

RESEARCH

Open Access



A novel centralized algorithm for constructing virtual backbones in wireless sensor networks

Chuanwen Luo¹, Wenping Chen¹, Jiguo Yu², Yongcai Wang¹ and Deying Li^{1*}

Abstract

Finding the minimum connected dominating set (MCDS) is a key problem in wireless sensor networks, which is crucial for efficient routing and broadcasting. However, the MCDS problem is NP-hard. In this paper, a new approximation algorithm with approximation ratio $H(\Delta) + 3$ in time $O(n^2)$ is proposed to approach the MCDS problem. The key idea is to divide the sensors in CDS into *core* sensors and *supporting* sensors. The core sensors dominate the supporting sensors in CDS, while the supporting sensors dominate other sensors that are not in CDS. To minimize the number of both the cores and the supporters, a three-phased algorithm is proposed. (1) Finding the base-core sensors by constructing independent set (denoted as S_1), in which the sensors who have the largest $\frac{|N^2(v)|}{|N(v)|}$ (number of two-hop neighbors over the number of one-hop neighbors) will be selected greedily into S_1 ; (2) Connecting all base-core sensors in S_1 to form a connected subgraph, the sensors in the subgraph are called cores; (3) Adding the one-hop neighbors of the core sensors to the supporter set S_2 . This guarantees a small number of sensors can be added into CDS, which is a novel scheme for MCDS construction. Extensive simulation results are shown to validate the performance of our algorithm.

Keywords: Wireless sensor network, Virtual backbone, Connected dominating set, Unit disk graph

1 Introduction

Wireless sensor networks (WSNs) play a critical role in many areas, such as environmental monitoring, disaster forecast, etc [1]. A key problem in WSN is multi-hop communication, because the communication range of a individual sensor is generally limited. In multi-hop communication, any two sensors that are within the communication range of each other are called *neighbors*, which can communicate to each other. Other sensors that are not within the communication range of each other and want to communicate, need intermediate sensors between them to forward their packets (for instance, sensory data [2, 3] and image data [4, 5]).

However, due to the broadcasting nature of the wireless communication, if there is not a specific routing path for packet forwarding, all neighbors are possible to become

intermediators for forwarding messages, which causes *message flooding* problem. The key way to avoid flooding is to find a communication backbone, so that the packets are relayed by the backbone sensors to save energy for the other sensors.

If modeling the WSN into an undirected graph, the connected dominating set (CDS) [6–8] is one of the good choices to construct virtual backbone of the network, because the sensor nodes in the CDS form a connected subgraph to forward messages from other sensors.

However, forwarding message may run into collision, which introduces retransmissions and increases end-to-end delays. As the number of sensors in the CDS grows, the negative effect of retransmissions increases greatly. Hence, CDS with smaller number of sensors is highly desired, which leads to the problem of finding the CDS with the minimum number of sensors, i.e., the minimum connected dominating set (MCDS) problem. However, it has been proved that the MCDS problem is NP-hard [9]. Therefore, approximation algorithms become the focus of addressing the MCDS problem. The majority of proposed

*Correspondence: deyingli@ruc.edu.cn

¹School of Information, Renmin University of China, Zhongguancun Road, 100872 Beijing, People's Republic of China

Full list of author information is available at the end of the article

algorithms in literatures follow a general two-phased approach [10–14]. In the first phase, a dominating set is constructed, and the sensors in the dominating set are called *dominators*. In the second phase, additional sensors are selected, called *connectors*. Together with the dominators, they induce a connected CDS topology.

In this paper, we design a three-phased approximation algorithm for the MCDS problem in WSNs. Firstly, we propose a novel method to construct an independent set S_1 for the graph G such that any pair of complementary sensor subsets for S_1 is separated by exactly three hops. Secondly, sensors in S_1 are connected by other sensors that are added into C to form a subtree. The number of sensors in C is an even number, since any pair of complementary sensor subsets in S_1 is separated by two sensors. A supporter set S_2 is constructed that neighbors of $S_1 \cup C$ are added into S_2 . $S_1 \cup C \cup S_2$ is connected dominated set. The performances of the proposed algorithms are thoroughly analyzed.

Our contributions are presented as follow:

- We propose a novel algorithm to generate the CDS and construct the virtual backbones in WSNs.
- We analyze the performance ratio and time complexity of our algorithm.
- We conduct extensive simulations to demonstrate the performance of the algorithms. Simulation results show that the algorithm generates CDS with smaller size than the state-of-the-art algorithms in [15].

The rest of the paper is organized as follows. Related work is reviewed in Section 2. Our novel centralized algorithm for constructing a CDS is presented in Section 3. The performance of the proposed algorithm is thoroughly analyzed in Section 4. Section 5 gives the results of simulations, which show the performance of the algorithm. Finally, we conclude this paper in Section 6.

2 Related works

In this section, we review the classical algorithms for constructing CDS. For more comprehensive approximation algorithms for CDS construction, one can refer to Du and Wan and Yu et al. [16, 17]. Since the MCDS problem in unit disk graph is NP-hard, many algorithms are proposed to compute approximation solutions. CDS construction algorithms can be divided into distributed algorithms and centralized algorithms.

2.1 Distributed algorithms

In the case of distributed algorithms, each node in the network only knows the local information and communicates with its neighbors. Recently, the popular methods for constructing CDS are to firstly construct an maximal independent set (MIS), then a CDS is formed by connecting the nodes in the MIS, such as [7, 8, 11–13].

In [7], Wan et al. proposed an ID-based distributed algorithm to construct a CDS with the performance ratio $8|opt| - 2$, where opt represents the minimum connected dominating set of the unit disk graph. In [8, 11, 12], some MIS-based algorithms are proposed and the first phase of these algorithms is to construct an MIS as shown in [7]. In the second phase of the algorithm in [8], Li et al. constructed a Steiner tree for connecting all nodes in MIS. The performance ratio of their algorithm is $(4.8 + \ln 5)|opt| + 1.2$. In [11] Min et al. improved the construction of Steiner tree to decrease the size of connectors. Consequently, they proved that the approximation ratio of the proposed algorithm is 6.8. In [12], Wan et al. proved the approximation ratio of [7] is 7.333 and proposed a new approximation algorithm with ratio 6.389. In [13], Misra et al. proposed a heuristic algorithm, called collaborative cover, to obtain an MIS. After that, they constructed a Steiner tree with minimum number of Steiner nodes to obtain a small CDS. The size of the CDS they got is at most $(4.8 + \ln 5)|opt| + 1.2$.

