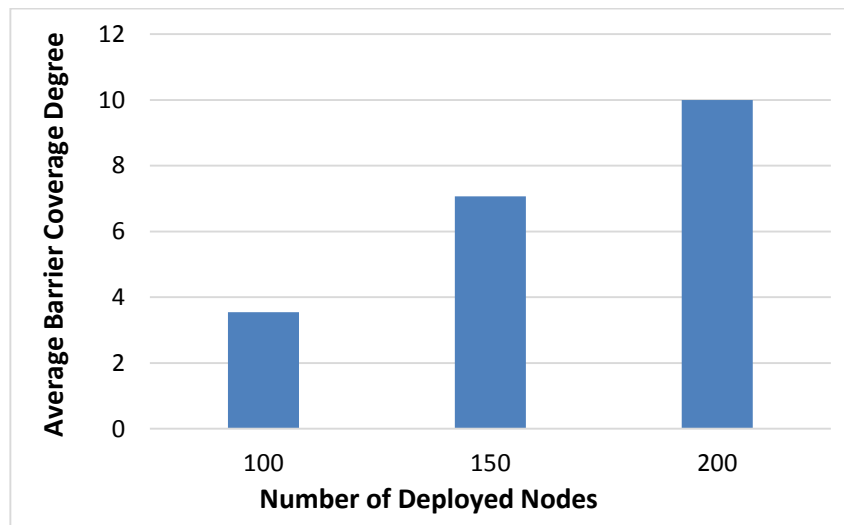
We conduct simulation experiments for the proposed OBCA algorithm, and compare the simulation results with those of the Global Determination Algorithm (GDA) [7] and the Optimal Nodes Selection Algorithm (ONSA) [22]. The algorithm GDA uses the maximum-flow algorithm to determine the highest degree of barrier coverage for a WSN with randomly deployed sensor nodes. The algorithm ONSA is an optimization algorithm with the goal to maximize the degree of barrier coverage with the minimum number of detecting nodes, and to make the detecting nodes sink-connected with the minimum number of forwarding nodes. We use C# language to incorporate the Google or-tool [28], which has the minimum-cost maximum-flow function call, to develop a simulator for evaluating the performance of OBCA, ONSA and GDA. In the simulations, 100, 150, or 200 sensor nodes are assumed to be deployed in a 150 m x 12 m rectangle-shaped ROI. Two sink nodes are assumed to be located at (50 m, 6 m), and (100 m, 6 m), respectively, which correspond to the middle position in the vertical direction and to the left, and right positions in the horizontal direction of the rectangular ROI. Sensor nodes are heterogeneous. Their sensing

1 ranges are arbitrarily set as 8 m or 10 m. All sensors, either sensors or sinks, are assumed to be equipped with the IEEE 802.15.4
 2 unslotted CSMA/CA network interface. The states of nodes and links are assumed to be fixed during the simulation duration.
 3 The transmitting power and receiving power of the radio module are set according to the off-the-shelf transceiver, Texas
 4 Instruments CC2420 [27], which is compliant with IEEE 802.15.4. Since a receiver in IEEE 802.15.4 has to acknowledge the
 5 receipt of a packet, we consider homogeneous transmission ranges for the sake of simplicity. Heterogeneous transmission
 6 ranges can be adopted. However, the packet acknowledgement should be omitted or the transmission graph should be
 7 constructed according to the smallest transmission range between sensors. Please refer to Table 1 for all simulation parameter
 8 settings.
 9
 10
 11
 12
 13

14 Table 1. Simulation Settings

15 Network Dimension	150 m x 12 m
16 Sensing Range	8 m or 10 m
17 Transmission Range	20 m
18 Transmission Rate	250 kbps
19 Packet Size	70 bytes
20 Number of Deployed Sensor Nodes	100, 150, 200
21 Number of Experiments	31 times/case
22 Number of Sink Nodes	2
23 Transmitting Power	19.8 mW
24 Receiving Power	35.5 mW

