

Received April 18, 2020, accepted May 3, 2020, date of publication May 6, 2020, date of current version May 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2992867

Barrier Coverage Mechanism Using Adaptive Sensing Range for Renewable WSNs

ZAIXIU DONG¹, CUIJUAN SHANG², CHIH-YUNG CHANG^{ID}², (Member, IEEE),
AND DIPTENDU SINHA ROY^{ID}³

¹School of Computer and Information Engineering, Chuzhou University, Chuzhou 239000, China

²Department of Computer Science and Information Engineering, Tamkang University, New Taipei City 25137, Taiwan

³Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong 793003, India

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

This work was supported in part by the Natural Science Foundation of Anhui Higher Education under Grant KJ2019A0648, in part by the Natural Science Foundation Project of Anhui Province under Grant 1808085MF170, in part by the Science and Technology Project of Chuzhou under Grant 2018ZN020, and in part by the Science and Technology Project of Sichuan under Grant 18YYJC0316.

ABSTRACT Barrier coverage aims at constructing defense barriers to detect the intruder crossing the predefined boundary in a given wireless sensor network. In literature, most studies considered the battery-powered sensors and applied the Boolean Sensing Model (*BSM*). The battery-powered sensors have the constraint of limited lifetime while the *BSM* will affect the actual surveillance quality evaluation because it cannot reflect the physical features of sensing. This paper applies the Probabilistic Sensing Model (*PSM*) and proposes an algorithm, called *BSAS*, which considers the solar-powered sensors with adjustable sensing radius to construct the defense barriers. Two main challenges should be overcome. The first one is the cooperative working between sensors to achieve the highest intruder detection probability for a given boundary curve. The *BSAS* identifies the bottleneck segment with the minimal surveillance quality and schedules as many as possible sensors to improve the bottleneck segment. In addition, a space-time transformation scheme which further adjusts the sensing radius of some sensors is proposed, aiming at improving the detection probability of the bottleneck segment. Consequently, the minimal surveillance quality of the barrier can be maximized. The second challenge is to maintaining the perpetual lifetime of WSNs. The *BSAS* takes into account the recharging and discharging ratio and time length of daytime in a day in its energy management which guarantees that the sensor energy can always satisfy the energy consumption for sensor working even in the nighttime. Experimental studies reveal that the proposed algorithm outperforms the existing studies in terms of surveillance quality and stability.

INDEX TERMS Solar-powered sensor, probabilistic sensing model, adjustable sensing radius, barrier coverage, wireless sensor networks.

I. INTRODUCTION

Wireless sensor networks (WSNs) have a wide range of potential applications, including smart home, precision agriculture and environmental monitoring [1], [2]. The coverage is a fundamental issue in wireless sensor networks, which has been widely investigated in recent years. According to different requirements of applications, the coverage problems are classified into target, barrier and area coverage categories [3]. The goal of target coverage is to monitor a number of given targets while the area coverage aims to surveil a given monitoring area. Difference from them,

the goal of barrier coverage [4] concerns the border surveillance aiming to detect the intruder's invalid crossing. In the barrier coverage, most studies considered that the boundary region has been deployed a large number of sensors and proposed how to organize the sensors to form the defense lines for detecting illegally crossing a boundary or a protected region. Each independent defense line is referred to a barrier. The barrier coverage is generally applied in very crucial or hazardous circumstance such as military, homeland and critical infrastructure security etc. For ensuring that the constructed barriers can effectively detect illegal crossing of border monitoring area, surveillance quality and lifetime of sensor barriers have been the main focuses of the previous studies.

The associate editor coordinating the review of this manuscript and approving it for publication was Guangjie Han ^{ID}.

The surveillance quality generally depends on a good scheduling mechanism which partitions the sensors into several groups and schedules the sensors in the same group to collaboratively establish the defense barriers. As a key factor of monitoring quality, the sensing ability of each sensor needs to be measured by applying some certain sensing model which directly influences the strategy of scheduling algorithm. There are two sensing models, including Boolean Sensing Model (*BSM*) and Probabilistic Sensing Model (*PSM*), often used in literature. A lot of studies [5]–[7] applied *BSM* which considered the coverage area of each sensor as a perfect disc. That is, each sensor can detect the event if it occurs in the sensing circle of the sensor. These studies aimed to propose scheduling mechanisms to construct defense barriers with k -coverage based on the *BSM*.

Study [5] identified the critical conditions for strong barrier coverage and proposed a distributed algorithm which partitions the sensors into several groups and schedules different groups working in turn to prolong the network lifetime. Study [6] proposed a self-organized deployment algorithm for mobile sensors aiming to construct k distinct complete barriers to guarantee the k -coverage. Study [7] investigated the issue of how to construct a defense barrier by scheduling a set of sensors for intruder detection. A decentralized algorithm, called *BCA*, is proposed to construct a maximum number of distinct k -barriers. Though most of these studies constructed the barriers with k -coverage to guarantee a certain level of surveillance quality, the *BSM* cannot reflect the facts of dynamic changes of sensing range and will reduce the detection accuracy in a real application.

Different from *BSM*, the *PSM* assumes that the detecting probability of a sensor is reduced when the distance between the sensor and the target increases. Compared with the *BSM*, the *PSM* is more practical, which can reflect the complex sensing effects of the real world. Based on the *PSM*, study [8] proposed a barrier construction algorithm aiming at minimizing the number of working sensors. The developed algorithm considered the practical functionalities of the real application scenarios and improved the intruder detection probability while reducing the false alarm probability as well. Study [9] considered the directional sensors and applied the *PSM* as its sensing model. It proposed a barrier construction algorithm which took the collaboration advantages of sensing directions and sensing ranges between neighboring sensors. Study [10] considered a number of parameters which can actually impact the sensing probability. Based on the developed probability sensing model, it proposed sensor placement schemes aiming at achieving the purpose of high coverage and low standard deviations of coverage. Though the abovementioned studies adopted the *PSM* for achieving better performance, most of the proposed barrier coverage mechanisms did not consider the issue of energy management.

Energy management is another important issue which determines the performance of the barrier coverage. In the past few years, the energy management has received

much attention. These studies can be further partitioned into two categories: Energy Conservation (EC) and Energy Harvesting (EH). The schemes of EC aimed to construct more sensor barriers by constructing as many as possible disjoint groups each of which consists of a minimal number of sensors. Different barriers can work in turn, aiming at achieving the purpose of energy conservation. Study [11] developed a learning automata scheme and proposed a distributed algorithm aiming to minimize the number of sensors in each barrier and hence maximize the number of barriers. The different barriers can stay in active mode in turn for the purpose of energy conservation. Study [12] proposed a scheduling algorithm for barrier coverage in heterogeneous sensor networks, aiming at maximizing the network lifetime. They identified several subsets with maximal network lifetime and scheduled them staying in sleep and active states alternatively such that the network lifetime of the constructed barriers can be maximized. Study [13] considered the hybrid sensor network, which consisted of two types of sensor nodes: the energy-scarce ground sensors and the energy-plentiful mobile sensors. Since the energy-scarce ground sensors are static, the goal of this work was to cooperatively reallocate the locations of the mobile sensors for achieving the both purposes of prolonging the network lifetime and improving the surveillance quality of the barrier coverage. Although these studies can increase the network lifetime of sensor barriers, it is inevitable that the battery of the sensors will drain as time goes by. As a result, the wireless sensor network cannot always work for monitoring the boundary region due to its limited network lifetime.

Some other studies considered the Energy Harvesting (EH) scheme to maintain the perpetual network lifetime. They assumed that the sensors are rechargeable for maintaining the perpetual network lifetime. Solar power was considered as one of the most promising environmental energy resources because the scale of sunlight is extensive and easily accessible [14]. The scheduling mechanisms of solar-powered WSNs usually scheduled some sensors staying in active (working) state and keep other sensors in sleeping (recharging) state to achieve the purpose of full coverage while the perpetual lifetime of the WSNs can be maintained. Study [15] discussed target coverage of solar-powered WSNs. An efficient greedy hill-climbing algorithm was developed to orderly switch sensors between recharging and active state, which aimed to maximize the overall surveillance quality of targets. Study [16] presented a reinforcement learning-based sleep scheduling for the solar-powered WSN. Based on the reinforcement learning, the proposed algorithm scheduled the sensor nodes in sleep or wakeup states, aiming to satisfy the desired area coverage. Although the abovementioned scheduling schemes developed for target or area coverage have been developed, the proposed algorithms cannot be applied to the barrier coverage because that the quality of barrier coverage is determined by the barrier with the lowest quality, rather than the total surveillance qualities of the barriers.

Different from studies [15] and [16], study [17] developed the barrier coverage mechanism in solar-powered wireless sensor networks. It proposed a barrier coverage algorithm, called *MSQ* which allocated the sensor having the largest contribution to monitor the space-time point with the minimal surveillance quality. However, study [17] did not take into account the parameter of variable sensing range which allows the sensing range to be adaptively adjusted.

This paper proposes a *Boundary Surveillance mechanism with Adjustable Sensing radius for solar-powered WSNs*, called *BSAS* algorithm. The proposed *BSAS* is a centralized mechanism and is executed by the sink node. The execution of the *BSAS* algorithm mainly consists of four phases. In the *Space-Time Partitioning Phase*, the proposed *BSAS* firstly divides the boundary curve into several equal-length segments. Similarly, the *BSAS* partitions the time line into a number of equal length cycles and then partitions each cycle into several equal-length time slots. Herein, we notice that all cycles have identical schedules for every day. Since sensors cannot be recharged in the nighttime, the *BSAS* considers two types of schedules, including daytime schedule and nighttime schedule. Each sensor stays in either recharging-only state or sensing & recharging state in each time slot of a daytime cycle. On the other hand, each sensor stays in either sleeping state or sensing-only state in each time slot of a nighttime cycle. In the *Contribution Calculation Phase*, the *BSAS* calculates the contribution of each sensor to each space-time point. The contribution of each sensor will be an important reference which will be used for scheduling in the later phase. In the *Sensors Scheduling Phase*, the *BSAS* considers the cooperative sensing between neighboring sensors, aiming to schedule all sensors such that the surveillance quality can be maximized while guaranteeing the perpetual lifetime of the WSN. To achieve this, the proposed *BSAS* treats each segment and time slot as a space-time point, finds the bottleneck points which have the minimal monitoring quality and then schedules as many as possible sensors to work for improving the surveillance qualities of those bottleneck points. Finally, in the *Space-Time Transformation Phase*, the *BSAS* dynamically adjusts the sensing radius of some potential sensors aiming to transfer the surveillance quality from space dimension to time dimension such that the monitoring qualities of the bottleneck points can be further improved.

Two main challenges should be overcome when designing the proposed *BSAS* algorithm. The first one is how to schedule each sensor for achieving the maximal surveillance quality. To cope with this problem, this work tries to identify “*the bottleneck space-time point*”. Then, the *BSAS* schedules as many as possible sensors for improving the surveillance qualities of these points. Another novel scheme, called space-time transformation scheme, for improving the qualities of these points is to dynamically adjust the sensing radius of some sensors. The second challenge is the perpetual lifetime constraint for energy management. Each sensor is

continuously recharged in daytime to satisfy the constraint that the recharged energy should be able to support the energy consumption for sensor working in both daytime and nighttime. In other words, the proposed *BSAS* can ensure sustained survival of sensor barriers. The main contributions of this paper are itemized as follows.

