Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2659901, 12 pages
https://doi.org/10.1155/2022/2659901

Research Article

Lifetime-Maximized Strong Barrier Coverage of 3D Camera Sensor Networks

Yi Hong[1,2] **Chuanwen Luo**[1,2] **Deying Li**[3] **Zhibo Chen**[1,2] **and Xiyun Wang**[1]

[1]*School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China*
[2]*Engineering Research Center for Forestry-Oriented Intelligent Information Processing of National Forestry and Grassland Administration, Beijing 100083, China*
[3]*School of Information, Renmin University of China, Beijing 100872, China*

Correspondence should be addressed to Chuanwen Luo; chuanwenluo@bjfu.edu.cn

Received 26 July 2022; Accepted 6 September 2022; Published 19 September 2022

Academic Editor: Yan Huang

Camera sensor networks (CSNs) have advantages on providing the precise and multimedia information for plenty of applications. The high coverage quality of CSNs especially satisfies the monitoring requirements of barrier coverage. In three-dimensional (3D) application scenarios, the tracking of the potential intruder in the monitored irregular spaces brings more difficulties and challenges on strong barrier coverage for CSNs. In this paper, we consider the strong barrier coverage problem in 3D CSNs and focus on the objective of monitoring the intruder with high resolution and maximizing the network lifetime. We firstly introduce the definition and hardness proof for the problem based on the irregular space model and the network model, which adopts the Region of Interest (ROI) sensing model with high effective resolution. Secondly, we design two sleep-and-awake scheduling algorithms for the problem in homogeneous and heterogeneous networks, respectively, which are based on the auxiliary graph transformation and the disjoint flows construction. To evaluate these algorithms' performance on the lifetime maximization, we conduct extensive simulation experiments and analyze their results on their advantages and applicable scenarios.

1. Introduction

Wireless sensor networks (WSNs) are being studied for a long time, which can be classified into sensor-based studies and data-based works. The most data-based works focused on the extracting kernel dataset and data query processing [1, 2]. The most sensor-based studies concentrated on the coverage and data transmission issues. With the high accuracy of monitoring information on coverage issue, camera sensor networks (CSNs) have been utilized for a wide range of applications which can be classified into indoor monitoring [3] and outdoor surveillance, e.g. military inspection and wild animal protection. For the military applications, CSNs can provide intrusion warning and action trend prediction for constructing the military boundaries to guarantee the quality of coverage service. For the wild animal protection, CSNs do not only prevent illegal personnel from entering the protection regions for illegal poaching but also avoid the protected animals from escaping from the regions. Thus barrier coverage has got a lot of attentions for research. To satisfy the coverage requirement of the practical applications, barrier coverage has the highest requirement on the sensed information in these coverage optimization problems.

Among the related theoretical research of barrier coverage scheduling in CSNs, the most concerned issues are the particularity of the monitored space and the accuracy of the sensing model. For the particularity of the monitored space, the space may have an irregular terrain structure in the applications like mountainous regions as shown in the orange boundary in Figure 1. The complex structure brings the difficulties and challenges for the sensor deployment and the sensing model construction, which should be considered in the sensor scheduling. For the accuracy of the sensing model, the most existing works adopted the full-view sensing model proposed in

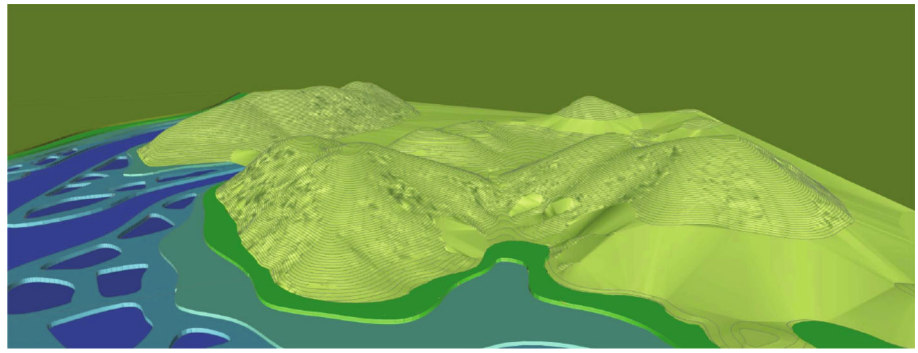


Figure 1: An instance of barrier coverage in forest region scenarios.

[4], which can guarantee the coverage for the target's all facing directions. Besides the sensing omnidirectivity, high image resolution is also considered in the sensing model in [5].

In this paper, we study on the barrier coverage problem in 3D CSNs for the application with irregular geometrical characteristics and strong coverage requirements. The goal of the problem is maximizing the network lifetime under the premise in strong barrier coverage, called the Lifetime-Maximized Strong Barrier Coverage problem for 3D CSNs (LifMax-BC Problem). To solve the problem, the modelling of the irregular space is our first consideration. And we secondly consider the sensing model of the camera sensors in [5] which is being modeled based on the combination of the sensing region and the image resolution. Thirdly, we focus on the problem in different network conditions, homogeneous networks and heterogeneous networks. The list of our contributions is as follows.

(i) We introduce LifMax-BC Problem for strong barrier coverage in 3D CSNs based on modelling the monitored space and the sensing region and give its hardness proof;

(ii) We propose two scheduling algorithms with the sleep-and-awake mode to solve the problem in the homogeneous networks and the heterogeneous networks, which are based on the auxiliary transformation and the disjoint maximum flow construction;

(iii) We conduct a large number of experiments and evaluate the performance of the proposed algorithms in terms of the constructed barrier number. Based on the simulation results, we analyze each scheduling algorithm's applicable scenarios.

The rest of the paper is organized as follows: Section 2 presents the related works. Section 3 introduces the preliminaries, the definition of our problem and the NP-hardness proof. The two scheduling algorithms are, respectively, proposed in Section 4 and Section 5. Performance evaluations are given in Section 6. concludes this paper and discusses the future work.

2. Related Works

The existing sensor-based research on wireless sensor networks (WSNs) can be classified into coverage problems and data transmission problems [6, 7], in which coverage includes target coverage, area coverage, and barrier coverage. Many mature sensing models of camera sensors have been formed from the contribution of the works on target and area coverage in WSNs.

Based on the sensing model of different kinds of camera sensors, there are more studies on barrier coverage with different optimization goals. Based on the sensing model of directional sensors, Wang and Cao [8] studied the construction problem for strong barrier coverage and presented redundancy reduction techniques. The authors proposed the algorithms to solve the barrier coverage problems with the minimum coverage cost based on modeling the full-view-covered regions in [9]. And Mohammad et al. [10] proposed a centralized barrier constructing algorithms based on distributed learning automata for adjustable-orientation directional sensor networks.

Among the optimization goals of barrier coverage, lifetime maximization and robustness guarantee have got attentions beside the coverage cost minimization. For lifetime maximization, Zhang et al. [11] designed a scheduling algorithm for maximizing the full-view coverage duration to solve the fairness-oriented coverage maximization problem, which is based on the full-view sensing model. For robustness guarantee, there exists lots of works to build k-barrier coverage. The k-barrier coverage algorithm for one-dimensional scenarios was proposed in [12], and the k-barrier coverage algorithm for one-dimensional scenarios was designed in [13]. For the sensors with the movement constraints, the strategy was presented for the maximum k -barrier coverage problem in [14]. There are some solutions for barrier coverage problem based on the classical theories: based on the divide and conquer theory, Wen et al. [15] proposed an efficient algorithm to construct k-barrier; based on the Dijkstra algorithm, Liu et al. [16] realized the minimum full-view coverage for mobile CSNs. And there are some algorithms for meeting special requirements in barrier coverage problem: the sensor interference issue was solved in the algorithm for k-barrier coverage in [17]; the one-way k -barrier algorithm was proposed in [18] to avoid one-way

invasion; the strategy for filling barrier coverage holes was designed to realize the goal of minimizing the energy consumption [19].

Considering the robustness guarantee, we focus on the strong barrier coverage problem with the goal of lifetime maximization and we will design two heuristics for CSNs with homogeneous networks and heterogeneous ones.

3. Preliminaries and Problem Formulations

3.1. Space Model. For the applications of CSNs in 3D scenarios, the monitored space can be modeled as a regular cube or cuboid in the most recent related works. The regular model of the space is beneficial to the deployment of the camera sensors and the construction of the sensing model of sensors. However, for the most applications with the rugged terrain or the mountain topography, the spaces are much different from those with the flat topography. The regular model of the space cannot provide precise position for the deployment of the sensors, which will affect the evaluation of the coverage quality.

