International Journal of Innovative Computing, Information and Control
Volume **12**, Number **5**, October **2016**

ICIC International ©2016 ISSN 1349-4198
pp. 1553–1564

AN IMPROVED TWO-PHASED APPROXIMATION OF MINIMUM CONNECTED DOMINATING SETS IN UNIT DISK GRAPHS USING TWO-HOP DEGREE CENTRALITY

Kun Zhao[1], Zheming Lu[2,*], Tingyuan Nie[1], Yansheng Li[1] and Chuanwang Song[1]

[1]School of Communication and Electronic Engineering
Qingdao University of Technology
No. 777, Jialingjiang Road, Qingdao 266520, P. R. China
sterling1982@163.com

[2]School of Aeronautics and Astronautics
Zhejiang University
No. 38, Zheda Road, Hangzhou 310027, P. R. China
[*]Corresponding author: zheminglu@zju.edu.cn

Received January 2016; revised July 2016

Abstract. *Minimum connected dominating set (MCDS) offers an optimized route in wireless networks. However, constructing an MCDS is an NP-complete problem. Many heuristics based approximation algorithms for MCDS problems have been previously reported. Nevertheless, results of these algorithms are not close enough to MCDS. In this paper, a new two-phased algorithm (THDCBTPS) is proposed for approximating an MCDS in unit disk graphs (UDGs) by using two-hop degree (D_2) centrality of the nodes. The main contributions of this paper are as follows. Firstly, a better approximation for MIS in the first phase is achieved by considering D_2. Secondly, an improved Steiner tree construction is proposed to obtain fewer connectors in the second phase, which leads to a smaller CDS. Thirdly, the upper bound of proposed approach is proved to be $6.075|opt| + 5.425$ which is better than most of existing methods. Simulation results show that THDCBTPS constructs better CDSs for networks with random distribution of nodes in UDG.*

Keywords: Two-hop degree centrality, Connected dominating set, Maximal independent set, Unit disk graph

1. Introduction.

Mobile ad hoc networks (MANETs) and wireless sensor networks (WSNs) are widely used for disaster control and geographical monitoring related applications. In the application of remote data acquisition, WSNs often use in-network data aggregation to optimize network communication [1]. Low loss in-network aggregation depends on optimized coverage of aggregating nodes. The set of aggregating nodes forms a dominating set (DS) of the network. These aggregation nodes are then organized in a Steiner tree to form a connected dominating set (CDS) as a data aggregation backbone. Ad hoc networks also use a CDS as a virtual backbone [2] for efficient routing and broadcasting operations. That is the main reason why so many network protocols, including media access, clustering, multicast/broadcast, power management and coverage/monitoring, are organized in CDSs. The effectiveness of the aggregation algorithm will be achieved when the underlying CDS tree is minimized. Because only the CDS nodes maintain routing information, the reduced CDS size could save storage spaces effectively.

Also, a smaller-sized CDS makes routing easier as it can reduce transmission interference and the number of control messages. Therefore, constructing a wireless backbone is modeled as the minimum connected dominating set (MCDS) problem in graph theory.

However, computing MCDS is an NP-complete problem [3]. Therefore, MCDS could only be approximated by heuristic algorithms. In this paper, we propose an improved construction of an MCDS in unit disk graphs (UDGs) using a two-hop degree centrality based two-phased scheme (THDCBTPS), which achieves the approximation factor of $6.075|opt| + 5.425$, where $|opt|$ is the size of an optimal CDS of the network.

The rest of this paper is organized as follows. In Section 2, we discuss related works on CDS construction algorithms. Section 3 gives the motivation and contributions of our work. Section 4 states the proposed THDCBTPS in detail. The simulation results and analysis are given in Section 5. Finally, we conclude the whole paper in Section 6.

2. **Related Work.** The existing CDS constructions in the context of MANETs and WSNs could be classified roughly into the following categories: greedy based [4], MIS based [5-7], pruning based [2,8,9], multipoint relaying based [10], Steiner tree based [11], probability based [12], k-CDS [13,28-30], considering energy-efficient [2,5,14,15] and nodal mobility [16,17]. When classifying and analyzing these approaches, the following characteristic parameters are taken into considerations: performance ratios/bounds; time and message complexities; degree of localization; energy-efficient topology; nodal movement; deterministic, probabilistic or hybrid scheme; backbone robustness and designing for k-CDS or d-hop clustering.

Instead of viewing the problem with the full-scale characteristic parameters that mentioned above, Wu et al. [18] focused on the evolution of MCDS approximation algorithms. Their survey uses the approximation factor as the main criterion. From their research we can learn that the MIS based approaches always get better approximation factors. These algorithms follow a general two-phased approach. The first phase constructs a DS usually by approximating an MIS. The nodes in the DS are called dominators. The second phase selects additional nodes, called connectors, inducing a connected topology together with dominators to form a CDS usually by optimizing a Steiner tree. Thus, the problem of MCDS approximation in a UDG breaks into two sub-problems: MIS approximation and Steiner tree with minimum number of Steiner nodes (ST-MSN). Accordingly, the final approximation factor can be obtained by an MIS approximation factor which is denoted as $|mis|$, and an STMSN approximation factor, namely $|st|$, which can be roughly calculated by $|mis| - 1$ according to [18].

About the upper bound of $|mis|$, there is a conjecture [18]: in a UDG, any MIS has the size $|mis|$ $3|opt| + 2$. Algorithms have been proposed to attack this conjecture. Wan et al. [6] firstly showed that the size of every MIS is $4|opt| + 1$ at most. Later, several efforts have been made to improve this bound [18-23]. Recently, Li et al. [19] employed a quite complicated geometry argument and pushed this bound further to $3.4306|opt|+4.8185$. In another direction, although Wan et al. [22] presented an example to show that $3|opt|+2$ is reachable, their algorithm cannot guarantee this. Very recently, Zhang et al. [24] claimed a much better bound $3.2833|opt| + 4.559$. However, there is no proof for this conclusion. Therefore, this bound is not recognized as proved. In [31], the upper bound is no more than $5(l_{2k}+2l_{3k}+1)|opt|$, where l_{2k} and l_{3k} are dominators with 2-hop and 3-hop distance away from each node, which means l_{2k} or l_{3k} is at least 1. Even in a best-case scenario, namely $l_{2k} = 1$ and $l_{3k} = 0$, the upper bound is still $10|opt|$. We simply list the upper bound evolution for MCDS approximation factor which has been clearly proved in Table 1.

