IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 17, 2018, accepted September 10, 2018, date of publication September 24, 2018, date of current version November 30, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2871963

Greedy Game Algorithms for Solving SET *K*-Cover Problem in HWSNs

WENJIE YAN, MENGJING CAO, YOUXI WU, (Member, IEEE), AND JUN ZHANG
[1]School of Artificial Intelligence, Hebei University of Technology, Tianjin 300401, China
[2]Hebei Province Key Laboratory of Big Data Calculation, Tianjin 300401, China

Corresponding author: Youxi Wu (wuc@scse.hebut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702157, in part by the Natural Science Foundation of China under Grant 51805141, in part by the Science and Technology Program of Hebei Province of China under Grant 15210506, and in part by the Natural Science Foundation of Tianjin City under Grant 16JCQNJC00400.

ABSTRACT Coverage is a fundamental problem in heterogeneous wireless sensor networks (HWSNs). Lifetime of the HWSNs is another important problem in this area. The K-Cover problem can solve both the coverage and lifetime issues. Especially, the lifetime of HWSNs can be prolonged by solving the SET K-Cover problem through partitioning the sensors into K sets. However, the SET K-Cover problem has not been widely researched in HWSNs. Hence, we introduce three kinds of simple pure distributed algorithms named greedy game algorithms (GGAs) based on game theory for solving the SET K-Cover problem in this paper, in which we consider the SET K-Cover problem as a multi-stage game process. The sensors in HWSNs are considered as the players, the cover sets chosen by N sensors as strategies, and the sensing area alone as the payoff function for each sensor. During the game process, each sensor wants to selfishly choose one of the K-covers at each iteration, with which it can maximize its payoff value. After the game process, the best strategies all the players have chosen constitute the Nash Equilibrium. Finally, extensive experiments have been conducted to show the superiority in coverage, convergence, robustness, and network design of the proposed GGAs comparing with other existing algorithms.

INDEX TERMS Greedy game algorithms (GGAs), SET K-Cover, heterogeneous wireless sensor networks (HWSNs), Nash equilibrium.

NOMENCLATURE

Acronyms

HWSNs	heterogeneous wireless sensor networks.
GGAs	greedy game algorithms.
WSNs	wireless sensor networks.
NPCGA	n person card game algorithm.
RA	random algorithm.
DKCA	distributed K-Cover algorithm.
SNECA	synchronous nash equilibria convergence algorithm.
MA	memetic algorithm.
HR	hit rate
TVLLA	time variant log-linear learning algorithm.
IDBGGA	identification based greedy game algorithm.
PRBGGA	pure random based greedy game algorithm.
TBGGA	time based greedy game algorithm.

Notations

N	The number of sensors in wireless sensor networks.
F	The rectangular area of the sensing field, which can be expressed as $F = p \times q$ grids.
R_s	The sensing radius of each sensor.
R_c	The communication radius of each sensor.
K	The number of disjoint cover sets can be partitioned in WSNs.
s_i	Denotes the cover set sensor i belongs to.
s_{-i}	A strategy vector chosen by all the other sensors except the i^{th} sensor.
i	the symbol of the one of the sensors.
$-i$	This symbol is used to represent all sensors in Δ except i itself.
ξ	Profile of the K-Cover sets, where $\xi = \{1, \ldots, K\}$.
s	A partition of N sensors into K-Cover sets, where $s = \{s_1, s_2, \ldots, s_N\}$.
S	The set of all possible K Covers, and the cardinality $\lvert S \rvert = K^N$.
R_i	The actual region covered by sensor i.
$c_j(s)$	The coverage by one of the coverage set $c_j(s)$.
$c(s)$	The average coverage ratio of the whole WSNs.
Δ	The players set constituted by N sensors.
V_i	The region covered alone by sensor i.
ΔV_i	The regret value of sensor i.

$N_{eig}(i)$	The neighbors of sensor i.
$d(i, W)$	The Euclidean distance between sensor i and the sensing point W.
$\eta_{xy}(i)$	The binary sensing model of a grid point W by the sensor i.
s^*	The Nash Equilibrium profile strategy of a non-cooperative game, which can be denoted as $s^* = \{s_1^*, \ldots, s_N^*\}$.
$s^*_{N_{eig}}(i)$	The Nash Equilibrium profile strategy of the neighbors of sensor i.
$N_{eig1}(i)$	The equal neighbors of sensor i.
$N_{eig2}(i)$	The explicit neighbors of sensor i.
$N_{eig3}(i)$	The potential neighbors of sensor i.
$N_{eig4}(i)$	The blind neighbors of sensor i.

I. INTRODUCTION

Coverage problem is a fundamental and important issue in wireless sensor networks(WSNs). In recent years, many researchers have focused on this key issue [1], [2], [4]–[6]. Lifetime of the WSNs is another crucial point. Sensors are often randomly deployed in the harsh area by air plane or other devices, so prolong the lifetime of WSNs is a crucial work and lots of researchers have paid attention to this problem [8], [9], [11]. Prolonging the lifetime as well as the coverage maximization of WSNs is a valuable but hard work to be done. Prior researchers have focused on this area, such as [14] and [15].

One efficient method to solve the coverage and lifetime maximization problem in WSNs is to divide the sensors into K sets and design an efficient scheduling algorithm to manage the sensors of K sets in a round-robin pattern during working stage. After that, the lifetime of the whole sensor networks can be extended up to K times. Prior researches on K coverage problem has taken two approaches. In the first, researchers consider each point of the monitored area is covered at least by K sensors, and design the K coverage algorithms to achieve the K-Covered property and prolong the lifetime of the sensor networks [3], [7], [10], [12], [13], [16]. In the second approach, all the deployed sensors are divided into K sets, and the critical issue is to design an efficient scheduling algorithm to schedule the K-group sensors. Finally, the lifetime of the sensor networks can be extended to K times while the average coverage ratio also can be maximized at the same time [17]–[19], [21]. The latter approach is more natural since we can not promise each point of the sensing field is K-Covered due to the sensors' random scattered property. Besides, the energy consumption is unbalanced if some sensing point is less than K-Covered. The latter approach is called the SET K-Cover problem [17]. We give a brief introduction to the SET K-Cover problem in HWSNs. As stated in the former part, the main purpose of solving the SET K-Cover problem is to maximize the average coverage ratio of all the HWSNs under the predefined network lifetime guarantee. In SET K-Cover problem, we mainly want to fully utilize the redundancy of the HWSNs and divide all of the sensors into K-Cover sets. For any predefined time slot, only the sensors in one of the K-Cover sets work, and the rest of the sensors sleep for saving energy. Following the latter approach, in our former research paper [20], a distributed n person card game algorithm(NPCGA) was proposed to solve the SET K-Cover problem in WSNs. By adopting NPCGA, the lifetime of WSNs is up to K times compared with the original random sensor deployment strategy, while the average coverage ratio can be maximized at the same time.

In this paper, we extend the SET K-Cover problem to the heterogeneous wireless sensor networks(HWSNs) based on our previous research. Further more, we introduce a heterogeneous sensing model and propose three kinds of efficient pure distributed algorithms named greedy game algorithms(GGAs) to solve the SET K-Cover problem in HWSNs. To the best of our knowledge, the SET K-Cover problem was not widely and deeply investigated for the HWSNs. Hence, we want to solve this hard problem in HWSNs through the proposed efficient pure distributed algorithms(GGAs) in this paper. In fact, GGAs are originally inspired by the potential games [24]. We show that the strategies of all the sensors will converge to the Nash equilibrium [25] by adopting GGAs finally. The main contributions of this research paper can be summarized as follows:

First, we propose three novel pure distributed Greedy Game Algorithms(GGAs) to solve the SET K-Cover problem in heterogeneous wireless sensor networks(HWSNs).

Second, two kinds of the general heterogeneous sensing models are introduced for the SET K-Cover problem in wireless sensor networks(WSNs).

Third, the classical game theory [26] is adopted to solve the SET K-Cover problem into the heterogeneous wireless sensor networks(HWSNs).

Fourth, extensive experiments have shown the superiority in coverage, convergence, robustness and network design performance of the proposed GGAs compared with the traditional Random Algorithm(RA), Distributed K-Cover Algorithm(DKCA) and Synchronous Nash Equilibria Convergence Algorithm(SNECA).

The rest paper is organized as follows. In section II, we mainly review the related works in this research area and express how our approach differs from the literatures. In section III, we state the coverage game problem and the heterogeneous sensing models in detail. GGAs are presented and analyzed in detail in section IV. In section V, we verify the superiority in coverage, convergence, robustness and network design performance of GGAs through extensive experiments. Finally, the conclusion is presented in section VI.

II. RELATED WORKS

Most researchers focus on maximizing the number of covers subject to the required coverage ratio, which is named as maximum set covers problem. In paper [27], Sasa and Miodrag prove that the maximum set covers problem is NP-Complete. In paper [28], Zhang and Hou derive a tight upper bound for α-lifetime, that is, at least α portion of the monitored region is covered during the working period. In paper [17], Abrams *et al.* try to maximize the coverage subject to the predefined lifetime requirement K, which is called the SET K-Cover problem, and it is also proved to be NP-Complete. Finally, the authors introduce three algorithms to solve this problem: (*i*)random algorithm, (*ii*)distributed greedy algorithm, (*iii*)global greedy algorithm. The random algorithm is the simplest one and also robust, however, its coverage performance is not so good. In the distributed greedy algorithm, each sensor chooses the cover set sequentially according to its (identification, ID) number. A sensor chooses one of the K-Cover sets to maximally increase the total coverage, and its decision is totally based on the decisions of all the previous sensors. The algorithm is also simple and can provide good coverage performance in most cases. However, there are some negative aspects in it. First, each sensor can only make a decision once due to the distributed algorithm, so it almost hardly provides the optimal solution to the SET K-Cover problem. Second, the algorithm is not a real purely distributed one because each sensor must know all of the previous sensors' decisions in the whole networks. It needs a central controller of WSNs to finish this job. Third, the running time of the distributed algorithm is proportional to the number of total sensors N, so it will take a long time to run in large-scale WSNs. Fourth, the algorithm requires synchronization for its execution. In global greedy algorithm, the coverage performance is better than that in random and distributed algorithms. However, the global greedy algorithm is indeed a centralized algorithm, and its communication and storage requirements are much higher than the random and distributed greedy algorithm. After that, Xin *et al.* [18] introduce SNECA to solve the SET K-Cover problem in wireless sensor networks(WSNs).

Following the prior researchers, NPCGA is provided in our previous work [20] to solve the SET K-Cover problem in WSNs. NPCGA shows the superiority in convergence and coverage performance than random, distributed greedy algorithm and SNECA. Interested readers can refer to [20] for detail information. In paper [23], they propose a novel memetic algorithm(MA) based on integer coded genetic algorithm and local search to solve the SET K-Cover problem in WSNs. The paper adapts the crossover and mutation operators to integer representation and, furthermore, design a new fitness function that consider both the number of covers and the contribution of each sensor to covers. Experimental results show that the proposed MA indeed outperforms five evolutionary algorithms in terms of the number of covers obtained, hit rate(HR), and running time. Unfortunately, the proposed MA is a pure centralized algorithm that can not be implemented in a distributed manner. This shortcoming can affect its application directly for practical situation in a large scale wireless sensor networks. Sun [22] proposes a time variant log-linear learning algorithm(TVLLA) that relies on local information only to solve the SET K-Cover problem in wireless sensor networks. The TVLLA guarantees convergence to the optimal solution with probability 1. Experimental results also reveal the better near-optimal solutions in a reasonable computation time than the state-of-the-art. However, the computational complexity of the proposed TVLLA is far more than other traditional algorithms based on the extensive experimental results. To be exactly, the TVLLA only has a small improvement in average coverage ratio at the expense of several tens of times time consumption. That is to say, the whole energy consumption of the TVLLA is far more than other traditional algorithms, especially in the large scale wireless sensor networks. Eg., NPCGA.