- (1) **Adopting the weakest-first policy for cooperative sensing.** The proposed *BSAS* algorithm calculates the cooperative sensing contribution of each sensor and identifies the bottleneck space-time point which has the weakest surveillance quality. Then the sensor with the largest contribution to the bottleneck space-time point will be prior scheduled, aiming to maximize the minimal quality of the barrier.
- (2) **Proposing the space-time transformation scheme by dynamically adjusting the sensing radius.** This paper dynamically reduces the sensing range of some certain sensors for saving the energy. The saved energies are scheduled to the time slots with the least surveillance quality. This scheme can be treated as the transformation of surveillance contribution from one space-time point to another, which improves the quality of bottleneck space-time point.
- (3) **Maximizing the utilization of sensors.** The proposed *BSAS* algorithm schedules the farthest sensors to improve the surveillance quality of the bottleneck space-time point in each cycle. This can provide more opportunities to those sensors whose locations are far away the boundary curve for participating in the monitoring task. As a result, their utilizations can be improved. Different with the existing studies [5]–[13], the proposed *BSAS* makes full utilizations of all the sensors, instead of just choosing the best sensor to construct the defense barriers.
- (4) **Maintaining the perpetual network lifetime.** The proposed *BSAS* algorithm takes into account the recharging and discharging rate of the sensors and partitions the time line into several cycles. All sensors are scheduled in sensing & recharging and recharging-only states in a way such that the recharged energy can always support the energy consumption in working state in each cycle. Hence it maintains the perpetual lifetime of the WSNs.
- (5) **Considering daytime and nighttime schedules.** The solar-powered sensors need to be recharged in each cycle of the daytime for working in both daytime and nighttime. In literature, study [15] only considered the schedule of daytime. Therefore, the proposed *BSAS* algorithm is more practical in real applications.

The rest of this paper is organized as follows. Section II presents the network environment and problem statement. Section III gives the detailed descriptions of *BSAS* algorithm. Section IV presents the simulation results. Finally, a conclusion of the proposed algorithms and future work are drawn.

II. NETWORK ENVIRONMENT AND PROBLEM STATEMENT

This section introduces the network environment of the considered WSNs. Then, the objectives and constraints of the investigated problem are described.

A. NETWORK ENVIRONMENT

This paper considers a rectangle region M which contains a boundary curve B . The size of M is $L \times W$, where L and W are the length and width of M , respectively. Inside the region M , there were n solar-powered sensors $S = \{s_1, s_2, \dots, s_n\}$ randomly deployed. The sensing radius of each sensor is identical and adjustable with several fixed levels. The energy consumption of each sensor will increase with its sensing radius. Each sensor is aware of its location. Through beacon exchanges with neighboring sensors, each sensor can collect the IDs, locations as well as remaining energies of its neighboring sensors. Figure 1 gives an example of the considered network scenario.

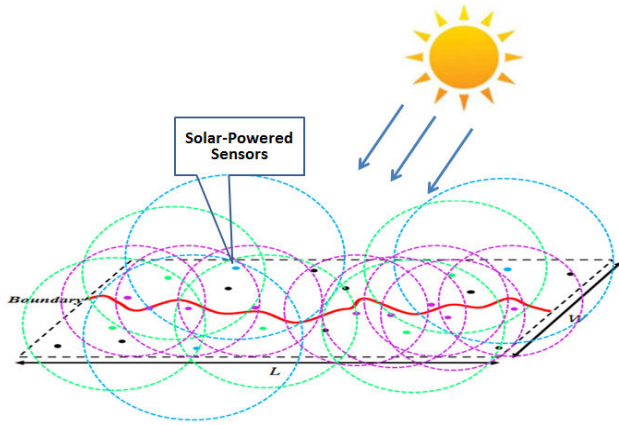


FIGURE 1. An example of the considered WSNs.

B. SENSING MODEL

In this paper, the probabilistic sensing model [18] is applied. In general, the target will be detected by the sensor with a higher probability if it is closer to the sensor. Figure 2 shows the probabilities of different sensing regions of the PSM (Probabilistic Sensing Model).

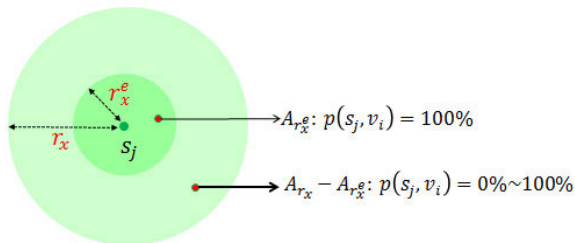


FIGURE 2. The probabilistic sensing model.

Let $R = \{r_1, r_2, \dots, r_q\}$ denote the set of q possible sensing radiuses of each sensor, where $r_i < r_j$ if $i < j$. When a sensor

adopts sensing radius r_x for executing monitoring task, there is a safe sensing radius r_x^e , where $r_x^e < r_x$. Let A_{r_x} denote the area of sensing circle formed by sensing radius r_x . The sensing area A_{r_x} can be divided into two sub-regions, the safe sensing region $A_{r_x^e}$ and the unsafe sensing region $A_{r_x} - A_{r_x^e}$. When an event occurs in the safe region $A_{r_x^e}$ of a sensor, say s_j , the detection probability of this event by sensor s_j is 100%. On the contrary, if an event occurs in the unsafe region $A_{r_x} - A_{r_x^e}$, the detection probability of this event by sensor s_j is reduced when the distance between sensor s_j and the event happening location is larger.

Let v_i denote a given point on a boundary curve. Let $s_{j,x}$ denote sensor s_j using sensing radius r_x , and use notation $s_{j,*}$ to represent the sensor s_j under all sensing radius. Let $d(s_{j,*}, v_i)$ denote the Euclidean distance between sensor s_j and point v_i . Let $p(s_{j,x}, v_i)$ denote the detection probability of the event happened in point v_i by sensor s_j with sensing radius r_x . The detection probability $p(s_{j,x}, v_i)$ is given by the following Exp. (1).

$$p(s_{j,x}, v_i) = \begin{cases} 1 & d(s_{j,*}, v_i) \leq r_x^e \\ e^{-\lambda(d(s_{j,*}, v_i) - r_x^e)^\gamma} & r_x^e < d(s_{j,*}, v_i) \leq r_x \\ 0 & d(s_{j,*}, v_i) > r_x \end{cases} \quad (1)$$

where λ and γ are the path-loss exponents of the sensing signal strength and could be adjusted with the physical properties of a sensor.

C. RECHARGING AND DISCHARGING MODEL

In the considered wireless sensor networks, all sensors are solar-powered. These sensors have four possible states, including recharging-only state, sensing & recharging state, sensing-only and sleep states. A sensor staying in recharging-only state or sensing & recharging state can be recharged from the solar energy resource when its power is not full. In sensing & recharging state, the sensors can perform sensing and recharging operations simultaneously. When a sensor stays in sensing & recharging state or sensing-only state, it will perform the monitoring task and hence consume energy. For simplicity, it is assumed that a sensor staying in sleeping state will not consume its remaining energy. In the daytime, the sensors can be recharged and be activated to work. They can stay in either recharging-only state or sensing & recharging state. In the nighttime, all sensors cannot be recharged and can only stay in sleeping or sensing-only states. Assume that the battery capacity of sensors have two levels, including E and E^{th} . The E is maximum capacity of the battery. That is, the initial battery power of all sensors is E . The E^{th} is the threshold of battery power, which is the basic energy required to support the basic operations of each sensor. This also indicates that the available energy for use is bounded by $E - E^{th}$.

Let u_x^{sen} denote discharging speed of sensor which adopts sensing radius r_x . According to study [19], the discharging speed u_x^{sen} can be measured by Exp. (2).

$$u_x^{sen} = \sigma r_x^2 \quad (2)$$

where σ is a constant and t_x^{sen} denotes the sensing time period that sensor can continuously perform the sensing operation with sensing radius r_x . The value of t_x^{sen} can be measured by the Exp. (3).

$$t_x^{sen} = \frac{E - E^{th}}{u_x^{sen}} = \frac{E - E^{th}}{\sigma r_x^2} \quad (3)$$

Recall that the radius set is $R = \{r_1, \dots, r_x, \dots, r_q\}$. The maximum sensing radius of a sensor is r_q . When sensing radius is adjusted to r_q , the time period for perform sensing operation will be shortest. This time period will be treated as a time slot which is the basic time unit for scheduling. Let notation τ denote a time slot. The value of τ can be calculated by Exp. (4).

$$\tau = t_q^{sen} = \frac{E - E^{th}}{u_q^{sen}} = \frac{E - E^{th}}{\sigma r_q^2} \quad (4)$$

Let u^{chg} denote the recharging speed of a sensor and t^{chg} denote the recharging time period. The value of t^{chg} can be calculated by

$$t^{chg} = \frac{E - E^{th}}{u^{chg}}.$$

Let notation μ_x denote the ratio of recharging and discharging when the sensor adopts sensing radius r_x . It is obvious that μ_q can be calculated by

$$\mu_q = \frac{u_q^{sen}}{u^{chg}} = \frac{t^{chg}}{t_q^{sen}} = \frac{t^{chg}}{\tau}.$$

The solar-powered sensors work intermittently and periodically. A day will be divided into several equal length cycles. Let notation T denote a working cycle of each sensor. A working cycle T consists of several slots in which sensors stay in either sensing & recharging or recharging-only states in the daytime and stay in either sensing-only or sleeping states in the nighttime. The recharged energy in the daytime should be able to support the consumed energy during both daytime and nighttime, because sensors can be recharged only in the daytime. Let L , $L^{daytime}$ and $L^{nighttime}$ denote time lengths of a day, daytime in a day and nighttime in a day. Let ζ^{-1} denote the ratio of daytime to a whole day. That is,

$$\zeta = \left\lceil \frac{L}{L^{daytime}} \right\rceil.$$

A sensor needs to be recharged for a time period $\mu_q \tau$ in the daytime to support the work of a time-slot τ . We have

$$T = \zeta t^{chg} = \zeta \mu_q \tau \quad (5)$$

as shown in Figure. 3. For example, if $L^{daytime} = L^{nighttime}$, the $\zeta = 2$. The battery of each sensor needs to be recharged for $2\mu_q \tau$ in one cycle T of daytime, one $\mu_q \tau$ is for sensing work in the cycle of daytime and the other $\mu_q \tau$ is reserved for supporting the sensing work in a cycle of nighttime. Let notation t_h denote the h -th time slot in a cycle, where $h = 1, 2, \dots, \zeta \mu_q$.

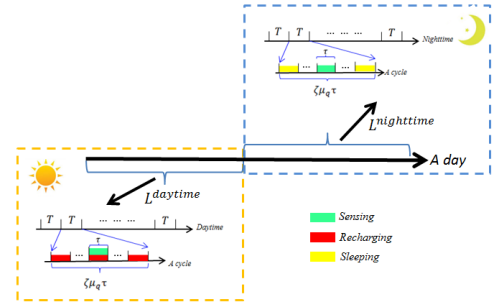


FIGURE 3. A recharging and discharging cycle T .

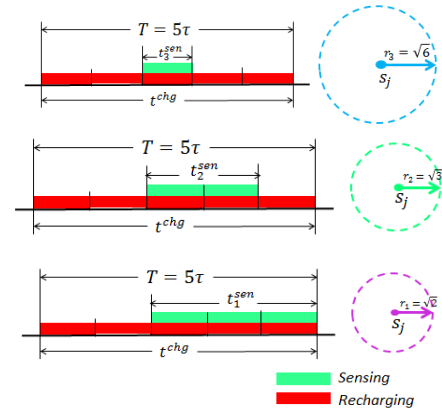


FIGURE 4. An example of the recharging and discharging cycle T and sensing radius r_x .