2.2 Centralized algorithms

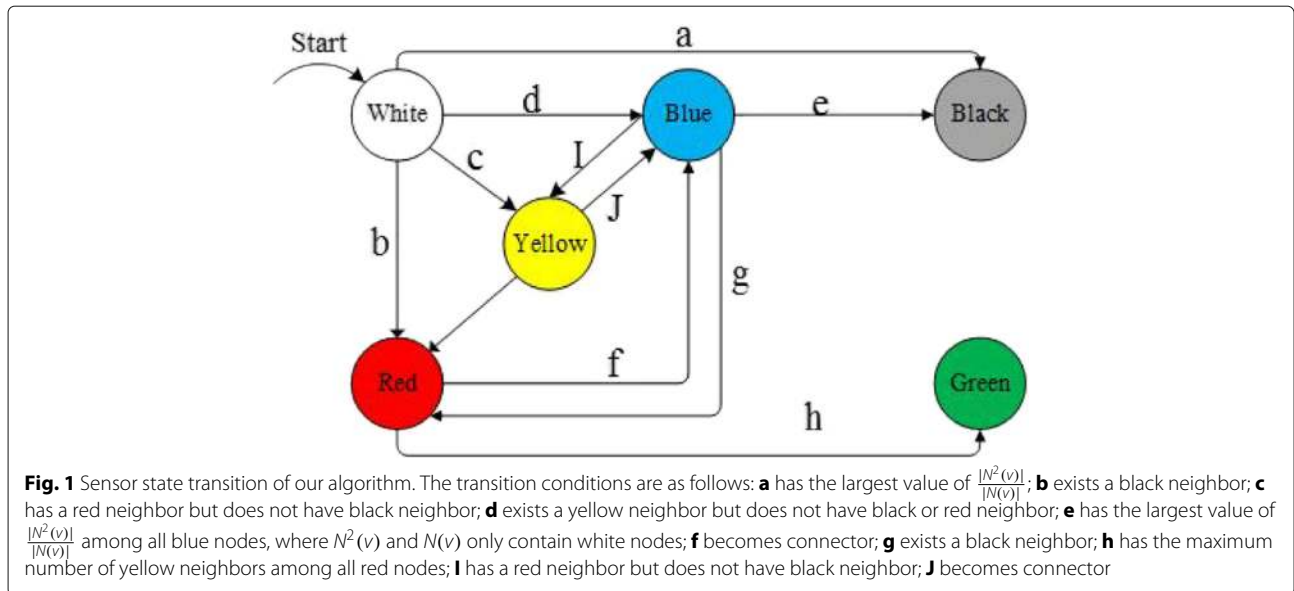
In the literature, Guha and Kuller [6] proposed the first approximation algorithm to construct an MCDS as a virtual backbone in a wireless network. They presented two centralized greedy algorithms for CDS construction with approximation factor $2H(\Delta) + 2$ and $H(\Delta) + 2$ respectively, where Δ is the maximum degree of the graph. In [18], Ruan et al. proposed another centralized algorithm with the approximation factor $\ln \Delta + 2$. In [19], Fu et al. proposed a centralized algorithm for CDS construction with the time complexity $O(n\Delta^2)$. Note that Δ can be as many as $O(n)$. Thus, the time complexity of the algorithm in [19] is $O(n^3)$.

In [15], Al-Nabhan et al. proposed three similar centralized algorithms to construct CDSs in wireless network with approximation factor of 5. These approximation algorithms outperform the existing state-of-the-art methods. Their algorithm contains four phases. The first phase is to construct a special independent set S_1 and any pair of complementary subsets of S_1 is separated by exactly three hops. The second phase is to compute an MDS for each disconnected component and all nodes in MDS form the set S_2 . The third phase is to connect S_2 nodes and S_1 nodes. The fourth phase is to connect all nodes in S_1 .

Some other centralized CDS construction algorithms also exist in the literatures [20–23].

The MCDS has many applications in the special network models, such as ad hoc networks [24, 25], energy harvest networks [26], battery-free networks [27], cognitive ratio networks [28], and others [29–31].

In this paper, we propose a three-phased approximation algorithm for CDS construction with approximation ratio $H(\Delta) + 3$ in time $O(n^2)$. To compare with the three



algorithms proposed in [15], extensive simulations are conducted, and the results show effectiveness of our algorithm. A preliminary version [32] was published in WASA 2017.

3 MCDS construction

3.1 Model

For simplicity, all sensors in WSN are randomly deployed in the two-dimensional plane. Assume that all sensors have the same transmission range in the network. The WSN is modeled as a unit disk graph $G(V, E)$, where V is the set of all sensors and E represents the set of links in the network. If the Euclidean distance between any two sensors u and v is less than or equal to 1, then there is an undirected edge e_{uv} between these two sensors. Each sensor $v \in V$ has a unique ID. Let $N(v)$ be the set of all neighbors of v and $d_v = |N(v)|$ be the degree of v . Denote $\Delta = \max\{d_v | \forall v \in V\}$ and $N^i(v)$ to be the i -hops neighbor set of v .

3.1.1 Connected dominating set (CDS)

A dominating set (DS) of a graph $G = (V, E)$ is a subset $V' \subseteq V$ such that each node in $V \setminus V'$ is adjacent to at least one node in V' , and a connected dominating set (CDS) is a dominating set which also induces a connected subgraph.

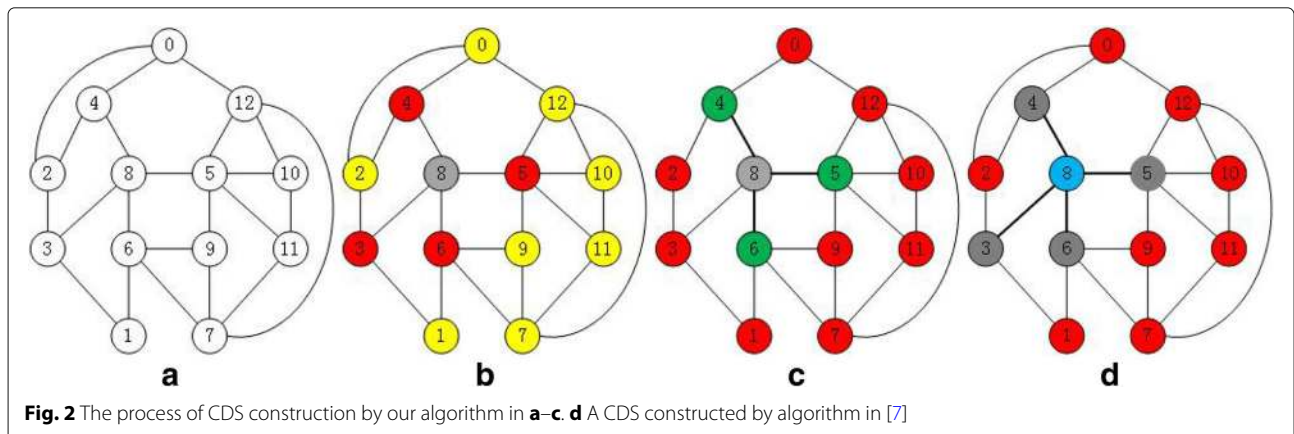
3.1.2 Minimum connected dominating set (MCDS) problem

Given a graph $G = (V, E)$, the minimum connected dominating set problem is to find the CDS in G such that the size of the CDS is minimized.

In this paper, we propose a novel approximation algorithm for solving the MCDS problem.

3.2 Algorithm overview

In this section, we overview the proposed approximation algorithm for the MCDS problem. The algorithm consists of three phases.



- In the first phase, we construct an independent set S_1 (sensors in S_1 called *base-cores*) for the graph G such that any pair of complementary sensor subsets in S_1 is separated by exactly three hops, which differs from the construction process of the first phase in [15].
- In the second phase, we select connectors from $V \setminus S_1$ to connect the base-core sensors in S_1 for obtaining a subtree, called all sensors on the subtree as *cores*.
- In the third phase, we construct a supporter set S_2 such that neighbors of $S_1 \cup C$ are added into S_2 . $S_1 \cup C \cup S_2$ forms a CDS.

For an illustrative purpose, we employ the different colors to differentiate sensor states during the construction process of our algorithm. Figure 1 shows the state transition process of sensors in the WSN.