Based on above analysis, most of the current algorithms to solve the SET K-Cover problem are subject to the homogeneous wireless sensor networks or suffer from some defects which are already described in the above. In fact, the heterogeneity is a very common characteristic in practical situation. Due to the different types and the unbalanced energy consumptions for each sensor in WSNs, the sensing radius R_s and the communication radius R_c are often different among each other. Inspired by potential games [24], the classical Nash equilibrium theory [25] and our previous work [20], three kinds of the greedy game algorithms(GGAs) are proposed to solve the SET K-Cover problem in HWSNs.

GGAs are the identification based greedy game algorithm(IDBGGA), pure random based greedy game algorithm(PRBGGA) and time based greedy game algorithm(TBGGA). IDBGGA is such a algorithm that sensors update their strategies totally based on their ID number. Detail information of IDBGGA will be demonstrated in section IV. This mechanism cause that each sensor must consider the strategies of all of its neighbors before updating its best strategy. Eg., the sensor with the smallest ID will update its strategy first among its local neighbors. That is to say, any sensor can not update its strategy until all the neighbor sensors with smaller ID have completed its strategy updating action. This indicates that a potential distributed synchronization mechanism exists among its neighbors in IDBGGA. PRBGGA is such a algorithm that each sensor updates its best strategy totally based on itself and never consider other neighbor sensors. It means that PRBGGA is a pure and true distributed algorithm. TBGGA is such a algorithm that each sensor updates its strategy totally based on itself and never need to consider its neighbor sensors. However, TBGGA must have a self synchronization mechanism that each sensor must finish its best strategy updating event within a fixed time period T at each iteration. The mechanism of TBGGA is easy to implement due to the single chip processor development embedded on the sensor. Based on above analysis, PRBGGA is the simplest algorithm to be conducted in practical situation. However, it will take relatively a little longer time to converge to the Nash equilibrium. TBGGA is the best algorithm, when considering the convergence time and the mechanism to be implemented together.

In GGAs, we assume that each sensor(player) is a selfish and rational decision maker. In order to get the maximum incomes for itself, every sensor will choose the best strategy to obtain the maximum sensing area which is sensed alone at each game iteration. Due to the basic principle of Nash equilibrium [25], the best strategy profile which is chosen by all the sensors will converge to the Nash equilibrium. The effects of the heterogeneity property of HWSNs on convergence time and coverage performance can be weakened during implementing the GGAs iteratively.

III. COVERAGE GAME PROBLEM AND HETEROGENEOUS SENSING MODELS

A. COVERAGE GAME PROBLEM

In this section, we consider there are N sensors randomly scattered in the sensing field F, which can be divided into $p \times q$ grids. Each sensor has its sensing radius R_s and communication radius R_c respectively. In this paper, we consider the idea sensing model, and a sensing field F with $p \times q$ grids. Assume that there are N sensors deployed randomly in it. Each sensor has a sensing radius R_s. Assume sensor i deployed at point(x_i, y_i). For any sensing point W at (x, y), we denote the Euclidean distance between i and W as $d(i, W)$, i.e., $d(i, W) = \sqrt{(x_i - x)^2 + (y_i - y)^2}$. Equation (1) shows the binary sensing model [29] that expresses the coverage $\eta_{xy}(i)$ of a grid point W by sensor i.

$$\eta_{xy}(i) = \begin{cases} 1 & \text{if } d(i, W) < R_s \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Based on this sensing model, each sensor has its sensing radius(eg., R_s as its physical property for each sensor) for practical situations. Hence, the practical sensing area within its sensing radius can be calculated as (2). Besides, the sensor can find its location and its neighbors by using GPS or sensor localization, and there are some algorithms in [30] and [31] for this task.

$$S = \pi \times R_s^2 \tag{2}$$

In order to take full advantage of the overlapped sensing area among all the sensors, much more sensors are randomly scattered in the sensing area than the common needs. In fact, the main purpose of this paper is to fully utilize the redundancy of the sensors, and prolong the lifetime as well as make the average coverage ratio maximized of HWSNs. To achieve such purpose, the SET K-Cover problem has been proposed in literature [17]. Let each sensor belongs to one of the K disjoint cover sets, and the lifetime of the whole networks is proportional to the parameter K. Our objective is to maximize the average coverage ratio with the given parameter K. For any sensor i, let s_i denote the cover set it belongs to, i.e., $s_i = j$, $j \in \xi$, and $\xi = 1, \ldots, K$ denotes the K-Cover sets. And then, $s_i \in s$, $s = \{s_1, s_2, \ldots, s_N\}$ represents a partition of N sensors into K-Cover sets by using the proposed algorithm. Let S be the set of all K-Cover possible, and the cardinality of S can be denoted as $|S| = K^N$. One of the coverage set $c_j(s)$ can be denoted as $c_j(s) = | \cup_{\{i:s_i=j\}} R_i |$, where R_i is the region covered by sensor i. Our purpose is to maximize (3) as follows:

$$Maximize\ c(s) = \frac{1}{K} \sum_{j=1}^{K} c_j(s), \quad s \in S \tag{3}$$

In order to achieve the maximum average coverage ratio, each sensor has to minimize the overlapped sensing area within its sensing radius R_s. Based on above analysis, we introduce the GGAs, which are originally extended from our previous work [20]. Basically speaking, GGAs are pure non-cooperative game approach, which has been well analyzed and proved by Nash in his classical paper [25]. Nash also proved the valuable and crucial conclusion that there must be an equilibrium point for every non-cooperative game. At each iteration of the game process, the selfish property allows each player(sensor) maximize its own incomes. It must make the best decision at each iteration based on the present strategies of its neighbors. Finally, we obtain the following assumptions: a game of cover sets chosen is an interactive decision making process among the selfish sensors, which can be expressed as $G(\Delta, s, V)$ and formally consists of the following elements.

(1) Δ is the players set, and $\Delta = \{1, \cdots, N\}$ with N, the number of players in the game.

(2) s is the cover set chosen strategies by all the N sensors. $s = \{s_1, s_2, \cdots, s_N\}$, where $s_i = j, j \in \xi$.

(3) The payoff $V_i \in V$, where $V = \{V_1, \cdots, V_i, \cdots, V_N\}$ has a close relationship with the strategy profile chosen by each selfish sensor.

The final objective of cover set chosen game G in HWSNs is to determine an optimal cover set strategy profile $s = \{s_1, \cdots, s_i, \cdots, s_N\} = \{s_i, s_{-i}\}$ to achieve a maximal payoff V_i for each sensor i, where s_i is the best cover set choice for sensor i, and s_{-i} is a strategy vector chosen by all the other sensors except the i^{th} sensor. The symbol $-i$ is used to represent all sensors in Δ except i itself. For coverage game model, we are mainly concerned with the sensing area V_i as the payoff function, which each player(sensor) wants to maximize.

$$V_i(s_i, s_{-i}) = |R_i \setminus \cup_{\{l:s_l=s_i, l \neq i\}} R_l| \tag{4}$$

The symbol of "\" means exclusion in (4). To be specific, it means that the difference value between the whole sensing area of sensor i and the overlapped sensing area with its neighbor sensors. That is the region covered alone by sensor i. Where V_i is the region covered alone by sensor i in (4), and it is shown clearly in Fig.1.

The purpose of this paper is to maximize the payoff function $V_i(s_i, s_{-i})$ for each sensor i so that we can get the maximized average coverage $c(s)$ in the whole HWSNs. The game process continues as the iteration n increases, and between the two iterations of the cover set chosen game, when sensor i changes its strategy from s_i to s_i' at a random iteration n, the change in the whole average coverage between the two K-Cover of the HWSNs is shown in (5).

$$\begin{aligned} \Delta c &= c(s_i', s_{-i}) - c(s_i, s_{-i}) \\ &= \frac{1}{K}(V_i(s_i', s_{-i}) - V_i(s_i, s_{-i})) \\ &= \frac{1}{K} \Delta V(i) \end{aligned} \tag{5}$$

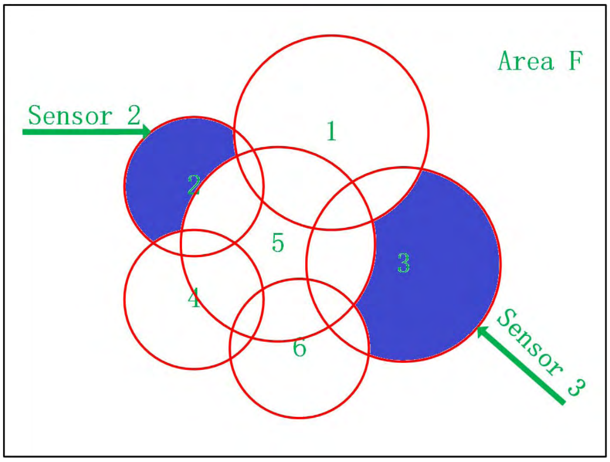


FIGURE 1. Payoff value of sensor 2 and 3 (V_2 and V_3) are shown by the blue areas in HWSNs.

where ΔV_i can be named as the regret value of sensor i. Obviously, all the sensors in the non-cooperative coverage game process are the rational and selfish decision makers, and they will maximize their own payoffs at each iteration of GGAs. The payoff V_i is only related to sensor i' neighbors $N_{eig}(i)$, with which the sensing area is overlapped. The relationship can be demonstrated as (6).

$$\forall i,\ V_i(s_i, s_{-i}) = V_i(s_i, s_{N_{eig}(i)}) \tag{6}$$

Definition 1: Nash Equilibrium of a non-cooperative game is a profile of strategy $s^* = \{s_1^*, \cdots, s_N^*\} \in s$. Especially, the sensor cannot obtain a higher payoff by unilaterally changing his strategy in the game process, and we can gain the (7) as follows:

$$\forall i, \quad and\ \forall s_i \in \xi,\ V_i(s_i^*, s^*_{N_{eig}(i)}) \geq V_i(s_i, s^*_{N_{eig}(i)}) \tag{7}$$

where its best strategy s_i^* can be expressed in (8) as follows:

$$s_i^*(s_i, s_{N_{eig}(i)}) = arg \max_{s_i=1,\ldots,K} V_i(s_i, s_{N_{eig}(i)}) \tag{8}$$

In (6), (7) and (8), $s_{N_{eig}(i)}$ represents the strategies of all the neighbors of sensor i, and then the best strategy of sensor i can be denoted as $s_i^*(s_i, s_{N_{eig}(i)})$.

Theorem 1: For the coverage game G in HWSNs, when the whole sensors(players) choose their best cover set strategies $s^* \in s$, and maximize the payoff functions $V(s^*)$, and the pure Nash Equilibrium can be constituted in the coverage game of HWSNs finally.