D. SENSING RADIUS ADJUSTING MODEL

Recall that $R = \{r_1, r_2, \dots, r_q\}$ denotes the set of q possible sensing radiuses of each sensor. If a sensor adopts larger sensing radius, it can contribute higher surveillance quality but consumes more energy, leading to shorter sensing time period t_x^{sen} in each cycle. Let δ_x denote the ratio between sensing time t_x^{sen} and a basic time slot τ , which is represented $t_x^{sen} = \delta_x \times \tau$. The fact of $\tau = t_q^{sen}$ implies $\delta_q = 1$. In this model, $\delta_1 : \delta_2 : \dots : \delta_{q-1} : \delta_q$ is limited to $q : q-1 : \dots : 2 : 1$. That is

$$\delta_x = q - x + 1.$$

According to Exps. (3) and (4), the relation between r_x and r_q can be derived by applying Exp. (6).

$$r_x = \frac{r_q}{\sqrt{\delta_x}} = \frac{r_q}{\sqrt{q - x + 1}} \quad (6)$$

Figure 4 gives an example to illustrate the relation between cycle T and sensing radius r_x . Assume that $q = 3$, $r_3 = \sqrt{6}$, $\zeta = 2$ and $\mu_q = 2.5$. We can calculate the length of each time slot

$$\tau = \frac{E - E^{th}}{6\sigma}.$$

According to the limitation of sensing radius adjusting model, the time length for sensing in a cycle using three different sensing radiuses are

$$t_1^{sen} = \delta_1 \times \tau = 3\tau,$$

$$t_2^{sen} = \delta_2 \times \tau = 2\tau \quad \text{and} \\ t_3^{sen} = \delta_3 \times \tau = \tau.$$

According to Exp. (6), each sensor has three adjustable sensing radius: $r_1 = \sqrt{2}$, $r_2 = \sqrt{3}$ and $r_3 = \sqrt{6}$. The length of a cycle is $T = 2\mu_q\tau = 5\tau$ according to Exp. (5).

E. PROBLEM STATEMENT

This paper presents a scheduling algorithm for the solar-powered sensors which can adjust sensing radius. The proposed scheduling algorithm aims to schedule the sensing radius and the state of each sensor such that the surveillance quality of boundary curve can be maximized. Let a denote one possible scheduling algorithm. The scheduling result applying algorithm a can be represented as a two dimensional matrix D . The value of element $D_{j,h}$ in matrix D represents the scheduling state of sensor s_j in the h^{th} time slot. Each sensor can stay at any of the four possible states: sensing & recharging, recharging-only, sensing-only and sleeping state. The element $D_{j,h}$ of matrix D has the following possible value:

$$D_{j,h} = \begin{cases} -1, & \text{in recharging-only state} \\ 0, & \text{in sleeping state} \\ r_x, & \text{in other states with radius } r_x \end{cases}$$

Figure. 5 gives an example where six sensors are scheduled in two time slots. The scheduling matrix in daytime and nighttime is

$$D = \begin{matrix} & t_1 & t_2 \\ \begin{matrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \end{matrix} & \begin{bmatrix} -1 & r_3 \\ -1 & r_1 \\ r_1 & -1 \\ r_2 & -1 \\ -1 & r_2 \\ r_2 & -1 \end{bmatrix} \end{matrix} \quad \begin{matrix} & t_1 & t_2 \\ \begin{matrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \end{matrix} & \begin{bmatrix} 0 & r_3 \\ 0 & r_1 \\ r_1 & 0 \\ r_2 & 0 \\ 0 & r_2 \\ r_2 & 0 \end{bmatrix} \end{matrix}$$

(a) In the daytime (b) In the nighttime

FIGURE 5. The scheduling matrix.

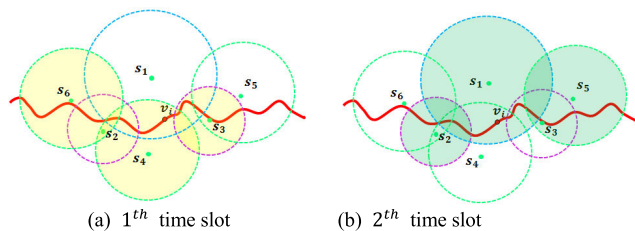


FIGURE 6. An example of sensors scheduled in the 1st and 2nd time slots.

In the matrix, rows represent sensors and the columns represent time slots. The element $D_{3,1} = r_1$ represents that sensor s_3 is scheduled to stay in sensing & recharging state and its sensing radius adopts r_1 in the 1st time slot, as shown in Figure 6(a). The element $D_{1,2} = r_3$ represents that sensor s_1 is scheduled to adopt sensing radius r_3 in 2nd time slot, as shown in Figure 6(b). Similarly, the scheduling results of other sensors are given in Figure 6.

The scheduling of each sensor is cycle-based. That is, each sensor has identical schedule in different cycles. This also indicates that the algorithm only needs to schedule each sensor for one cycle. Let $S_{i,h}^a$ denote the set of sensors which are scheduled by algorithm a to monitor point v_i in the h^{th} time slot. The detection probability $p(s_{j,x}, v_i)$ can be further calculated, according to Exp. (1). In case of $s_j \in S_{i,h}^a$, the sensing radius r_x of s_j is recorded in the element $D_{j,h}$ of the scheduling matrix D . Let $P_{i,h}^a$ denote the surveillance quality of point v_i in the h^{th} time slot by applying scheduling algorithm a . The $P_{i,h}^a$ can be calculated by applying Exp. (7).

$$P_{i,h}^a = 1 - \prod_{s_{j,x} \in S_{i,h}^a} (1 - p(s_{j,x}, v_i)) \quad (7)$$

Given a boundary curve B , the boundary surveillance quality, denoted by U^a , is represented by the weakest surveillance quality of all points on boundary curve B . That is,

$$U^a = \arg \min_{v_i \in B, t_h \in T} P_{i,h}^a \quad (8)$$

The proposed algorithm aims to achieve the goal of maximal surveillance quality, while guaranteeing the perpetual lifetime of WSNs. Let Θ denote set of all possible scheduling algorithms. The following presents the objective function of this work.

Object Function:

$$\text{Max}_{a \in \Theta} (U^a = \min_{v_i \in B, t_h \in T} P_{i,h}^a) \quad (9)$$

Some constraints given below should be further satisfied when developing the scheduling algorithm. Let Boolean notations $\beta_{j,h}^{sen}$, $\beta_{j,h}^{chg}$ and $\beta_{j,h}^{slp}$ denote whether or not sensor s_j performs sensing, recharging and sleeping operations in time slot t_h , respectively. That is

$$\beta_{j,h}^{sen} = \begin{cases} 1, & \text{if } D_{j,h} > 0 \\ 0, & \text{otherwise} \end{cases} \\ \beta_{j,h}^{chg} = \begin{cases} 1, & \text{if } D_{j,h} < 0 \\ 0, & \text{otherwise} \end{cases} \\ \beta_{j,h}^{slp} = \begin{cases} 1, & \text{if } D_{j,h} = 0 \\ 0, & \text{otherwise} \end{cases}$$

The following state constraint should be satisfied. Each sensor can only stay in one of the sensing & recharging and sensing-only states at any given time in the daytime, and can only stay in one of recharging-only and sleeping states at any given time at night.

1) STATE CONSTRAINT

$$(\beta_{j,h}^{chg} + \beta_{j,h}^{sen}) \geq 1 \quad \& \quad (\beta_{j,h}^{chg} + \beta_{j,h}^{slp}) = 1 \quad \& \quad (\beta_{j,h}^{sen} + \beta_{j,h}^{slp}) = 1 \\ \& \quad \left(\sum_h \left\lceil \frac{T}{\tau} \right\rceil \beta_{j,h}^{chg} = \zeta \mu_q \vee \sum_h \left\lceil \frac{T}{\tau} \right\rceil (\beta_{j,h}^{sen} + \beta_{j,h}^{slp}) = \zeta \mu_q \right), \forall j, \forall h \quad (10)$$

The second constraint guarantees that each scheduled sensor should perform sensing operation for at least one time slot and perform recharging operation for at least one time slot. Let S^a denote the set of scheduled sensors by algorithm a . Exp. (11) reflects this requirement.

2) WORKING CONSTRAINT

$$\sum_{h=1}^{\lceil \frac{T}{\tau} \rceil} \beta_{j,h}^{sen} \geq 1, \quad \forall s_j \in S^a \quad (11)$$

The third constraint ensures that the total amount of recharged energy is not smaller than that of discharged energy in each scheduling cycle. This constraint guarantees the perpetual lifetime of each sensor.

3) PERPETUAL LIFETIME CONSTRAINT

$$\sum_{t=1}^{\lceil \frac{L_{daytime}}{T} \rceil} \sum_{h=1}^{\lceil \frac{T}{\tau} \rceil} \beta_{j,h}^{sen} \times u_x^{sen} \leq \sum_{t=1}^{\lceil \frac{L}{T} \rceil} \sum_{h=1}^{\lceil \frac{T}{\tau} \rceil} \beta_{j,h}^{rec} \times u^{chg}, \quad \forall s_j \in S^a \quad (12)$$

Section III further presents the proposed barrier coverage algorithm which schedules the sensors aiming to achieve the goal depicted in (9), while guaranteeing the satisfactory of constraints (10), (11) and (12).

III. THE PROPOSED SCHEDULING SCHEME

The proposed *BSAS* algorithm mainly consists of four phases: *space-time partitioning phase*, *contribution calculation phase*, *sensors scheduling phase* and *space-time transformation phase*. In the first phase, the boundary region is partitioned into several grids such that the boundary curve can be also partitioned into line segments. In addition, this phase also partitions the time line into several equal-length time slots. Through space and time partitions, the two-dimensional space-time points are formed in order to present the surveillance quality of each line segment of boundary curve in each slot. In the second phase, the proposed *BSAS* calculates the space and time contributions of each sensor. Based on each specific sensing radius, the space contribution of each sensor refers to the coverage qualities of each sensor to the line segments of boundary curve while the time contribution refers to the converge time of each sensor to the line segments of boundary curve. In the third phase, in the sensor scheduling phase, sensors are scheduled to perform recharging or cooperatively sensing, aiming at achieving the goal of maximum surveillance quality under the constraint of perpetual lifetime. Finally, the *BSAS* dynamically adjusts the sensing radius of some potential sensors aiming to transfer the surveillance quality from space dimension to time dimension such that the surveillance qualities of the bottleneck points can be further improved.

A. SPACE-TIME PARTITIONING PHASE

This phase consists of two tasks: space and time partitioning tasks. The following describes details of the two tasks.

TASK I: SPACE PARTITIONING TASK

In order to reduce the computational complexity, the *Space Partitioning Task* initially partitions the boundary curve into several line segments. Then the calculation complexity of coverage quality can be transformed from infinite number of points to the fixed number of line segments. The space partitioning task partitions the rectangular areas M into several equal-sized grids. Hence, the boundary curve B in rectangular region M can be further partitioned by these grids. The grids which are passed by the boundary curve can be labeled with an ordered ID from left to right, denoted by $G = \{g_1, g_2, \dots, g_m\}$. A larger grid size can reduce the computational complexity but reduce the calculation accuracy of coverage quality. In the proposed *BSAS* algorithm, the edge length of each grid is set by $\frac{1}{\sqrt{2}}r_1$ where r_1 is the minimum radius in radius set R . This grid size should ensure that any sensor falling in the grid can cover the entire grid, as shown in Figure 7. Let l_k denote each line segment where k denotes the number ordered from left to right. Therefore, the boundary curve B can be represented as the set of line segments. That is, we have $B = \{l_1, l_2, \dots, l_m\}$.