We illustrate the CDS construction process of our algorithm by Fig. 2, which is the same network example $G(V, E)$ in [7]. Initially, all sensors are marked as white and each sensor has a unique ID, as shown in Fig. 2a. In the first phase, we can know that node 8 has the largest value of $\frac{|N^2(v)|}{|N(v)|}$ among all sensors in the graph. Hence, sensor 8 is colored black and all neighbors in $N(8)$ are colored red and all sensors in $N^2(8)$ are colored yellow. As shown in Fig. 2b, sensors 3, 4, 5, and 6 are colored red and sensors 0, 1, 2, 7, 9, 10, 11, and 12 are colored yellow. None of sensors become connector in the second phase since only one black sensor 8 is added into independent set S_1 . In the third phase, we need to select supporters (added into S_2) from red sensors to dominate all yellow sensors. For all red sensors, sensor 5 has the maximum number of yellow neighbors, then sensor 5 is marked green and its yellow neighbors 9, 10, 11, and 12 are colored red. After that, sensors 6 and 4 have the same number of yellow neighbors and the ID of sensor 6 is larger than sensor 4; therefore, sensor 6 is marked green and its yellow neighbors 1 and 7 are colored red. Then sensor 4 is marked green and sensors 0 and 2 are colored red. Finally, sensors with black and green form a CDS that contains sensors 4, 5, 6, and 8, as shown in Fig. 2c. Figure 2d shows a CDS (blue and black sensors) obtained by the algorithm in [12].

3.3 Independent set S_1 construction

In this section, we construct the set S_1 such that the hop distance between any two complementary sensor subsets in S_1 is exactly three hops. The details of S_1 construction process as shown in the following steps.

First, a sensor $v \in V$ with the largest value of $\frac{|N^2(v)|}{|N(v)|}$ initiates the S_1 construction by coloring itself black. Then, the black sensor v dominates its neighbors in $N(v)$ and all sensors in $N(v)$ are marked red. After that, we color all sensors in $N^2(v)$ as yellow and all sensors in $N^3(v)$ are colored blue. Last, each blue sensor u deletes red sensors

from the set $N^2(u)$ and deletes yellow sensors from the set $N(u)$.

Then select black sensor from the current blue sensors, for this purpose, the algorithm repeats the following steps, until no blue/white sensors is left in the graph.

We select a blue sensor v and color it black when the value of $\frac{|N^2(v)|}{|N(v)|}$ is largest among all blue sensors. If more than one sensor node have the same value of $\frac{|N^2(v)|}{|N(v)|}$, then the algorithm selects the blue sensor with the maximum number of sensors in $N(v)$. If more than one blue sensor have the same value of $|N(v)|$, then the algorithm selects the blue sensor with the highest ID value among these blue sensors.

After that, the algorithm executes the following operations:

- All sensors in $N(v)$ are colored red
- All sensors in $N^2(v)$ are colored yellow
- All sensors in $N^3(v)$ are colored blue
- Each blue sensor u deletes red sensors from the set $N^2(u)$ and deletes yellow sensors from the set $N(u)$

The detail illustration as shown in Algorithm 1.

Algorithm 1 S_1 Construction

- 1: **Input:** $G(V, E)$
 - 2: **Output:** S_1
 - 3: Sets of $S_1 \leftarrow \emptyset$;
 - 4: All sensors in V are marked white;
 - 5: Choose an initiator $v \in V$ with the maximum $\frac{|N^2(v)|}{|N(v)|}$ among all sensors in V ;
 - 6: $Color_v = black$; $S_1 = S_1 \cup \{v\}$;
 - 7: **for** each sensor $u \in N(v)$ **do**
 - 8: $Color_u = red$;
 - 9: **end**
 - 10: **for** each sensor $u \in N^2(v)$ **do**
 - 11: $Color_u = yellow$;
 - 12: **end**
 - 13: **for** each sensor $u \in N^3(v)$ **do**
 - 14: $Color_u = blue$;
 - 15: **end**
 - 16: **for** each blue sensor w **do**
 - 17: Delete red sensors from the set $N^2(w)$ and delete yellow sensors from the set $N(w)$;
 - 18: **end**
 - 19: Select blue sensor w with the largest value $\frac{|N^2(w)|}{|N(w)|}$ among all blue sensors and set $v = w$;
 - 20: Repeat line 4-17 until all sensors in V are black or red or yellow;
 - 21: **return** S_1 ;
-

After Algorithm 1 terminates, the sensors in V are either black, red, or yellow. We obtain an independent set S_1 that

is composed of black sensors and any red sensor is definitely dominated by a black sensor and any yellow sensor has two hops distance from a black sensor. We can prove that any pair of complementary sensor subsets of S_1 is separated by exactly three hops. The sensors in the set S_1 are called *base-cores*.

3.4 Connector set C construction

In this section, we propose a novel algorithm to find a set of connectors C such that $S_1 \cup C$ forms a subtree.

Before we describe the algorithm, we introduce some terms and notations. For any subset $U \subseteq V$, let $q(U)$ be the number of connected components in $G(U)$. The set U is initially equal to S_1 , and the initial value of $q(U)$ is $|S_1|$. Let $M = \{e | e \in E \text{ and the endpoints are red and yellow}\}$ and C be the set of connectors. Let W be the subset of S_1 such that any pair of sensors of W is connected by other sensors in C .

To begin our algorithm, first, we select an arbitrary black sensor $s_1 \in S_1$ to start selection of connectors and set $W = \{s_1\}$. The algorithm repeats the following steps, until the condition $q(U) = 1$ is satisfied:

- Select a sensor $s_i \in W$ such that there exists a sensor $s_j \in N^3(s_i) \cap (S_1 - W)$
- Select an edge $e_{xy} \in M$ such that $x \in N(s_j)$ and $y \in N(s_i)$
- Delete the edge e_{xy} from M , then sensors x and y are marked blue and added into C
- For each yellow sensor w , if $w \in N(x)$ or $w \in N(y)$, then it is marked red
- Execute operations $U = U \cup \{u\}$, $U = U \cup C$ and $q(U) = q(U) - 1$

The detail illustration as shown in Algorithm 2.

After Algorithm 2 terminates, any two black sensors are connected by a path that consists of black sensors and blue sensors. That is, we obtain a subtree and all sensors on the subtree are called *cores*.

3.5 Supporter set S_2 construction

After executing Algorithm 2, we have got a subtree over on $S_1 \cup C$. However, there are still some yellow sensors not being dominated since they have two hops distance from black sensor or blue sensor.

In this section, we propose a novel greedy algorithm for acquiring a supporting set S_2 , in which the sensors are used to dominate remaining yellow sensors. Sensors in the set S_2 are called *supporter*.

Let RD be the set $\{s | s \in V, Color_s = red\}$ and YL be the set $\{s | s \in V, Color_s = yellow\}$. In each iteration, we select a red sensor $s \in RD$ with the maximum number of yellow sensors in $N(s)$. If more than one red sensors

Algorithm 2 Connecting S_1 sensors

```

1: Input:  $G(V, E), S_1$ 
2: Output:  $C$  (The set of connectors)
3: Let  $U = S_1, W, C \leftarrow \emptyset, M = \{e | e \in E \text{ and the endpoints are red and yellow}\}$ ;
4:  $q(U) = |S_1|$ ;
5: Select arbitrary sensor  $s_1 \in S_1$ ;
6:  $W = W \cup \{s_1\}$ ;
7: while  $q(U) > 1$  do
8: Select a sensor  $s_i \in W$  such that there exists a sensor  $s_j \in N^3(s_i) \cap (S_1 - W)$ ;
9: Select an edge  $e_{xy} \in M$  such that  $x \in N(s_j)$  and  $y \in N(s_i)$ ;
10:  $M = M \setminus e_{xy}, Color_x = blue, Color_y = blue, C = C \cup \{x, y\}$ ;
11: for each sensor  $w \in N(x)$  or  $w \in N(y)$  do
12:  $Color_w = red$ ;
13: end
14:  $W = W \cup \{s_j\}, U = U \cup C, q(U) = q(U) - 1$ ;
15: end
16: return  $C$ ;

```

have the same number of the yellow neighbors, then the algorithm selects the red sensor with the highest ID.