With the consideration of the topographic complexity of the monitored space in real scenarios, we model the monitored space into an irregular 3D curve strip $\mathcal{ST}$ instead of a regular cube or cuboid. The 3D curve strip space $\mathcal{ST}$ has two terminal sections $\mathcal{S}$, $\mathcal{D}$ and the ceiling and the ground planes $\mathcal{T}$, $\mathcal{B}$, which can be indicated as a quadruple $\mathcal{ST} = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B})$ as shown in Figure 2. Note that if the irregular 3D strip is a cyclic annular or a zigzag band, it can be decomposed into multiple curve strips which are similarly modeled in the paper.

3.2. Sensing Model Based ROI and Network Model. For the sensing model of camera sensors, the full-view coverage model has been widely applied for the most two-dimensional scenarios, which was introduced in [4]. The full-view coverage model can provide the omnidirectional coverage based on the facing directions of the targets, which can guarantee high coverage accuracy.

In this paper, considering the accuracy of the capturing information and 3D application scenarios, we adopt the Region of Interest (ROI) sensing model with high effective resolution, which was proposed in the research [5] as the 3D sensing model of camera sensors. The ROI sensing model of camera sensors dose not only consider the monitored target's facing direction and position height but also construction the relationship between the 3D coverage space and the 2D projection area for the camera sensor. The model can satisfy the coverage requirement of application and the strategy design of sensor scheduling.

The ROI sensing model is applied in a 3D curve strip space $\mathcal{ST}$ in our paper. Considering a pair of a camera sensor v and a target t in $\mathcal{ST}$, we focus on three groups of parameters: (1) the heights of v and t, denoted as H and h, respectively; (2) the length of target t, denoted as L, which is decide by the target itself; and (3)the angle between the t's facing direction and the v's viewing direction in the vertical plane (Effective vertical angle), denoted by β. To guarantee the effective coverage, the parameter has a range with the minimum effective vertical angle $\beta_{\min}$ and the maximum effective vertical angle $\beta_{\max}$, which is determined by the required resolution and can be predefined. Based on the conclusion in [5], the angle between the target's facing direction and the sensor's viewing direction in the horizontal plane is out of the consideration because of the instability of its value. The definition of ROI model for the camera sensor is given as follows:

Definition 1 (The ROI Sensing Model of Camera Sensors). Consider a 3D curve strip space $\mathcal{ST} = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B})$, a camera sensor v located at $(\mathrm{X}, \mathrm{Y}, \mathrm{H})$ and a target t located at $(\mathrm{x}, \mathrm{y}, \mathrm{h})$ in $\mathcal{ST}$, the effective projection sensing area of v for t is a sector-shaped ring or an annular-sector domain $\mathrm{SecRing} = (\mathrm{X}, \mathrm{Y}, \mathrm{r}, \mathrm{R})$ on $\mathcal{B}$ with the inside radius $\mathrm{r} = H - h + (L/2)/\tan\beta_{\max}$ and the external radius $\mathrm{R} = H - h + (L/2)/\tan\beta_{\min}$ as shown in Figure 3.

Based on the ROI sensing model of the camera sensor, we consider the target or intruder in the barrier coverage with the known height and length, i.e., h and L. For example, if the monitored intruder is a person, the person's height and face length can be set as 1.7 meters and 0.5 meters, respectively.

The camera sensor network considered in our paper is composed of N randomly-deployed nodes, which are candidate for barrier coverage scheduling for a 3D curve strip space $\mathcal{ST} = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B})$. These N nodes are collected in the set V. For each node v_i in V, it has its own position (X_i, Y_i, H_i) and the maximum working duration l_i. With the consideration of the complexity and irregularity of the monitored space, the heights of the sensors are different. And we will discuss the cases with the same working duration and the different working durations later.

For the network, the camera sensors can be modeled as a node set $V = \{v_1, v_2, \cdots, v_N\}$. And we consider the connectivity between each pair of nodes based on their sensing ranges. Based on the ROI sensing model of nodes, there is an edge $e_{ij} = (v_i, v_j)$ between v_i and v_j if their sector-shaped rings intersect, i.e., $\mathrm{SecRing}_i \cap \mathrm{SecRing}_j \neq \varnothing$. All the connected edges are collected into the edge set E. Then the original network is modeled as $G = (V, E)$ as shown in Figure 4. And it is assumed that the transmission radius is at least twice of the sensing radius for each camera sensor, then the network $G = (V, E)$ is a connected graph.

3.3. Problem Definitions and Hardness. We focus on the strong barrier coverage in 3D CSNs, which can guarantee to detect intruders without any constraint on crossing paths in the boundary space. Based on the preliminaries, we propose the Lifetime-Maximized Strong Barrier Coverage problem for 3D CSNs (LifMax-BC Problem), whose formal definition is as follows.

Definition 2 (LifMax-BC Problem) Given.

(i) A 3D continuous curve strip space $\mathcal{ST} = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B})$ where $\mathcal{S}$, and $\mathcal{D}$ are $\mathcal{ST}$'s two terminal

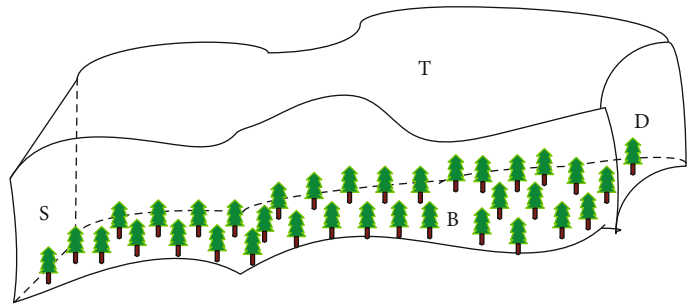


Figure 2: The 3D irregular space model for barrier coverage.

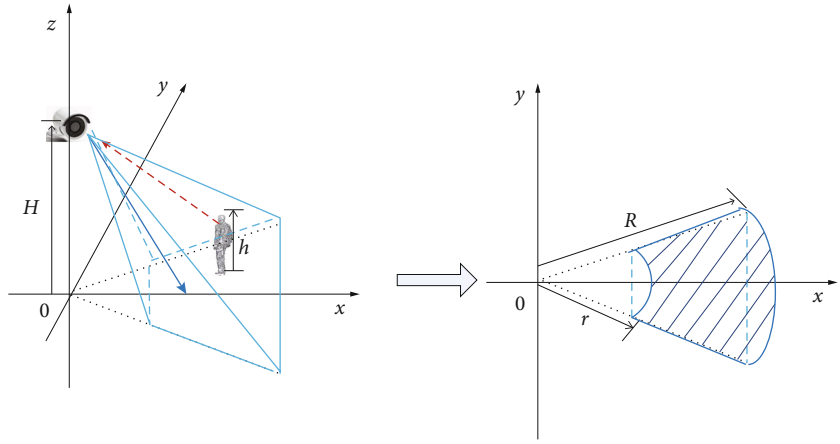


Figure 3: The illustration of camera sensing model.