Proof 1: We assume that there is a cover set chosen optimal strategy profile $s^{opt} \in S$, considering any sensor i and the strategy profile $s' = (s_i, s_{-i}^{opt})$. Since s^{opt} is the optimal strategy profile of the whole strategy profile set s, we can derive $c(s^{opt}) \geq c(s')$. Thus, according to (7), we can obtain $V_i(s_i^{opt}, s_{-i}^{opt}) \geq V_i(s_i, s_{-i}^{opt})$, and the equation holds for any sensor i and its corresponding strategy $s_i \in \xi$. Based on the definition of Nash Equilibrium and (8), the pure Nash Equilibrium is constituted after the optimal cover set strategy $s^* = s^{opt}$ has been chosen by the whole sensors in HWSNs.

B. HETEROGENEOUS SENSING MODEL

By neighborhood, it means that the sensors have overlapped sensing area among each other. In detail, the neighborhood of sensors can be divided into four kinds. First, if sensor i and j can communicate with each other directly, and hence they are considered as equal neighbors($N_{eig1}(i)$ or $N_{eig1}(j)$). Second, if sensor i and j cannot communicate with each other, and hence they are considered as blind neighbors($N_{eig4}(i)$ or $N_{eig4}(j)$). Third, if sensor i is in the communication radius of sensor j, but converse is not true. Hence, we can consider sensor i is a potential neighbors($N_{eig3}(j)$) of sensor j, while sensor j is a explicit neighbors($N_{eig2}(i)$) of sensor i. Generally speaking, the above four kinds of neighbors can be constituted by the following two kinds of heterogeneous sensing models.

1) HOMOGENEOUS R_C VERSUS HETEROGENEOUS R_S

By homogeneous R_c and heterogeneous R_s, we mean that each sensor with the same communication radius R_c and the different kinds of sensing radius R_s. First, we assume that there are two kinds of sensing radius, such as $R_c \geq 2R_s$ or $R_c \leq R_s$. Two sensors are taken as an example, and the detail information is as follows:

a) $R_s(i) = R_s(j) = R_c$.

When the distance between sensor i and j is $d(i,j) \leq R_c$, Fig.2(a) shows that sensor i and j are equal neighbors with each other. When $R_c < d(i,j) \leq 2R_c$, they are blind neighbors, which is shown in Fig.2(b).

b) $R_s(i) = R_s(j) = R_c/2$.

When the distance between sensor i and j is $d(i,j) \leq R_c$, Fig.2(c) shows that they can communicate with each other and they are considered as equal neighbors.

c) $R_s(i) = 2R_s(j) = R_c$.

When the distance between sensor i and j is $R_c < d(i,j) \leq (Rc + \frac{1}{2}R_c)$, Fig.2(e) shows that they are blind neighbors. When $d(i,j) \leq R_c$, they are equal neighbors as Fig. 2(d) shows.

d) $2R_s(i) = R_s(j) = R_c$.

When the distance between sensor i and j is $R_c < d(i,j) \leq (Rc + \frac{1}{2}R_c)$, Fig.2(g) shows that they are blind neighbors. When $d(i,j) \leq R_c$, sensor i and j are equal neighbors as Fig.2(f) shows.

2) HOMOGENEOUS R_S VERSUS HETEROGENEOUS R_C

By homogeneous R_s and heterogeneous R_c, we mean that each sensor with the same sensing radius R_s and the different kinds of communication radius R_c. We consider that there are two kinds of communication radius, such as $R_s \leq \frac{1}{2}R_c$ or $R_s \geq R_c$. The detail information is shown as follows:

a) $R_c(i) = R_c(j) = 2R_s$

Fig.3(a) shows that the distance between i and j is $d(i,j) \leq 2R_s$, and hence they are equal neighbors.

b) $R_c(i) = R_c(j) = R_s$

Fig.3(b) shows that the distance between sensor i and j is $d(i,j) \leq R_s$ and hence they are equal neighbors.

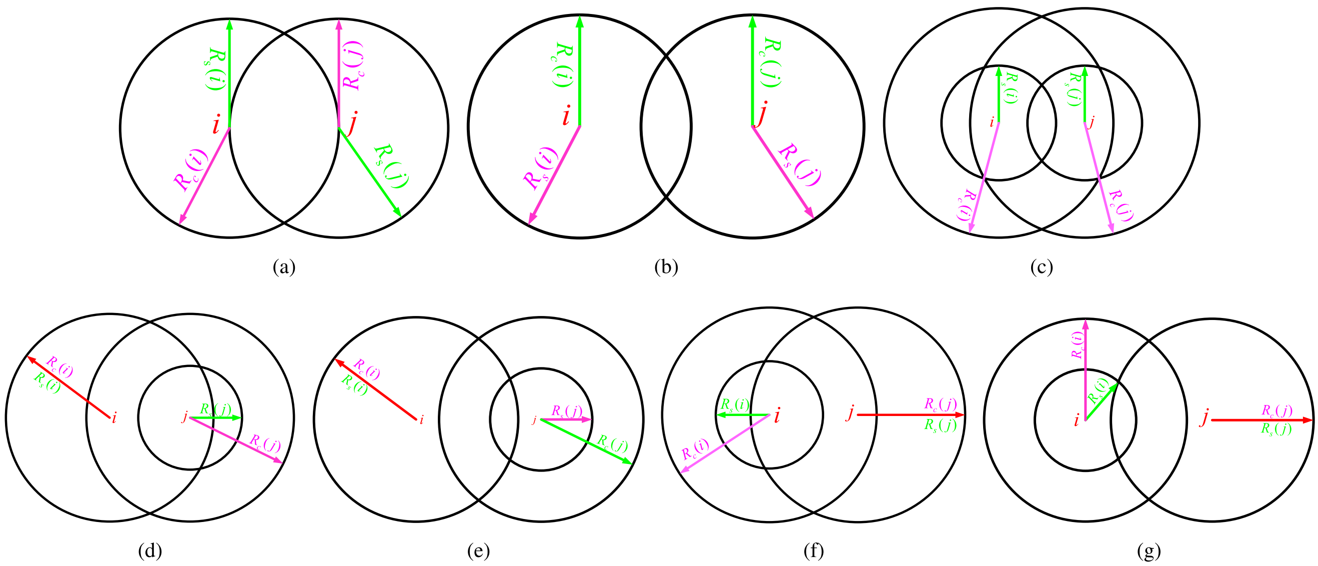


FIGURE 2. Neighborhood of sensors with the homogeneous R_C and the heterogeneous R_S. (a) Equal neighbors. (b) Blind neighbors. (c) Equal neighbors. (d) Equal neighbors. (e) Blind neighbors. (f) Equal neighbors. (g) Blind neighbors.

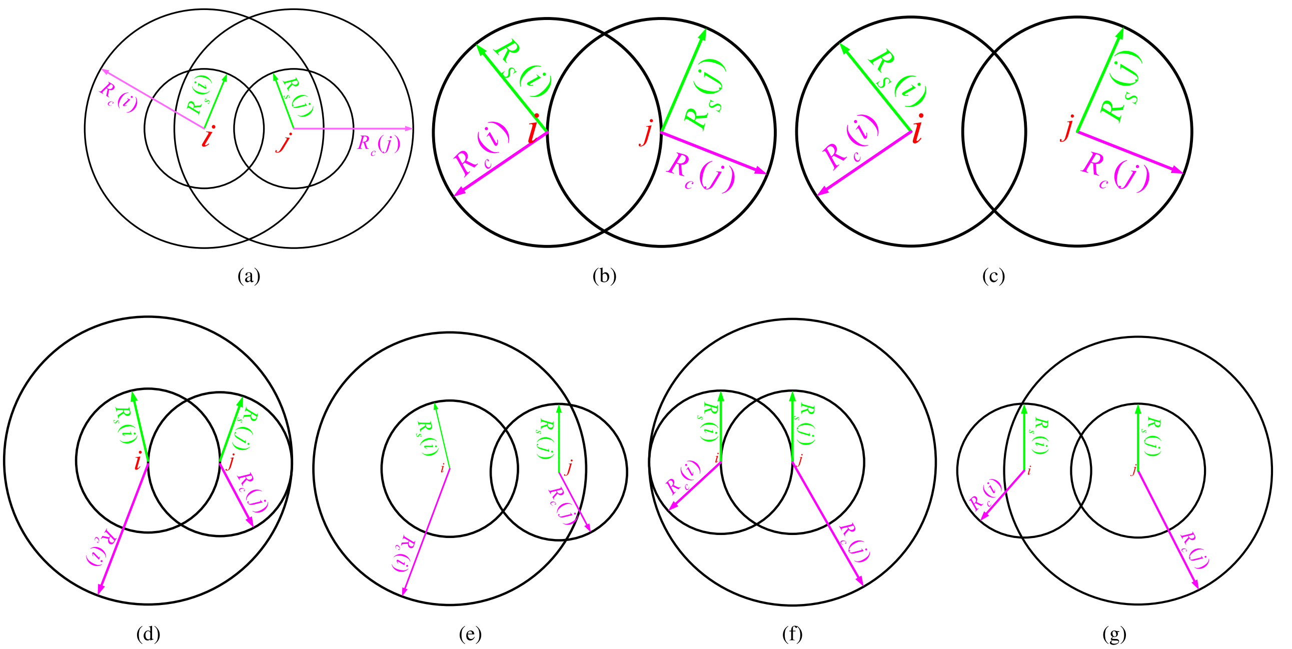


FIGURE 3. Neighborhood of sensors with the homogeneous R_S and the heterogeneous R_C. (a) Equal neighbors. (b) Equal neighbors. (c) Blind neighbors. (d) Equal neighbors. (e) Explicit neighbor. (f) Equal neighbors. (g) Explicit neighbors.

When $R_s < d(i, j) \leq 2R_s$, Fig.3(c) shows that they are blind neighbors.

c) $R_c(i) = 2R_s$ and $R_c(j) = R_s$

When $d(i, j) \leq R_s$, Fig.3(d) shows that they are equal neighbors. When $R_s < d(i, j) \leq 2R_s$, sensor j is a potential neighbor of sensor i, while sensor i is a explicit neighbor of j. The detail information is shown in Fig.3(e).

d) $R_c(i) = R_s$ and $R_c(j) = 2R_s$

When $d(i, j) \leq R_s$, Fig.3(f) shows that they are equal neighbors. When $R_s < d(i, j) \leq 2R_s$, and hence sensor i is a potential neighbor of sensor j, while sensor j is a explicit neighbor of sensor i. The detail information is shown in Fig.3(g).

In practical situation, whether we deploy different types of sensors or the same type sensors with different initial energies in the sensing area, the heterogeneity of WSNs can be considered as the mentioned two types in some cases.

IV. GGAS EXPLANATION AND CONVERGENCE ANALYSIS

A. GGAS EXPLANATION

In this section, we give detail information about the proposed algorithms. There is just a simple example of the initial topology of the HWSNs with 14 heterogeneous sensors as

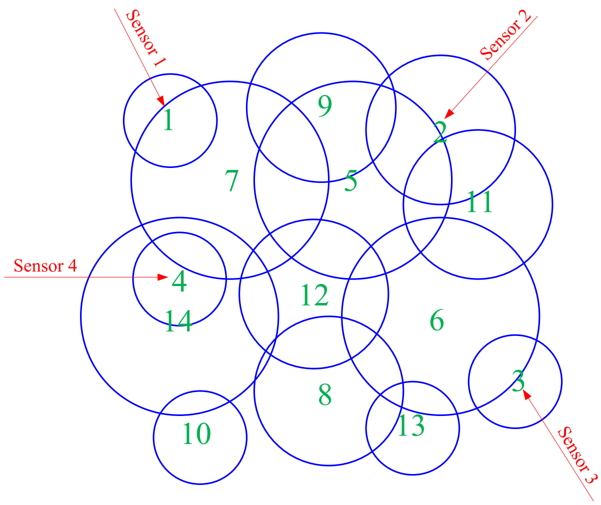


FIGURE 4. Initial sensing area of HWSNs with 14 heterogeneous sensors.