The algorithm repeats the following steps, until the condition $YL = \emptyset$ is satisfied:

- Select a red sensor $s \in V$ with the maximum number of yellow neighbors
- Sensor s is marked green and its yellow neighbors in $N(s) \cap YL$ are marked red
- Delete sensors of $N(s) \cap YL$ from YL

The detail illustration as shown in Algorithm 3.

Algorithm 3 S_2 Construction

```

1: Input:  $G(V, E), S_1, C$ 
2: Output:  $S_2$ 
3: Let  $S_2 \leftarrow \emptyset, RD = \{s | s \in V, Color_s = red\}, YL = \{s | s \in V, Color_s = yellow\}$ ;
4: while  $YL \neq \emptyset$  do
5: Select a red sensor  $s \in RD$  with the maximum number of yellow neighbors;
6:  $Color_s = green, S_2 = S_2 \cup \{s\}$ ;
7: for each sensor  $u \in N(s) \cap YL$  do
8:  $Color_u = red, RD = RD \cup \{u\}$ ;
9:  $YL = YL \setminus \{u\}$ ;
10: end
11: end
12: return  $S_2$ ;

```

3.6 CDS construction

In this section, we propose our approximation algorithm for solving MCDS problem. The algorithm consists of four steps, and the first three steps correspond to Algorithms 1–3, respectively. The last step is to compute union of S_1 , C , and S_2 . The detail illustration as shown in Algorithm 4.

After this algorithm terminates, we obtain a CDS that is union of S_1 (black sensors), C (blue sensors), and S_2 (green sensors). For a given graph $G(V, E)$, we give the executing process of the Algorithm 4, as shown in Fig. 3(a)–(d).

Algorithm 4 CDS Construction

Input: $G(V, E)$

Output: CDS

step 1: Obtain independent set S_1 by executing Algorithm 1;

step 2: Obtain connector set C by executing Algorithm 2;

step 3: Obtain supporter set S_2 by executing Algorithm 3;

step 4: $CDS = S_1 \cup C \cup S_2$;

return CDS;

4 Performance analysis

In this section, we analyze the performance ratio and time complexity of our algorithm. Let $H(n) = \sum_{i=1}^n \frac{1}{i}$ be the harmonic function and MCDS be an optimal CDS.

Lemma 1 *The set S_1 found by Algorithm 1 is an independent set, and any pair of complementary sensor subsets of S_1 is separated by exactly three hops.*

Proof We use $\{s_1, s_2, \dots, s_k\}$ to denote the set S_1 . Any two sensors $s_i, s_j \in S_1$ are not adjacent to each other according to the process of S_1 construction by Algorithm 1. Therefore, the set S_1 is an independent set of G .

Let $T_j = \{s_1, s_2, \dots, s_j\}$ and $H_j = (T_j, E_j)$ for any $1 \leq j \leq k$. For arbitrary two sensors $s_i, s_l \in T_j$, an edge $(s_i, s_l) \in E_j$ if and only if their distance in G is three. We prove by induction on j that H_j is connected. Since H_1 contains a single sensor, it is connected obviously. Assume that H_j is connected for $1 \leq j \leq k-1$, when the sensor s_{j+1} is marked black, according to the Algorithm 1, there exists $s_i \in T_i$ ($1 \leq i \leq j$) such that the distance between s_{j+1} and s_i in G is three, which means there exists an edge between s_i and s_{j+1} in H_{j+1} . Due to H_j is connected, H_{j+1} is also connected. Therefore, H_j is connected for any $1 \leq j \leq k$. This implies that any pair of complementary subsets of S_1 is exactly three hops. \square

Lemma 2 *The CDS = $S_1 \cup C \cup S_2$ got by Algorithm 4 is a connected dominating set.*

Proof According to lemma 1, we know that S_1 is an independent set and $S_1 \cup C$ is connected.

According to Algorithm 3, each sensor in S_2 is adjacent to at least one sensor in $S_1 \cup C$. Therefore, the set CDS is connected. Since the distance between any sensor not in $S_1 \cup C$ and $S_1 \cup C$ in G is at most 2, all other sensors not in CDS are dominated by sensors in CDS according to the selection process of S_2 . Therefore, for any sensor $v \in V$, it belongs to the set CDS or has at least a neighbor in CDS, which means CDS is a connected dominating set. \square

Lemma 3 *The size of S_1 is less than or equal to $|MCDS|$.*

This lemma has been proved by lemma 2 in [15].

Lemma 4 *The size of the set C obtained by Algorithm 2 is at most $2|MCDS| - 2$.*

Proof Let S_1 be the set $\{s_1, s_2, \dots, s_k\}$. According to lemma 1, we obtain that auxiliary graph H_k over S_1 is a tree. Hence, H_k contains $k-1$ edges. According to Algorithm 2, any two endpoints of an edge in H_k are two sensors in S_1 . Therefore, two connectors are added into C to connect the two sensors.

Therefore, the size of set C is $2|S_1| - 2$. By lemma 3, we get $|C|$ is at most $2|MCDS| - 2$. \square

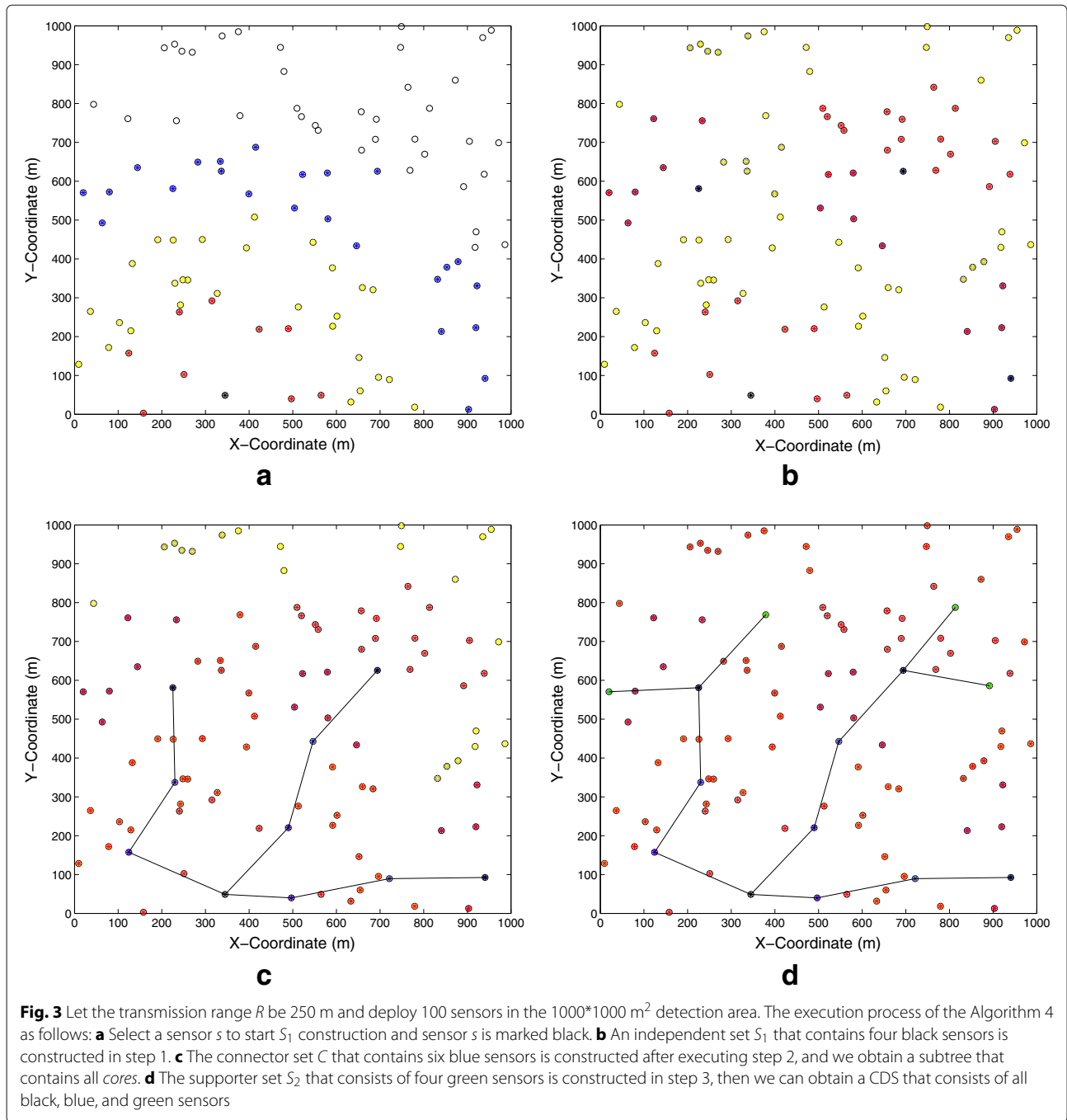
Lemma 5 *The size of the set S_2 obtained by Algorithm 3 is less than $H(\Delta)|MCDS|$.*