25
 26 Fig. 10 shows the barrier coverage degrees returned by OBCA, ONSA and GDA. They are the same for the three algorithms,
 27 so only one bar is depicted in Fig. 10. Fig. 11 shows the comparisons of OBCA, ONSA and GDA in terms of the detection
 28 quality. By Fig. 11 we can observe that OBCA has the highest average detection quality, and ONSA and GDA have similar
 29 average detection quality. This is because OBCA optimizes the detection quality by removing the edges with the quality lower
 30 than the best ever found quality to re-run the max-flow algorithm while retaining the same detection degree. ONSA executes the
 31 minimum-cost maximum-flow algorithm for obtaining the maximum detection degree with the minimum number of nodes.
 32 GDA executes the maximum-flow algorithm only once for obtaining the maximum detection degree. Both ONSA and GDA
 33 do not consider the detection quality. Fig. 12 shows the comparisons of OBCA, ONSA and GDA in terms of the number of
 34 detecting nodes involved in constructing the barrier coverage. Since ONSA tries to minimize the number of detecting nodes, it
 35 has the minimum number of nodes in the barrier coverage. GDA arbitrarily uses sensor nodes to obtain the maximum detection
 36 degree, it has slightly more detecting nodes than ONSA. OBCA aims at maximizing the detection quality, and it thus uses the
 37 maximum number of detection nodes in constructing the barrier coverage.
 38
 39
 40
 41



42
 43
 44
 45
 46
 47
 48
 49
 50
 51
 52
 53
 54
 55
 56
 57
 58
 59
 60
 61
 62
 63
 64
 65
 Figure 10. The barrier coverage degrees of GDA, ONSA and OBCA.

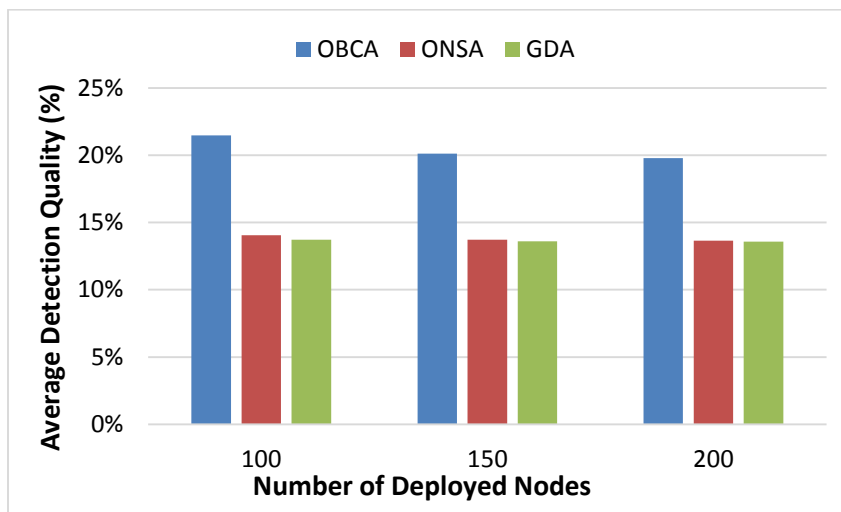


Figure 11. Comparisons of GDA, ONSA and OBCA in terms of the detection quality.

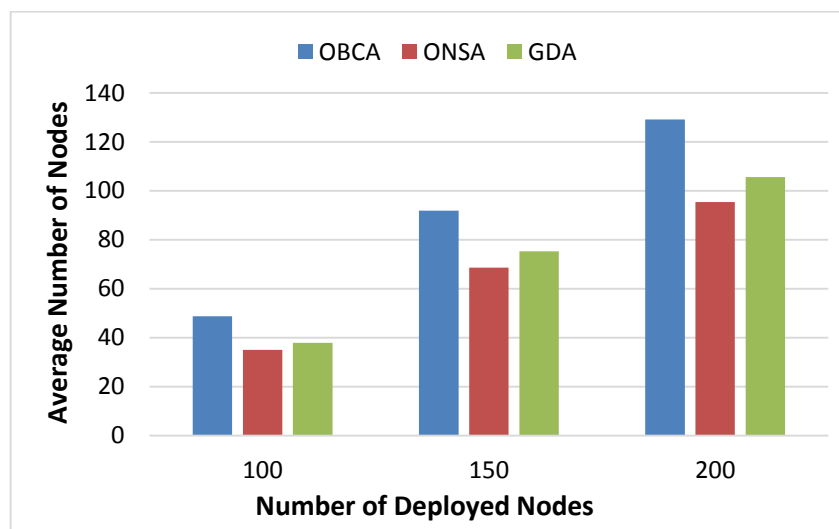


Figure 12. Comparisons of OBCA, ONSA and GDA in terms of the average number of nodes.