Table 1. Upper bound evolution for MCDS approximation factor in UDG

Algorithm	Upper Bound for MIS	Upper Bound for MCDS
Ren et al. [31]	$5\|opt\|$	$10\|opt\|$
Wan et al. [6]	$4\|opt\| + 1$	$6.862\|opt\|$
Wu et al. [20]	$3.8\|opt\| + 1.2$	$7.8\|opt\|$
Min et al. [11]	$3.8\|opt\| + 1.2$	$6.8\|opt\|$
Funke et al. [21]	$3.748\|opt\| + 9$	$6.91\|opt\|$
Wan et al. [22]	$3.6667\|opt\| + 1.3333$	$6.389\|opt\|$
Gao et al. [23]	$3.453\|opt\| + 4.839$	$6.453\|opt\|$
Li et al. [19]	$3.4306\|opt\| + 4.8185$	$6.075\|opt\| + 5.425$
Proposed approach	$3.4306\|opt\| + 4.8185$	$6.075\|opt\| + 5.425$
Wu et al. [18]	$3\|opt\| + 2$	–

From Table 1 we can learn that not all the algorithms with less upper bound for approximating MIS lead to less upper limits for the final MCDS approximation factor, which also depends on tighter Steiner sets obtained in the second phase. With the same upper bound for approximating MIS as in [20], and even more than that in [21], the algorithm proposed in [11] reaches a less upper limit for MCDS approximation factor. This promotion is owed to a "polynomial-time 3-approximation algorithm" in the second phase for the ST-MSN problem. By using this approach, the upper bound for ST-MSN approximation factor reduces from $|mis| - 1$ (more than $3|opt|$) to $3|opt|$. [19] obtains a tighter result which is only $2.6444|opt| + 0.6065$.

The upper bound can only evaluate the performance of an algorithm in its worst case. Instead of giving deterministic upper limits for their approximation with theoretically strict proof, several studies like [25-27] compare their work with other approaches by simulating data in a more general sense.

In this paper, we proposed a novel algorithm to achieve a better approximation for MIS in the first phase (see Algorithm 1 in Sub-section 4.1) and an ST-MSN approximation approach in the second phase (see Algorithm 2 in Sub-section 4.2). The distinguishing difference of our work from the other literature is the using of the 2-hop degree of each node, which leads the same upper bound as in [19]. The simulation results are also shown in this paper to show the improvement of the proposed approach.

3. **Motivation and Contributions.** As we have discussed in Section 2, the majority of the distributed algorithms which obtain tighter deterministic upper bounds follow a general two-phased approach: one phase approximates an MIS, and the other selects connectors. When approximating an MIS in the first phase, most approaches use a degree based algorithm which selects dominator with maximum degree at each iteration. Degree, namely one-hop degree, is the sum of one-hop neighbors of the current node. Selecting the node with maximum one-hop degree as a dominator can guarantee the current node cover most dominatees, which intuitionally lead to an optimal approximation for MIS. However, in a large scale network, it is very common that many nodes have the same one-hop degree. In case of a tie, the node with least ID is often given preference. Actually, that introduces some uncertainty for the approximation, which may lead to a bigger CDS.

Figure 1 shows the case of a symmetrical network. When using a one-hop degree based approach, a DS which contains 14 nodes is obtained in Figure 1(a). Figure 1(b) shows a smaller DS with 8 dominators. To obtain it, we considered both the one-hop and the two-hop degree centrality of the current node when looking for a dominator. The latter

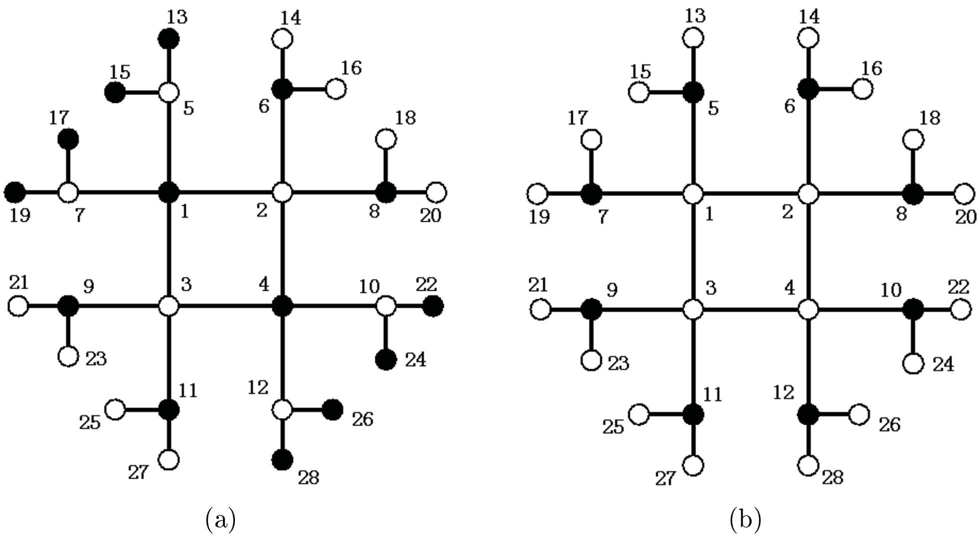


FIGURE 1. Different DSs using different MIS approximation algorithms. (a) uses only one-hop degree information. (b) uses both one-hop and two-hop degree information. Dominators in DS are colored as black and dominatees are colored white.