Proof For a sensor $v \in MCDS$, let P_v be the sensors set including v in which each sensor is dominated by v . According to Algorithm 3, when a red sensor v is marked green, all yellow neighbors of v are dominated by v .

We will prove that the total number of sensors in P_v for any node v is at most $H(\Delta)$.

Assume that when we pick a sensor v from RD to add to S_2 , y yellow sensors turn to red. We obtain that each of y yellow sensors spends at most $\frac{1}{y}$.

Assume that the number of yellow sensors is initially $y_0 < \Delta$ in P_v , and finally drops to 0. Let y_j denote the number of yellow sensors in P_v after step j . Here, we assume that some yellow sensors in P_v are marked red at each step. Therefore, the number of yellow sensors in P_v decreases at each step. After the first step, the number of sensors which changed color is $y_0 - y_1$. In the j th step, the number of sensors that change color in set P_v is $y_{j-1} - y_j$, and the cost of each sensor which changed color is at most $\frac{1}{y_j}$. Let $y_h = 0$. We can get the total number of sensors in P_v is



$$\begin{aligned}
 \sum_{j=1}^h \frac{1}{y_{j-1}} (y_{j-1} - y_j) &= \sum_{j=1}^h \sum_{i=y_j}^{y_{j-1}} \frac{1}{y_{j-1}} \\
 &\leq \sum_{j=1}^h \sum_{i=y_j}^{y_{j-1}} \frac{1}{i} \\
 &= \sum_{j=1}^h \left(\sum_{i=1}^{y_{j-1}} \frac{1}{i} - \sum_{i=1}^{y_j} \frac{1}{i} \right) \\
 &= H(y_0) < H(\Delta).
 \end{aligned}$$

Therefore,

$$|S_2| \leq \cup_{v \in MCDS} |P_v| < H(\Delta) |MCDS|.$$

This lemma is proved. □

We know that $CDS = S_1 \cup C \cup S_2$. According to lemma 3–5, we obtain the following theorem.

Theorem 1 The number of sensors in CDS found by Algorithm 4 is less than $(H(\Delta) + 3) |MCDS| - 2$.

Lemma 6 *The time complexity of Algorithm 1 is $O(n^2)$.*

Proof According to Algorithm 1, we need $|S_1|$ iterations for obtaining the set S_1 . In the first iteration, we need at most n steps to choose a sensor v with the largest value of $\frac{|N^2(v)|}{|N(w)|}$ from V . Since any black sensor comes from blue sensor, we need at most n steps to select a black sensor from blue sensors in i th iteration. Therefore, the total number of black sensor selection over all iterations is $O(n^2) = O(n|S_1|)$, since $|S_1| < n$, and we obtain that the time complexity of Algorithm 1 is $O(n^2) + O(n) = O(n^2)$. \square

Lemma 7 *The time complexity of Algorithm 2 is $O(n^2)$.*

Proof Firstly, we pick out all edges with a red endpoint and a yellow endpoint from set E . Therefore, the operation needs the running time of $O(|E|)$.

Secondly, due to the initial value of $q(U)$ is $|S_1|$, the number of iterations is less than $|S_1|$ by lemma 3. In the interior of the loop, first, we need $|W|$ steps to select a sensor $s_i \in W$ such that there exists a sensor $s_j \in N^3(s_i) \cap (S_1 - W)$. The maximum value of $|W|$ is equal to $|S_1|$. Second, we select an edge $e_{xy} \in M$ for connecting s_i and s_j such that e_{xy} is composed of an endpoint $x \in N(s_i)$ and an endpoint $y \in N(s_j)$. Therefore, we need at most 2Δ steps to select edge e_{xy} . Last, 2Δ steps are needed for coloring all sensors in $N(x) \cup N(y)$.

Therefore, the time complexity of Algorithm 2 is $O(|E| + (|W| + 2\Delta + 2\Delta) \times |S_1|) = O(n^2)$. \square

Lemma 8 *The time complexity of Algorithm 3 is $O(n^2)$.*

Proof We need n steps to pick out red sensors (added into RD) and yellow sensors (added into YL) from V . Algorithm 3 executes at most $|YL|$ iterations. In a single iteration, due to the size of RD is less than n , we need at most n steps a red sensor $v \in RD$ with the maximum number of yellow neighbors among all sensors in RD . And at most Δ steps are needed to mark all yellow neighbors of v .

Therefore, the time complexity of Algorithm 3 is $O((n + \Delta + n) \times |YL|) = O(n^2)$, since $|YL| < n$. \square

We know that Algorithm 4 consists of four steps and the first three steps correspond to Algorithms 1, 2, and 3, respectively. The last step needs single time to compute the union of S_1 , C , and S_2 . According to lemmas 6–8, we obtain the following theorem.

Theorem 2 *The time complexity of Algorithm 4 is $O(n^2)$.*

5 Simulation

In this section, we evaluate the performance of our algorithm through simulations. In the simulations, N sensors

are randomly deployed in the two-dimension plane. All sensors are assumed to have the same transmission range R . Each experimental result is the average of 100 runs. We first evaluate how the network configuration, such as the number of the sensors, the transmission range, and the area of the deployment, impact on the size of CDS, as shown in Section 5.1. After that, we compare the performance of our algorithm with the performance of the three algorithms (Approach I, Approach II, and Approach III) in [15], as shown in Section 5.2. We used MATLAB R2013a for all simulations.