Fig.4 shows. They will cover the maximum sensing area after using the proposed distributed algorithms which are listed in the following part.

The GGAs prolong the lifetime of HWSNs to K times while the average coverage ratio of the whole monitored area gets as high as possible. In GGAs, each iteration generally consists several steps. First, each sensor i calculates its payoff $V(i)$ for every possible cover chosen sets. Second, based on the selfish property of each sensor, it will change to the best strategy according to the current strategy of its neighbors at each iteration. Third, each sensor broadcasts its updated strategy to its neighbors. Fourth, the next game process starts after all the N sensors finish the current game iterations. When the game process is completed, each sensor i chooses its best strategy s_i^*, which will constitute the pure Nash Equilibrium finally.

The main difference among IDBGGA, PRBGGA and TBGGA lies in strategy updating time of the basic game rule. First, in IDBGGA, the rule of updating and broadcasting its best strategy of each sensor is totally based on its identification(ID) at each game iteration. Second, each sensor updates and broadcasts its best strategy in a purely random way in PRBGGA, and it never concern other neighbor sensors. Third, in TBGGA, each sensor updates and broadcasts its best strategy at a pure random time point but in a fixed time period T for each iteration.

We continue to take the simple and small sensor networks which has been shown in Fig.4 as an example. For IDBGGA, the sensors with ID of 1, 2, 3, and 4 will update and broadcast the strategies simultaneously to their neighbors. More importantly, they are the first one which can update their coverage strategies among their local neighbors based on the IDBGGA principle. That is to say, the sensor with the smallest ID can update its coverage strategy first among its local neighbors, and the other sensors will update their coverage strategies one by one among their local neighbors based their ID numbers. After that, IDBGGA does not finish until the cover chosen strategies of all the sensors constitute the Nash Equilibrium. For PRBGGA, all the 14 sensors start to run the algorithm independently without considering other neighbor sensors among their local neighbors respectively. The PRBGGA does not terminate until the strategy profile of all the 14 sensors constitutes the Nash Equilibrium. For TBGGA, all the 14 sensors start to implement the algorithm independently within a fixed time threshold T at each iteration among their local neighbors respectively. The TBGGA does not end until all the 14 sensors get the Nash Equilibrium strategies successfully. The detail information can be demonstrated in algorithm 1, 2 and 3 respectively.

Next, the convergence property of IDBGGA, PRBGGA and TBGGA are discussed as follows.

Algorithm 1 IDBGG Algorithm(IDBGGA)

Input Initial strategy profile:$s^0 = (s_1^0, \cdots, s_N^0)$.
Output Nash Equilibrium s^*.

For sensor i, its practical neighbors can be concerned as $N_{eig}(i) = N_{eig1}(i) \cup N_{eig2}(i) \cup N_{eig3}(i) \cup N_{eig4}(i)$.
$N_{eigx} = \{\forall j \in N_{eig}(i) | ID(j) < ID(i)\}$;
$N_{eigy} = \{\forall k \in N_{eig}(i) | ID(k) > ID(i)\}$;
IDBGGA begin: s^{n-1} strategy profile at iteration $n-1$.
In each local game group and for each sensor $i \in \Delta$:
while Nash Equilibrium strategy profile does not constitute **do**
  if $N_{eigx} \neq \varnothing$ **then**
    if $\forall j \in N_{eigx}(i)$ and $\forall k \in N_{eigy}(i)$, they have sent best strategies to sensor i **then**
      Sensor i will do the followings:
      (I)Calculating its payoff $V(i)$ for every possible covers, and deciding its best strategy $s_i^*(s_i^n, s_{N_{eigx}}^n \cup s_{N_{eigy}}^{n-1})$;
      (II)Changing to its best strategy $s_i^*(s_i^n, s_{N_{eigx}}^n \cup s_{N_{eigy}}^{n-1})$;
      (III)Broadcasting its best strategy s_i^n to $N_{eig1}(i)$ and $N_{eig3}(i)$ through local communications.
    else
      Waiting for each sensor $j \in N_{eigx}$ and $k \in N_{eigy}$ has sent its best strategy to i at iteration n and $n-1$.
    end if
  else
    if $\forall k \in N_{eigy}(i)$, it has sent its best strategy to i at iteration $n-1$ **then**
      Then sensor i will follow the same steps as the former (I), (II) and (III).
    else
      Waiting until $\forall k \in N_{eigy}(i)$ has sent the best strategy to i at iteration $n-1$, then sensor i will do the procedure as (I), (II) and (III).
    end if
  end if
end while
return Nash Equilibrium strategy profile $s^n = s^*$

Algorithm 2 PRBGG Algorithm(PRBGGA)

Input Initial strategy profile:$s^0 = (s_1^0, \cdots, s_N^0)$.
Output Nash Equilibrium s^*.
For sensor i, its practical neighbors can be concerned as $N_{eig}(i) = N_{eig1}(i) \cup N_{eig2}(i) \cup N_{eig3}(i) \cup N_{eig4}(i)$.
Initial $t = 0$ and iteration counter $n = 0$, $\forall i \in \Delta$ *and* $t_i(0) = 0 + \eta_i T$.
PRBGGA begin:In each local game group and for each sensor $i \in \Delta$:
Generating an uniform random variable $\eta_i \in \bigcup(a-b, a+b)$ and $a \geq b$
while Nash Equilibrium strategy profile does not constitute **do**
  if $\forall i \in \Delta$ and $t = t_i(n)$ **then**
    Sensor i will do the followings:
    (I)Calculating its payoff $V(i)$ for every possible covers and deciding its best strategy $s_i^*(s_i^n, s_{N_{eig}(i)}^{\varphi})$ with uncertain iteration number φ;
    (II)Changing to its best strategy $s_i^*(s_i^n, s_{N_{eig}(i)}^{\varphi})$;
    (III)Broadcasting its best strategy $s_i^*(s_i^n, s_{N_{eig}(i)}^{\varphi})$ to $N_{eig1}(i)$ and $N_{eig3}(i)$ through local communications;
    (IV)Waiting for a random time $\eta_i T$ for the next game process. In detail, we set $t_i(n+1) = t_i(n) + \eta_i T$ and $n = n+1$.
  end if
end while
return Nash Equilibrium strategy profile $s^n = s^*$

Algorithm 3 TBGG Algorithm(TBGGA)

Input Initial strategy profile:$s^0 = (s_1^0, \cdots, s_N^0)$; A fixed time period T
Output Nash Equilibrium s^*.
For sensor i, its practical neighbors can be concerned as $N_{eig}(i) = N_{eig1}(i) \cup N_{eig2}(i) \cup N_{eig3}(i) \cup N_{eig4}(i)$.
Initial $t = 0$, and iteration counter $n = 0$, $\forall i \in \Delta$ *and* $t_i(0) = 0 + m_i T$.
TBGGA begin: In each local game group and for each sensor $i \in \Delta$:
Generating an uniform random variable $m_i \in \bigcup(0, 1)$.
while Nash Equilibrium strategy profile does not constitute **do**
  if $\forall i \in \Delta$ *and* $t = t_i(n)$ **then**
    Sensor i will do the followings:
    (I) Calculating its payoff $V(i)$ for every possible covers deciding its best strategy$s_i^*(s_i^n, s_{N_{eig}(i)}^{r})$ with uncertain iteration number r;
    (II) Changing to its best strategy $s_i^*(s_i^n, s_{N_{eig}(i)}^{r})$;
    (III) Broadcasting its best strategy $s_i^*(s_i^n, s_{N_{eig}(i)}^{r})$ to $N_{eig1}(i)$ and $N_{eig3}(i)$.
    (IV) Waiting for a random time $m_i T$ for the next game process. In detail, we set $t_i(n+1) = nT + m_i T$ and $n = n+1$.
  end if
end while
return Nash Equilibrium strategy profile $s^n = s^*$

Theorem 2: IDBGGA converges to the Nash Equilibrium.

Proof 2: Consider c^* as the optimal average coverage among all of the average coverage set that the whole N sensors provide. As we know, since the whole sensing area that all the wireless sensors cover is a finite value and the upper bound should be a constant, which can be denoted as $c_{max} < +\infty$. We use s^n to denote the cover set chosen strategy that all of the N sensors choose at iteration n. Based on the principle of IDBGGA, there is at most one sensor can change its strategy among all of its neighbors $N_{eig}(i)$ at iteration n. Therefore, Δc can be expressed as (9) as follows:

$$\begin{aligned} \Delta c &= c(s^n) - c(s^{n-1}) \\ &= \frac{1}{K} \sum_{i: s_i^n \neq s_i^{n-1}} \Delta V_i(s_i^{n-1}, s_{N_{eig}(i)}^{\tau}) \geq 0 \end{aligned} \tag{9}$$

Based on the principle of IDBGGA, when all the N sensors finish their strategies changing event from s^{n-1} to s^n, the whole average coverage can be improved by $\Delta c = \frac{1}{K}\sum_{i:s_i^n \neq s_i^{n-1}} \Delta V_i$. In IDBGGA, broadcasting and changing strategy sequence of each player are totally based on its ID in the local game group. When sensor i is changing its strategy from $(n-1)^{th}$ to n^{th} iteration, its neighbors with smaller ID must be at the iteration n and the neighbors with bigger ID must be at the iteration $(n-1)$. The value of τ in (9) is n or $(n-1)$ for the neighbors of sensor i. Based on the greedy updating rule, the average coverage is strictly increasing as the game iteration n increases. As we have stated before, the optimal average coverage c^* is a finite value and the proposed algorithm must converge finally. That is, ΔV_i will be equal to zero for each sensor i. In short, IDBGGA converges to the pure Nash Equilibrium finally.

Theorem 3: PRBGGA and TBGGA converge to the Nash Equilibrium.

Proof 3: The proof method is similar with the former one. Hence, we only omit the procedure here due to the restricted length of this paper.

B. CONVERGENCE ANALYSIS OF THE PROPOSED GGAS

In this section, we mainly investigate the convergence performance of the proposed GGAs in HWSNs. We adopt Θ to denote the overall iterations needed for GGAs, with which the strategies of all the sensors converge to the Nash Equilibrium point. ζ is used to denote the maximum number of the equal, explicit and potential neighbors of sensor i and plus itself. Then ζ can be expressed as (10):

$$\zeta = \arg \max_{\forall i \in \Delta}(N_{eig1}(i) + N_{eig2}(i) + N_{eig3}(i)) + 1 \tag{10}$$

According to the definition of ζ, the overall iterations needed for GGAs can be considered as follows.