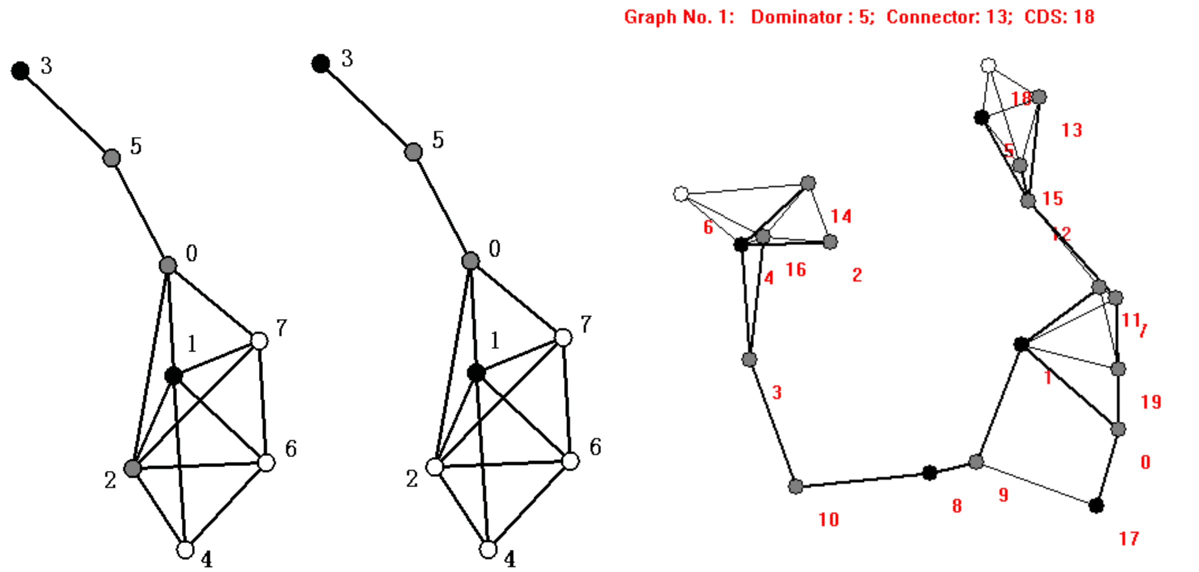
of it is defined as the sum of nodes that are able to reach with exact two hops by the current node. In order to avoid ambiguity, the following cases are not taken into account when calculating the two-hop degree of a node: 1) the current node itself; 2) the nodes that can also be reached by one hop. Thus, we can obtain the one-hop degree and the two-hop degree for each node of the network in Figure 1.

In Figure 1(b), the weight of each node is used to displace one-hop degree when choosing dominators. The weight is calculated by consideration of both one-hop and two-hop degree of each node. The term of one-hop degree should be a positive term to guarantee that a dominator covers dominatees as much as possible. Considering that there is much possibility for a two-hop neighbour of a dominator to also be a dominator, the candidate which has the least two-hop degree should be a priority choice when one-hop degrees of which are quite close. Any two nodes in an IS are either two-hop neighbours or three-hop neighbours. When the node with less two-hop degree is chosen as a dominator, the number of its two-hop neighbours likely to be dominators is less than other options. Thus, when one-hop degrees are close, the option of less two-hop degree implies a smaller IS. We set the two-hop degree to be a negative term. In order to make sure that it works only when one-hop degrees are close, the two-hop degree should be normalized by one-hop degree. Thus, we obtain the weight W of node v by Equation (1), where $D_1(v)$ and $D_2(v)$ denote the one-hop degree and two-hop degree of v respectively.

$$W(v) = \begin{cases} D_1(v) - D_2(v)/D_1(v) & \text{if } D_1(v) \neq 0 \\ 0 & \text{if } D_1(v) = 0 \end{cases} \tag{1}$$

The weight of node 1, 7 and 17 in Figure 1 are 1.75, 2 and -1 respectively according to Equation (1). Thus, node 7 with the most weight is chosen as a dominator at the first iteration, not node 1 with the most one-hop degree.

Besides the promotion with probable smaller IS in the first phase, we also noted that the two-hop degree can be expanded to the second phase to form a smaller CDS. When looking for connectors using a Steiner tree, existing algorithms like [19] and [27] always choose the one in the neighbourhood of most connected sub-graphs to make the connectors



(a) (b) (c)

Figure 2. Different CDSs using different connector searching algorithms. (a) uses the algorithm in [27]. (b) uses proposed algorithm. Connectors are coloured as gray. (c) is a real case using the algorithm in [27] with our platform.

as little as possible. We denote the sum of the connected sub-graph neighbours of a node as *one-hop connected sub-graph degree* (D_{csg1}). The *two-hop connected subgraph degree* (D_{csg2}) could be also defined. The consideration of using two-hop connected sub-graph degree when nodes have the same one-hop connected sub-graph degree could turn the search of connector in the right direction. Figure 2(a) shows the connector (gray nodes) searching result using the algorithm in [27], which sets the node degree to be the second priority condition. The search order is $2 \to 0 \to 5$. By using the two-hop connected sub-graph degree priority (node 0) to replace the node degree priority (node 2), less connectors are obtained in Figure 2(b). Figure 2(c) shows a real case of algorithm proposed in [27], which leads to a connector disaster (a mass of gray nodes). The rank of priority condition for searching a connector in [27] is: $D_{csg1}(Max) \to D_1(Max) \to ID(Min)$. While, the rank in the proposed algorithm is: $D_{csg1}(Max) \to D_{csg2}(Max) \to D_1(Max) \to ID(Min)$. Different from the first phase, the node with the maximum two-hop connected sub-graph degree, not the least one, is chosen to be a connector when its one-hop connected subgraph degree is the same as other nodes.

We are motivated towards dealing with the issues cited above to reduce the CDS size further at an optimal trade-off in the number of messages exchanged. Thus, in this paper, a **Two-Hop Degree Centrality Based Two-Phased Scheme (THDCBTPS)** which contributes towards improving the CDS size further than previous approximation algorithms is proposed.

4. Proposed Approach.

The proposed approach consisted of two phases which are described in Sub-sections 4.1 and 4.2. In Sub-section 4.3, we discuss the upper bound of proposed approach briefly.

4.1. Phase 1: IS construction algorithm.

In the first phase, an IS is constructed to approximate an MIS greedily according to Algorithm 1.

Algorithm 1. IS Construction Using Two-hop Degree Information

1. Initialize <*CurrentState* = **DOMINATEE**> for each node in the original graph G_0, $<IS = \emptyset>$ for G_0.
2. Calculate $< W(v) = D_1(v) - D_2(v)/D_1(v) | D_1(v) \neq 0 >$, $< W(v) = 0 | D_1(v) = 0 >$ for each node v, where $W(v)$, $D_1(v)$ and $D_2(v)$ denote the weight, one-hop degree and two-hop degree of v respectively.
3. Find the node u with the maximal weight, set <*CurrentState* = **DOMINATOR**> for u, $<IS = IS \cup \{u\}>$. In case of a tie, mark the node with the maximum one-hop degree as **DOMINATOR**. If there is still a tie, the node with least *ID* is marked as **DOMINATOR**.
4. Remove u and its one-hop neighbours $N[u]$ with the edges on them from the current graph G. Update D_1 and D_2 for each remaining node.
5. Calculate the maximum one-hop degree Δ for the current graph G. If $\Delta \neq 0$, go back to Step 2; else <*CurrentState* = **DOMINATOR**> for all remaining nodes *RN*, $<IS = IS \cup RN>$.
6. Recover the original graph G_0 with all removed nodes and edge. An *IS* is obtained.