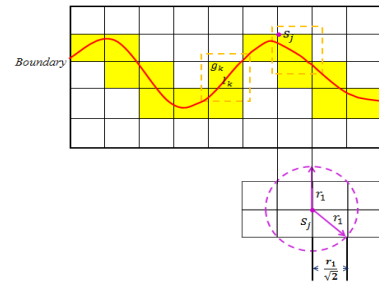


FIGURE 7. Partitioning boundary curve with grid size $\frac{r_1}{\sqrt{2}}$.

TASK II: TIME PARTITIONING

This task mainly partitions the time line into several fixed-length time slots. The length of each time slot is denoted by τ which can be derived by Exp. (4). Then, the length of each cycle T can be calculated according to Exp. (5). Since all cycles have identical schedule, the proposed *BSAS* only needs to schedule the sensors for one cycle. Each cycle consists of ζu_q time slots, where u_q is the recharging and discharging ratio of sensors adopting sensing radius r_q . Therefore, the cycle T can be expressed $T = \{t_1, \dots, t_h, \dots, t_{\zeta u_q}\}$. The Z-axis in Figure 8 depicts the partitioned time line and cycles.

The line segment l_k in time slot h is represented as a space-time point $\alpha_{k,h}$, as shown in Figure 8. The proposed *BSAS* algorithm focuses on improving surveillance quality of each space-time point $\alpha_{k,h}$. The next phase will calculate out contribution of each sensor to boundary curve B in space and time, firstly.

B. CONTRIBUTION CALCULATION PHASE

This phase consists of two tasks: the space contribution calculation and time contribution calculation. The following illustrates details of each task.

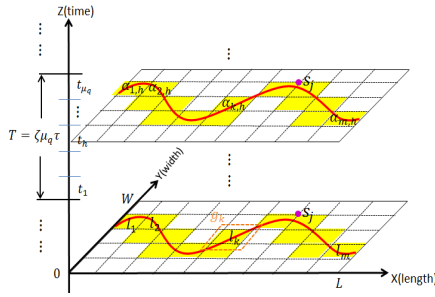


FIGURE 8. The space-time points $\alpha_{k,h}$.

TASK I: SPACE CONTRIBUTION CALCULATION TASK

Assume that sensor s_j covers line segment l_k . The space contribution of sensor s_j to line segment l_k refers to the coverage quality of line segment l_k contributed from sensor s_j . Recall that this paper applies the probability sensing model given in Exp. (1). Therefore, one important feature which impacts the space contribution is the distance between sensor s_j and line segment l_k . That is, a shorter distance between sensor s_j and line segment l_k can result in larger space contribution of s_j to l_k . Another important feature which impacts the space contribution is the sensing radius. That is, sensor s_j adopting larger sensing radius can increase the space contribution of line segment l_k .

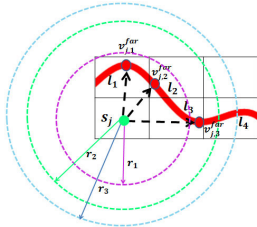


FIGURE 9. The sensing probability of sensor s_j to line segment is calculated by the farthest point from sensor s_j .

Let $c_{j,x,k}^{space}$ denote the space contribution of sensor s_j to line segment l_k at sensing radius r_x . Let $p(s_{j,x}, l_k)$ denote sensing probability of sensor s_j to line segments l_k . The space contribution of sensor s_j to line segments l_k is measured by sensing probability $p(s_{j,x}, l_k)$. Without losing fairness, the farthest point on line segment l_k from sensor s_j is used to calculate $p(s_{j,x}, l_k)$, as shown in Figure 9. Let $v_{j,k}^{far}$ denote the farthest point from sensor s_j to line segment l_k . That is,

$$v_{j,k}^{far} = \arg \max_{v_i \in l_k} d(s_{j,*}, v_i). \quad (13)$$

We have,

$$p(s_{j,x}, l_k) = p(s_{j,x}, v_{j,k}^{far}) \quad (14)$$

and

$$c_{j,x,k}^{space} = p(s_{j,x}, l_k) \quad (15)$$

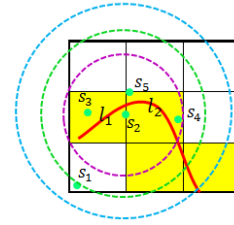


FIGURE 10. Five sensors are deployed around line segments l_1 and l_2 .

Let C_j^{space} denote the space contributions of sensor s_j provided to boundary curve. The C_j^{space} is the space contribution set of sensor s_j to all line segments covered by s_j under sensing radius r_x . That is

$$C_j^{space} = \{c_{j,x,k}^{space} \mid c_{j,x,k}^{space} > 0, l_k \in B \cap r_x \in R\}. \quad (16)$$

In this task, the space contribution $c_{j,k,x}^{space}$ of sensor s_j to line segment l_k should be calculated for each sensor $s_j \in S$.

TASK II: TIME CONTRIBUTION CALCULATION TASK

The time contribution of sensor s_j to boundary curve refers to the number of time slots that sensor s_j can cover boundary curve in a cycle. According to the sensing radius adjusting model, sensors using different sensing radiuses can result in different sensing time. It is obvious that sensors using a smaller sensing radius can consume less energy, leading to longer sensing time in a cycle. According to Exp. (6), the variable δ_x which is a ratio of sensing time t_x^{sen} and a time slot τ can be calculated based on sensing radius r_x . That is, we can obtain the length of t_x^{sen} , which is $\delta_x \tau$. The sink node will evaluate the time contribution of each sensor s_j by calculating the sensing time t_x^{sen} for each sensing radius r_x in a cycle. Let $c_{j,x}^{time}$ denote time contribution of sensor s_j to boundary curve when the sensing radius is set at r_x . The time contribution $c_{j,x}^{time}$ is

$$c_{j,x}^{time} = \delta_x \tau \quad (17)$$

in each cycle $T = \{t_1, \dots, t_h, \dots, t_{\zeta u_q}\}$. Let C_j^{time} denote the set of time contributions that sensor s_j can provide to the boundary curve. We have

$$C_j^{time} = \{c_{j,x}^{time} \mid \forall r_x \in R\} \quad (18)$$

The following considers an example as shown in Figure 10. There are five sensors s_1, s_2, s_3, s_4, s_5 and two line segments l_1 and l_2 . The characteristics of sensors in sensing radiuses, recharging rate and discharging rate are similar as those in Figure 4.

The space contributions of sensors s_1, s_2, s_3 and s_4 are calculated by Exp. (16). The time contributions of s_1, s_2, s_3 and s_4 are calculated by Exp. (18). The Figure 11 depicts the results of contribution calculation.

Until now, the space and time contributions of each sensor s_j to each line segment l_k covered by s_j have calculated.

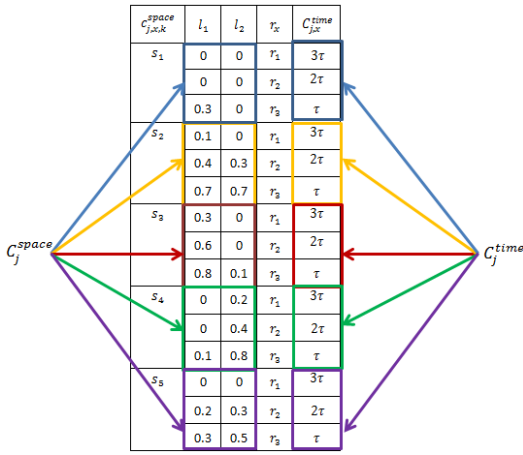


FIGURE 11. Space and time contributions of the five sensors.

The next phase aims to schedule sensors such that the surveillance quality of the boundary curve can maximize.

C. SENSORS SCHEDULING PHASE

The major goal of this phase is to schedule the deployed sensors for achieving the maximal monitoring quality. The idea behind the proposed algorithm is to schedule the proper sensors to monitor the space-time point with the least surveillance quality. These points are called *bottleneck space-time points*. However, the constraints given in Exps. (10), (11) and (12) should be satisfied.

The proposed BSAS mechanism is cycle-based. This indicates that all cycles should have the same schedule. Hence, the scheduling mechanism only pays attention to all space-time points in one cycle. This phase is further partitioned into two tasks. The first task evaluates the surveillance quality of each space-time point and then finds the bottleneck points. The second task aims to schedule the proper sensors for improving the bottleneck space-time points. To achieve this, the dynamic adjustment of sensing radius is proposed. In what follows, the details of the proposed two tasks are presented.

TASK I: IDENTIFICATION OF THE "BOTTLENECK POINTS"

Let S^c denote set of sensors which can cover the boundary curve. That is

$$S^c = \{s_{j,*} \mid d(s_{j,*}, l_k) < r_q\} \quad l_k \in B \quad (19)$$

Similarly, let $\tilde{S}^c$ denote set of sensors which are unable to cover boundary curve. We have,

$$\tilde{S}^c = \{s_{j,*} \mid s_{j,*} \in S \setminus S^c\}.$$

The deployed sensors around boundary curve are divided into two disjoint sensor sets S^c and $\tilde{S}^c$. The following schedule calculation only considers sensors in set S^c .

The process of scheduling sensors in set S^c is divided into two categories: unscheduled sensors and scheduled sensors. Let S^{unsch} denote the set of unscheduled sensors. Initially, we have $S^{unsch} = S^c$. Let $S_{k,h}^{sch}$ denote the set of sensors which

are scheduled in sensing & recharging state in time-slot h and are able to coverage line segment l_k . The goal of scheduling is to arrange all the sensors in S^{unsch} to the corresponding $S_{k,h}^{sch}$ for obtaining the maximum surveillance quality of boundary curve. In order to know the sensors scheduled in each space-time point, the sink node will establish a matrix to store the sensor set $S_{k,h}^{sch}$. Let Φ denote the sensor-scheduling matrix, as shown in Figure 12. Each element $\Phi[k, h]$ stores the scheduled sensor set $S_{k,h}^{sch}$.

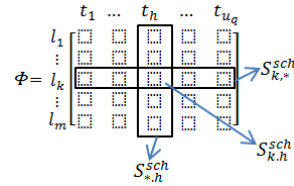


FIGURE 12. The sensor-scheduling matrix.

A row element in the matrix Φ , expressed as $S_{k,*}^{sch}$, is a set of sensors which have been scheduled to take care segment l_k . A column element in the matrix Φ , expressed as $S_{*,h}^{sch}$, is a set of sensors which have been scheduled in time-slot t_h .

Let $u_{k,h}$ denote the surveillance quality of space-time point $\alpha_{k,h}$. The value of $u_{k,h}$ is determined by the cooperative detection probability from each sensor $s_{j,x} \in S_{k,h}^{sch}$. We have

$$u_{k,h} = 1 - \prod_{s_{j,x} \in S_{k,h}^{sch}} (1 - p(s_{j,x}, l_k)) \quad (20)$$

The sink will establish another matrix to store the surveillance quality of each space-time point, which is expressed as U , as shown in Figure 13.

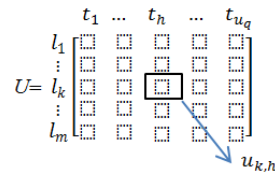


FIGURE 13. The surveillance quality matrix.

The surveillance quality of boundary curve depends on the weakest space-time point. Let $a_{\tilde{k},\tilde{h}}$ denote the bottleneck space-time point. Exp. (21) can be applied to derive each $a_{\tilde{k},\tilde{h}}$.

$$a_{\tilde{k},\tilde{h}} = \arg \min_{1 \leq k \leq m, 1 \leq h \leq \mu_q} u_{k,h} \quad (21)$$

The proposed scheduling algorithm aims to improve the surveillance quality of each $a_{\tilde{k},\tilde{h}}$.