Case one: $K \geq \zeta$:

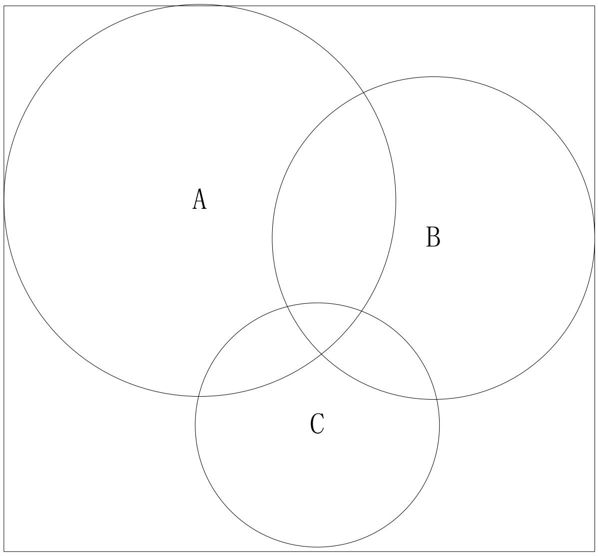


FIGURE 5. Initial topology for whole HWSNs.

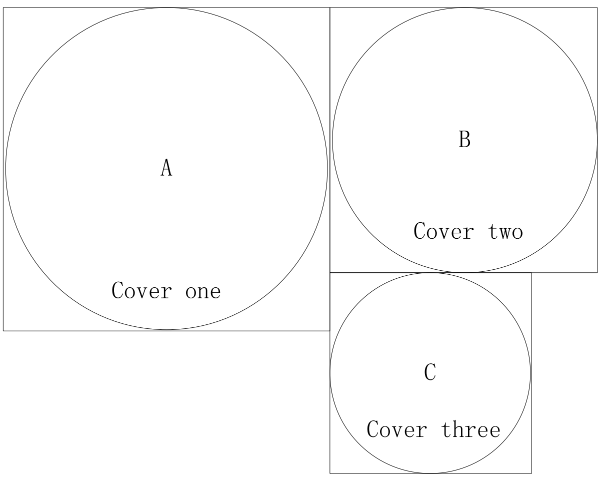


FIGURE 6. Whole iterations by GGAs with $K \geq \zeta$.

Theorem 4: When $K \geq \zeta$, the total iterations needed for GGAs(including IDBGGA, PRBGGA and TBGGA) can be expressed as (11):

$$\Theta = 1 \tag{11}$$

Proof 4: Under such condition, due to the number of the cover sets K is not less than ζ in the whole HWSNs, therefore, each sensor can choose one of the cover sets freely without hurting the payoff values of its equal, explicit and potential neighbors. Hence, all the sensors can obtain their best cover set chosen strategies, and constitute the Nash Equilibrium. In order to make it better understood, a simple example is shown in Fig.5 and 6. There are three sensors(A, B and C) deployed in the sensing field randomly.

Case two: $K \leq \zeta$:

Theorem 5: When $K \leq \zeta$, the total iterations needed for GGAs(including IDBGGA, PRBGGA and TBGGA) can be expressed as (12):

$$\Theta \leq N\binom{K}{1} = K^N \tag{12}$$

Proof 5: Under such condition, due to the number of the cover sets K is less than ζ in the whole HWSNs, therefore, each sensor cannot choose one of the cover sets freely without hurting the payoff values of its equal, explicit and potential neighbors in most cases. Hence, any sensor will choose one of the covers which can maximize its payoff value at each iteration, and at the same time, the game process starts. Each sensor can choose one of the K-Cover sets with its selfish property at each iteration, and the total number of sensors is N, therefore, the maximum number of the iterations needed for GGAs can be expressed as (13):

$$\begin{aligned}\Theta &= \underbrace{\binom{\delta_1}{1}\cdots\binom{\delta_i}{1}\cdots\binom{\delta_N}{1}}_{N} \\ &\leq \underbrace{\binom{K}{1}\cdots\binom{K}{1}\cdots\binom{K}{1}}_{N} \\ &= N\binom{K}{1} = K^N \end{aligned} \tag{13}$$

Where δ_i is one of the covers chosen by sensor i during the whole iterations and $\forall i \; \delta_i \leq K$. In fact, the total number of iterations needed for GGAs is much less than K^N(only several iterations needed for GGAs to converge to the Nash Equilibrium) through the extensive experiments in the following section.

V. PERFORMANCE EVALUATION

Extensive experiments are conducted to evaluate the effectiveness of the GGAs and compare them to the existing classic distributed algorithms, including random algorithm(RA), distributed greedy algorithm [17] and SNECA [18]. The experimental environment is set in table1 and the parameters are different from each other in each experiment. Without loss of generality, we randomly scatter N sensors in a rectangle region with $p \times q$ grid points. The sensing radius of each sensor is R_s grids, and thus, each sensor can cover a disk area centered at itself with the sensing radius R_s. The communication radius of each sensor is R_c. All the sensors are divided into K covers, and the coverage is measured by the number of covered grid points, e.g., 100×100.

A. COVERAGE PERFORMANCE

In this section, we verify the coverage performance of the proposed GGAs in different aspects. We assume that the monitored area F is divided by 100×100 grids. The heterogeneous ratio(HC_{Ratio}) is from 0 to 0.4 for the following three kinds of experiments. Parameters of the coverage performance, namely, N, K and R_s are chosen to make GGAs are comparable.

TABLE 1. Experimental platform description.

Experimental Software Platform	Software Development Environment	Matlab 2014b
	Developing Language	Matlab
Experimental Hardware Platform	CPU	Intel(R)Core(TM)i7-6700HQ
	RAM	8GB
	Operating System	Microsoft Windows 7(64bit)

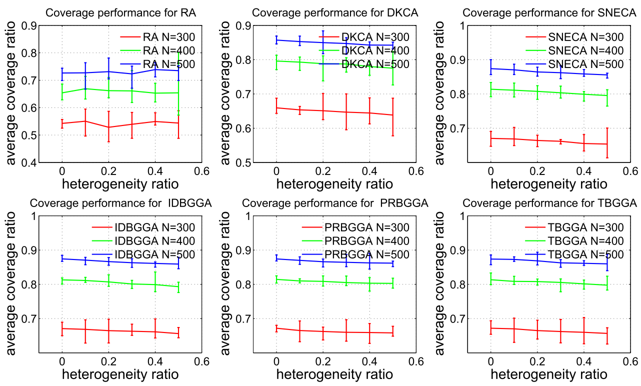


FIGURE 7. Coverage performance of different algorithms versus N.

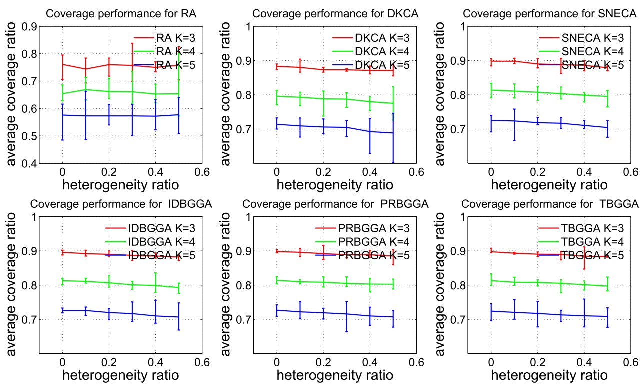


FIGURE 8. Coverage performance of different algorithms versus K.

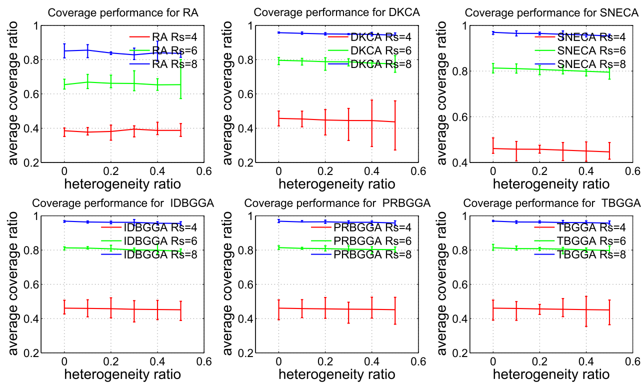


FIGURE 9. Coverage performance of different algorithms versus R_s.

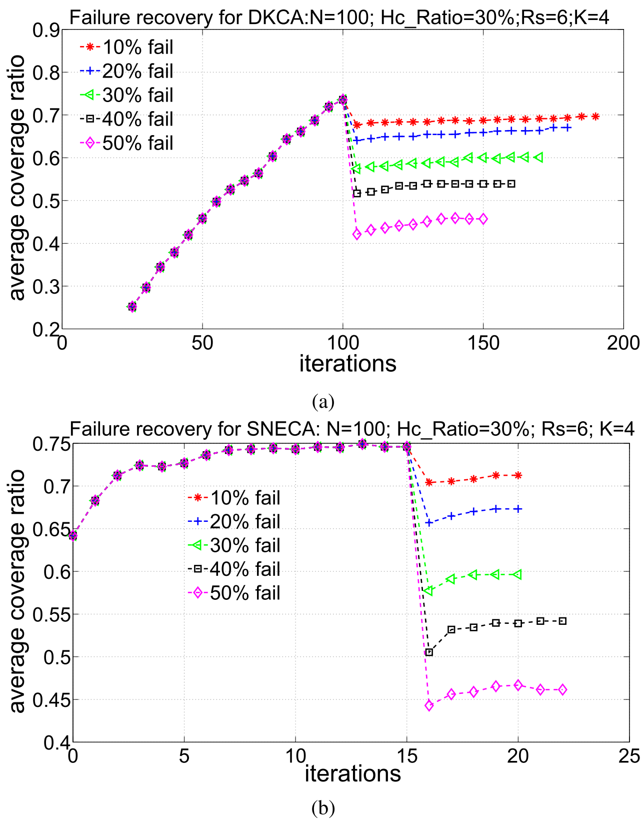


FIGURE 10. Failure Recovery performance with various sensors failure ratios by traditional algorithms. (a) Failure Recovery for DKCA. (b) Failure Recovery for SNECA.

First, we set $K = 4$, $R_s = 6$ and N ranges from 300 to 500. In Fig.7, we observe that the average coverage performance of IDBGGA, PRBGGA and TBGGA are almost the same and a little better than SNECA. The superiority of coverage performance for GGA is obvious compared with RA and DKCA under the same parameter settings. For GGAs, the coverage difference among all the covers is almost not affected by variation of the heterogeneous ratio for the same N. However, the coverage difference indeed becomes more and more smaller for the same heterogeneous ratio as N increases. It is concluded that GGAs are stable and very robust to the variation of the heterogeneous ratio.

Second, we set $N = 400$, $R_s = 6$ and K ranges from 3 to 5. Fig.8 shows that the average coverage performance of IDBGGA, PRBGGA and TBGGA are almost the same. Besides, coverage performance of GGAs is a little better than SNECA and much better than RA and DKCA under the same parameter settings. For GGAs, the differential of the coverage ratio is relatively stable as the heterogeneous ratio increases among all the covers. In a word, GGAs are relatively stable and robust to the variation of the heterogeneous ratio with other parameters fixed in HWSNs.