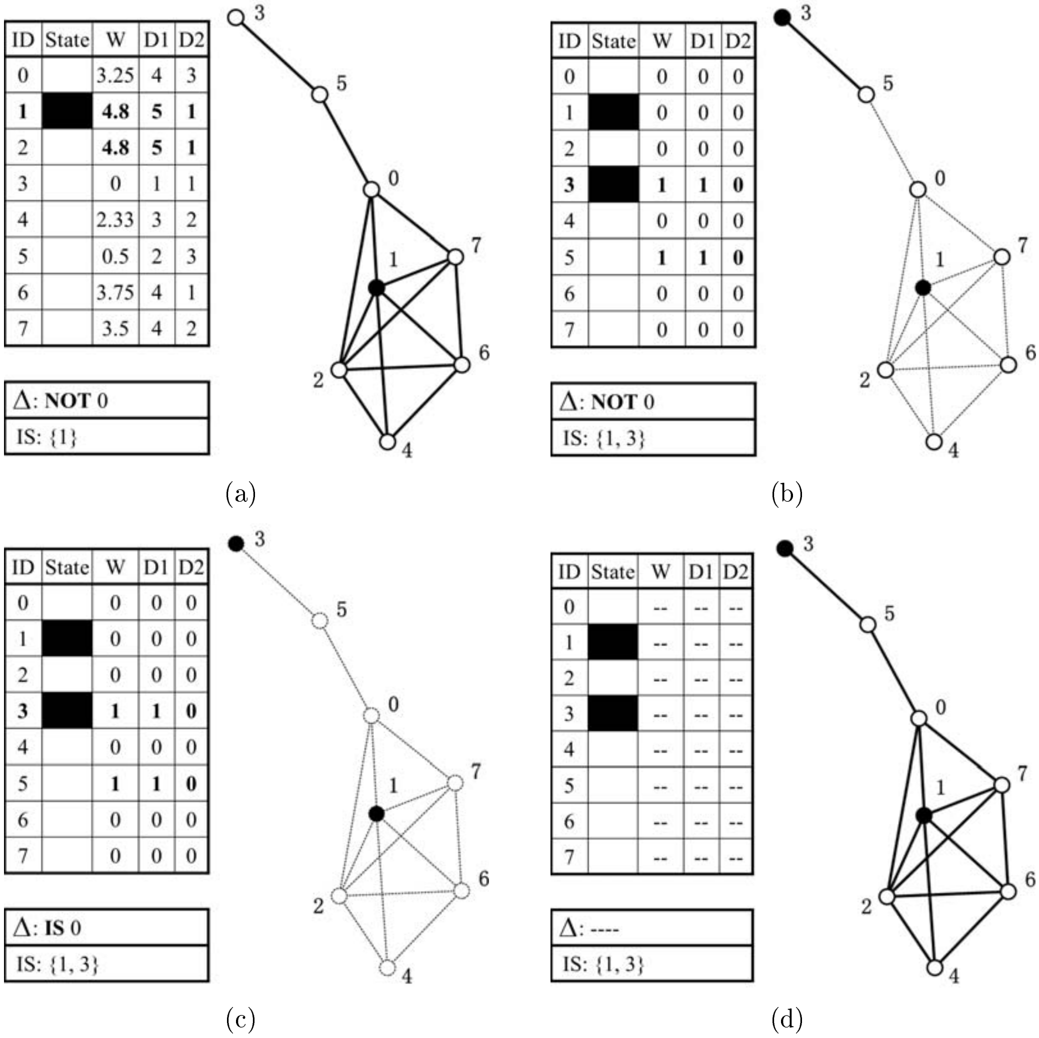
In order to make it easier to understand, Figure 3(a) to Figure 3(d) show the results of Algorithm 1 step by step that run on the case in Figure 1(b). The current states of nodes are marked in different colors both in the graph and the table beside it: black represents the DOMINATOR and white represents the DOMINATEE. The nodes and edges in dotted lines meant that they have been removed from the graph temporarily.

4.2. **Phase 2: CDS construction algorithm.** In the second phase, CDS is constructed to approximate an MCDS greedily according to Algorithm 2.

Algorithm 2. CDS Construction Using Two-hop Connected Sub-graph Degree

1. Initialize every single dominator in the original graph G_0 to be a connected sub-graph: $<CSG_i = \{u\} | CurrentState_u =$ **DOMINATOR**>. Initialize *CDS* to be the *IS* obtained in last phase: $<CDS = IS>$ for G_0.
2. Calculate $<D_{csg1}(v)$, $D_{csg2}(v)$, $D_1(v) | CurrentState_v =$ **DOMINATEE**> for each dominatee v.
3. Find the node u with the maximal $D_{csg1}(u)$, set <*CurrentState* = **CONNECTOR**> for u, $<CDS = CDS \cup \{u\}>$. In case of a tie, mark the node with the maximum D_{csg2} as **CONNECTOR**. If there is still a tie, the dominatee with maximum D_1 is marked as **CONNECTOR**. If still a tie, then choose the one with the least *ID*.
4. Combine u and its one-hop *CSG* neighbours $N_{csg}[u]$ into a new *CSG*.
5. Count the sum of CSGs S for the current graph G. If $S \neq 1$, go back to Step 2; else end the algorithm and obtain the final *CDS* with all the dominators and connectors.

In order to make it easier to understand, Figure 4(a) to Figure 4(b) show the results of Algorithm 2 step by step that run on the case in Figure 3. The CONNECTORs are colored in gray. The areas in dotted circles are connected-sub-graphs (CSGs).