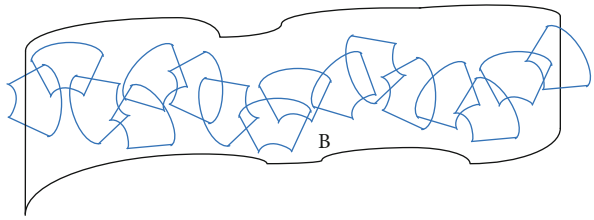


Figure 4: The illustration of network modeling.

sections and $\mathcal{T}$ and $\mathcal{B}$ are the ceiling and the ground planes of $\mathcal{ST}$

(ii) The camera sensor set V deployed in the space $\mathcal{ST}$, $\{v_1, v_2, \cdots, v_N\}$, in which each node v_i has its position (X_i, Y_i, H_i) and maximum working duration l_i

(iii) A potential target or intruder crossing the space $\mathcal{S}$ $\mathcal{T}$ with the predefined height h and face length L

(iv) LifMax-BC Problem is to find a collection of subsets of $VB = \{\text{barrier}_1, \text{barrier}_2, \text{barrier}_3, \cdots\}$ in which each barrier guarantees the strong barrier coverage of the potential target, and schedule these barriers in sleep-and-awake mode with the barrier lifetimes $\{\text{lifetime}_1, \text{lifetime}_2, \text{lifetime}_3, \cdots\}$

(v) The constraint is that each camera sensor cannot be scheduled to exceed its maximum working duration l_i, i.e., $\sum_{\text{barrier}_k \in B} \text{lifetime}_k \cdot x_i^k \le l_i (1 \le i \le N)$, where x_i^k is a binary variable to denote whether v_i is scheduled in barrier_k (if $v_i \in \text{barrier}_k$, $x_i^k = 1$; otherwise $x_i^k = 0$)

(vi) The goal is maximizing the network lifetime $\sum_{\text{barrier}_k \in B} \text{lifetime}_k$

To analyze the hardness of our problem, we review a classical NP-hard problem in graph theory, Minimum Weighted Set Cover (MWSC) Problem, which mathematical formulation is as follows:

Given a set A composed of n elements, a collection C of m subsets of $A (C = \{A_1, A_2, \cdots, A_m\})$ where each $A_j \in C$ $(1 \le j \le m)$ with a weight $w(A_j)$, the problem is to find the minimum weighted subcollection $C_0 \subseteq C$ such that $\bigcup_{A_j \in C_0} A_j = A$ and $\sum_{A_j \in C_0} w(A_j)$ is minimized.

Based on the definition of MWSC Problem, the hardness proof of our problem is given as follows.

Theorem 3. *The LifMax-BC Problem is NP-hard.*

Proof 1. In order to prove the hardness of LifMax-BC Problem, we consider the special case of the problem: each node only contributes to only one barrier, i.e., its working

duration overall contributes to the barrier it belongs to and $\sum_k x_i^k = 1$. Based on the ROI sensing model, we can construct candidate node-disjoint barriers $\{\text{barrier}_1, \text{barrier}_2, \cdots, \text{barrier}_{k'}\}$ on the structural parameters of the space $\mathcal{ST}$ and the known height and face length of the potential target. With the predefined maximum working duration l_i of each sensor, we can calculate the lifetime of each barrier, i.e., $\text{lifetime}_\text{k} = \min_{v_i \in \text{barrier}_k} l_i$. If we assign the inverse of the lifetime as a weight to each barrier, i.e., $\text{weight}(\text{barrier}_k) = 1/\text{lifetime}_k$, we can rewrite the problem in the case with different l_is as follows:

Given a sensor set $V = \{v_1, v_2, \cdots, v_N\}$ and a barrier set $C = \{\text{barrier}_1, \text{barrier}_2, \cdots, \text{barrier}_{K'}\}$ in which each barrier is a subset of V and can guarantee the barrier coverage for the space $\mathcal{ST}$, the problem is to find a subset of C, e.g. $C_0 = \{\text{barrier}_1, \text{barrier}_2, \cdots, \text{barrier}_k\}$, such that $\sum_{k=1}^{K} \text{weight}(\text{barrier}_k)$ is minimized and $\cup_{\text{barrier}_k \in C_0} \text{barrier}_k = V$.

Since the special version of LifMax-BC Problem is equivalent to MWSC Problem which is proven to be NP-hard [20]. Therefore, LifMax-BC Problem is NP-hard in general. □

To solve LifMax-BC Problem, we firstly consider the problem in homogeneous networks (denoted as Homo-LifMax-BC Problem), i.e., the camera sensors have the uniform working duration l_0. We design a barrier coverage scheduling algorithm with disjoint barriers, Robust Barrier Coverage Algorithm. Secondly, we propose the scheduling algorithm with intersecting barriers for the problem in heterogeneous networks (denoted as Hetero-LifMax-BC Problem), i.e., the camera sensors have different maximum working duration l_is, which is called as Enhancing Barrier Coverage Algorithm. The descriptions and analysis of these two algorithms are presented in the next two sections.

4. Robust Barrier Coverage Algorithm for Homo-LifMax-BC Problem

Consider the case of homogeneous camera sensors with the same maximum working duration l_0, the robustness of the network is important. And the pivotal key is avoiding the exhausted situation of some sensor, which will lead to the failure of the barriers that the sensor works for. Thus the scheduling should balance each sensor's function in the coverage barriers. With the goal of maximizing the network lifetime, we adopt the sleep-and-awake mode for scheduling, i.e., there is one barrier working and the other barriers are in sleep mode for each round. The sleep-and-awake mode can transform the original goal into maximizing the number of batches of the constructed coverage barriers. For Homo-LifMax-BC Problem, we design Robust Barrier Coverage Algorithm which is composed of two phases, Auxiliary Graph Transformation and Barrier Coverage Scheduling.

4.1. Auxiliary Graph Transformation in Robust Barrier Coverage. To design a sleep-and-awake scheduling for barrier coverage, the first phase is to give an equivalent transformation for the network model $G = (V, E)$ illustrated in subsection 3.2. The 3D graph has the node set of camera sensors $V = \{v_1, v_2, \cdots, v_N\}$ and is connected by the intersection of the sensing ranges among the sensors, i.e., $\text{SecRing}_i \cap \text{SecRing}_j \neq \varnothing (1 \le i, j \le N)$. For each node in V, we define its neighbor set and degree as $\text{Neighb}(v_i) = \{v_j | v_j \in V \wedge (v_i, v_j) \in E\}$ and $\deg(v_i) = |\text{Neighb}(v_i)|$, respectively. Then, we transform the undirected and unweighted graph G into a directed and edge-weighted graph G^* according to the following steps:

Step 1. Virtual source and destination introducing. To guarantee the strong barrier coverage of any potential target in the space $\mathcal{ST} = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B})$, it is important to construct a consecutive barrier without interval. To the end, we introduce two virtual nodes on the terminal sections $\mathcal{S}$ and $\mathcal{D}$, respectively, i.e., $V \longleftarrow V \cup \{s, t\}$, as shown in Figure 5. For the additional source s on $\mathcal{S}$, we add new edges to connect s and the nodes with the sensing range intersecting with the terminal section $\mathcal{S}$, i.e., $E \longleftarrow E \cup \{(s, v_i) | v_i \in V \wedge \text{SecRing}_i \cap \mathcal{S} \neq \varnothing\}$. For example, v_1's sensing range intersects with $\mathcal{S}$ in G, and then the edge (s, v_1) can be added into E, as shown in Figure 5. In a similar way, for the additional destination d on $\mathcal{D}$, the new edges are added to connect d and the nodes with the sensing range intersecting with the terminal section $\mathcal{D}$, i.e., $E \longleftarrow E \cup \{(v_j, d) | v_j \in V \wedge \mathit{SecRing}_j \cap \mathcal{D} \neq \varnothing\}$. Then, we update the network graph as $G = (V, E)$ by introducing s and t and we construct the auxiliary graph G^* in the next steps.

Step 2. Node-to-directed-edge converting. Based on the updated graph $G = (V, E)$, we give an equivalent transformation for each node v_i in V (with the exception of s and d): v_i is converted into a directed edge $\langle v_i, v'_i \rangle$ with the weight $\text{weight}(\langle v_i, v'_i \rangle) = \deg(v_i)$. For example, v_1 with the degree 3 in the original G corresponds to the directed edge $<v_1, v'_1>$ with $\text{weight}(\langle v_1, v'_1 \rangle) = 3$ in G^* as shown in Figure 5.

Note that there is a clear division of each pair v_i and v'_i on the function of connecting directed edges in Step 3. v_i will be the destination of all the ingoing edges to v_i and v'_i will be the source of all the outgoing edges from v_i in the original G which will give the detailed examples in Step 3. Then for the auxiliary graph G^*, the node set $V^* = V \cup \{v'_i | v_i \in V\}$, the edge set $E^* = \{<v_i, v'_i> | v_i \in V\}$, and the edge-weight set $W^* = \{\mathit{weight}(\langle v_i, v'_i \rangle) | < v_i, v'_i > \in E^*\}$.

Step 3: Undirected-edge-to-directed-edge duplexing. For the auxiliary graph G^*, we will transform the edges in the original G and assign the weights for them, which are considered into the following three cases:

(i) The outgoing edges from s: For each edge (s, v_i) in E, it is converted into a directed edge $\langle s, v_i \rangle$ with the weight $\mathit{weight}(\langle s, v_i \rangle) = 1$. Then $E^* = E^* \cup \{\langle s, v_i \rangle | (s, v_i) \in E\}$ and $W^* = W^* \cup \{\mathit{weight}(\langle s, v_i \rangle) | (s, v_i) \in E\}$. For example, the edge (s, v_1) in G has a corresponding edge $\langle s, v_i \rangle$ with $\mathit{weight}(\langle s, v_i \rangle) = 1$ in G^* as shown in Figure 5.