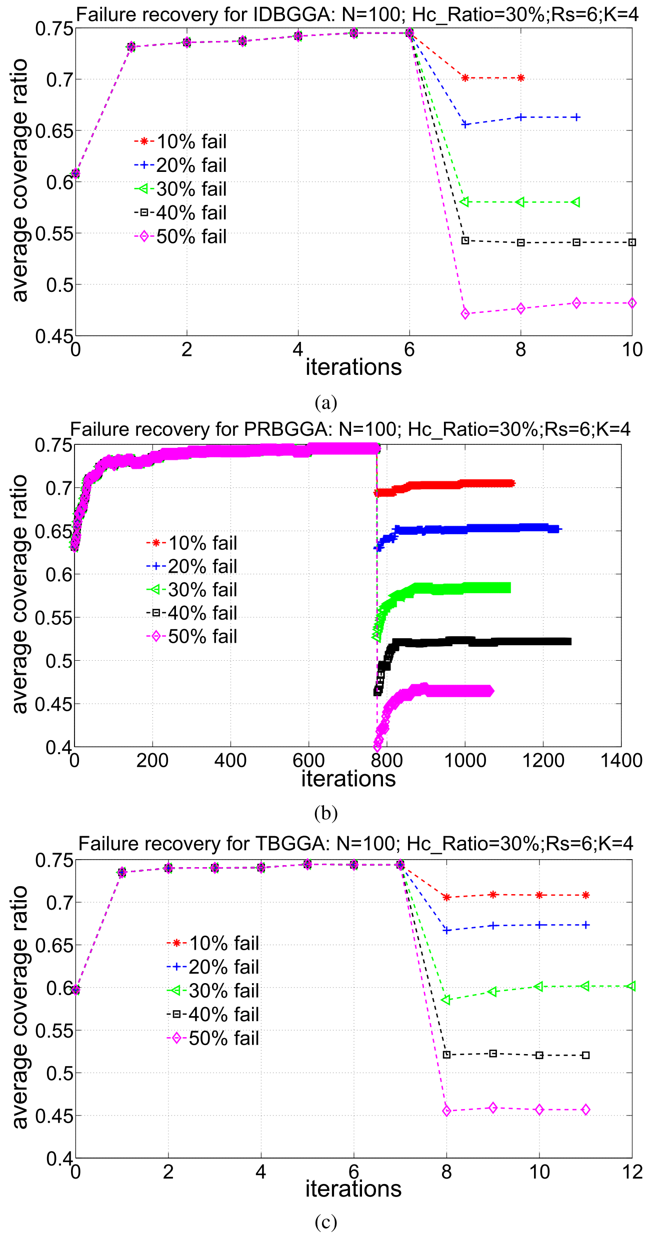


FIGURE 11. Failure Recovery performance with various sensors failure ratios by GGAs. (a) Failure Recovery for IDBGGA. (b) Failure Recovery for PRBGGA. (c) Failure Recovery for TBGGA.

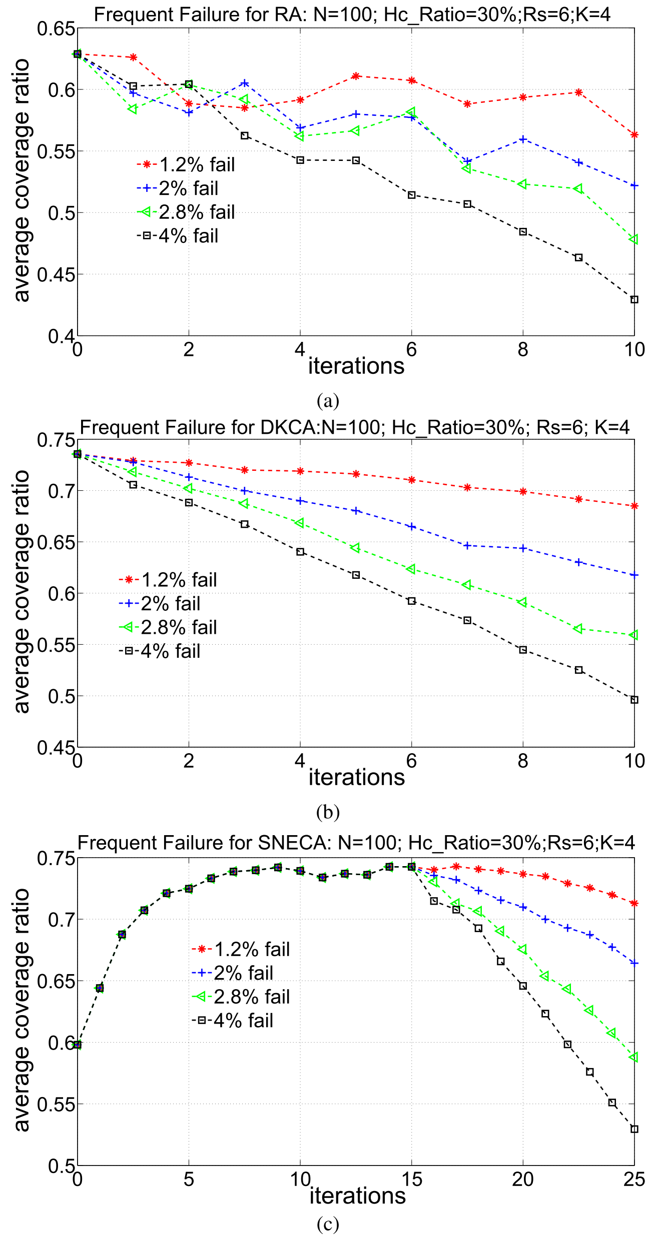


FIGURE 12. Frequent failure performance of traditional algorithms. (a) Frequent failure performance of RA. (b) Frequent failure performance of DKCA. (c) Frequent failure performance of SNECA.

Third, we set $N = 400$, $K = 4$ and R_s ranges from 4 to 8. In Fig.9, it is observed that the average coverage performance of IDBGGA, PRBGGA and TBGGA are still almost the same and a little better than SNECA. Besides, coverage performance of GGAs is much better than RA and DKCA under the same situations. For GGAs, the coverage difference among all the covers is almost not affected by variation of the heterogeneous ratio for same R_s. However, the coverage difference indeed becomes more and more smaller for same heterogeneous ratio as R_s increases. It means that GGAs are stable and very robust to the variation of the heterogeneous ratio with other parameters fixed.

B. CONVERGENCE AND ROBUSTNESS

In this section, we mainly focus on the convergence and robustness property of GGAs. By convergence, we mean how many iterations they need until the proposed algorithms converge to the Nash Equilibrium. By robustness, we mean how well it can recover from failures in sensor networks. RA is not a kind of the iteration algorithm, so it is not used as the comparative algorithm in all the following experiments. Since sensors are expected to be fault prone especially in the harsh environment, this is an critical aspect to be investigated. To be convenient, network parameters are set as follows: $F = 50m \times 50m$, $N = 100$, $K = 4$, $R_s = 6$ and $HC_{Ratio} = 0.3$.

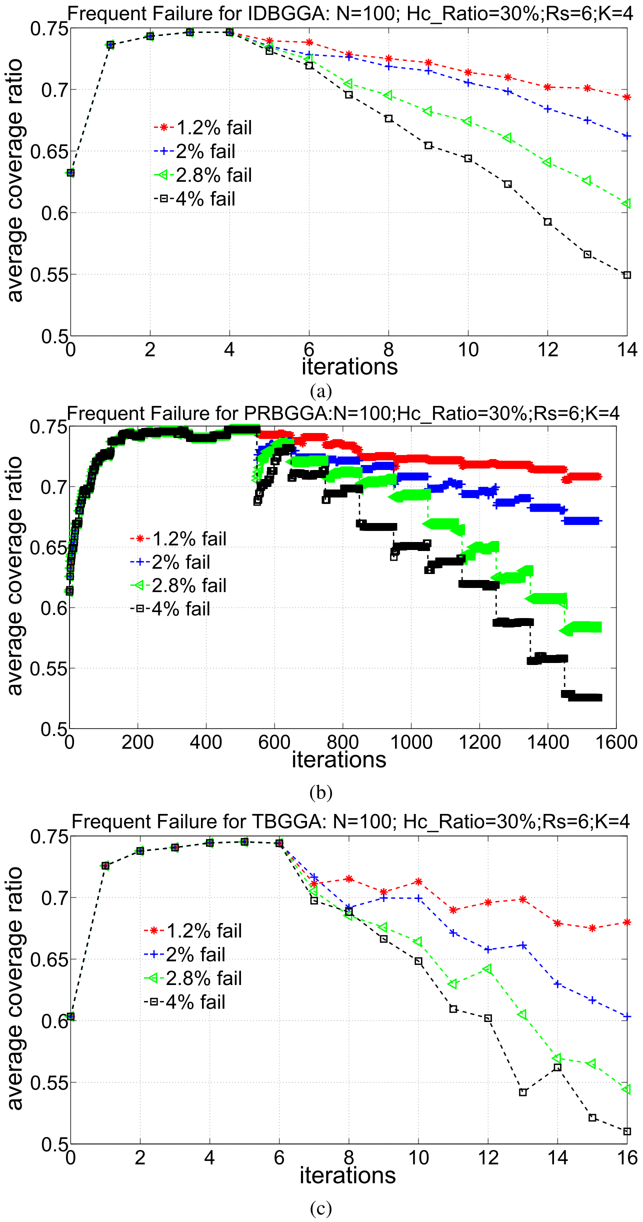


FIGURE 13. **Frequent failure performance of GGAs. (a) Frequent failure performance of IDBGGA. (b) Frequent failure performance of PRBGGA. (c) Frequent failure performance of TBGGA.**

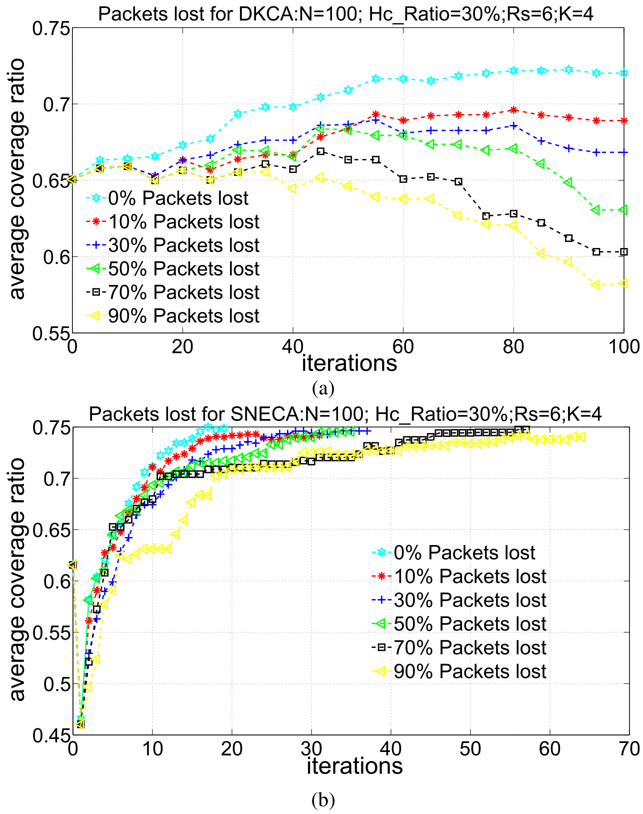


FIGURE 14. **Packets lost performance of DKCA and SNECA. (a) Packets lost performance of DKCA. (b) Packets lost performance of SNECA.**

First, we assume there is some catastrophic failure, whereby a large fraction of the sensors in network fail suddenly at a particular iteration. Fig.10(a) shows that at most 90 iterations are needed for the remaining sensors to run DKCA, while at most 7 iterations are needed(after all the sensors converge to the Nash Equilibrium by using 16 iterations) for the remaining sensors to reach the new Nash Equilibrium by SNECA shown in Fig.10(b). In Fig.11(b), we make the recovery process start at iteration 9, after which the Nash Equilibrium constitutes for all the N sensors(The iterations should be divided by the number N for calculating the convergence time for PRBGGA), there are 10%, 20%, 30%, 40% and even 50% of sensor fail respectively. At most 5 iterations are required for the remaining sensors to converge to the new Nash Equilibrium again and achieve the new-optimal solutions in all the five cases. The recovery process is the same with IDBGGA and TBGGA, which are presented in Fig.11(a) and 11(c) respectively. From above analysis, the convergence speed of the proposed IDBGGA and TBGGA(at most 9 iterations needed) are much faster than the traditional DKCA and SNECA(at least 16 iterations needed)when they constitute to the Nash Equilibrium. However, the proposed PRBGGA may need much more iterations to reach the Nash Equilibrium for its randomized property in nature. Besides, the same result can be gained when some fraction of the sensors fail suddenly at a particular iteration. In such cases, the remained sensors still can reach the Nash Equilibrium faster by adopting the proposed IDBGGA and TBGGA than DKCA and SNECA for all the five situations.