(a)

ID	State	W	D1	D2
0		3.25	4	3
1		**4.8**	**5**	**1**
2		**4.8**	**5**	**1**
3		0	1	1
4		2.33	3	2
5		0.5	2	3
6		3.75	4	1
7		3.5	4	2

Δ: **NOT** 0
IS: {1}

(b)

ID	State	W	D1	D2
0		0	0	0
1		0	0	0
2		0	0	0
3		**1**	**1**	**0**
4		0	0	0
5		**1**	**1**	**0**
6		0	0	0
7		0	0	0

Δ: **NOT** 0
IS: {1, 3}

(c)

ID	State	W	D1	D2
0		0	0	0
1		0	0	0
2		0	0	0
3		**1**	**1**	**0**
4		0	0	0
5		**1**	**1**	**0**
6		0	0	0
7		0	0	0

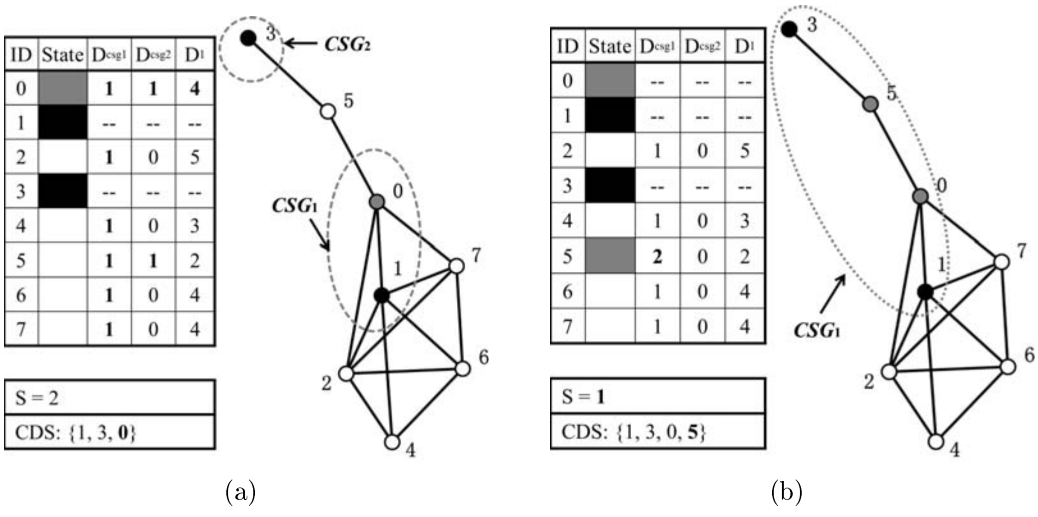
Δ: **IS** 0
IS: {1, 3}

(d)

ID	State	W	D1	D2
0		--	--	--
1		--	--	--
2		--	--	--
3		--	--	--
4		--	--	--
5		--	--	--
6		--	--	--
7		--	--	--

Δ: ----
IS: {1, 3}

FIGURE 3. A case of running Algorithm 1: (a) after steps 1-3, round 1; (b) after steps 4-5, round 1 and steps 1-3, round 2; (c) after steps 4-5, round 2; (d) after step 6, round 2



(a)

ID	State	D_{csg1}	D_{csg2}	D_1
0		**1**	**1**	**4**
1		--	--	--
2		**1**	0	5
3		--	--	--
4		**1**	0	3
5		**1**	**1**	2
6		**1**	0	4
7		**1**	0	4

S = 2
CDS: {1, 3, **0**}

(b)

ID	State	D_{csg1}	D_{csg2}	D_1
0		--	--	--
1		--	--	--
2		1	0	5
3		--	--	--
4		1	0	3
5		**2**	0	2
6		1	0	4
7		1	0	4

S = **1**
CDS: {1, 3, 0, **5**}

FIGURE 4. A case of running Algorithm 2 on the graph in Figure 3: (a) after steps 1-4, round 1; (b) after steps 2-5, round 2

4.3. **Performance analysis for the proposed approach.** According to [19], the upper bound of $|mis|$ for the general two-phased approach is $3.4306|opt| + 4.8185$. From the detailed discussions in [19], it follows that THDCBTPS also produces the same upper bound of $|mis|$ in the first phase. The present tightest upper bound for the number of connectors in the second phase that has been proved strictly is $|mis|/3 + 3|opt|/2 - 1$, which is obtained by using a greedy Steiner node searching algorithm in [22] and proved in [19]. Here we simply state that THDCBTPS achieves the same upper bound in the second phase as that in [22].

In [22], they choose the node $w \in V - (IS \cup ST)$ with maximum gain with respect to ST and add w to ST in each connector searching step while $q(ST) > 1$, where V is the set of all nodes in the graph, IS is DS obtained in the first phase, ST is the set of connectors, $q(ST)$ is the number of connected components in $IS \cup ST$ and the gain $\Delta wq(ST)$ is obtained by Equation (2).

$$\Delta wq(ST) = q(ST) - q(ST) \cup \{w\} \tag{2}$$

From Equation (2) we can learn that the more CSGs are adjacent to w, the larger the gain is. In THDCBTPS, the dominatee u with the maximum D_{csg1} is selected to be a connector, which can guarantee the maximum $\Delta uq(ST)$ is obtained. Lemma 9 in [22] implies that for any set $ST \subseteq V - IS$, the set $IS \cup ST$ is a CDS if and only if $q(ST) = 1$, which holds further if and only if every node has zero gain. In THDCBTPS, the iteration of connector searching stops when there is only one CSG in the graph, which is in line with the case when $q(ST) = 1$, when $IS \cup ST$ is a CDS. Thus, the CDS construction algorithm in the second phase of THDCBTPS can guarantee the same upper bound for the sum of connectors as the one in [22], which is proved to be $|mis|/3 + 3|opt|/2 - 1$ by [19]. Furthermore, THDCBTPS can guarantee the connector searching in the right direction when a tie case appears by using the D_{csg2} information. However, the improvement is not visible in the worst case. Therefore, it has no effect on the upper bound. Thus, the final upper bound of the approximation factor of THDCBTPS could be deduced as Equation (3).

$$\begin{aligned} |cds| &= |mis| + |st| \\ &\le |mis| + |mis|/3 + 3|opt|/2 - 1 \\ &= 4|mis|/3 + 3|opt|/2 - 1 \\ &\le 4 \times (3.4306|opt| + 4.8185)/3 + 3|opt|/2 - 1 \\ &\le 6.075|opt| + 5.425 \end{aligned} \tag{3}$$

Even the final upper bound is as the same as that in [19], the improvement could be obviously found by data simulation in Section 5. In order to obtain a more global result, THDCBTPS abandons the framework of multiple leaders. This results in a time complexity of $O(n)$ time and $O(D)$ rounds, where n is the sum of nodes and D is the network diameter.

5. **Simulations.** In this section, we present the simulation results to measure the performance of THDCBTPS. The simulation experiments are as the following.

- Performance comparison in the first phase with related two-phased techniques.
- Performance comparison in the second phase with related two-phased techniques.
- Whole performance comparison with the related techniques.