In the following, the line segment $l_{\tilde{k}}$ and time-slot $t_{\tilde{h}}$ of bottleneck time-space point is called as bottleneck line segment and bottleneck time-slot, respectively. In the process of scheduling, the number of bottleneck space-time points might be larger than one. Let A^{wst} denote set of bottleneck space-time points. Initially, no sensors are scheduled. That is,

surveillance quality of each space-time point is zero. We have

$$S^{unsch} = S^c, \Phi = \begin{bmatrix} \emptyset & \dots & \emptyset \\ \vdots & \ddots & \vdots \\ \emptyset & \dots & \emptyset \end{bmatrix} \text{ and } U = \begin{bmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}.$$

In this initial stage, all space-time points are bottleneck space-time points which belong to set A^{wst} . The next task aims to select sensors in S^{unsch} and schedule them for improving the surveillance qualities.

TASK II: SCHEDULING SENSORS BASED ON "BOTTLENECK POINTS"

This task aims to schedule all sensors in set S^{unsch} for improving the surveillance quality. To achieve this, three policies are applied:

- (1) The sensors that can take care of the bottleneck space-time points in set A^{wst} should be scheduled first.
- (2) The sensor that has the farthest distance to the boundary curve should be scheduled first. This policy allows all sensors in set S^{unsch} to have opportunities to contribute their coverage to the boundary curve.
- (3) The farthest sensor adopts the maximal sensing radius r_q to maximize its space contribution.

Let S^b denote the set of sensors that can cover a bottleneck line l_k and is unscheduled. That is

$$S^b = \{s_{j,x} \mid s_{j,*} \in S^{unsch} \cap \exists c_{j,x,\tilde{k}}^{space} > 0, \text{ where } r_x \in R\}.$$

Let $d(s_{j,*}, B)$ denote the distance of sensor s_j to the closest point of boundary curve. The value of $d(s_{j,*}, B)$ adopts the distance of sensor $s_{j,*}$ to the nearest line segment on boundary curve. That is

$$d(s_{j,*}, B) = \min_{l_k \in B} d(s_{j,*}, l_k).$$

Let $s_{j,q}^{far}$ denote the farthest sensor from the boundary curve among all sensors which can cover a bottleneck line segment and are unscheduled. The farthest sensor $s_{j,q}^{far}$ is represented as

$$s_{j,q}^{far} = \max_{s_{j,*} \in S^b} d(s_{j,*}, B). \quad (22)$$

If the number of sensors $s_{j,q}^{far}$ is more than one, the one with the smallest ID of $\hat{j}$ is selected as the farthest sensor.

Let $u_{k,h}^{\hat{j},q}$ denote the surveillance quality of the space-time point $a_{k,h}$ after sensor $s_{j,q}^{far}$ joins the monitoring task. The value of $u_{k,h}^{\hat{j},q}$ can be calculated by applying Exp. (23).

$$u_{k,h}^{\hat{j},q} = 1 - \prod_{s_{a,b} \in S_{k,h}^{sch} \cup \{s_{j,q}^{far}\}} (1 - p(s_{a,b}, l_k)) \quad (23)$$

The surveillance qualities $u_{k,h}^{\hat{j},q}$ of all bottleneck space-time points will recalculate after sensor $s_{j,q}^{far}$ joins to monitor boundary curve according to Exp. (23). The maximal improved bottleneck space-time point, denoted by $a_{\tilde{k},\tilde{h}}^{far}$,

which can have the maximal improvement of surveillance quality from the help of sensor $s_{j,q}^{far}$, will be identified, as shown in Exp. (24).

$$a_{\tilde{k},\tilde{h}}^{far} = \arg \max_{a_{k,h} \in A^{wst}} u_{k,h}^{\hat{j},q} \quad (24)$$

Let $\hat{A}^{wst}$ denote set of space-time point $a_{\tilde{k},\tilde{h}}^{far}$ satisfying Exp. (24). If the number of elements in set $\hat{A}^{wst}$ is more than one, the space-time point $a_{\tilde{k},\tilde{h}}^{far}$ will be further checked to guarantee the maximal improved surveillance quality of all line segments in time $t_{\tilde{h}}$. Let $a_{\hat{k},\hat{h}}$ denote the space-time points left after filtering by Exp. (25). We have

$$a_{\hat{k},\hat{h}} = \arg \max_{a_{k,h} \in \hat{A}^{wst}} \sum_{k=1}^m u_{k,h}^{\hat{j},q} - u_{k,\tilde{h}} \quad (25)$$

Up to now, the BSAS schedules sensor $s_{j,q}^{far}$ in time-slot $t_{\tilde{h}}$ for helping bottleneck space-time point $a_{\hat{k},\hat{h}}$. Next, the sensor set S^{unsch} will be updated as shown in the following.

$$S^{unsch} = S^{unsch} \setminus \{s_{j,*}^{far}\}.$$

Since the sensor $s_{j,q}^{far}$ can improve surveillance quality of other line segments besides $l_{\hat{k}}$ in time slot $t_{\tilde{h}}$, this sensor will also be included in other sensor set $S_{k,h}^{sch}$ if the segment l_k can be covered by $s_{j,q}^{far}$. We have

$$S_{k,h}^{sch} = S_{k,h}^{sch} \cup \{s_{j,q}^{far}\}, \text{ if } c_{j,q,k}^{space} > 0.$$

The surveillance quality $u_{k,h}$ is updated if its sensor set $S_{k,h}^{sch}$ is updated by applying Exp. (20). The new weakest space-time points in set A^{wst} are also updated by applying Exp. (21). The next round will start to find another farthest sensor among all sensors that can help bottleneck space-time points, and then scheduled it in most suitable bottleneck time slot. The task II will be repeatedly executed until no sensor can improve the bottleneck space-time points in set A^{wst} . In other words, we have $S^b = \emptyset$.

$$\Phi = \begin{bmatrix} \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \quad U = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$S^{unsch} = \{s_{1,*}, s_{2,*}, s_{3,*}, s_{4,*}, s_{5,*}\}$$

$$S^b = \{s_{1,*}, s_{5,*}, s_{2,*}, s_{3,*}, s_{4,*}\} \text{ sort by distance from boundary curve}$$

FIGURE 14. Initial values of matrix Φ , U and set S^{unsch} for the example of Figure 10.

Continuously consider the example given in Figure 10. Figure 14 shows the initial values of the matrix Φ , U , and set S^{unsch} .

No sensors are scheduled. All space-time points are bottleneck points in set A^{wst} because each $u_{k,h} = 0$. The first farthest sensor $s_{1,*}$ from boundary curve is selected to improve the surveillance quality of bottleneck space-time point $a_{1,1}$. Then, the matrix Φ , U and set S^{unsch} are updated,

$$\Phi = \begin{bmatrix} \{s_{1,3}\} & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \quad U = \begin{bmatrix} 0.3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$S^{unsch} = \{s_{5,*}, s_{2,*}, s_{3,*}, s_{4,*}\}$$

$$S^b = \{s_{5,*}, s_{2,*}, s_{3,*}, s_{4,*}\} \text{ sort by distance from boundary curve}$$

FIGURE 15. The first farthest sensor $s_{1,3}$ is scheduled in time-slot t_1 for improving the quality of the bottleneck space-time point $a_{1,1}$.

as shown in Figure 15. After the execution of this round, the new bottleneck space-time points in set A^{wst} are formed. The space-time $a_{1,1}$ is no longer a bottleneck space-time.

$$\Phi = \begin{bmatrix} \{s_{1,3}\} & \{s_{5,3}\} & \emptyset & \emptyset & \emptyset \\ \emptyset & \{s_{5,3}\} & \emptyset & \emptyset & \emptyset \end{bmatrix} \quad U = \begin{bmatrix} 0.3 & 0.3 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \end{bmatrix}$$

$$S^{unsch} = \{s_{2,*}, s_{3,*}, s_{4,*}\}$$

$$S^b = \{s_{2,*}, s_{3,*}, s_{4,*}\} \text{ sort by distance from boundary curve}$$

FIGURE 16. The second farthest sensor $s_{5,3}$ is scheduled in time-slot t_2 for improving quality of bottleneck space-time point $a_{2,2}$.

In second round, all space-time points are in A^{wst} except $a_{1,1}$. All sensors in the set S^{unsch} are sensors that can help the bottleneck space-time points. The farthest sensor from the boundary curve in set S^{unsch} is $s_{5,*}$. The sensor $s_{5,3}$ will help bottleneck space-time point $a_{2,2}$ according to Exps. (24) and (25). The Figure 16 shows the scheduling result of second round.

$$\Phi = \begin{bmatrix} \{s_{1,3}\} & \{s_{5,3}\} & \{s_{2,3}\} & \{s_{3,3}\} & \{s_{4,3}\} \\ \emptyset & \{s_{5,3}\} & \{s_{2,3}\} & \{s_{3,3}\} & \{s_{4,3}\} \end{bmatrix} \quad U = \begin{bmatrix} 0.3 & 0.3 & 0.7 & 0.8 & 0.1 \\ 0 & 0.5 & 0.7 & 0.1 & 0.8 \end{bmatrix}$$

$$S^{unsch} = \{\emptyset\}$$

$$S^b = \{\emptyset\} \text{ sort by distance from boundary curve}$$

FIGURE 17. The scheduling result of the last round.

Figure 17 shows the scheduling result of the last round. The five sensors in set S^{unsch} are finally scheduled.

D. SPACE-TIME TRANSFORMATION PHASE

This task aims to locally adjust the scheduled sensors such that the surveillance quality of boundary curve can be further enhanced. The key idea is to transfer the space contribution to time contribution for a scheduled sensor, aiming to improve the weakest bottleneck space-time point. To achieve this, a scheduled sensor can reduce its sensing radius such that it can reserve more energy to monitor for other bottleneck time slot. This further provides an opportunity to improve the surveillance quality of some bottleneck space-time points remained in set A^{wst} .