Second, we consider the situation where a fraction of the sensors fail at each iteration(after all the sensors have reached the Nash Equilibrium). We use this phenomenon to indicate there are some sensors with little energy left at that time. We assume that there are 1.2%, 2%, 2.8% and 4% of the sensors fail at the end of each iteration respectively. Fig.12(c) and Fig.13(a), 13(b), 13(c) show that, when the failure ratio is smaller than 2%, the coverage performance of SNECA and GGAs degrades slowly. While the failure ratio is bigger than 2%, the slope of the degradation of

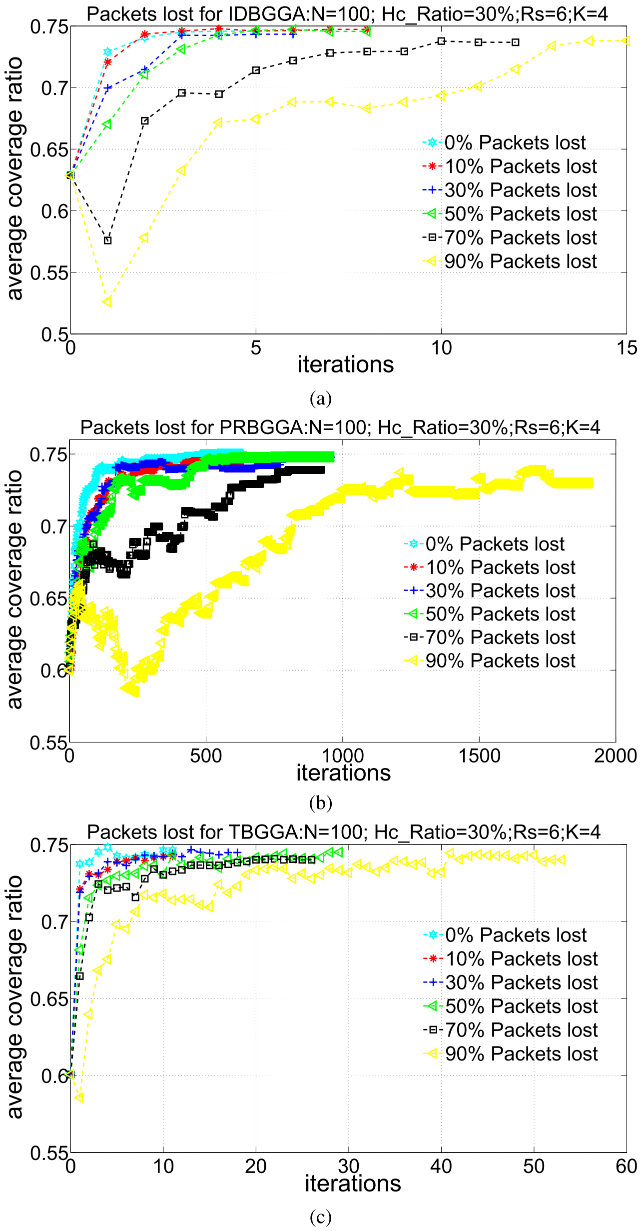


FIGURE 15. **Packets lost performance of GGAs. (a) Packets lost performance of IDBGGA. (b) Packets lost performance of PRBGGA. (c) Packets lost performance of TBGGA.**

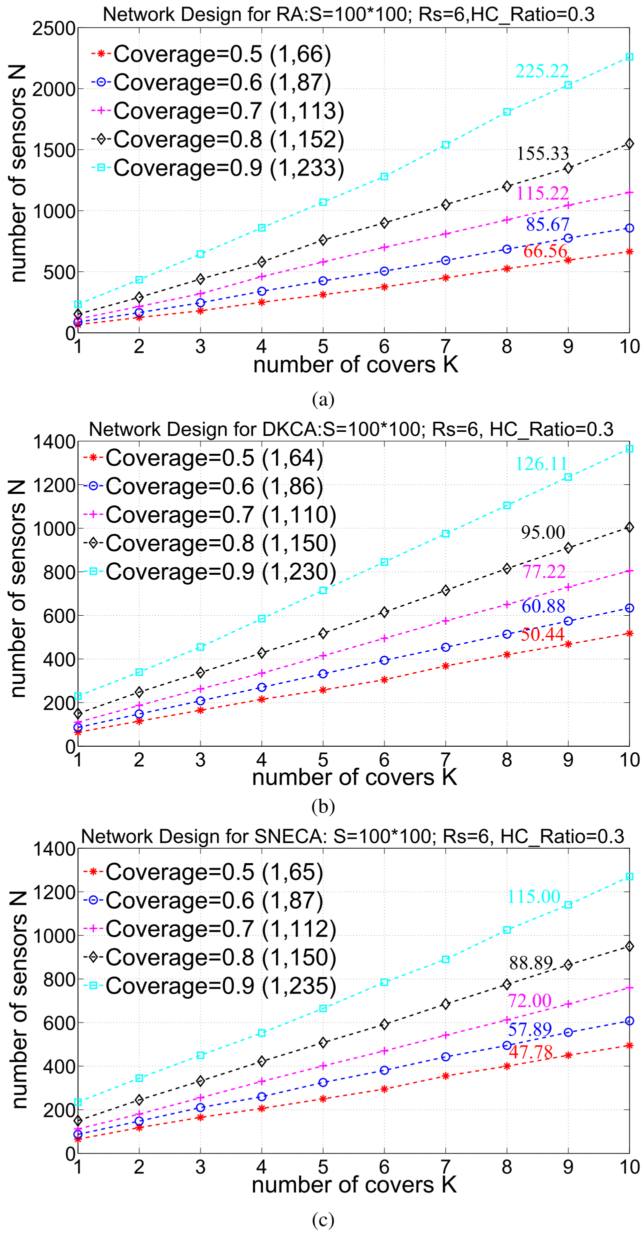


FIGURE 16. **Network design performance of the traditional algorithms. (a) Network design performance of RA. (b) Network design performance of DKCA. (c) Network design performance of SNECA.**

the coverage performance becomes relatively bigger than the small failure ratios. The coverage ratio fluctuates during the sensor failure process. This phenomenon indicates that the heterogeneity becomes more and more obviously as more sensors die. In short, degradation ratios of the coverage performance is relatively slow during the whole sensor failure process for all the four cases. The performance of GGAs is much better than that of RA and DKCA, which is shown in Fig.12(a) and 12(b). That indicates IDBGGA, PRBGGA(For fair comparison, according to the mechanism of PRBGGA and N equals to 100, we make 100 iterations as one cycle of the sensors' failure for PRBGGA.) and TBGGA are robust to the frequent sensor failures. The frequent sensor failures' performance of GGAs is comparable with SNECA.

Third, we investigate the performance of GGAs and compare it with DKCA and SNECA when the information exchanged is lost with probability p. In Fig.14 and 15, we demonstrate the coverage performance of DKCA, SNECA and GGAs with $p = 0, 0.1, 0.3, 0.5, 0.7$ and even up to 0.9 respectively. It is obvious that SNECA and GGAs can achieve nearly the same coverage ratio for all the p values, and the coverage performance of them are much better than that of DKCA. However, the convergence time of SNECA and GGAs increase smoothly as the packets lost

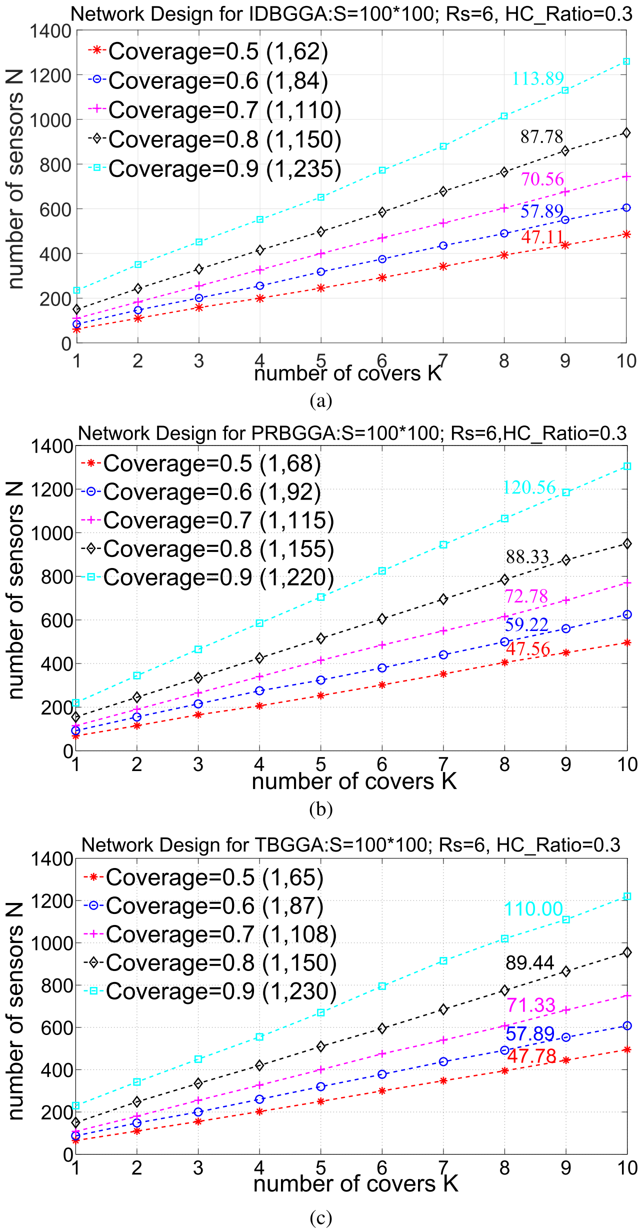


FIGURE 17. Network design performance of GGAs. (a) Network design performance of IDBGGA. (b) Network design performance of PRBGGA. (c) Network design performance of TBGGA.

ratio p increases. When the packets lost ratio p equals to 0.9, the convergence time required maybe a little longer than the smaller p value. More and more exchanged information lost as p increases, and more and more sensors will get the cover set chosen strategies' information of its neighbors which is out of data, and make wrong decisions at that time. That is the true reason why it causes the longer convergence time. In a word, the convergence time increases relatively slowly as the packets lost ratio increases and the coverage performance is almost not affected by the packets lost phenomenon. Besides, The convergence time of GGAs is much shorter than that of SNECA under all the packet lost situations.