In the experimental setup, a set of nodes was deployed randomly in a fixed square of dimension 100×100 square units, known as deployment area M, to model the wireless ad hoc network. In M, N hosts are randomly generated by choosing each of their abscissa and ordinate using a uniform random number generator. We further assume that each node has a uniform transmission range r where $r^2 = (d \times M)/(\pi \times N)$, d being the network density. The induced graph of underlying network is a UDG. We also ensure that the networks considered are connected. We run the algorithm 500 times for each of the different network sizes. The averaged results are shown in the accompanying figures. All the results are worked out by using a platform that we developed with C++ in VS2015 (see Figure 2(c)). The simulation is carried out in MATLAB2015.

In the first experiment (Figure 5), we compare the size of IS obtained in the first phase of two typical one-hop degree based algorithms, level-based [6] and PSCASTS [27] with the proposed THDCBTPS. Transmission range varies from 25 to 50 units. Network size is chosen as 20, 50 and 100 nodes. Note that we take connected graph into consideration. The algorithm was run 500 times on each different set of parameters. The averaged results are shown in Figure 5. The simulation results reveal that the proposed approach reduces the size of DS in the first phase by 58% and 23% comparing with level-based and PSCASTS respectively when $N = 20$. The reductions are 74% and 19% when $N = 50$, 84% and 16% when $N = 100$.

Next (Figure 6), we compare the performance of Steiner nodes generation algorithm in the second phase of THDCBTPS still with level-based and PSCASTS using the same parameter setting as the first experiment. The averaged results are shown in Figure 6. The reduction of the size of ST nodes stands out when comparing with the level-based algorithm by 43% when $N = 20$, 51% when $N = 50$ and 56% when $N = 100$. It is also obvious by 17% when comparing with PSCASTS in a small network ($N = 20$). However,

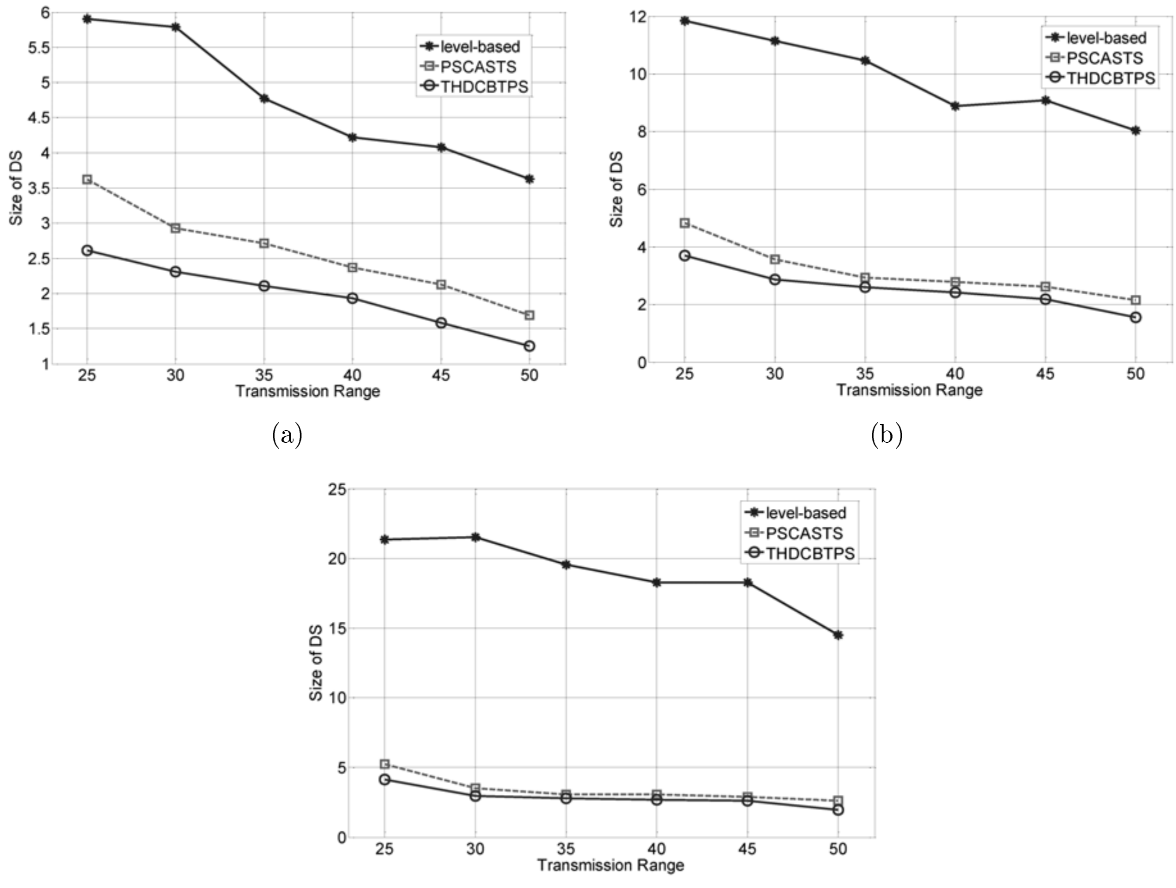


FIGURE 5. Comparison of the size of DS of three algorithms: (a) $N = 20$, (b) $N = 50$ and (c) $N = 100$