(ii) The bidirectional edges between (v_i, v_j)s: For each edge (v_i, v_j) in E, it is transformed into two directed

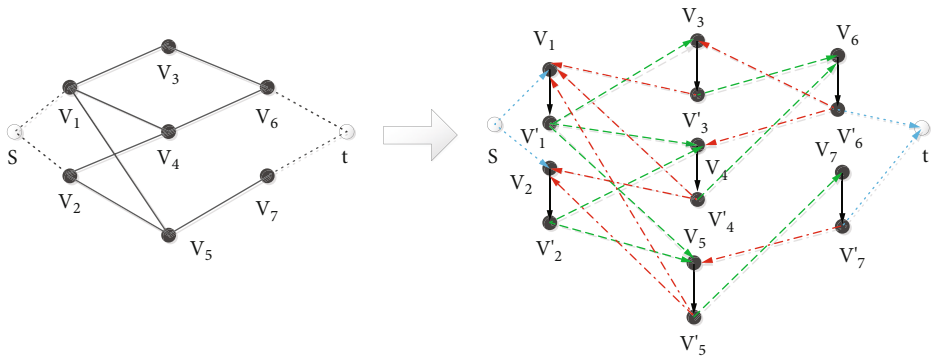


Figure 5: An instance of auxiliary graph transformation.

edges $\langle v'_i, v_j\rangle$ and $\langle v'_j, v_i\rangle$ which both have the weight 1. Then $E^* = E^* \cup \{\langle v'_i, v_j\rangle, \langle v'_j, v_i\rangle | (v_i, v_j) \in E\}$ and $W^* = W^* \cup \{weight(\langle v'_i, v_j\rangle), weight(\langle v'_j, v_i\rangle) | (v_i, v_j) \in E\}$. For example, if there is an edge (v_5, v_7) in G, the directed edges $\langle v'_5, v_7\rangle$ and $\langle v'_7, v_5\rangle$ are connected in G^* as shown in Figure 5.

(iii) The ingoing edges to d: For each edge (v_j, d) in E, it is converted into a directed edge $<v'_j, d>$ with the weight $weight(\langle v'_j, d\rangle) = 1$. Then $E^* = E^* \cup \{\langle v'_j, d\rangle | (v_j, d) \in E\}$ and $W^* = W^* \cup \{weight(\langle v'_j, d\rangle) | (v_j, d) \in E\}$. For example, the edge (v_6, t) in G has a corresponding edge $\langle v'_6, t\rangle$ with $weight(\langle v'_6, t\rangle) = 1$ in G^* as shown in Figure 5.

After the above three steps, we construct an equivalent auxiliary graph $G^* = (V^*, E^*, W^*)$ for the next barrier constructing in subsection 4.2.

4.2. Barrier Coverage Scheduling in Robust Barrier Coverage. Based on $G^* = (V^*, E^*, W^*)$, we construct the feasible barriers with the maximum number based on the maximum flow algorithm as follows.

Firstly, by taking G^*, s and t as the source and destination as the input of the Stint Algorithm in [21], K node-disjoint flows can be obtained which can be restored into a path set $\mathscr{P} = \{\text{path}_1, \text{path}_2, \cdots, \text{path}_K\}$. The paths in $\mathscr{P}$ are node-disjoint which can satisfy the requirement of LifMax-BC Problem in homogeneous network, i.e., it can avoid the situation that some sensor's expiration causes the failure of the barriers which this sensor works for.

Secondly, since the path set is constructed in the auxiliary graph G^*, we need to reduce these paths in G^* back into the barriers in G. Here we take one path as an example to explain the reduction process and the reduction of other paths are in the same way. For each $path_k = \{s, v_1, v'_1, v_2, v_2', \cdots, v_p, v'_p, d\}$ in $\mathscr{P}(1 \le k \le K)$, it can be reduced back to a coverage barrier $barrier_k$ according to the following process: for the edge $\langle s, v_i\rangle$, it is restored into the source v_i of $barrier_k$ in G; for the edge $\langle v'_i, v_j\rangle$, it is reduced into the undirected edge (v_i, v_j) of $barrier_k$ in G; for the edge $\langle v'_j, d\rangle$, it is restored into the destination v_j of $barrier_k$ in G. As shown in Figure 6, two node-disjoint paths in the auxiliary graph are reduced into two barriers $\{v_1, v_3, v_6\}$ and $\{v_2, v_5, v_7\}$. And the lifetime of each barrier is l_0 in the homogeneous network, thus the network lifetime is $K \cdot l_0$.

The detailed description of Robust Barrier Coverage Algorithm for Homo-LifMax-BC Problem is given in Algorithm 1.

To analyze the time complexity of Algorithm 1, we review the two phases of the algorithm: for auxiliary graph transformation, the virtual nodes introducing, node converting, and edge duplexing take the time of $O(1)$, $O(|V|)$ and $O(|E|)$, respectively. Thus the time complexity of Phase 1 is $O(|V|)$ under the assumption that $|V| > |E|$ in barrier coverge; for barrier coverage scheduling, it adopts the Stint Algorithm for constructing the node-disjoint flows with the maximum number, which has the time complexity of $O(|V| \cdot |E|^2)$. And the reduction process takes the time of $O(|V| \cdot |E|)$. To sum up, the time complexity of Algorithm 1 is $O(|V| \cdot |E|^2)$.

5. Enhancing Barrier Coverage Algorithm for Hetero-LifMax-BC Problem

Consider the heterogeneous camera sensors with the different maximum working duration l_is, increasing the coverage efficiency of the sensors with high l_is is beneficial to maximize the network lifetime. If applying Algorithm 1 to solve Hetero-LifMax-BC Problem, disjoint barriers will lead to the situation that there is plenty of extended sleep time for the sensors with high l_is, which goes against the problem goal of maximizing network lifetime. Thus it is necessary to design another algorithm to efficiently schedule the sensors with high durations from the perspective of making contribution to as many barriers as possible. Then, we design Enhancing Barrier Coverage Algorithm for Hetero-LifMax-BC Problem, whose main idea is to schedule the sensors with longer working durations and higher degrees in more barriers to maximize the network lifetime. There are also two phases in this algorithm, Auxiliary Graph Transformation and Barrier Coverage Scheduling.

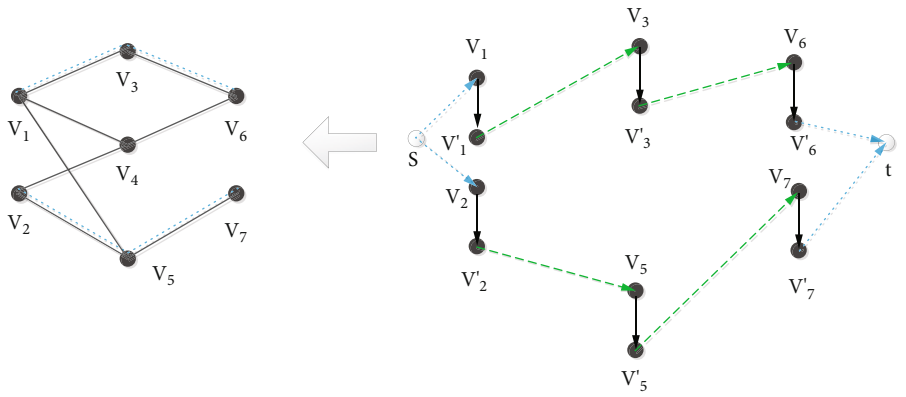


Figure 6: An instance of barrier reduction.