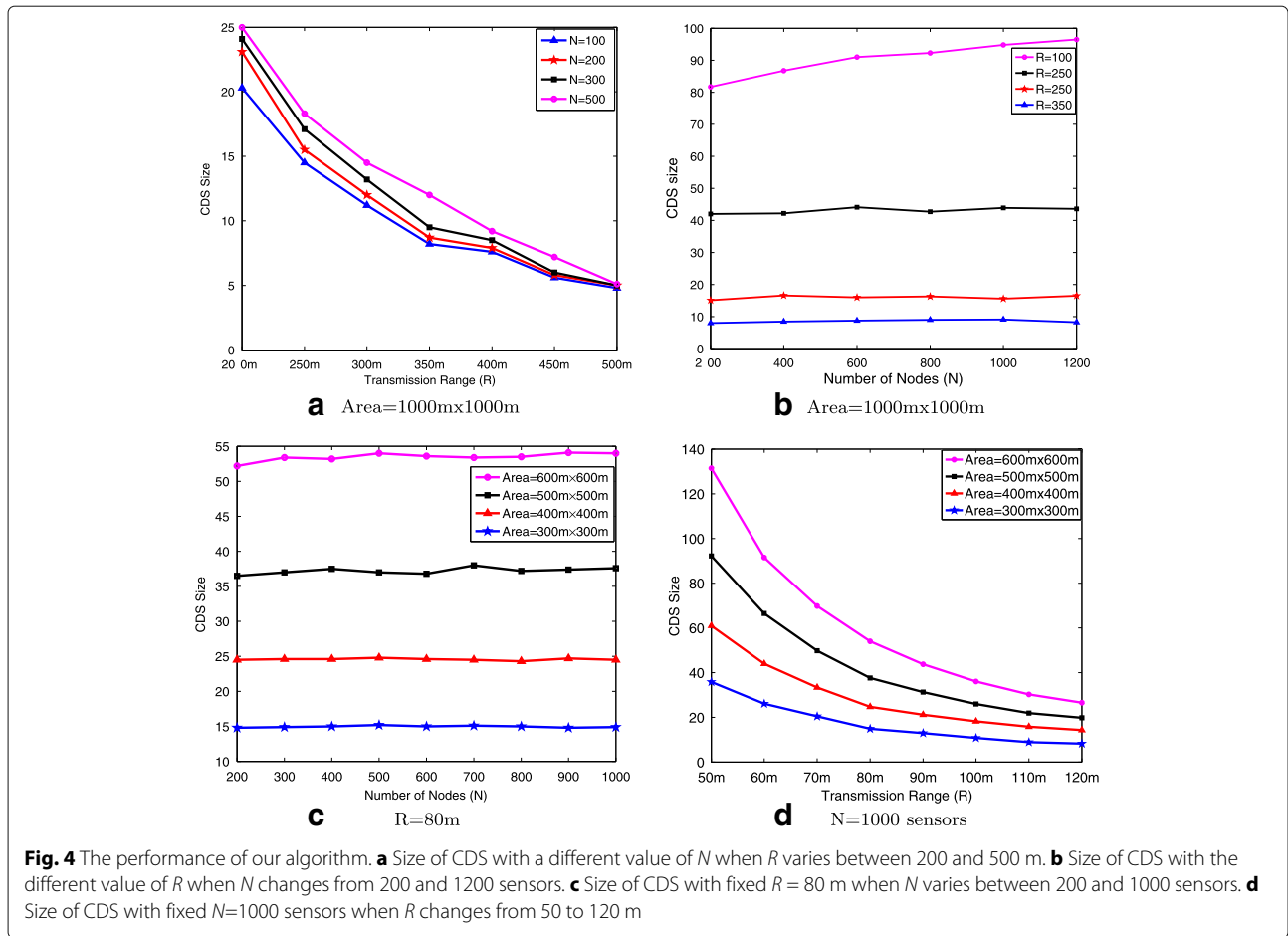
5.1 Impact of network configuration

In this section, we evaluate the impact of the different parameter settings on the size of CDS.

Firstly, Fig. 4a illustrates the impact of the transmission range R on the size of CDS with different number of sensors. We randomly deploy N sensors in a 1000×1000 m² area, and measure the size of CDS when the transmission range R varies from 200 to 500 m increased by 50 m. As shown in Fig. 4a, we can observe that the size of CDS decreases as the transmission range R increases. This is because when the transmission range becomes longer, the number of neighbors of sensors increase. That is to say, a backbone sensor is able to dominate more non-backbone sensors. When the transmission range R is large enough and the number of sensors reaches to some number, the CDS size is almost same no matter how big the number of sensors N is. It is because the some sensors can cover the whole detection area when the transmission range R is large enough. From Fig. 4a, when $R = 500$ m, the CDS sizes are almost the same. We can also find that the ratio of CDS size to the total number of sensors in the network decreases with the increasing of the density of network deployment. For example, we fix R to 300 m, when $N = 100$ and $N = 500$, the size of CDSs are 11.2 and 14.5, respectively, and the ratio of the former is 11.2% and the latter is 2.9%.

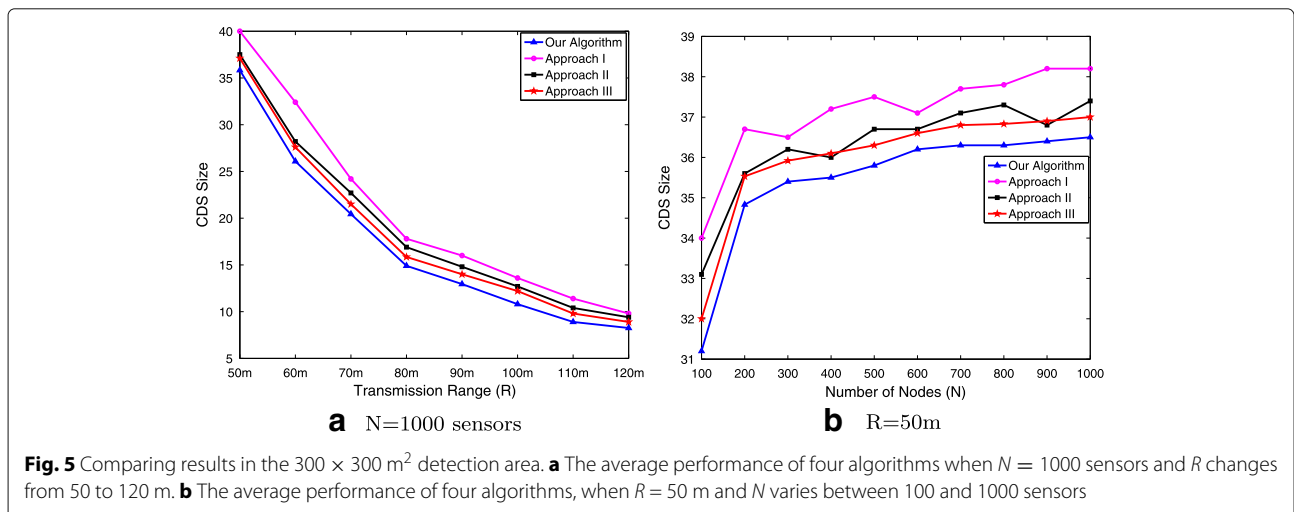
Secondly, we evaluate the impact of the number of sensors N on the size of CDS with the different transmission range R . In the 1000×1000 m² monitor area, the number of sensors N changes from 200 to 1200 sensors, we can find that the size of CDS increases with the number of sensors increasing when $R = 100$ m and that the size of CDS levels off as R is more than 250 m, as shown in Fig. 4b. We also obtain that, when N is fixed, the size of CDS decreases more and more slowly with the increasing of the transmission range when the transmission range reaches to some value.