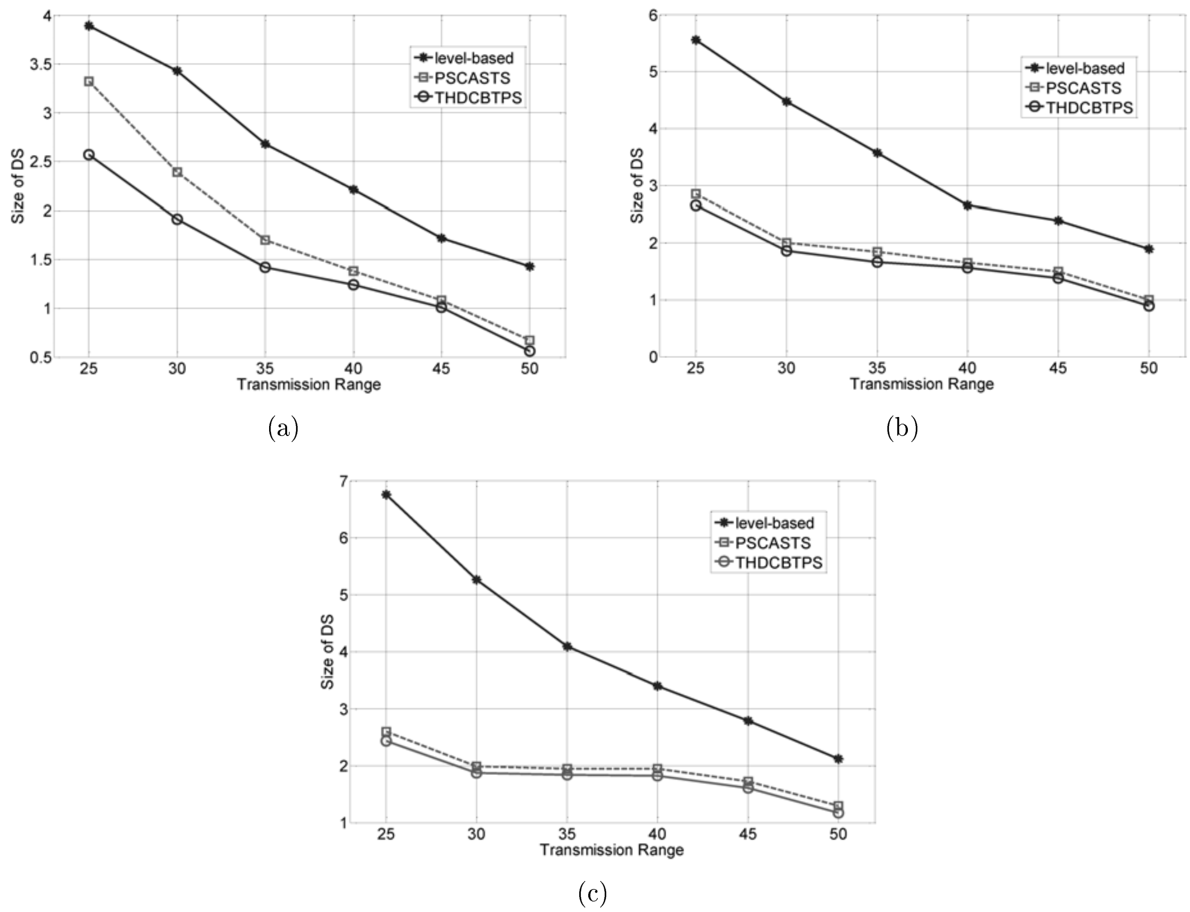


FIGURE 6. Comparison of the size of ST of three algorithms: (a) $N = 20$, (b) $N = 50$ and (c) $N = 100$

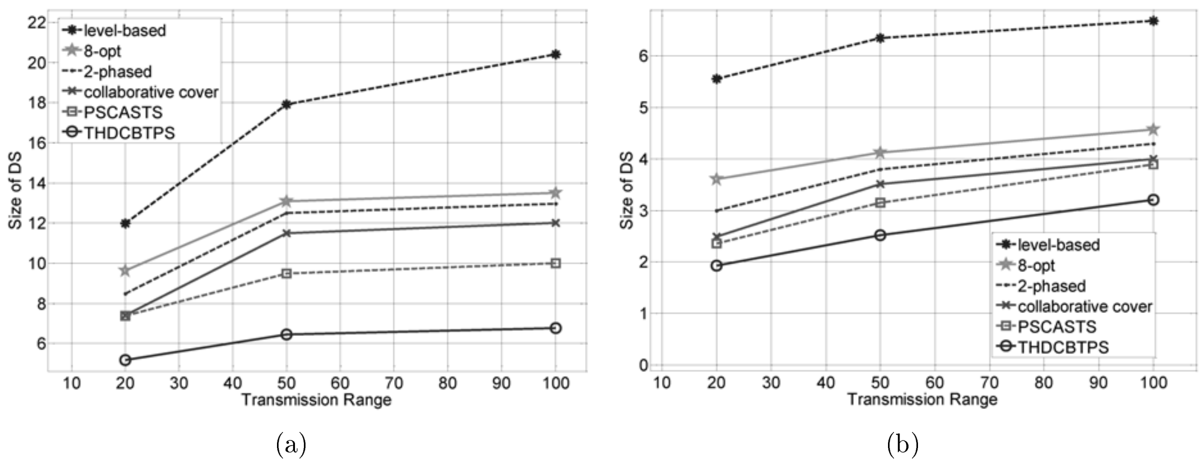


FIGURE 7. Comparison of the size of CDS of different algorithms: (a) $r = 25$, (b) $r = 50$

it is not as obvious as the reduction in the first phase comparing with PSCASTS when the size of the network grows over 50 nodes (7% and 6% when $N = 50$ and $N = 100$ respectively).

At last (Figure 7), we compare the performance of THDCBTPS with the CDS construction techniques reported in [6,7,22,26,27] by setting the number of nodes N to be 20, 50 and 100, and the transmission range r to be 25 and 50. The result is presented in Figure 7. Figure 7 demonstrates that THDCBTPS outperforms the level-based [6], 8-opt [7], 2-phased [22], collaborative cover [26] and PSCASTS [27] in identifying a smaller size CDS. The results reveal that our approach reduces the CDS size by 32% and 19% compared with the previous best approach PSCASTS for the case of $r = 25$ and $r = 50$ respectively. Furthermore, the collaborative cover, 2-phased, 8-opt and the level-based algorithms result in 40%, 46%, 49% and 63% higher CDS sizes respectively when compared with THDCBTPS for the case of $r = 25$, 23%, 31%, 38% and 59% for the case of $r = 50$.

6. **Conclusions.** In this paper, a two-phased approximation algorithm which identifies a minimal size connected dominating set has been described by using the information of two-hop degree centrality, whose approximation factor is at most $6.075|opt| + 5.425$, where $|opt|$ is the size of an optimal CDS of the network. A two-hop connected subgraph degree based priority condition identifies the Steiner nodes leading to a Steiner tree for independent set nodes. It improves the existing approximation for reported CDS algorithms. In order to obtain a more global result, the proposed approach abandons the framework of multiple leaders. This results in a time complexity of $O(n)$ time and $O(D)$ rounds, where n is the sum of nodes and D is the network diameter.