1: Set $V^*, E^*, W^* \longleftarrow \varnothing$, $Lifetime = 0$
2: **for** each sensor v_i in V **do**
3: $Neighb(v_i) = \{v_j | v_j \in V \wedge (v_i, v_j) \in E\}$, $\deg(v_i) = |Neighb(v_i)|$
4: **//Phase 1: Auxiliary Graph Transformation**
5: **//Step 1: Virtual source and destination introducing**
6: $V \longleftarrow V \cup \{s, t\}$
7: $E \longleftarrow E \cup \{(s, v_i), (v_j, d) | v_i, v_j \in V \wedge SecRing_i \cap \mathcal{S} \neq \varnothing \wedge SecRing_j \cap \mathcal{D} \neq \varnothing\}$
8: **//Step 2: Node-to-directed-edge converting**
9: $V^* = V \cup \{v_i' | v_i \in V\}$
10: $E^* = \{\langle v_i, v_i' \rangle | v_i \in V\}$
11: $W^* = \{weight(\langle v_i, v_i' \rangle) = \deg(v_i) | \langle v_i, v_i' \rangle \in E^*\}$
12: **//Step 3: Undirected-edge-to-directed-edge duplexing**
13: **for** each edge (s, v_i) in E **do**
14: $E^* = E^* \cup \{\langle s, v_i \rangle\}$, $W^* = W^* \cup \{weight(\langle s, v_i \rangle) = 1\}$
15: **for** each edge (v_i, v_j) in E **do**
16: $E^* = E^* \cup \{\langle v_i', v_j \rangle, \langle v_j', v_i \rangle\}$, $W^* = W^* \cup \{weight(\langle v_i', v_j \rangle) = 1, weight(\langle v_j', v_i \rangle) = 1\}$
17: **for** each edge (v_j, d) in E **do**
18: $E^* = E^* \cup \{\langle v_j', d \rangle\}$, $W^* = W^* \cup \{weight(\langle v_j', d \rangle) = 1\}$
19: $G^* = (V^*, E^*, W^*)$
20: **//Phase 2: Barrier Coverage Scheduling**
21: Apply Stint Algorithm in [21] to (G^*, s, t) and obtain K node-disjoint paths collected in $\mathcal{P} = \{path_1, path_2, \cdots, path_K\}$
21: **for** each path $path_k$ in $\mathcal{P}$ **do**
23: **for** each directed edge on $path_k$ **do**
24: Case 1: for $\langle s, v_i \rangle$, it is restored into the source v_i of $barrier_k$.
25: Case 2: for each $\langle v_i', v_j \rangle$, it is reduced into the undirected edge (v_i, v_j) of $barrier_k$.
26: Case 3: for $\langle v_i', d \rangle$, it is restored into the destination v_j of $barrier_k$.
27: $Lifetime = K \cdot l_0$
28: $\{barrier_1, barrier_2, \cdots, barrier_K\}$, $Lifetime$.

Algorithm 1: Robust Barrier Coverage Algorithm for Homo-LifMax-BC Problem $(\mathcal{SFT} = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B}), \boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E}))$

5.1. Auxiliary Graph Transformation in Enhancing Barrier Coverage. This phase is composed of three steps, and we adopt the same Step 1. Virtual source and destination introducing as that in Section 4 and obtain the updated graph $G = (V, E)$.

To construct an auxiliary graph for barrier scheduling in the heterogeneous network, it is necessary to balance two important parameters for each node: the maximum working duration l_i and the neighborhood scale $\deg(v_i)$. To this end, we introduce a new measure for each node

as lifdeg$(v_i) = l_i/\deg(v_i)$, which represents the node's possible average working duration for each neighbor. Furthermore, to analyze each node's contribution for barrier coverage, we need to give the criterion for parameters l_i and $\deg(v_i)$. (1) For the node degree, v_i's degree is identified as **high-degree** if $\deg(v_i) > 1$; otherwise, it is regarded as low-degree. (2) For the maximum working duration, l_i is identified as high-lifetime if $l_i \geq AvgL$, where $AvgL = \sum_{1 \leq i \leq N} l_i/N$; otherwise, it is regarded as low-lifetime. Based on the above preliminaries, we explain the process for Step 2 as follows.

Step 2. Node-to-directed-edge converting. Based on the graph $G = (V, E)$ added with s and t, the transformation for the nodes in V (with the exception of s and d) is divided into the following three cases:

(i) $V_1 = \{v_i | v_i \text{ is low} - \text{degree}\}$. Since $\deg(v_i) = 1$ which stands for that v_i has only one neighbor, v_i will contribute on only one barrier if v_i is scheduled in Phase 2, which is regardless of whether v_i is high-lifetime or low-lifetime. In this case, $\boldsymbol{v_i}$ is converted into 1 directed edge $<v_i, v'_i>$ with the weight $weight(<v_i, v'_i>) = l_i$

(ii) $V_2 = \{v_i | v_i \text{ is high} - \text{degree and low} - \text{lifetime}\}$. In this case, v_i will give high contribution for several barriers and we divide its working duration equally to each connectivity relationship. In details, v_i is converted into $\deg(v_i)$ directed edges $\langle v_i^d, v'^d_i \rangle$s $(1 \leq d \leq \deg(v_i))$ with the uniform weight weight$(\langle v_i^d, v'^d_i \rangle) = \text{lifdeg}(v_i)$

(iii) $V_3 = \{v_i | v_i \text{ is high} - \text{degree and low} - \text{lifetime}\}$. This case is the most complicated because of its possible unbalanced contributions to several barriers in scheduling. To avoid the unbalance, we propose a trade-off approach to guarantee the reasonably efficient scheduling for such nodes.

Firstly, we sort v_i's neighbors in nonincreasing order on the values of lifdeg(v_j), where $v_j \in Neighb(v_i)$. Secondly, we calculate the maximum value of the sum of the first $\text{sum}(v_i)$ neighbors' lifdeg(v_j)s, which is no more than l_i. In other words, we maximize the contribution of v_i on a part of neighbors (the first $\text{sum}(v_i)$ neighbors) rather than all the neighbors (Neighb(v_i)). Then we update Neighb(v_i) as $\{\text{neighb}_i^1, \text{neighb}_i^2, \cdots, \text{neighb}_i^{\text{sum}(v_i)}\}$ by only retaining the first $\text{sum}(v_i)$ neighbors and eliminating other neighbors. Then, $\deg(v_i) = \text{sum}(v_i)$. Thirdly, we perform the transformation of v_i: v_i is converted into $\text{sum}(v_i)$ directed edges $\langle v_i^u, v'^u_i \rangle$ with the different weights weight$(\langle v_i^u, v'^u_i \rangle) = \text{lifdeg}(\text{neighb}_i^u)(1 \leq u \leq \text{sum}(v_i))$.

To conclude the above three cases, there is also a clear division of each pair v_i and v'_i on the function of connecting directed edges in Step 3. v_i is in charge of all the ingoing edges to v_i and v'_i is responsible for all the outgoing edges from v_i in the original G. And the construction of the auxiliary graph G^* in Step 2 is as follows:

(a) The node set $V^* = \bigcup_{v_i \in V_1} \{v_i, v'_i\} \cup \bigcup_{v_i \in V_2} \{v_i^d, v'^d_i | 1 \leq d \leq \deg(v_i)\} \cup \bigcup_{v_i \in V_3} \{v_i^u, v'^u_i | 1 \leq u \leq \text{sum}(v_i)\}$

(b) The edge set $E^* = \bigcup_{v_i \in V_1} \{<v_i, v'_i>\} \cup \bigcup_{v_i \in V_2} \{<v_i^d, v'^d_i> | 1 \leq d \leq \deg(v_i)\} \cup \bigcup_{v_i \in V_3} \{<v_i^u, v'^u_i> | 1 \leq u \leq \text{sum}(v_i)\}$

(c) The edge-weight set $W^* = \{\text{weight}(\langle v_i, v'_i \rangle) | <v_i, v'_i> \in E^*\}$

Step 3. Undirected-edge-to-directed-edge duplexing. Based on the partial of G^* constructed in Step 2, we will add new directed edges by transforming the undirected edges in the original G as follows:

(i) The outgoing edges from s are as follows: for each edge (s, v_i) in E, it is converted into a directed edge $\langle s, v_i \rangle$ with weight$(\langle s, v_i \rangle) = 0$, if $v_i \in V_1$; it is converted into $\deg(v_i)$ directed edges $\langle s, v_i^d \rangle$ with weight$(\langle s, v_i^d \rangle) = 0$, if $v_i \in V_2$; it is converted into $\text{sum}(v_i)$ directed edges $\langle s, v_i^u \rangle$ with weight$(\langle s, v_i^u \rangle) = 0$, if $v_i \in V_3$. Then $E^* = E^* \cup \bigcup_{v_i \in V_1 \wedge (s,v_i) \in E} \{<s, v_i>\} \cup \bigcup_{v_i \in V_2 \wedge (s,v_i) \in E} \{<s, v_i^d> | 1 \leq d \leq \deg(v_i)\} \cup \bigcup_{v_i \in V_3 \wedge (s,v_i) \in E} \{<s, v_i^u> | 1 \leq u \leq \text{sum}(v_i)\}$ and $W^* = W^* \cup \{weight(\langle s, v_i \rangle) | <s, v_i> \in E^*\}$