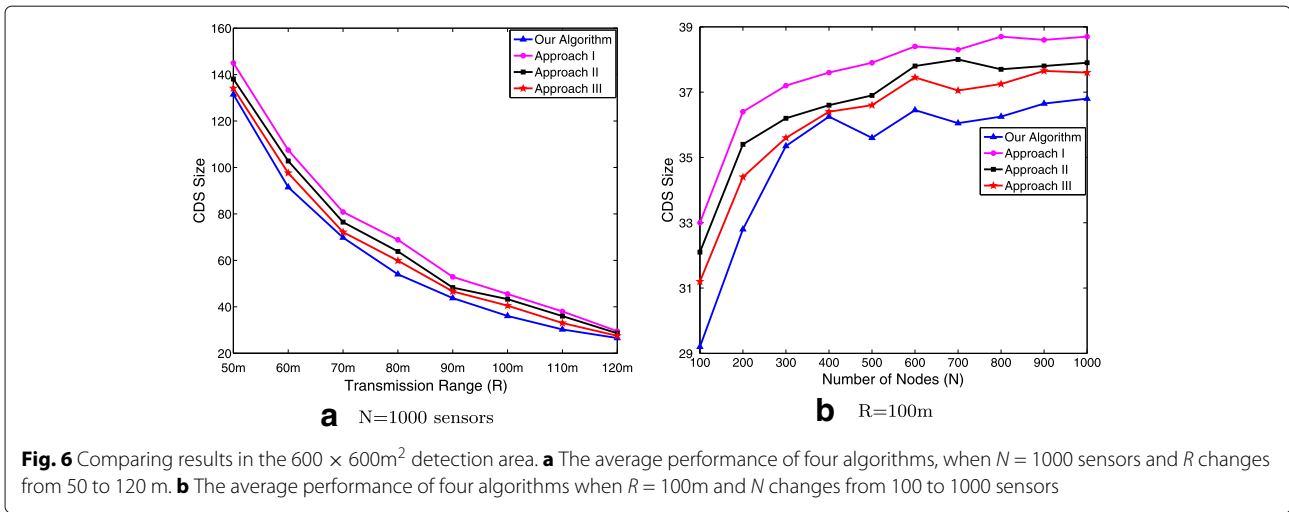
Thirdly, we measure the effect of the size of the deployment area on the size of CDS. We deploy network sensors in the detection areas 300×300 m², 400×400 m², 500×500 m², and 600×600 m², respectively. First, we evaluate



the impact of the number of sensors N on the size of CDS in different detection areas, as shown in Fig. 4c. When we fix the transmission range to 80 m and the number of sensors N (from 200 to 1000), we can notice that the CDS size increases as the deployment area grows. Afterwards, we

evaluate the impact of the transmission range on the size of CDS in different detection areas, as shown in Fig. 4d. When we fix the number of sensors N to 1000 and R (from 50 to 120 m), we can notice that the CDS size increases as the deployment area grows.





5.2 Performance evaluation

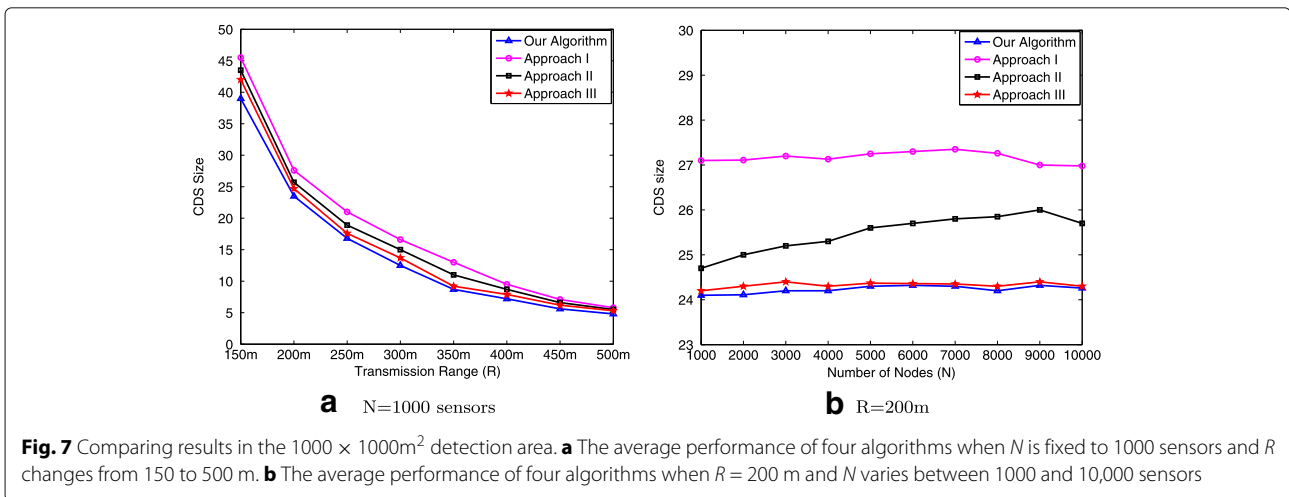
In this section, we compare the performance of our algorithm with the performance of the three algorithms (Approach I, Approach II, and Approach III) in [15]. To compare the performance of our algorithm with the three algorithms, we set the same value of the experiment parameters of our algorithm as the other three algorithms in [15].

Firstly, we give the comparison of the algorithms when the sensors are randomly deployed in the $300 \times 300 m^2$ area, as shown in Fig. 5. When the number of sensors $N = 1000$ and the transmission range R is increased by 10 m from 50 to 120 m, we give the comparative results of the four algorithms in Fig. 5a. The results show that the size of CDS got by our algorithm is always better than the other three algorithms as the transmission range becomes longer. And CDS sizes decrease with the transmission range increasing, which is because the transmission range

is bigger, the coverage area is larger, and the network area size is finite. Similarly, we fix the transmission range R to 50 m and change N from 100 to 1000 sensors increased by 100. The comparative results in Fig. 5b illustrate that our algorithm outperforms the other three algorithms.

Secondly, for $600 \times 600 m^2$ monitor area, Fig. 6 shows the performance of the compared algorithms. If setting the number of sensors $N = 1000$ and changing the transmission range R between 50 and 100 m, our algorithm is better than the other three algorithms and the gap between the four results is getting smaller and smaller with increasing of the transmission range. By setting $R = 100 m$, Fig. 6b gives the comparison in terms of CDS size through increasing the number of sensors from 100 to 1000. We can observe that our algorithm is still better than the other three algorithms.

Finally, to better illustrate the superiority of our algorithm, we deploy the sensors $1000 \times 1000 m^2$ area



randomly, as shown in Fig. 7. In Fig. 7a, when the number of sensors N is fixed to 1000 and R varies from 150 to 500 m, we can observe that our algorithm also outperforms the other three algorithms in the larger detection area. And the size of CDS of the four algorithms tends to be stable when transmission range is big enough. According to Fig. 7b, if we set $R = 200$ m and vary N from 1000 and 10,000, our algorithm still outperforms other three algorithms and the CDS sizes of the algorithms level off as the number of sensors increases, which means that our algorithm is also suitable in dense networks.

6 Conclusions

This paper proposes an approximation algorithm for the MCDS problem in wireless sensor networks. The key idea is to separate sensors in CDS into *core* sensors and *supporting* sensors. The core sensors dominate the supporting sensors in CDS and some sensors are not in CDS, while the supporting sensors dominate remaining sensors that are not in CDS. Simulation results show that the algorithm generates CDS with smaller size than the state-of-the-art algorithms.

Abbreviations

CDS: Connected dominating set; DS: Dominating set; MCDS: Minimum connected dominating set; MIS: Maximal independent set; WSN: Wireless sensor network

Funding

This work was supported in part by the National Natural Science Foundation of China under Grants 11671400, 61672524; the Fundamental Research Funds for the Central University, and the Research Funds of Renmin University of China, 2015030273.