Fig. 13 shows the comparisons of OBCA and ONSA in terms of the transmission quality, i.e., the average ETT value. We do not compare GDA, since it does not consider intrusion notification transmission. By Fig. 13 we can observe that OBCA has shorter ETT than ONSA. This is because OBCA executes the min-cost max-flow algorithm with the cost being the ETT, but ONSA executes the min-cost max-flow algorithm with the cost being 1 for passing one node. That is to say, OBCA tries to minimize the ETT value, while ONSA tries to minimize the number of nodes per route for forwarding intrusion notification towards the sink. Thus, OBCA is better than the ONSA in terms of the average ETT.

Fig. 14 shows the comparisons of OBCA and ONSA in terms of the energy consumption per intrusion event. The comparisons only consider the energy consumed when sensor nodes are sending/receiving/forwarding packets at the appearance of an intrusion event. By Fig. 14 we can observe that OBCA has lower total energy consumption than ONSA. This is because the ETT of OBCA is smaller than of ONSA, and the successful probability of transmitting packets of OBCA is thus higher. Therefore, OBCA sends fewer packets than ONSA. Although OBCA involves more sensor nodes than ONSA, OBCA outperforms ONSA in terms of the energy consumption.

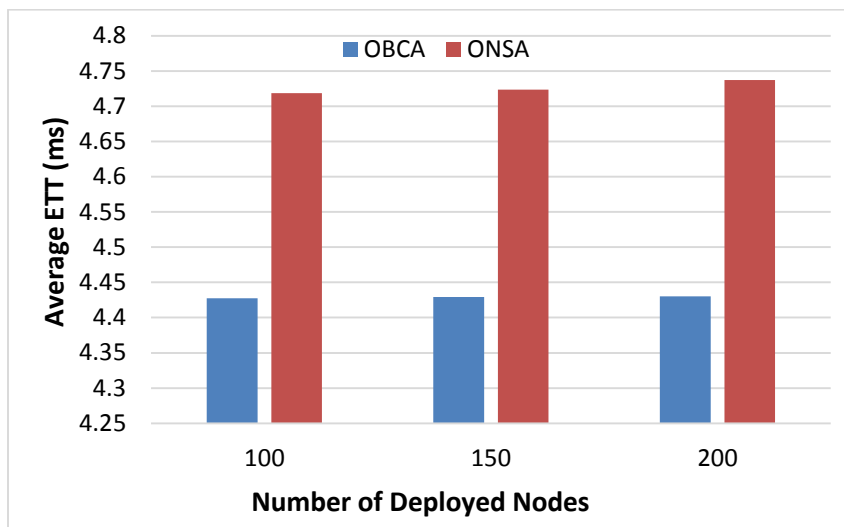


Figure 13. Comparisons of OBCA and ONSA in terms of the average ETT.