(ii) The bidirectional edges between (v_i, v_j)s: For each edge (v_i, v_j) in E, it is transformed into $\deg(v_i) \cdot \deg(v_j)$ pairs of directed edges $\langle v'^{d_i}_i, v_j^{d_j} \rangle$ and $\langle v'^{d_j}_j, v_i^{d_i} \rangle (1 \leq d_i \leq \deg(v_i)$ and $1 \leq d_j \leq \deg(v_j))$ which all have the weight 0. Then $E^* = E^* \cup \bigcup_{(v_i,v_j) \in E} \{\langle v'^{d_i}_i, v_j^{d_j} \rangle, \langle v'^{d_j}_j, v_i^{d_i} \rangle | 1 \leq d_i \leq \deg(v_i)$ and $1 \leq d_j \leq \deg(v_j)\}$ and $W^* = W^* \cup \{\text{weight}(\langle v'_i, v_j \rangle), \text{weight}(\langle v'_j, v_i \rangle) | \langle v'_i, v_j \rangle, \langle v'_j, v_i \rangle \in E^*\}$.

(iii) The ingoing edges to d are as follows: For each edge (v_j, d) in E, the transformation is similar with that of (i), i.e., $E^* = E^* \cup \bigcup_{v_j \in V_1 \wedge (v_j,d) \in E} \{\langle v'_j, d \rangle\} \cup \bigcup_{v_j \in V_2 \wedge (v_j,d) \in E} \{\langle v'^d_j, d \rangle | 1 \leq d \leq \deg(v_j)\} \cup \bigcup_{v_j \in V_3 \wedge (v_j,d) \in E} \{\langle v'^u_j, d \rangle | 1 \leq u \leq \text{sum}(v_j)\}$ and $W^* = W^* \cup \{\text{weight}(\langle v'_j, d \rangle) = 0 | <v'_j, d> \in E^*\}$

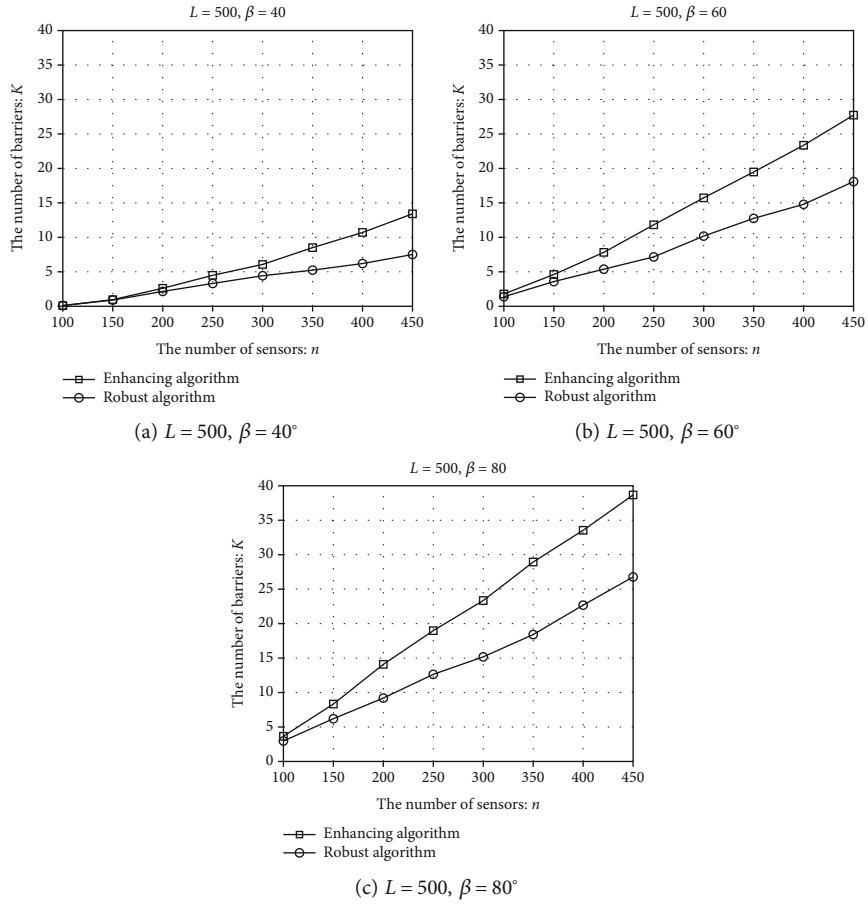
5.2. Barrier Coverage Scheduling in Enhancing Barrier Coverage. For this phase, we input the constructed auxiliary graph (G^*, s, t) to Stint Algorithm [21] and generate K node-disjoint flows which are collected in $\mathcal{P} = \{path_1, path_2, \cdots, path_K\}$. Note that since we divide the nodes with high possible contributions into several independent directed edges in Step 2 of Phase 2, we can also apply the node-

Set $V^*, E^*, W^* \longleftarrow \varnothing$, $Lifetime = 0$, $AvgL = \sum_{1 \le i \le N} l_i / N$
for each sensor v_i in V **do**
  $Neighb(v_i) = \{v_j | v_j \in V \wedge (v_i, v_j) \in E\}$, $\deg(v_i) = |Neighb(v_i)|$
  $lifdeg(v_i) = l_i / \deg(v_i)$
//**Phase 1: Auxiliary Graph Transformation**
Set $V_1, V_2, V_3 \longleftarrow \varnothing$
for each sensor v_i in V **do**
  Case 1: if $\deg(v_i) = 1$, $V_1 \longleftarrow V_1 \cup \{v_i\}$
  Case 2: if $\deg(v_i) > 1$ and $l_i \ge AvgL$, $V_2 \longleftarrow V_2 \cup \{v_i\}$
  Case 3: if $\deg(v_i) > 1$ and $l_i < AvgL$, $V_3 \longleftarrow V_3 \cup \{v_i\}$
//**Step 1: Virtual source and destination introducing**
$V \longleftarrow V \cup \{s, t\}$
$E \longleftarrow E \cup \{(s, v_i), (v_j, d) | v_i, v_j \in V \wedge SecRing_i \cap \mathcal{S} \ne \varnothing \wedge SecRing_j \cap \mathcal{D} \ne \varnothing\}$
//**Step 2: Node-to-directed-edge converting**
for each sensor v_i in V_1 **do**
  $V^* = V^* \cup \{v_i, v_i'\}$, $E^* = E^* \cup \{\langle v_i, v_i' \rangle\}$, $W^* = W^* \cup \{weight(\langle v_i, v_i' \rangle) = l_i\}$
for each sensor v_i in V_2 **do**
  $V^* = V^* \cup \{v_i^d, v'^d_i | 1 \le d \le \deg(v_i)\}$, $E^* = E^* \cup \{<v_i^d, v'^d_i> | 1 \le d \le \deg(v_i)\}$, $W^* = W^* \cup \{weight(<v_i^d, v'^d_i>) = lifdeg(v_i) | 1 \le d \le \deg(v_i)\}$
for each sensor v_i in V_3 **do**
  Calculate the maximum value of the sum of the first $sum(v_i)$ neigh-bors' $lifdeg(v_j)$s, which is no more than l_i.
  $Neighb(v_i) = \{neighb_i^1, neighb_i^2, \cdots, neighb_i^{sum(v_i)}\}$, $\deg(v_i) = |Neighb(v_i)|$
  $V^* = V^* \cup \{v_i^u, v'^u_i | 1 \le u \le sum(v_i)\}$, $E^* = E^* \cup \{<v_i^u, v'^u_i> | 1 \le u \le sum(v_i)\}$, $W^* = W^* \cup \{weight(<v_i^u, v'^u_i>) = lifdeg(neighb_i^u) | 1 \le u \le sum(v_i)\}$
//**Step 3: Undirected-edge-to-directed-edge duplexing**
for each edge (s, v_i) in E **do**
$E^* = E^* \cup \bigcup_{v_i \in V_1 \wedge (s, v_i) \in E} \{<s, v_i>\} \cup \bigcup_{v_i \in V_2 \cup V_3 \wedge (s, v_i) \in E} \{\langle s, v_i^d \rangle | 1 \le d \le \deg(v_i)\}$, $W^* = W^* \cup \{weight(\langle s, v_i \rangle) = 0 | < s, v_i > \in E^*\}$
for each edge (v_i, v_j) in E **do**
$E^* = E^* \cup \bigcup_{(v_i, v_j) \in E} \{\langle v'^{d_i}_i, v_j^{d_j} \rangle, \langle v'^{d_j}_j, v_i^{d_i} \rangle | 1 \le d_i \le \deg(v_i) \text{ and } 1 \le d_j \le \deg(v_j)\}$, $W^* = W^* \cup \{weight(\langle v_i', v_j \rangle) = 0, weight(\langle v_j', v_i \rangle) = 0 | < v_i', v_j >, < v_j', v_i > \in E^*\}$
for each edge (v_j, d) in E **do**
$E^* = E^* \cup \bigcup_{v_j \in V_1 \wedge (v_j, d) \in E} \{\langle v_j', d \rangle\} \cup \bigcup_{v_j \in V_2 \cup V_3 \wedge (v_j, d) \in E} \{v'^d_j, d | 1 \le d \le \deg(v_j)\}$, $W^* = W^* \cup \{weight(\langle v_j', d \rangle) = 0 | < v_j', d > \in E^*\}$
$G^* = (V^*, E^*, W^*)$
//**Phase 2: Barrier Coverage Scheduling**
Apply the maximum flow algorithm in [21] to (G^*, s, t) and obtain K node-disjoint paths collected in $\mathcal{P} = \{path_1, path_2, \cdots, path_K\}$
for each path $path_k$ in $\mathcal{P}$ **do**
  for each directed edge on $path_k$ **do**
    Case 1: for $\langle s, v_i \rangle$ or $\langle s, v_i^d \rangle$, it is restored into the source v_i of $barrier_k$.
    Case 2: For each $\langle v'_i, v_j \rangle$ or $\langle v'^{d_i}_i, v_j^{d_j} \rangle$, It Is reduced into the undirected edge (v_i, v_j) of $barrier_k$.
    Case 3: For $\langle v'_j, d \rangle$ or $\langle v'^d_j, d \rangle$, It Is Restored into the Destination v_j of $barrier_k$.
  $lifetime_k = \min_{<v_i, v_i'> \in path_k} weight(\langle v_i, v_i' \rangle)$
$Lifetime = \sum_{1 \le k \le K} lifetime_k$
Return $\{barrier_1, barrier_2, \cdots, barrier_K\}$, $Lifetime$.