Authors' contributions

CWL has contributed towards the algorithms, the analysis, and the simulations and written the paper. DYL has contributed towards the algorithms, and the analysis. As the supervisor of CWL, she has proofread the paper several times and provided guidance throughout the whole preparation of the manuscript. WPC, JGY, and YCW have revised the equations, helped writing the introduction and the related works, and critically revised the paper. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹School of Information, Renmin University of China, Zhongguancun Road, 100872 Beijing, People's Republic of China. ²School of Information Science and Engineering, Qufu Normal University, Rizhao, 276826 Shandong, People's Republic of China.

Received: 23 September 2017 Accepted: 27 February 2018

Published online: 07 March 2018

References

- Blum, M, Ding, X, Cheng, X: Applications of connected dominating sets in wireless networks. *Handb. Comb. Optim.* **42**, 329–369 (2004)
- Cheng, Z, Cai, J, Li, H, Gao, G: Dataset from big sensory data in wireless sensor networks. *IEEE Trans. Knowl. Data Eng.* **29**(4), 813–827 (2017)
- Li, J, S, Cheng, Z, Cai, J, Yu, C, Wang, Y, Li, Y: Approximate holistic aggregation in wireless sensor networks. *ACM Trans. Sens. Netw.* **13**(2), 1–24 (2017)
- Wu, H, Z, Miao, Y, Wang, M, Lin, L: Optimized recognition with few instances based on semantic distance. *Vis. Comput.* **31**(4), 367–375 (2015)
- Wu, H, Z, Miao, Y, Wang, J, Chen, C, Ma, T, Zhou, T: Image completion with multi-image based on entropy reduction. *Neurocomputing.* **159**(7), 157–171 (2015)
- Guha, S, Khuller, A: Approximation algorithms for connected dominating sets. *Algorithmica.* **20**(4), 374–387 (1998)
- Wan, P-J, Alzoubi, K, Frieder, O: in *Proceedings IEEE, INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 3*. Distributed construction of connected dominating set in wireless ad hoc networks (IEEE, New York, 2002), pp. 1597–1604
- Li, Y, Thai, M, Thai, F, Wang, C-W, Yi, P-J, Wan, D-Z, Du, D: On greedy construction of connected dominating sets in wireless networks. *Wirel. Commun. Mob. Comput.* **5**(8), 927–932 (2005)
- Clark, B, Colbourn, C, Johnson, D: Unit disk graphs. *Discret. Math.* **86**(1-3), 165–177 (1990)
- Funke, S, Kesselman, A, Meyer, U, Segal, M: A simple improved distributed algorithm for minimum cds in unit disk graphs. *ACM Trans. Sens. Netw.* (TOSN). **2**(3), 444–453 (2006)
- Min, M, Du, H, Du, X, Jia, X, Huang, C, Huang, S-H, Wu, W: Improving construction for connected dominating set with steiner tree in wireless sensor networks. *J. Glob. Optim.* **35**(1), 111–119 (2006)
- Wan, P-J, Wang, L, Wang, F, Yao, F: in *Distributed Computing Systems, 2008. ICDCS'08. The 28th International Conference On*. Two-phased approximation algorithms for minimum cds in wireless ad hoc networks (IEEE, Hang Zhou, 2008), pp. 337–344
- Misra, R, Mandal, C: Minimum connected dominating set using a collaborative cover heuristic for ad hoc sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **21**(3), 292–302 (2010)
- Tang, Q, Luo, Y-S, Xie, M-Z, Li, P: Connected dominating set construction algorithm in wireless networks based on connected subset. *J. Commun.* **11**(1), 50–57 (2016)
- Al-Nabhan, N, Zhang, B, Zhang, X, Cheng, M, Al-Rodhaan, A, Al-Dhelaan, A: Three connected dominating set algorithms for wireless sensor networks. *Int. J. Sens. Netw.* **21**(1), 53–66 (2016)
- Du, D-Z, Wan, P-J: Connected dominating set: theory and applications. *Springer Sci. Bus. Media.* **77** (2012)
- Yu, J, Wang, N, Wang, G, Yu, D: Connected dominating sets in wireless ad hoc and sensor networks—a comprehensive survey. *Comput. Commun.* **36**(2), 121–134 (2013)
- Ruan, L, Ruan, H, Du, X, Jia, X, Wu, W, Li, Y, Li, K-I, Ko, A: A greedy approximation for minimum connected dominating sets. *Theor. Comput. Sci.* **329**(1-3), 325–330 (2004)
- Fu, D, Han, L, Liu, L, Gao, Q, Z, Feng, Z: An efficient centralized algorithm for connected dominating set on wireless networks. *Procedia Comput. Sci.* **56**, 162–167 (2015)
- Cheng, X, Ding, M, Ding, D, Chen, D: in *Proc. of International Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor, and Peer-to-Peer Networks (TAWN), vol. 2*. An approximation algorithm for connected dominating set in ad hoc networks, (Washington, 2004)
- Du, H, Wu, W, Wu, Q, Ye, D, Li, W, Lee, X, Xu, X: Cds-based virtual backbone construction with guaranteed routing cost in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **24**(4), 652–661 (2013)
- Wang, W, Liu, B, Liu, D, Kim, D, Li, J, Wang, W, Gao, A: A new constant factor approximation to construct highly fault-tolerant connected dominating set in unit disk graph. *IEEE/ACM Trans. Netw.* (TON). **25**(1), 18–28 (2017)
- Shi, Y, Shi, Z, Zhang, Y, Mo, D-Z, Du, D: Approximation algorithm for minimum weight fault-tolerant virtual backbone in unit disk graphs. *IEEE/ACM Trans. Netw.* **25**(2), 925–933 (2017)
- Li, D, Li, X, Jia, H, Liu, L: Energy efficient broadcast routing in static ad hoc wireless networks. *IEEE Trans. Mob. Comput.* **3**(2), 144–151 (2004)
- Hong, Y, Bradley, D, Kim, D, Li, A, Tokuta, Z, Ding, D: Construction of higher spectral efficiency virtual backbone in wireless networks. *Ad. Hoc. Netw.* **25**, 228–236 (2015)
- Shi, T, Shi, S, Cheng, Z, Cai, Y, Li, J, Li, L: Exploring connected dominating sets in energy harvest networks. *IEEE/ACM Trans. Netw.* (TON). **25**(3), 1803–1817 (2017)

27. T Shi, S Cheng, J Li, Z Cai, in *The 36th Annual IEEE International Conference on Computer Communications INFOCOM*. Constructing connected dominating sets in battery-free networks (IEEE, Atlanta, 2017), pp. 1–9
28. J Yu, W Li, X Cheng, M Atiquzzaman, H Wang, L Feng, Connected dominating set construction in cognitive radio networks. *Pers. Ubiquit. Comput.* **20**(5), 757–769 (2016)
29. J Yu, N Wang, G Wang, Constructing minimum extended weakly-connected dominating sets for clustering in ad hoc networks. *J. Parallel Distrib. Comput.* **72**(1), 35–47 (2012)
30. S Cheng, Z Cai, J Li, Curve query processing in wireless sensor networks. *IEEE Trans. Veh. Technol.* **64**(11), 5198–5209 (2015)
31. Z He, Z Cai, S Cheng, X Wang, Approximate aggregation for tracking quantiles and range countings in wireless sensor networks. *Theor. Comput. Sci.* **607**, 381–390 (2015)
32. C Luo, Y Wang, J Yu, W Chen, D Li, in *International Conference on Wireless Algorithms, Systems, and Applications (WASA)*, Guilin, China. A new greedy algorithm for constructing the minimum size connected dominating sets in wireless networks (Springer, Cham, 2017), pp. 109–114

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