Acknowledgment. Part of this work was supported by following grants respectively: A Project of Shandong Province Higher Educational Science and Technology Program: No. J14LN25; Shandong Province Natural Science Foundation, China: No. ZR2014FP003; Shandong Province Natural Science Foundation, China: No. ZR2015FQ13; Shandong Province Natural Science Foundation, China: No. ZR2014FQ012; National Science Foundation of China: No. 61501278; National Science Foundation of China: No. 61572269 and the National Science & Technology Pillar Program during the 12th Five-Year Plan Period of China: No. 2014BAK12B06-01.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, Wireless sensor networks: A survey, *Computer Networks*, vol.38, no.4, pp.393-422, 2002.
[2] B. Chen, K. Jamieson et al., Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks, *Wireless Networks*, vol.8, no.5, pp.481-494, 2002.
[3] B. N. Clark, C. J. Colbourn and D. S. Johnson, Unit disk graphs, *Annals of Discrete Mathematics*, vol.48, pp.165-177, 1991.
[4] B. Das and V. Bharghavan, Routing in ad-hoc networks using minimum connected dominating sets, *Proc. of IEEE International Conference on Communications*, pp.376-380, 1997.
[5] L. Bao and J. J. Garcia-Luna-Aceves, Topology management in ad hoc networks, *Proc. of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp.129-140, 2003.
[6] P. J. Wan, K. M. Alzoubi and O. Frieder, Distributed construction of connected dominating set in wireless ad hoc networks, *Mobile Networks and Applications*, vol.9, pp.141-149, 2004.
[7] M. Cardei, M. X. Cheng, X. Cheng and D. Z. Du, Connected domination in multihop ad hoc wireless networks, *Proc. of the 6th Joint Conference on Information Science*, CiteSeer, pp.251-255, 2002.
[8] J. Wu and H. Li, On calculating connected dominating set for efficient routing in ad hoc wireless networks, *Proc. of the 3rd ACM International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp.7-14, 1999.
[9] F. Dai and J. Wu, An extended localized algorithm for connected dominating set formation in ad hoc wireless networks, *IEEE Trans. Parallel and Distributed Systems*, vol.15, no.10, pp.908-920, 2004.
[10] A. Qayyum, L. Viennot and A. Laouiti, Multipoint relaying for flooding broadcast messages in mobile wireless networks, *Proc. of the 35th IEEE Annual Hawaii International Conference on System Sciences*, pp.3866-3875, 2002.

[11] M. Min, H. Du, X. Jia, C. X. Huang, S. C. H. Huang and W. Wu, Improving construction for connected dominating set with Steiner tree in wireless sensor networks, *Journal of Global Optimization*, vol.35, no.1, pp.111-119, 2006.

[12] Z. J. Haas, J. Y. Halpern and L. Li, Gossip-based ad hoc routing, *IEEE/ACM Trans. Networking*, vol.14, no.3, pp.479-491, 2006.

[13] F. Dai and J. Wu, On constructing k-connected k-dominating set in wireless networks, *Proc. of the 19th IEEE International Symposium on Parallel and Distributed Processing*, pp.81a-81a, 2005.

[14] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, *Proc. of the 33rd IEEE Annual Hawaii International Conference on System Sciences*, vol.2, pp.10-20, 2000.

[15] J. Wu, F. Dai, M. Gao and I. Stojmenovic, On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks, *Journal of Communications and Networks*, vol.4, no.1, pp.59-70, 2002.

[16] A. B. McDonald and T. F. Znati, A mobility-based framework for adaptive clustering in wireless ad hoc networks, *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1466-1487, 1999.

[17] P. Basu, N. Khan and T. D. Little, A mobility based metric for clustering in mobile ad hoc networks, *Proc. of the 21st IEEE International Conference on Distributed Computing Systems Workshops*, pp.413-418, 2001.

[18] W. Wu, X. Gao, P. M. Pardalos and D. Z. Du, Wireless networking, dominating and packing, *Optimization Letters*, vol.4, no.3, pp.347-358, 2010.

[19] M. Li, P. J. Wan and F. Yao, Tighter approximation bounds for minimum CDS in unit disk graphs, *Algorithmica*, vol.61, no.4, pp.1000-1021, 2011.

[20] W. Wu, H. Du, X. Jia, Y. Li and S. C.-H. Huang, Minimum connected dominating sets and maximal independent sets in unit disk graphs, *Theoretical Computer Science*, vol.352, no.1, pp.1-7, 2006.

[21] S. Funke, A. Kesselman et al., A simple improved distributed algorithm for minimum CDS in unit disk graphs, *ACM Trans. Sensor Networks*, vol.2, no.3, pp.444-453, 2006.

[22] P. J. Wan, L. Wang and F. Yao, Two-phased approximation algorithms for minimum CDS in wireless ad hoc networks, *Proc. of the 28th IEEE International Conference on Distributed Computing Systems*, pp.337-344, 2008.

[23] X. Gao, Y. Wang, X. Li and W. Wu, Analysis on theoretical bounds for approximating dominating set problems, *Discrete Mathematics, Algorithms and Applications*, vol.1, no.01, pp.71-84, 2009.

[24] J. Zhang, L. Xu, S. M. Zhou and W. Wu, Constructing connected dominating set based on crossed cube in WSN, *Proc. of the 5th IEEE International Conference on Intelligent Networking and Collaborative Systems*, pp.443-447, 2013.

[25] J. A. Torkestani and M. R. Meybodi, An intelligent backbone formation algorithm for wireless ad hoc networks based on distributed learning automata, *Computer Networks*, vol.54, no.5, pp.826-843, 2010.

[26] R. Misra and C. Mandal, Minimum connected dominating set using a collaborative cover heuristic for ad hoc sensor networks, *IEEE Trans. Parallel and Distributed Systems*, vol.21, no.3, pp.292-302, 2010.

[27] A. Das, C. Mandal, C. Reade and M. Aasawat, An improved greedy construction of minimum connected dominating sets in wireless networks, *Proc. of IEEE Wireless Communications and Networking Conference*, pp.790-795, 2011.

[28] X. Li, X. Gao and C. Zhao, A new greedy algorithm for D-hop connected dominating set, *Proc. of the 10th IEEE International Conference on Mobile Ad-hoc and Sensor Networks*, pp.54-57, 2014.

[29] J. Li, X. Zhu, X. Gao et al., A novel approximation for multi-hop connected clustering problem in wireless sensor networks, *Proc. of the 35th IEEE International Conference on Distributed Computing Systems*, pp.696-705, 2015.

[30] Z. Zhang and Y. Shi, Approximation algorithm for minimum weight fault-tolerant virtual backbone in homogeneous wireless sensor network, *Proc. of IEEE Conference on Computer Communications*, pp.1080-1085, 2015.

[31] S. Ren, P. Yi, Y. Wu and J. Li, An asynchronous distributed algorithm for constructing a connected dominating set optimized by minimum-weight spanning tree, *Proc. of the 17th IEEE International Conference on Computational Science and Engineering*, pp.1205-1212, 2014.