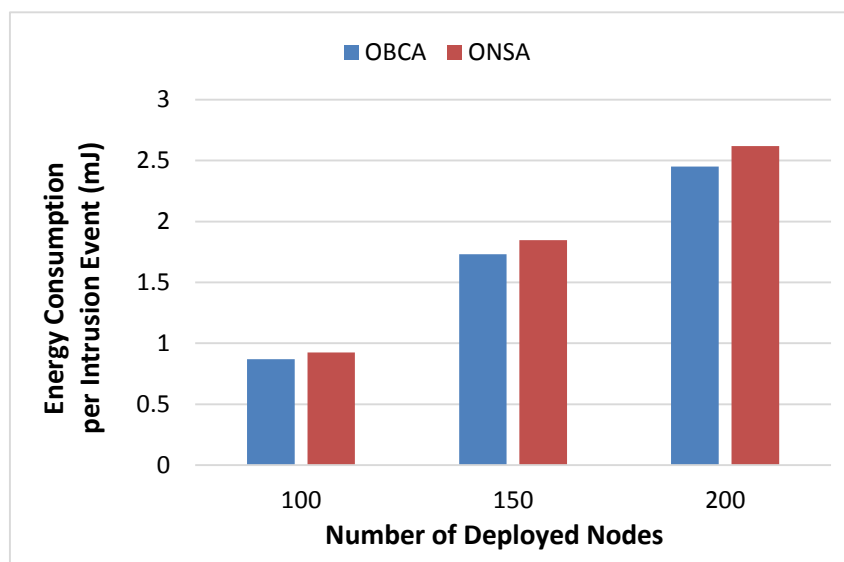


Figure 14. Comparisons of OBCA and ONSA in terms of the energy consumption per intrusion event.

6. CONCLUSIONS

In this paper, we formulate the (k, q, t) -barrier coverage optimization problem considering how to construct barrier coverage in WSNs for (1) maximizing the detection degree k of the barrier coverage, (2) maximizing the detection quality q of detecting intruders crossing the ROI, and (3) minimizing the expected transmission time t for sensors to send sensed data to sinks. An algorithm, called optimal barrier coverage algorithm (OBCA), is proposed to solve the problem on the basis of the max-flow algorithm and the min-cost max-flow algorithm running for flow networks with the polynomial time complexity. The algorithm is formally proved to solve the problem correctly.

We also propose a model to formulate the minimum detection probability between two WSN sensors with different sensing ranges. With the model, OBCA can be applied to heterogeneous WSNs whose sensors have various sensing ranges. We simulate OBCA and compare the simulation results with those of related algorithms, namely ONSA and GDA. The comparisons show that OBCA outperforms ONSA and GDA in terms of the detection quality, expected transmission time, and energy consumption for sending/forwarding notifications per intrusion event. OBCA needs more than ONSA and GDA. This is not problematic since some sensors, such as PIR sensors [29], just incur very low energy consumption.

Acknowledgments

This work was supported in part by the Ministry of Science and Technology (MOST), Taiwan, under Grant Nos. 104-2221-E-008-017-, 105-2221-E-008-078- and 105-2218-E-008-008-.