Let $f_{s_{j,x}}^{st} : S_{*,h}^{sch} \{s_{j,x}\} \rightarrow S_{*,h}^{sch} \{s_{j,x-1}\} + S_{*,\tilde{h}}^{sch} \{s_{j,x-1}\}$ denote a transformation function from space to time applied to sensor $s_{j,x}$. In function $f_{s_{j,x}}^{st}$, the $S_{*,h}^{sch} \{s_{j,x}\}$ denotes the set of sensors scheduled in time-slot t_h and sensor $s_{j,x}$ is in set $S_{*,h}^{sch}$. Since sensor $s_{j,x}$ reduces its sensing range from r_x to r_{x-1} ,

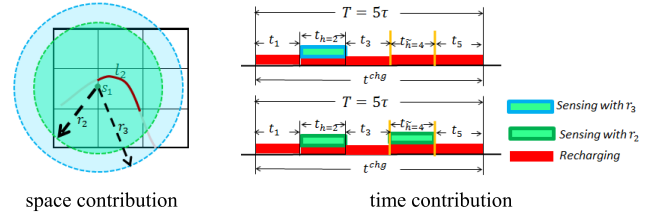


FIGURE 18. Sensor s_1 adjusts its sensing radius from r_3 to r_2 , reducing its space contribution but increasing its time contribution from time slot $t_{h=2}$ to time slots $t_{h=2}$ and $t_{h=4}$.

it reserves more energy to perform monitoring task in another bottleneck time-slot $t_{\tilde{h}}$. Therefore, in addition to the original time-slot t_h , the new time-slot $t_{\tilde{h}}$ is also taken care by sensor $s_{j,x-1}$. The function $f_{s_{j,x}}^{st}$ represents the transformation from $S_{*,h}^{sch} \{s_{j,x}\}$ to $S_{*,h}^{sch} \{s_{j,x-1}\} + S_{*,\tilde{h}}^{sch} \{s_{j,x-1}\}$. If a line segment l_k in time slot $\tilde{h}$ has low surveillance quality, the proposed algorithm BSAS can ask sensor $s_{j,x}$ to perform the operation $f_{s_{j,x}}^{st}$ which reduces its sensing radius from r_x to r_{x-1} and the reserved energy can be used to additionally monitor line segment l_k in time slot $t_{\tilde{h}}$. That is, the transformation function $f_{s_{j,x}}^{st}$ can transfer $S_{*,h}^{sch} \{s_{j,x}\}$ to $S_{*,h}^{sch} \{s_{j,x-1}\} + S_{*,\tilde{h}}^{sch} \{s_{j,x-1}\}$. Figure 18 gives an example that the proposed BSAS aims to adjust the schedule of s_1 for improving the surveillance quality of the bottleneck space-time point $a_{2,4}$. Assume that sensor s_1 can cover line segment l_2 . The proposed BSAS applies $f_{s_1}^{st}$ function to transfer $S_{k=2,h=2}^{sch} \{s_{1,3}\}$ to $S_{k=2,h=2}^{sch} \{s_{1,2}\}$ and $S_{k=2,\tilde{h}=4}^{sch} \{s_{1,2}\}$. That is, sensor s_1 reduces its sensing radius from r_3 to r_2 , which transforms its space contribution to the time contribution, additionally performing the sensing operation at slot $t_{\tilde{h}} = t_4$. As a result, sensor $s_{1,3}$ original in the set $S_{k=2,h=2}^{sch}$ is transformed to $s_{1,2}$ and is additionally included in the sets $S_{k=2,\tilde{h}=4}^{sch}$.

Let $s_{j,x}^{adj}$ denote the scheduled sensor which is asked to adjust its sensing radius by the proposed BSAS. The sensor $s_{j,x}^{adj}$ should satisfy the following three criteria:

- (1) **Helpfulness Property:** Sensor $s_{j,x}^{adj}$ adopts the reduced sensing radius r_{x-1} can cover the bottleneck segment l_k . In other words, the space contribution should satisfy the following condition.

$$c_{j,x-1,\tilde{k}}^{space} > 0$$

- (2) **Non-Bottleneck Pre-Condition Property:** Sensor $s_{j,x}^{adj}$ should not be scheduled in a bottleneck space-time point. Let $t_{\tilde{h}}$ denote a non-bottleneck time slot. The sensor $s_{j,x}^{adj}$ should satisfy the following condition.

$$\forall \alpha_{k,\tilde{h}} \notin A^{wst}$$

Let $u_{k,\tilde{h}}^{j,x-1}$ denote the surveillance quality of space-time point $a_{k,\tilde{h}}$ when a schedule of sensor $s_{j,x}^{adj}$ has been

changed to $s_{j,x-1}^{adj}$. We have

$$u_{k,h}^{j,x-1} = 1 - \prod_{s_{a,b} \in (S_{k,h}^{sch} \setminus \{s_{j,x}\}) \cup \{s_{j,x-1}\}} (1 - p(s_{a,b}, l_k)), \quad \forall l_k \in B \quad (26)$$

- (3) **Non-Bottleneck Post-Condition Property:** Reducing sensing radius of sensor $s_{j,x-1}^{adj}$ should not cause itself as a new bottleneck space-time points. Let U^{bef} denote surveillance quality of boundary curve before adjusting sensing radius of sensor $s_{j,x}^{adj}$. Exp. (27) reflects this property.

$$\min_{c_{j,x,k}^{space} > 0} (u_{k,h}^{j,x-1}) > U^{bef} \quad (27)$$

Exp. (27) indicates that the monitoring qualities of all line segments covered by sensor $s_{j,x}^{adj}$ in time-slot t_h should still be higher than U^{bef} after reducing sensing radius of the sensor $s_{j,x}^{adj}$.

It is noticed that the number of adjustable sensor $s_{j,x}^{adj}$ which satisfies the above three conditions may be more than one. Let S^{adj} denote the set of adjustable sensors. The set S^{adj} needs to be further checked to ensure that the selected $s_{j,x}^{adj}$ has maximum space contribution to bottleneck line segment l_k . That is

$$s_{j,x}^{adj} = \arg \max_{a_{k,h} \in A^{wst}, s_{j,x} \in S_{j,x,k}^{space}} c_{j,x,k}^{space} \quad (28)$$

Up to now, the adjustable sensor $s_{j,x}^{adj}$ can be determined. The surveillance quality of time slots t_h and t_{h-1} will be changed. The operating result of the function $f_{j,x}^{st}$: $S_{*,h}^{sch} \{s_{j,x}^{adj}\} \rightarrow S_{*,h}^{sch} \{s_{j,x-1}^{adj}\} + S_{*,h}^{sch} \{s_{j,x-1}^{adj}\}$ can be obtained. The monitoring qualities of space-time points in time slots t_h and t_{h-1} are recalculated according to Exp.(20). The bottleneck space-time points in A^{wst} are also recalculated. The next round of Task III will again be applied until $S^{adj} = \emptyset$. That is to say, Task III will be finished if there is no adjustable sensor which can help bottleneck space-time points.

Continuously consider the example given in Figure 17. The bottleneck space-time point is $a_{2,1}$. According to *Non-Bottleneck Pre-Condition Property*, the adjustable sensor is in set $S_{2,2}^{sch} = \{s_{5,3}\}$, $S_{2,3}^{sch} = \{s_{2,3}\}$, $S_{2,4}^{sch} = \{s_{3,3}\}$ or $S_{2,5}^{sch} = \{s_{4,3}\}$. Take $s_{2,3}$ as an example. When its sensing radius is reduced to r_2 , it satisfies the *helpfulness property*. That is, it has space contribution to bottleneck line segment l_2 since it satisfies condition $c_{2,2,2}^{space} > 0$. The sensor $s_{2,3}$ has been scheduled in time-slot t_3 . In time-slot t_3 , the monitoring qualities $u_{1,3}^{2,2} = 0.4$ and $u_{2,3}^{2,2} = 0.3$ are still higher than $U^{bef} = 0$ after adjusting $s_{2,3}$ into $s_{2,2}$. That is to say, sensor $s_{2,3}$ satisfies *Non-Bottleneck Post-Condition Property*. The sensor $s_{2,3}$ is included in set S^{adj} . The similar calculations can be applied such that the sensor $s_{5,3}$ is also included in set S^{adj} . Hence, we have $S^{adj} = \{s_{2,3}, s_{5,3}\}$. According to Exp. (28), the sensor that performs the space-time transformation in this

$$S^{adj} = \{s_{5,3}, s_{2,3}\} \xrightarrow{f_{2,3}^{st}, S_{*,3}^{sch} \{s_{2,3}^{adj}\}} S_{*,3}^{sch} \{s_{2,2}^{adj}\} + S_{*,1}^{sch} \{s_{2,2}^{adj}\}$$

$$\Phi = \begin{bmatrix} \{s_{1,3}, s_{2,2}\} & \{s_{5,3}\} & \{s_{2,2}\} & \{s_{3,3}\} & \{s_{4,3}\} \\ \{s_{2,2}\} & \{s_{5,3}\} & \{s_{2,2}\} & \{s_{3,3}\} & \{s_{4,3}\} \end{bmatrix} \quad U = \begin{bmatrix} 0.58 & 0.3 & 0.4 & 0.8 & 0.1 \\ 0.3 & 0.5 & 0.3 & 0.1 & 0.8 \end{bmatrix}$$

FIGURE 19. Adjustment result of shcheduled sensors in the first round.

round is $s_{2,3}$ because its space contribution $c_{2,3,2}^{space}$ to bottleneck line segment l_2 is greater than $c_{5,3,2}^{space}$ of the sensor $s_{5,3}$. The scheduled result of this round is shown in Figure 19.

Figure 20 shows the adjusted result of the second round. The sensor $s_{5,3}$ is adjusted to help bottleneck space-time $a_{2,4}$.

$$S^{adj} = \{s_{5,3}\} \xrightarrow{f_{5,3}^{st}, S_{*,2}^{sch} \{s_{5,3}^{adj}\}} S_{*,2}^{sch} \{s_{5,2}^{adj}\} + S_{*,4}^{sch} \{s_{5,2}^{adj}\}$$

$$\Phi = \begin{bmatrix} \{s_{1,3}, s_{2,2}\} & \{s_{5,2}\} & \{s_{2,2}\} & \{s_{3,3}, s_{5,2}\} & \{s_{4,3}\} \\ \{s_{2,2}\} & \{s_{5,2}\} & \{s_{2,2}\} & \{s_{3,3}, s_{5,2}\} & \{s_{4,3}\} \end{bmatrix} \quad U = \begin{bmatrix} 0.58 & 0.2 & 0.4 & 0.84 & 0.1 \\ 0.3 & 0.3 & 0.3 & 0.37 & 0.8 \end{bmatrix}$$

FIGURE 20. Adjustment result of shcheduled sensors in the second round.

As shown in Figure 21, the task is terminated in last round because $S^{adj} = \emptyset$. No sensor can be adjusted.

$$S^{adj} = \{\emptyset\}$$

$$\Phi = \begin{bmatrix} \{s_{1,3}, s_{2,2}\} & \{s_{3,1}, s_{5,2}\} & \{s_{2,2}\} & \{s_{3,1}, s_{5,2}\} & \{s_{3,1}\} \\ \{s_{2,2}, s_{4,2}\} & \{s_{5,2}\} & \{s_{2,2}\} & \{s_{5,2}\} & \{s_{4,2}\} \end{bmatrix} \quad U = \begin{bmatrix} 0.58 & 0.36 & 0.4 & 0.36 & 0.3 \\ 0.58 & 0.3 & 0.3 & 0.3 & 0.4 \end{bmatrix}$$

FIGURE 21. Adjustment result of shcheduled sensors in the last round.

IV. SIMULATION

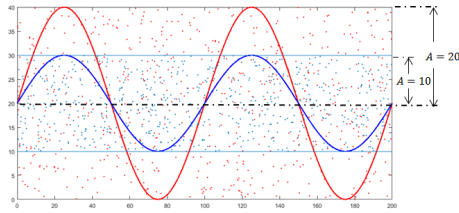
This section measures the performance improvements of the proposed BSAS against the existing Maximizing Surveillance Quality Mechanism (MSQ). The proposed BSAS algorithm has two important steps: scheduling new sensors for bottleneck space-time points and adjusting sensing radius of scheduled sensors for bottleneck space-time points. In the first step, different policies can be employed. There are two policies can be employed: the *farthest-first* and the *nearest-first* policies. This paper adopts the *farthest-first* policy which prior schedules the sensor farthest to the boundary curve, aiming to fully utilize the sensors. The proposed BSAS applying the *farthest-first* policy is called as *BSAS_Far*. On the contrary, the proposed BSAS applying the *nearest-first* policy which prior schedules the nearest sensor is called *BSAS_Near*. The *BSAS_Far* and *BSAS_Near* algorithms are compared in simulation. The existing MSQ only schedules the sensors without adjusting the sensing range. The following firstly presents the simulation model and then discusses the simulation results.

A. SIMULATION MODEL

The simulation parameters are given in Table 1. Five boundaries with different amplitudes are considered in a rectangle monitoring area as shown. The length L of rectangle monitoring area is 200m. The width $W = 2 \times A$ varies depending on the value of A , which varies ranging from 20m to 100m. The number of sensors is varied from 200 to 1000. These sensors

TABLE 1. Simulation settings.