Finally, we discuss the impact on network design by adopting GGAs, RA, DKCA and SNECA. For a fixed $F = 100 \times 100, R_s = 6, HC_{Ratio} = 0.3$ and the specific coverage requirement, we investigate how N varies with K. As Fig.16 and Fig.17 show, we observe that N increases almost linearly with K and the slopes of four curves. The number of sensors required when $K = 1$(denoted by $N1$) are also denoted in the legend of Fig.16 and Fig.17. The linear nature of the curves is not surprising, what we are really interested in is the slope of the cures.

For RA, e.g., (1, 66) in the legend implies that, when $K = 1$, 66 sensors are required for a coverage ratio of 0.5, while the actual number of sensors required which represents the slope over the line is 66.56 in Fig.16(a). We observe that the slope value of each line is nearly the same as the value in legend for each coverage requirement. This indicates that the sensors needs increases several times over for each coverage requirement as the number of K increases. For DKCA, (1,64) in the legend implies that, when $K = 1$, 64 sensors are required for a coverage ratio of 0.5, while the actual number of sensors required which represents the slope over the line is 50.44 in Fig.16(b). The same meanings are presented for SNECA, IDBGGA, PRBGGA and TBGGA, which are shown in Fig.16(c), Fig.17(a), 17(b), and 17(c) respectively. We observe that the sensor needs for each special coverage requirement are nearly the same among SNECA and GGAs. That is, the above proposed GGAs and SNECA have the same optimal property for network design, and the performance of them is much better than that of RA and DKCA.

VI. CONCLUSION AND FUTURE WORK

In this paper, we mainly introduce three kinds of novel and pure distributed algorithms(GGAs) to solve the SET K-Cover problem in HWSNs. Firstly, the classic heterogeneous sensing models are established and deeply investigated. Secondly, we deeply studied the GGAs, including the convergence property and Nash Equilibrium existence. Specially, the GGAs are totally pure distributed optimized algorithms based on the classic non-cooperative game theory. From the point of view of the selfish sensors, each one wants to maximize its payoff value(sensing area alone) as well as maximize the average coverage ratio, which is considered as the potential game behavior. When the Nash Equilibrium strategies constitute, the maximum average coverage ratio reaches at the same time. Thirdly, extensive experiments are conducted to verify the excellent performance of GGAs, including the coverage performance, convergence, robustness and network designing properties in HWSNs, in which comparing with the RA, DKCA and SNECA. Our future work mainly focus on the following aspect to better solve the SET K-Cover problem in HWSNs. In practical situation, there are four kinds of sensors which exist in the practical *HWSNs*. In this research paper, we named them as equal neighbor, explicit neighbor, potential neighbor and blind neighbor. During implementing *GGAs*, only the former three kinds of neighbors can exchange information among each other. Though we have conducted extensive experiments to verify that *GGAs* have shown the superiority in the coverage performance, convergence property, robustness and network design characteristic. Besides that, the average coverage ratio can almost remain the same comparing with the homogeneous sensor networks as the heterogeneous ratio increases. We believe that the blind neighbors' effects can still exist during the procedure of *GGAs*. Especially, the blind neighbors effects on the average coverage ratio. Though the bad effects of the blind neighbors can be weakened by conducting the *GGAs* iteratively and have been verified in the extensive experiments. However, we still definitely to say that, the solution which we obtained is a sub-optimal point of the whole optimal solution space. We believe that there are some other methods to solve the information exchanging problems between the unequal neighbor sensors.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers with their instructive comments for improving the quality of this paper.

REFERENCES

[1] K. Sakai, M.-T. Sun, W.-S. Ku, T. H. Lai, and A. V. Vasilakos, "A framework for the optimal k-coverage deployment patterns of wireless sensors," *IEEE Sensors J.*, vol. 15, no. 12, pp. 7273–7283, Dec. 2015.

[2] G. J. Han, L. Liu, J. Jiang, L. Shu, and G. Hancke, "Analysis of energy-efficient connected target coverage algorithms for industrial wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 135–143, Feb. 2017.

[3] J. Yu, S. Wan, X. Cheng, and D. Yu, "Coverage contribution area based k-coverage for wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 9, pp. 8510–8523, Sep. 2017.

[4] C. Yang and K.-W. Chin, "On nodes placement in energy harvesting wireless sensor networks for coverage and connectivity," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 27–36, Feb. 2017.

[5] F. Li, J. Luo, S. Xin, and Y. He, "Autonomous deployment of wireless sensor networks for optimal coverage with directional sensing model," *Comput. Netw.*, vol. 108, pp. 120–132, Oct. 2016.

[6] O. M. Alia and A. Al-Ajouri, "Maximizing wireless sensor network coverage with minimum cost using harmony search algorithm," *IEEE Sensors J.*, vol. 17, no. 3, pp. 882–896, Nov. 2017.

[7] Z. Zhang, J. Willson, Z. Lu, W. Wu, X. Zhu, and D.-Z. Du, "Approximating maximum lifetime k-coverage through minimizing weighted k-Cover in homogeneous wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 24, no. 6, pp. 3620–3633, Dec. 2016.

[8] F. Carrabs, R. Cerulli, C. D'Ambrosio, and A. Raiconi, "A hybrid exact approach for maximizing lifetime in sensor networks with complete and partial coverage constraints," *J. Netw. Comput. Appl.*, vol. 58, pp. 12–22, Dec. 2015.

[9] Q. Yang, S. He, J. Li, J. Chen, and Y. Sun, "Energy-efficient probabilistic area coverage in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 1, pp. 367–377, Jan. 2015.

[10] J. N. Al-Karaki and A. Gawanmeh, "The optimal deployment, coverage, and connectivity problems in wireless sensor networks: Revisited," *IEEE Access*, vol. 5, pp. 18051–18065, Aug. 2017.

[11] M. Mukherjee, L. Shu, L. Hu, G. P. Hancke, and C. Zhu, "Sleep scheduling in industrial wireless sensor networks for toxic gas monitoring," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 106–112, Aug. 2017.

[12] H. M. Ammari, "CSI: An energy-aware cover-sense-inform framework for k-covered wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 4, pp. 651–658, Apr. 2012.

[13] H. P. Gupta, P. K. Tyagi, and M. P. Singh, "Regular node deployment for k-coverage in m-connected wireless networks," *IEEE Sensors J.*, vol. 15, no. 12, pp. 7126–7134, Dec. 2015.

[14] S. Selina *et al.*, "α-overlapping area coverage for clustered directional sensor networks," *Comput. Commun.*, vol. 109, pp. 89–103, Sep. 2017.

[15] Y. Wang, S. Wu, Z. Chen, X. Gao, and G. Chen, "Coverage problem with uncertain properties in wireless sensor networks: A survey," *Comput. Netw.*, vol. 123, pp. 200–232, Aug. 2017.

[16] M. Chenait, B. Zebbane, C. Benzaid, and N. Badache, "Energy-efficient coverage protocol based on stable and predictive scheduling in wireless sensor networks," *Comput. Netw.*, vol. 127, pp. 1–12, Nov. 2017.

[17] Z. Abrams, A. Goel, and S. Plotkin, "Set k-cover algorithms for energy efficient monitoring in wireless sensor networks," in *Proc. 3rd Int. Symp. Inf. Process. Sensor Netw.*, Berkeley, CA, USA, Apr. 2004, pp. 424–432.

[18] X. Ai, V. Srinivasan, and C. K. Tham, "Optimality and complexity of pure Nash equilibria in the coverage game," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 7, pp. 1170–1182, Sep. 2008.

[19] S. Fujita, "Distributed algorithm for set k-cover problem and its worst case analysis," in *Proc. 3rd Int. Conf. Netw. Comput.*, Okinawa, Japan, Dec. 2012, pp. 33–40.

[20] Q. Wang, W. J. Yan, and Y. Shen, "N-person card game approach for solving SET K-COVER problem in wireless sensor networks," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 5, pp. 1522–1535, Feb. 2012.

[21] P. Nagarathna and R. Manjula, "Genetic algorithm with a new fitness function to enhance WSN lifetime," in *Proc. Int. Conf. Appl. Theor. Comput. Commun. Technol.*, Davangere, India, Oct. 2015, pp. 95–100.

[22] C. Sun, "A time variant log-linear learning approach to the SET K-COVER problem in wireless sensor networks," *IEEE Trans. Cybern.*, vol. 48, no. 4, pp. 1316–1325, Apr. 2017.

[23] C.-C. Liao and C.-K. Ting, "A novel integer-coded memetic algorithm for the setk-cover problem in wireless sensor networks," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2245–2258, Aug. 2018.

[24] D. Monderer and L. S. Shapley, "Potential games," *Games Econ. Behavior*, vol. 14, no. 1, pp. 124–143, 1996.

[25] J. F. Nash, Jr., "Equilibrium points in n-person games," *Proc. Nat. Acad. Sci. USA*, vol. 36, no. 1, pp. 48–49, 1950.

[26] T. Basar, G. J. Olsder, G. Clsder, T. Basar, T. Baser, and G. J. Olsder, *Dynamic Noncooperative Game Theory*, vol. 200. Philadelphia, PA, USA: SIAM, 1995, pp. 77–159.

[27] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," in *Proc. IEEE Int. Conf. Commun.*, Helsinki, Finland, Jun. 2001, pp. 472–476.

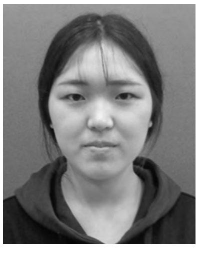
[28] H. Zhang and J. C. Hou, "On the upper bound of α-lifetime for large sensor networks," *ACM Trans. Sensor Netw.*, vol. 1, no. 2, pp. 272–300, Nov. 2005.

[29] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, "Grid coverage for surveillance and target location in distributed sensor networks," *IEEE Trans. Comput.*, vol. 51, no. 12, pp. 1448–1453, Dec. 2002.

[30] A. Howard, M. J. Mataric, and G. Sukhatme, "Relaxation on a mesh: A formalism for generalized localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Expanding Societal Role Robot. Next Millennium*, Maui, HI, USA, Oct./Nov. 2001, pp. 1055–1060.

[31] C. Savarese, J. M. Rabaey, and J. Beutel, "Location in distributed ad-hoc wireless sensor networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Salt Lake City, UT, USA, May 2001, pp. 2037–2040.

WENJIE YAN received the Ph.D. degree from the Harbin Institute of Technology. He is currently a Lecturer with the School of Artificial Intelligence, Hebei University of Technology. His research interests include wireless sensor networks, game theory, machine learning, and big data analysis.



MENGJING CAO received the bachelor's degree from Hebei Agriculture University. She is currently pursuing the master's degree with the School of Artificial Intelligence, Hebei University of Technology. Her research interests include wireless sensor networks, game theory, machine learning, and big data analysis.

YOUXI WU received the Ph.D. degree in theory and new technology of electrical engineering from the Hebei University of Technology, Tianjin, China. He is currently a Ph.D. Supervisor and a Professor with the Hebei University of Technology. His current research interests include data mining and machine learning. He is a Senior Member of CCF.

JUN ZHANG received the Ph.D. degree from the School of Electrical Engineering, Hebei University of Technology. He is currently an Associate Professor with the School of Artificial Intelligence, Hebei University of Technology. His current research interests include embedded system and intelligent computing.