REFERENCES

- [1] M. G. Ball, B. Qela, and S. Wesolkowski, "A Review of the Use of Computational Intelligence in the Design of Military Surveillance Networks," *Recent Advances in Computational Intelligence in Defense and Security*, Springer, pp. 663-693, 2016.
- [2] N. Suri, M. Tortonese, J. Michaelis, P. Budulas, G. Benincasa, S. Russell, and R. Winkler, "Analyzing the applicability of Internet of Things to the battlefield environment," in *Proc. of IEEE International Conference on Military Communications and Information Systems (ICMCIS)*, pp. 1-8, 2016.
- [3] E. Cavalcante, J. Pereira, M. P. Alves, P. Maia, R. Moura, T. Batista, and P. F. Pires, "On the interplay of Internet of Things and Cloud Computing: A systematic mapping study," *Computer Communications*, Vol. 89-90, No. 1, pp. 17-33, 2016.
- [4] A. Botta, W. de Donato, V. Persico, and A. Pescapé, "Integration of cloud computing and internet of things: a survey," *Future Generation Computer Systems*, Vol. 56, pp. 684-700, 2016.
- [5] Priyanka Rawat, Kamal Deep Singh, Hakima Chaouchi, and Jean Marie Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *The Journal of Supercomputing*, Volume 68, Issue 1, pp 1-48, April 2014.
- [6] F. J. Oppermann, C. A. Boano and K. Römer, "A decade of wireless sensing applications: survey and taxonomy," *The Art of Wireless Sensor Networks*, Springer, pp. 11-50, 2014.
- [7] S. Kumar, T.-H. Lai, and A. Arora, "Barrier coverage with wireless sensors," in *Proc. of the 11th ACM annual international conference on Mobile computing and networking (MobiCom05)*, pp. 284-298, 2005.
- [8] G. Yang, and D. Qiao, "Barrier information coverage with wireless sensors," in *Proc. of IEEE INFOCOM*, 2009.
- [9] A. Chen, S. Kumar and T. H. Lai, "Designing localized algorithms for barrier coverage," in *Proc. of ACM Mobicom07*, pp. 63-74, 2007.
- [10] A. Chen, S. Kumar and T. H. Lai, "Local barrier coverage in wireless sensor networks," *IEEE Transactions on Mobile Computing (TMC)*, vol.9, no.4, April 2010.
- [11] A. Saipulla, C. Westphal, B. Liu, and J. Wang, "Barrier coverage of line-based deployed wireless sensor networks," in *Proc. of IEEE INFOCOM*, 2009.
- [12] S. He, X. Gong, J. Zhang, J. Chen, and Y. Sun, "Barrier coverage in wireless sensor networks: from lined-based to curve-based deployment," in *Proc. of IEEE INFOCOM*, 2013.
- [13] A. Saipulla, B. Liu, and J. Wang, "Finding and mending barrier gaps in wireless sensor networks," in *Proc. of IEEE Globecom*, 2010.
- [14] A. Saipulla, B. Liu, G. Xing, X. Fu, and J. Wang, "Barrier Coverage with Sensors of Limited Mobility," in *Proc. of ACM MobiHoc*, pp. 201-210, 2010.
- [15] J. Chen, B. Wang, W. Liu, L. T. Yang, and X. Deng, "Rotating Directional Sensors to Mend Barrier Gaps in a Line-Based Deployed Directional Sensor Network," *IEEE Systems Journal*, Vol. PP, Issue 99, pp. 1-12, 2014.
- [16] D. Tao, S. Tang, H. Zhang, X. Mao, X. Li, and H. Ma, "Strong barrier coverage detection and mending algorithm for directional sensor networks," *Ad Hoc Sensor Wireless Networks*, vol. 18, no. 1-2, p. 17, 2013.
- [17] X. Deng, B. Wang, C. Wang and W. Liu, "Barrier coverage in wireless sensor networks with adjustable sensing ranges," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 15, no. 1-3, pp. 121-132, 2014.
- [18] S. Kumar, T. H. Lai, M. E. Posner, and P. Sinha, "Maximizing the Lifetime of a Barrier of Wireless Sensors," *IEEE Transactions On Mobile Computing*, Vol. 9, No. 8, 2010.
- [19] H. Luo, H. Du, D. Kim, Q. Ye, R. Zhu, and J. Jia, "Imperfection Better Than Perfection: Beyond Optimal Lifetime Barrier Coverage in Wireless Sensor Networks," in *Proc. of 10th IEEE International Conference on Mobile Ad-hoc and Sensor Networks (MSN 2014)*, 2014.
- [20] J. DeWitt and H. Shi, "Maximizing Lifetime for k-Barrier Coverage in Energy Harvesting Wireless Sensor Networks," in *Proc. of 2014 Globecom*, 2014.
- [21] P. Balister, B. Bollobas, A. Sarkar, and S. Kumar, "Reliable density estimates for coverage and connectivity in thin strips of finite length," in *Proc. of ACM Mobicom*, pp.75-86, 2007.
- [22] Y. L. Lai, and J. R. Jiang, "Optimizing sink-connected barrier coverage in wireless sensor networks," *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 20, No. 1, pp. 39-48, 2015.
- [23] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*, MIT Press, 2001.
- [24] A. Elfes, "Occupancy grids: A stochastic spatial representation for active robot perception," in *Proc. of the 6th Annual Conference on Uncertainty in Artificial Intelligence (UAI-90)*, pp. 60-70, 1991.
- [25] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Prof. of ACM MobiCom*, 2004.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

[26] J. B Orlin, R.K. Ahuja, "New scaling algorithms for assignment and minimum cycle mean problems," *Mathematical Programming*, vol. 54, no. 1-3, pp. 41-56, 1992.

[27] Texas Instruments, "CC2420 datasheet," URL: www.ti.com/lit/ds/symlink/cc2420.pdf, last accessed in August 2016.

[28] Google, "An introduction to or-tools, Google's software suite for combinatorial optimization," URL: <https://developers.google.com/optimization/>, last accessed in December 2016.

[29] Panasonic, "PIR motion sensors (Passive Infrared or Pyroelectric) from Panasonic for optimal usability and reliability," URL: http://www3.panasonic.biz/ac/cdn/e/control/sensor/human/catalog/bltn_eng_papirs.pdf, last accessed in December 2016.