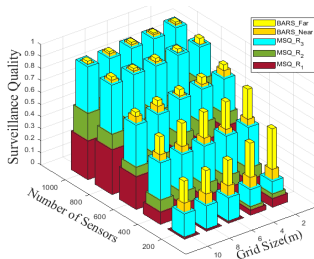
Parameters	Values
The Length Of Monitoring Area	200m
The amplitude	10~50m
The Width Of Monitoring Area	20m~100m
Number of Sensors	200 – 1000
Sensing Radius R_3	5m~25m
Communication Radius	Twice of sensing radius
Deployment	Randomly
Grid Size	2m~10m
Recharging Rate	20 units /hour
Discharging Rate	80 units/hour

**FIGURE 22.** The boundaries with amplitudes $A = 10$ and $A = 20$.

are randomly deployed in monitoring area. Figure 22 depicts two boundaries of amplitudes 5m and 10m.

Each sensor has three types of sensing radius: R_1 , R_2 and R_3 . The relationship between R_1 , R_2 and R_3 is $R_2 = \frac{1}{\sqrt{2}}R_3$ and $R_1 = \frac{1}{\sqrt{3}}R_3$. The value of R_3 varies ranging from 5m to 20m. In the experiments, the existing *MSQ* [17] adopting sensing radiuses R_1 , R_2 and R_3 are called as *MSQ_R1*, *MSQ_R2* and *MSQ_R3*, respectively. The proposed *BSAS* dynamically adjusts the sensing radius aims to improve the surveillance quality of the bottleneck space-time point. The larger sensing radius can achieve greater space contribution but smaller time contribution.

The communication radius is as long as the twice of the sensing radius. The edge of each grid varies from 2m to 10m. The rates for energy recharging and discharging are 20 units/hour and 80 units/hour, respectively.

**FIGURE 23.** Performance comparison of *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1* in terms of surveillance quality by deploying different numbers of sensors and varying different grid sizes.

B. SIMULATION RESULTS

Figure 23 compares the *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1* in terms of surveillance quality.

The number of deployed sensors varies ranging from 200 to 1000 while the grid size varies ranging from 2m to 10m. The amplitude is 10m and sensing radius R_3 is 20m. As shown in Figure 23, a common trend that the surveillance qualities increase with number of sensors. This occurs because more sensors can increase the number of active sensors in each time slot, resulting in higher surveillance qualities. Another trend found in Figure 23 is that the surveillance quality decreased with the grid size. This occurs because that a large grid results in a long line segment, which reduces the number of sensors enabling to fully cover the segment. In comparison, the proposed *BSAS* algorithms, including *BSAS_Far* and *BSAS_Near*, outperform the existing *MSQ* algorithms, including *MSQ_R3*, *MSQ_R2* and *MSQ_R1*. The existing *MSQ* algorithms just schedule sensors to take care of bottleneck space-time points. Unlike *MSQ* algorithm, the proposed *BSAS* algorithms further dynamically adjust the sensing ranges to improve surveillance quality of bottleneck space-time point. As a result, the proposed *BSAS_Far* and *BSAS_Near* algorithms achieve better performance, as compared with the existing *MSQ_R3*, *MSQ_R2* and *MSQ_R1* algorithms.

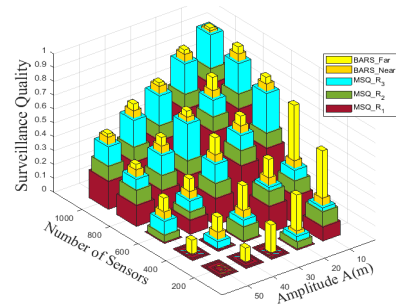
**FIGURE 24.** Performance comparison of *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1* in terms of surveillance quality by considering different amplitude boundaries and deploying different numbers of sensors.

Figure 24 compares the proposed algorithms, including *BSAS_Far* and *BSAS_Near*, and the existing *MSQ_R1*, *MSQ_R2* and *MSQ_R3* in terms of the surveillance quality. The number of sensors varies ranging from 200 to 1000 while the amplitude of boundary curves varies ranging from 10m to 50m. The sensing radiuses of R_1 , R_2 and R_3 are $\frac{20}{\sqrt{3}}$ m, $\frac{20}{\sqrt{2}}$ m and 20m, respectively. The grid size is set at 2m. As shown in Figure 24, a common trend that the surveillance qualities significantly decrease with the amplitude of boundary curve. This occurs because of two reasons. One is that a larger amplitude leads to a large monitoring area where the number of deployed sensors is a constant. This causes that the deployment density of sensors to be small, resulting in a low surveillance quality. Another reason is that a large amplitude of boundary increases the length of boundary curve. As a result, the number of segments is increased, which requires more sensors to maintain the surveillance quality. In comparison, the proposed *BSAS_Far* achieves the

best performance. This occurs because *BSAS Far+Adj* algorithm employs farthest-first policy, which increases the sensor utilization. The algorithms, including *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1*, schedule sensors based on the largest contribution first policy, which firstly schedule the nearest sensor to be active.

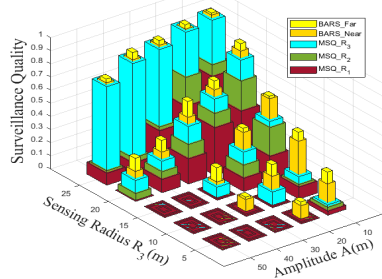


FIGURE 25. Performance comparison of *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1* in terms of surveillance quality by considering different amplitude boundaries and varying sensing radius.

Figure 25 compares the proposed algorithms, including *BSAS_Far* and *BSAS_Near*, and the existing *MSQ_R1*, *MSQ_R2* and *MSQ_R3* in terms of the surveillance quality. The sensing radius varies ranging from 5m to 25m. The number of sensors is set at 600 and grid size is set at 2m. As shown in Figure 25, a common trend that the surveillance qualities significantly increase with sensing radius. This occurs because that the space contribution of sensors to the boundary curve increases with sensing radius. In comparison, the proposed *BSAS* algorithms, including *BSAS_Far* and *BSAS_Near*, outperform the existing *MSQ* algorithms, including *MSQ_R3*, *MSQ_R2* and *MSQ_R1* under the same amplitude. This occurs because *BSAS* algorithms can take complementary advantages of time and space. That is, the proposed *BSAS* algorithms further adjust sensing radius of some sensors to improve the surveillance quality of the bottleneck space-time points.

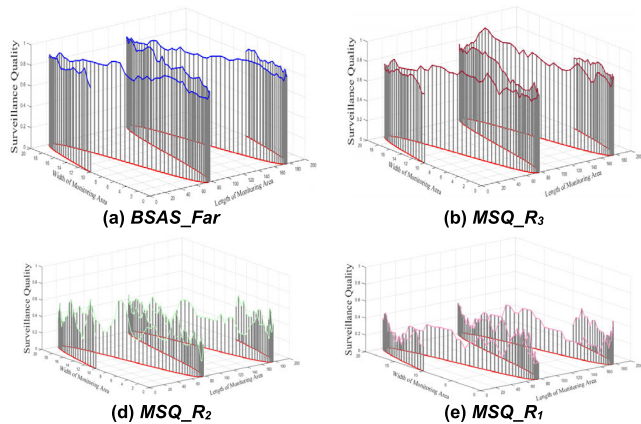


FIGURE 26. Surveillance quality of each location on whole boundary curve with amplitude 10m by applying *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1* algorithms.

Figures 26 investigate surveillance qualities of each location of boundary curve where the amplitude is set at 10m. The surveillance qualities of *BASA_Far*, *MSQ_R3*, *MSQ_R2*

and *MSQ_R1* algorithms are shown in Figures 26 (a), (b), (c) and (d), respectively. There are 600 sensors deployed in $20\text{m} \times 200\text{m}$ rectangle monitoring area. The sensing radius of each sensor is 20m. The proposed *BASA_Far* algorithm steady keep a high surveillance quality without large fluctuation, as compared with the other three compared algorithms. It implies that the proposed *BSAS_Far* achieves better stability of surveillance qualities. The main reason is that the proposed *BSAS_Far* adopts space-time transformation operations to further improve the surveillance qualities.

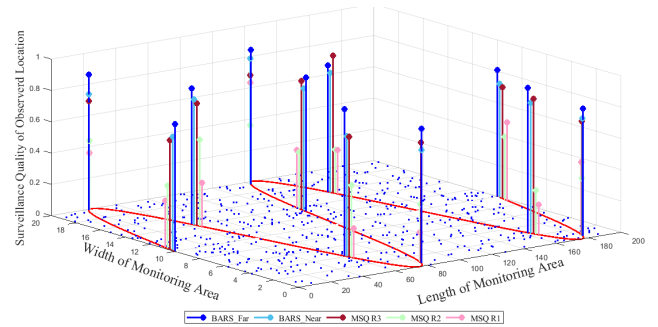
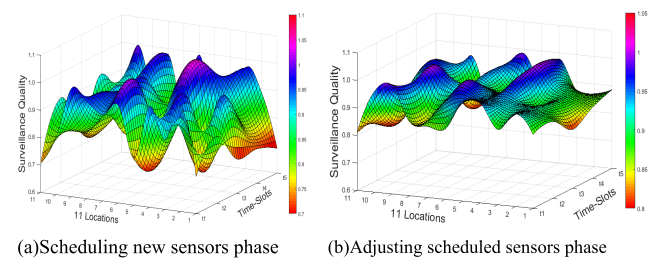


FIGURE 27. Surveillance quality comparison of randomly selecting 11 locations for boundary curve with amplitude 10m by applying *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1* algorithms.

Figure 27 investigates the surveillance qualities of the compared algorithms in Figure 26 by randomly selecting 11 locations. In comparison, the proposed *BSAS_Far* algorithm outperforms *MSQ_R3*, *MSQ_R2* and *MSQ_R1* in all cases in terms of the surveillance qualities. This occurs because the proposed *BSAS_Far* algorithm employs the space-time transformation operation which further improve the weakest space-time points.



(a) Scheduling new sensors phase (b) Adjusting scheduled sensors phase

FIGURE 28. Surveillance quality of 11 selected locations for 5 time slots. The comparison of *BASA_Far* algorithm with and without applying space-time transformation phase.

Figures 28 (a) and (b) further observe the surveillance qualities of selected space-time points before and after applying the space-time transformation operations of the proposed *BSAS_Far* algorithm, respectively. The observed time period includes one cycle which consists of five time slots. There are 11 selected locations of the boundary curve to be observed. As shown in Figure 28 (a), the highest and lowest surveillance qualities are 0.69 and 0.999, respectively. This result is obtained only applying the *sensors scheduling phase* of the proposed algorithm. Figure 28 (b) depicts the improved

surveillance quality after applying the *space-time transformation phase* of the proposed algorithm. The space-time transformation further adjusts the sensing radius of the scheduled sensors according to the bottleneck space-time points. As a result, the highest and lowest surveillance qualities are 0.814 and 1, respectively. In comparison, the space-time transformation operations, as proposed in task III, further improve 12.4% surveillance quality of the lowest surveillance quality.