Algorithm 2: Enhancing Barrier Coverage Algorithm for Hetero-LifMax-BC Problem $(\mathcal{SDTB}) = (\mathcal{S}, \mathcal{D}, \mathcal{T}, \mathcal{B}), \boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$

disjoint flows algorithm, which can realize the scheduling of such nodes in different barriers.

These node-disjoint paths in $\mathcal{P}$ are constructed in the auxiliary graph G^* which are needed to be reduced back into the barriers in G. Since the nodes in V_2 or V_3 may be scheduled in multiply flows in $\mathcal{P}$, the reduction process is different from that of Algorithm 1. For each path $path_k$ in $\mathcal{P}(1 \le k \le K)$, it can be reduced back to a coverage barrier $barrier_k$ based on three kinds of edges: for the edge $\langle s, v_i \rangle$ or $\langle s, v_i^d \rangle$, it is restored into the source v_i of $barrier_k$ in G; for the edge $\langle v'_i, v_j \rangle$ or $\langle v'^{d_i}_i, v_j^{d_j} \rangle$, it is reduced into the undirected edge (v_i, v_j) of $barrier_k$ in G; and for the edge $\langle v'_j, d \rangle$ or $\langle v'^d_j, d \rangle$, it is restored into the destination v_j of $barrier_k$ in G. Finally, the minimum nonzero weight on the corresponding path is the lifetime of each



(a) $L = 500$, $\beta = 40^\circ$

(b) $L = 500$, $\beta = 60^\circ$

(c) $L = 500$, $\beta = 80^\circ$

FIGURE 7: The number of Barriers K vs. number of nodes n.

barrier in the heterogeneous network, i.e., $\text{lifetime}_k = \min_{<v_i, v_i'> \in \text{path}_k} \text{weight}(\langle v_i, v_i' \rangle)$.

The whole description of Enhancing Barrier Coverage Algorithm for LifMax-BC Problem is given in Algorithm 2.

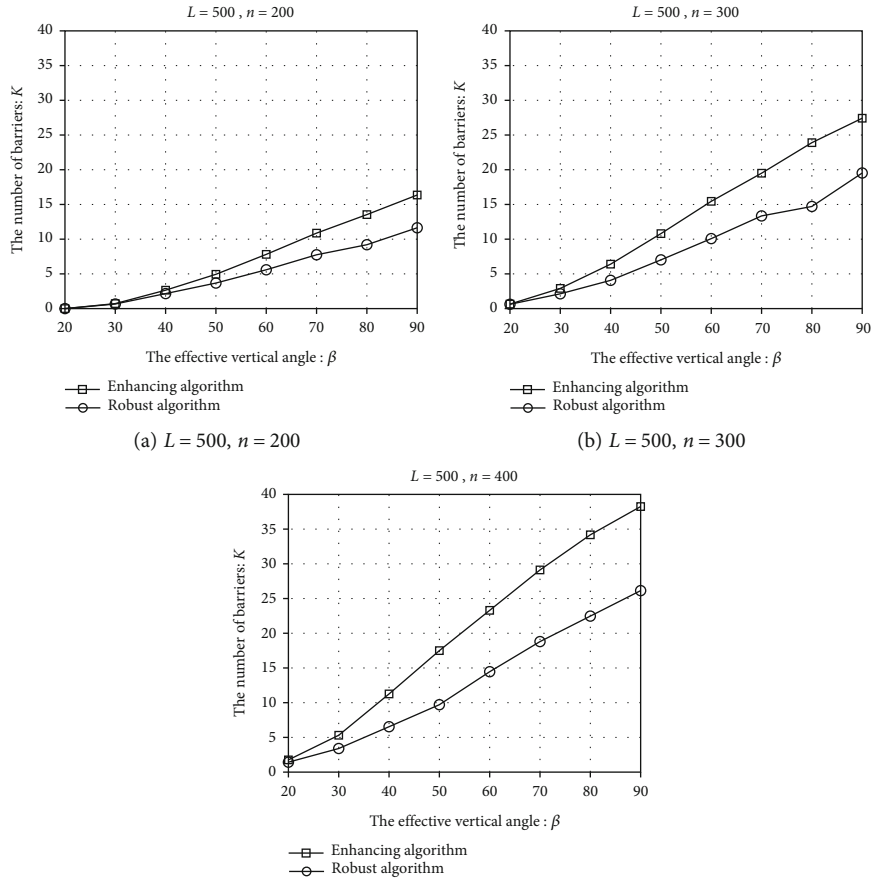
Similarly with the analysis of Algorithm 1, Algorithm 2 also has the time complexity of $O(|V| \cdot |E|^2)$. Thus the running times of our strategies are both polynomial. They are the feasible solutions of LifMax-BC Problem for homogeneous networks and heterogeneous networks.

6. Performance Evaluation

6.1. Experiment Plan. To evaluate the performance of the proposed algorithms for LifMax-BC Problem, we perform a series of experiments to compare their performance by JAVA. The optimization goal of LifMax-BC Problem is maximizing the network lifetime and we solve it for two cases (the same working duration and the different ones). Instead of the network lifetime, we choose the number of barriers K as the evaluation criterion. It is because that the number of constructed barriers stands for the number of scheduling rounds, which is more objective and fairer than the length of network lifetime, especially in the case that there is a big difference in the sensors' working duration. Here, we denote the two algorithms as Robust Algorithm and Enhancing Algorithm for short.