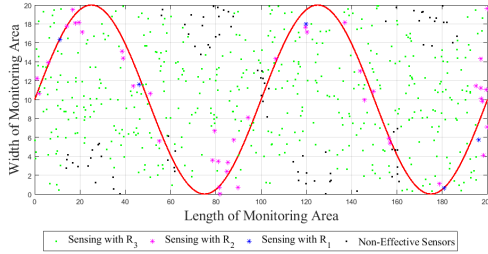


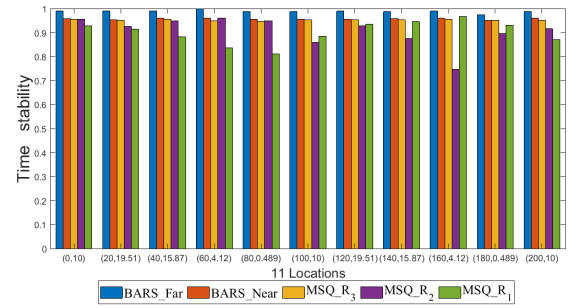
FIGURE 29. The distribution of scheduled sensors in the proposed *BSAS_Far* algorithm.

Figure 29 further observes the scheduling details of experimental result obtained from Figure 28. The black dots represent unscheduled sensors in the experiment of Figure 28. The rest are scheduled sensors. The scheduled sensors consist of three types with different sensing radiuses. The sensors working at sensing radius R_1 , R_2 and R_3 are marked with blue, pink and green dots, respectively. The sensors adopting sensing radius R_1 , R_2 and R_3 are active for 3, 2 and 1 time-slots, respectively. Initially, the proposed algorithm schedules sensors with sensing radius R_3 . Then, the sensing radiuses of the sensors, marked with blue and pink colors, are adjusted to R_2 and R_1 in adjusting sensing radius of scheduled sensors phase. As shown in Figure 29, those sensors which are adjusted their sensing radiuses are located close to the boundary curve. This occurs because these sensors originally have larger space contribution and thus can be adjusted to transform the space contribution to time contribution by reducing their sensing radiuses. In other words, the sensors can save energy aiming at increasing the sensing time by reducing their sensing ranges.

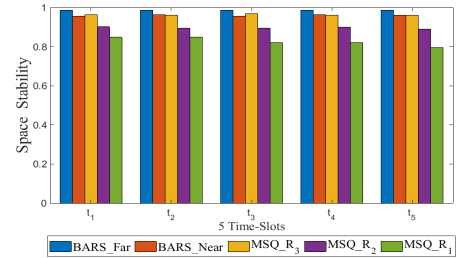
Figures 30 (a) and (b) further compare the *time stability* and *space stability*, respectively, of surveillance quality of the five compared algorithms. Figure 30(a) shows the performance of 11 selected locations, in terms of *time stability* of surveillance quality, are observed for $n = 5$ time slots. Let $u_{i,j}$ denote the surveillance quality of the i -th location at the j -th time slot. Let ξ_i be the time stability of the i -th location, which can be measured by the following Exp. (29).

$$\xi_i = \frac{\left(\sum_{j=1}^n u_{i,j}\right)^2}{n \sum_{j=1}^n u_{i,j}^2}. \quad (29)$$

A value of ξ_i closer to 1 represents that the surveillance qualities of the i -th location are similar (or stable) for $n=5$ time slots.



(a) Time stability of each observed location (ξ_i , for $1 \leq i \leq 11$).



(b) Space stability of each observed time-slot (ξ_j , for $1 \leq j \leq 5$).

FIGURE 30. Time and Space stabilities comparison of observed 11 locations and 5 time-slots in *BSAS_Far*, *BSAS_Near*, *MSQ_R3*, *MSQ_R2* and *MSQ_R1*.

Similarly, let ξ_j denote the space stability of the j -th time slot on m locations, where ξ_j is defined by Exp. (30).

$$\xi_j = \frac{\left(\sum_{i=1}^m u_{i,j}\right)^2}{m \sum_{i=1}^m u_{i,j}^2}. \quad (30)$$

Figure 30 (b) further compares the space stability of each time slot.

In comparison, the proposed *BSAS_Far* achieves the best performance and almost achieves 1, in terms of time and space stabilities. This occurs because that some sensors are adjusted their sensing range to enhance the qualities of the bottleneck points. This help further balance the surveillance qualities of all space-time points.

V. CONCLUSIONS

This paper presents a centralized barrier coverage algorithm, called *BSAS*, which schedules the solar-powered sensors aiming at maximizing the surveillance quality of a given boundary curve. The sink schedules sensors by applying the proposed *BSAS* algorithm and distributes the scheduling results to each sensor node. The proposed *BSAS* applies the probabilistic sensing model to evaluate the cooperative sensing contribution of each sensor and identify the bottleneck space-time points during the construction process of the barriers. Then the space-time transformation scheme, which adjusts the sensing radiuses of the sensors with the largest quality contribution and then schedules them to monitor those points, aims to maximize the surveillance quality of these bottleneck points. Experimental experiments show that the proposed *BSAS* outperforms the existing studies in terms of

surveillance quality and stability. Future work will further consider the communication protocol to report the emergent events when the constructed barriers detect the intruders. Another work will extend the WSNs to the heterogeneous WSNs which consists of different types of sensors.

REFERENCES

- [1] C.-T. Chang, C.-Y. Chang, S. Zhao, J.-C. Chen, and T.-L. Wang, "SRA: A sensing radius adaptation mechanism for maximizing network lifetime in WSNs," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9817–9833, Dec. 2016.
- [2] M. Li, W. Cheng, K. Liu, Y. He, X. Li, and X. Liao, "Sweep coverage with mobile sensors," *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1534–1545, Nov. 2011.
- [3] M. Cardei and J. Wu, "Energy-efficient coverage problems in wireless ad hoc sensor networks," *Comput. Commun.*, vol. 29, no. 4, pp. 413–420, Feb. 2006.
- [4] S. Kumar, T. Lai, and A. Arora, "Barrier coverage with wireless sensors," in *Proc. ACM MobiCom*, Cologne, Germany, Aug. 2005, pp. 284–298.
- [5] B. Liu, O. Dousse, J. Wang, and A. Saipulla, "Strong barrier coverage of wireless sensor networks," in *Proc. 9th ACM Int. Symp. Mobile ad hoc Netw. Comput. (MobiHoc)*, Hong Kong, May 2008, pp. 411–420.
- [6] S. Silvestri and K. Goss, "MobiBar: An autonomous deployment algorithm for barrier coverage with mobile sensors," *Ad Hoc Netw.*, vol. 54, pp. 111–129, Jan. 2017.
- [7] C.-I. Weng, C.-Y. Chang, C.-Y. Hsiao, C.-T. Chang, and H. B. Chen, "On-supporting energy balanced k-barrier coverage in wireless sensor networks," *IEEE Access*, vol. 6, pp. 13261–13274, 2018.
- [8] X. Zhang, M. L. Wymore, and D. Qiao, "An iterative method for strong barrier coverage under practical constraints," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Lumpur, Malaysia, May 2016, pp. 1–7.
- [9] X. Fan, F. Hu, T. Liu, K. Chi, and J. Xu, "Cost effective directional barrier construction based on zooming and united probabilistic detection," *IEEE Trans. Mobile Comput.*, early access, Apr. 23, 2019, doi: 10.1109/TMC.2019.2912749.
- [10] V. Akbarzadeh, C. Gagne, M. Parizeau, M. Argany, and M. A. Mostafavi, "Probabilistic sensing model for sensor placement optimization based on Line-of-Sight coverage," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 2, pp. 293–303, Feb. 2013.
- [11] H. Mostafaei, M. U. Chowdhury, and M. S. Obaidat, "Border surveillance with WSN systems in a distributed manner," *IEEE Syst. J.*, vol. 12, no. 4, pp. 3703–3712, Dec. 2018.
- [12] Z. Zhang, W. Wu, J. Yuan, and D.-Z. Du, "Breach-free sleep-wakeup scheduling for barrier coverage with heterogeneous wireless sensors," *IEEE/ACM Trans. Netw.*, vol. 26, no. 5, pp. 2404–2413, Oct. 2018.
- [13] D. Kim, W. Wang, J. Son, W. Wu, W. Lee, and A. O. Tokuta, "Maximum lifetime combined barrier-coverage of weak static sensors and strong mobile sensors," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1956–1966, Jul. 2017.
- [14] W.-K. Lee, M. J. W. Schubert, B.-Y. Ooi, and S. J.-Q. Ho, "Multi-source energy harvesting and storage for floating wireless sensor network nodes with long range communication capability," *IEEE Trans. Ind. Appl.*, vol. 54, no. 3, pp. 2606–2615, May 2018.
- [15] S. Tang, X.-Y. Li, X. Shen, J. Zhang, G. Dai, and S. K. Das, "Cool: On coverage with solar-powered sensors," in *Proc. IEEE ICDSC*, Minneapolis, MN, USA, Jun. 2011, pp. 488–496.
- [16] H. Chen, X. Li, and F. Zhao, "A reinforcement learning-based sleep scheduling algorithm for desired area coverage in solar-powered wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 8, pp. 2763–2774, Apr. 2016.
- [17] Z. Dong, C.-Y. Chang, G. Chen, I.-H. Chang, and P. Xu, "Maximizing surveillance quality of boundary curve in solar-powered wireless sensor networks," *IEEE Access*, vol. 7, pp. 77771–77785, 2019.
- [18] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," *ACM Trans. Embedded Comput. Syst.*, vol. 3, no. 1, pp. 61–91, Feb. 2004.
- [19] J. Wang and S. Medidi, "Energy efficient coverage with variable sensing radii in wireless sensor networks," in *Proc. 3rd IEEE Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, White Plains, NY, USA, Oct. 2007, p. 61.



ZAIXIU DONG received the B.S. degree in computer science and technology from Anhui Agricultural University, Hefei, China, in 2003, and the M.E. degree in computer technology from the Hefei University of Technology, Hefei, in 2013. She is currently a Lecturer with the School of Computer and Information Engineering, Chuzhou University, Chuzhou, Anhui, China. Her current research interests are in intelligent control technologies, wireless sensor networks, and healthcare for the elderly.



CUIJUAN SHANG received the B.S. degree in biomedical engineering from Zhengzhou University, Zhengzhou, China, in 2008, and the M.S. degree in biomedical engineering from Shandong University, Jinan, China, in 2011. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Information Engineering, Tamkang University, New Taipei, Taiwan. She is currently a Lecturer with the School of Computer and Information Engineering, Chuzhou University, Chuzhou, China. Her current research interests include wireless sensor network and healthcare.



CHIH-YUNG CHANG (Member, IEEE) received the Ph.D. degree in computer science and information engineering from National Central University, Taiwan, in 1995. He is currently a Full Professor with the Department of Computer Science and Information Engineering, Tamkang University, New Taipei City, Taiwan. His current research interests include the Internet of things, wireless sensor networks, artificial intelligence, and deep learning. He has served as an Associate Guest Editor for several SCI-indexed journals, including the *International Journal of Ad Hoc and Ubiquitous Computing* (IJAHUC), from 2011 to 2019, the *International Journal of Distributed Sensor Networks* (IJDSN), from 2012 to 2014, *IET Communications*, in 2011, *Telecommunication Systems* (TS), in 2010, the *Journal of Information Science and Engineering* (JISE), in 2008, and the *Journal of Internet Technology* (JIT), from 2004 to 2008.



DIPTENDU SINHA ROY received the Ph.D. degree in engineering from the Birla Institute of Technology, Mesra, India, in 2010. He joined the Department of Computer Science and Engineering, National Institute of Technology (NIT) Meghalaya, India, as an Associate Professor, in 2016, where he has been the Chair with the Department of Computer Science and Engineering, since January 2017. Prior to his stint at NIT Meghalaya, he was with the Department of Computer science and Engineering, National Institute of Science and Technology, Berhampur, India. His current research interests include software reliability, distributed and cloud computing, and the IoT, specifically applications of artificial intelligence and machine learning for smart integrated systems. He is a member of the IEEE Computer Society.

...