The experiments are performed in an irregular 3D space which is a 3D curve strip space with the length of 500 units, the width of 300 units, and the height in the range of $[50, 80]$ units, i.e., the ceiling plane of the space is an irregular curved surface. And the boundary located in the space has the length of L. For the camera sensor network, n camera sensors are randomly deployed on the ceiling plane of the space, i.e., their positions (X_i, Y_i, H_i) are randomly valued in the scope of the space. And the process of deployment is successfully finished when the network graph is connected. For each camera sensor, it has the sensing radius of 100 units, the Field-of-Vision 60° and the effective vertical angle β. And the maximum working duration l_i of each sensor is uniformed as 10 for Robust Algorithm and valued in the range of $[5, 30]$ for Enhancing Algorithm. Based on the ROI sensing model, each sensor has the minimum effective vertical angle $\beta_{\min} = 0^\circ$ and the maximum effective vertical angle $\beta_{\max} = \beta$; for the potential target/intruder, we set the height as 17 units and the face length as 2 units for general situations.

In the experiments, we will investigate the performance of the scheduling algorithms from two important parameters: the number of camera sensors n and the effective vertical angle β, which are corresponding to two groups of



(a) $L = 500$, $n = 200$

(b) $L = 500$, $n = 300$

(c) $L = 500$, $n = 400$

FIGURE 8: The number of barriers K vs. effective vertical angle β.

settings: Group 1—n varies from 100 to 450 by the step of 50 (a) $L = 500$, $\beta = 40°$; (b) $L = 500$, $\beta = 60°$; and (c) $L = 500$, $\theta = 80°$. Group 2—β varies from 20° to 90° by the step of 1 0° (a) $L = 500$, $n = 200$; (b) $L = 500$, $n = 300$; and (c) $L = 500$ $n = 400$. For each parameter setting, we run 100 instances and compute their average for evaluation.

6.2. Experiment Result Analysis. As the results in Figure 7 shown, it can be observed that the number of constructed barriers from the proposed algorithms present rising trend with the enlargement of the networks n. Between the two algorithms, Enhancing Algorithm is more influenced by n, i.e., K obtained by the algorithm grows faster with the increasing of n; Robust Algorithm is less effected by n, and the gap between the results from the two algorithms becomes larger with the growth of n. The increasing of the network scale can satisfy more requirements of strong barrier coverage and Enhancing Algorithm utilizes some sensor for multiple barriers, which increases the number of barriers. Furthermore, from Figures 7(a) and 7(c), the better performance of Enhancing Algorithm becomes more significant when the effective vertical angle $\beta = 80°$. It can be explained that the enlargement of the effective vertical angle improves sensors' coverage range which increases the probability of barrier coverage.

Investigating the effect of the effective vertical angles on scheduling algorithms in Figure 8, we can find that β's change has less significant influence when network scale is relatively small and the number of barriers increases slowly in Figures 8(a) and 8(b), while β has more influence on the algorithms' performance from Figure 8(c). Furthermore, the gap between the results from the two algorithms presents smaller than that obtained by varying the number of sensors. Seen from these three subfigures, Robust Algorithm's results presents the rising trend with the increasing of β, which is less than that presented by Enhancing Algorithm. It can be concluded that the sensing conditions have less influence on the coverage quality when the network is homogeneous, and the difference on the coverage range is more beneficial for enhancing the coverage efficiency in the heterogeneous network.

From the above two groups of experiment results, we can conclude that the proposed algorithms are both efficient on maximizing the network lifetime and can be utilized into the solution for LifMax-BC Problem in the homogeneous and heterogeneous networks.

7. Conclusions

In this paper, we investigated the camera sensor scheduling problem for strong barrier coverage in 3D CSNs with the

goal of maximizing the network lifetime, LifMax-BC Problem. The problem has been considered and analyzed for homogeneous networks (all the sensors have the uniform working duration) and heterogeneous networks (the sensors have different working durations). Based on the ROI sensing model, we, respectively, proposed two heuristic algorithms via the auxiliary graph construction and the maximum flow algorithm. The algorithm for homogeneous networks aims to increase the network robustness by constructing the disjoint barriers; and the algorithm for heterogeneous networks realizes the lifetime maximization via enhancing the utilization of the sensors with high working duration. By evaluating the performance of the proposed algorithms, the simulation results were analyzed in terms of the number of the scheduled barriers, which show that the algorithms have high efficiency on maximizing the network lifetime and can adapt to different network types. We will design the distributed strategies for the related optimization problems in the future.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China under Grant (62002022 and 62202054) and the Fundamental Research Funds for the Central Universities (No. BLX201921, No. 2021ZY88).

References

[1] S. Cheng, Z. Cai, J. Li, and H. Gao, "Extracting kernel dataset from big sensory data in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 4, pp. 813–827, 2017.

[2] S. Cheng, Z. Cai, and J. Li, "Curve query processing in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 11, pp. 5198–5209, 2015.

[3] Y. Hong, Y. Wang, Y. Zhu, D. Li, Z. Chen, and J. Li, "3D camera sensor scheduling algorithms for indoor multi-objective tracking," *Journal of Combinatorial Optimization*, vol. 39, no. 3, pp. 899–914, 2020.

[4] Y. Wang and G. Cao, "On full-view coverage in camera sensor networks," in *Proceedings of IEEE INFOCOM*, pp. 1781–1789, Shanghai, China, 2011.

[5] P. Si, W. Chengdong, Y. Zhang, Z. Jia, P. Ji, and H. Chu, "Barrier coverage for 3D camera sensor networks," *Sensors*, vol. 17, no. 8, p. 1771, 2017.

[6] Z. He, Z. Cai, S. Cheng, and X. Wang, "Approximate aggregation for tracking quantiles and range countings in wireless sensor networks," *Theoretical Computer Science*, vol. 607, no. 3, pp. 381–390, 2015.

[7] J. Li, S. Cheng, Z. Cai, Y. Jiguo, C. Wang, and Y. Li, "Approximate holistic aggregation in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 13, no. 2, pp. 1–24, 2017.

[8] Y. Wang and G. Cao, "Barrier coverage in camera sensor networks," in *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 1–10, Paris, France, 2011.

[9] H. Ma, M. Yang, D. Li, Y. Hong, and W. Chen, "Minimum camera barrier coverage in wireless camera sensor networks," in *2012 Proceedings IEEE INFOCOM*, p. 217, Orlando, FL, 2012.

[10] M. Khanjary, M. Sabaei, and M. R. Meybodi, "Barrier coverage in adjustable-orientation directional sensor networks: a learning automata approach," *Computers and Electrical Engineering*, vol. 72, pp. 859–876, 2018.

[11] Q. Zhang, S. He, and J. Chen, "Toward optimal orientation scheduling for full-view coverage in camera sensor networks," in *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Washington, DC, USA, 2016.

[12] L. Li, B. Zhang, and J. Zheng, "A study on one-dimensionalk-coverage problem in wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 13, no. 1, 11 pages, 2013.

[13] J. Tian, W. Zhang, G. Wang, and X. Gao, "2D k-barrier duty-cycle scheduling for intruder detection in wireless sensor networks," *Computer Communications*, vol. 43, no. 5, pp. 31–42, 2014.

[14] H. Ma, D. Li, W. Chen, Q. Zhu, and H. Yang, "Energy efficient k-barrier coverage in limited mobile wireless sensor networks," *Computer Communications*, vol. 35, no. 14, pp. 1749–1758, 2012.

[15] D.-S. B. J. Wen, J. Jiang, and W.-H. Dou, "Constructing *k*-barrier coverage in mobile wireless sensor networks," *Journal of Software*, vol. 22, no. 9, pp. 2089–2103, 2011.

[16] X. Liu, B. Yang, and G. Chen, "Full-view barrier coverage in mobile camera sensor networks," *Wireless Networks*, vol. 25, no. 8, pp. 4773–4784, 2019.

[17] Y. Zhu, M. Mei, and Z. Zheng, "Scheduling algorithms for k-barrier coverage to improve transmission efficiency in WSNs," *Multimedia Tools and Applications*, vol. 79, no. 15-16, pp. 10505–10518, 2020.

[18] J. Luo and S. Zou, "Strong k-barrier coverage for one-way intruders detection in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 12, no. 6, Article ID 3807824, 2016.

[19] X. Fan, S. Wang, Y. Wang, X. Jinshan, and K. Chi, "Energy-efficient barrier lifetime prolonging scheme based on repairing in directional sensor networks," *IEEE Systems Journal*, vol. 14, no. 4, pp. 4943–4954, 2020.

[20] M. R. Garey and D. S. Johnson, ""Strong" NP-completeness results," *Journal of ACM*, vol. 25, no. 3, pp. 499–508, 1978.

[21] 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, pp. 1161–1172, 2010.