# Minimum Total Communication Power Connected Dominating Set in Wireless Networks*

Deying Li[1], Donghyun Kim[2,**], Qinghua Zhu[1], Lin Liu[1], and Weili Wu[3]

[1] School of Information, Renmin University of China, Beijing 100872, China
{deyingli,qinghuazhu,spl}@ruc.edu.cn
[2] Dept. of Mathematics and Computer Science,
North Carolina Central University 1801 Fayetteville St. Durham NC 27707, USA
donghyun.kim@nccu.edu
[3] Dept. of Computer Science,
University of Texas at Dallas, Richardson, TX, 75080, USA
weiliwu@utdallas.edu

**Abstract.** A virtual backbone of a wireless network is a connected subset of nodes responsible for routing messages in the network. A node in the subset is likely to be exhausted much faster than the others due to its heavy duties. This situation can be more aggravated if the node uses higher communication power to form the virtual backbone. In this paper, we introduce the *minimum total communication power connected dominating set (MTCPCDS)* problem, whose goal is to compute a virtual backbone with minimum total communication power. We show this problem is NP-hard and propose two distributed algorithms. Especially, the first algorithm, MST-MTCPCDS, has a worst case performance guarantee. A simulations is conducted to evaluate the performance of our algorithms.

## 1 Introduction

A *virtual backbone (VB)* of a wireless network is a connected subset of nodes such that each node outside the subset is adjacent to a node in the subset. It is well-known that the substructure can be exploited to improve efficiency of wireless networks. A VB causes less overhead and becomes more effective if its size is small. The *minimum connected dominating set (MCDS)* problem is to

* This work was supported in part by the NSFC under Grants No. 61070191 and 91124001, the Fundamental Research Funds for the Central Universities,and the Research Funds of Renmin University of China 12XNH179, and Research Fund for the Doctoral Program of Higher Education of China No. 20100004110001. This work was also supported in part by US National Science Foundation (NSF) CREST No. HRD-0833184 and by US Army Research Office (ARO) No. W911NF-0810510. This work was partially supported by the NSF under Grant No. HRD-0833184, IIS-0513669, CCF-0621829, and CNS-0524429.

** Corresponding Author.

X. Wang et al. (Eds.): WASA 2012, LNCS 7405, pp. 132–141, 2012.

find a connected subset of nodes such that all nodes outside the subset has a neighbor in the subset, and frequently used to compute a quality VB. Since it is NP-hard, several approximation algorithms [1–3] and a full polynomial-time approximation scheme (FPTAS) [4] are introduced for MCDS in unit disk graph (UDG). In [5–7], the authors introduced distributed algorithms for MCDS. In [8], Kim et al. studied MCDS in unit ball graph (UBG). In [9], Thai et al. studied MCDS in disk graph (DG). The minimum node-weight dominating set (or connected dominating set) problem is also extensively studied [10–13].

Due to their heavy duties, the nodes in a VB are likely to be exhausted much faster than the other nodes. In addition, this situation can be further aggravated if the nodes use higher communication power to form the VB. Based on this observation, we claim a VB with smaller total (or equivalently average) communication power to form a CDS is more energy-efficient. In the literature, topology control of a wireless network via communication power adjustment is frequently used to improve the energy-efficiency of a protocol running over the network without compromising its performance [14–17]. To the best of our knowledge, however, no effort has been made to find a CDS in a wireless network of nodes with adjustable communication power, and *our work is the first one making an effort toward this direction.* In fact, it has been implicitly assumed that every node of a wireless network has a fixed transmission power when computing a VB.

In this paper, we introduce the *minimum total communication power connected dominating set (MTCPCDS)* problem, whose goal is to find a CDS of a wireless network such that the sum of communication power of the nodes in the CDS becomes minimum. The formal definition of MTCPCDS is in Definition 1. Note that MTCPCDS problem can be considered as a generalization of the problem models in [10–13]. The summary of the contributions is as follow. First, we propose MTCPCDS and show it is NP-hard. Second, we introduce a simple distributed approximation algorithm, *a minimum spanning tree (MST) based distributed algorithm for MTCPCDS (MST-MTCPCDS)*, prove its performance ratio, and analyze its time and message complexities. Third, we introduce *a new greedy heuristic algorithm for MTCPCDS (GREEDY-MTCPCDS)*, and analyze its time and message complexities. At last, we study the average performance of the proposed algorithms via simulation.

The rest of this paper is organized as follows. Section 2 presents the notations, definitions, and important assumptions. Section 3 and Section 4 introduce MST-MTCPCDS and GREEDY-MTCPCDS, respectively. Our simulation result and corresponding discussions are given in Section 5. Finally, Section 6 concludes this paper.

## 2 Notations, Assumptions, and Problem Definition

In this paper, $V$ is the set of the nodes in a given wireless network and $n$ is the number of the nodes. Given $V$ and corresponding communication power assignment of the nodes, $G[V]$ is the communication graph induced by the nodes.

For simplicity, we will use $G = (V, E)$ to represent the communication graph. Therefore, the meaning of $G$ is highly dependent on the context. $G(V, E)$ is a communication graph with a node set $V$ and an edge set $E$. In many cases, a graph in this paper is edge-weighted and we use $w_E(u, v)$ to represent the edge weight between two nodes $u, v \in V$. Each node $u$ can adjust its communication power $p(u)$ such that $0 \leq p(u) \leq p_{max}(u)$, where $p_{max}(u)$ is the maximum communication power of $u$. $P_{max} = \bigcup_{u \in V} p_{max}(u)$.

As like [15–17], we assume the energy $\mathcal{E}$ consumed to transmit a bit of message is $\mathcal{E} = \beta \cdot d^{\alpha}$, where $d$ is the travel distance of the message, $\alpha$ is a power attenuation factor, a constant between 2 and 5, and $\beta$ is some constant. $Hopdist(u, v)$ and $Eucdist(u, v)$ are the hop and euclidean distance between $u$ and $v$, respectively.

**Definition 1 (MTCPCDS).** *Given a pair $\langle V, P_{max} \rangle$, MTCPCDS is to determine the communication power of each node and find a subset $D \subseteq V$ such that 1) each node is either in $D$ or is (bidirectionally) connected to a node in $D$, 2) $G[D]$ is connected, and 3) the total communication power assigned to $D$ is minimum. More formally, it is to find $\langle D \subseteq V, \{p(u) | u \in D\} \rangle$ such that 1) $\forall u \in D$, $0 < p(u) \leq p_{max}(u)$, 2) both of $G(D, E_1)$ and $G(V, E_1 \cup E_2)$ are bidirectionally connected, where $E_1 = \{(u, v) | \min\{p(u), p(v)\} \geq \beta \cdot Eucdist(u, v)^{\alpha}, \forall u, v \in D\}$, and $E_2 = \{(u, v) | \min\{p(u), p_{max}(v)\} \geq \beta \cdot Eucdist(u, v)^{\alpha}, \forall u \in D, v \notin D\}$, and 3) $\sum_{v \in D} p(v)$ is minimum, respectively.*

**Theorem 1.** *The MTCPCDS problem is NP-hard.*

*Proof.* Imagine a grid graph such that the euclidean distance between any two neighbors is exactly 1. Clearly, such grid graph is a special case of UDG. Next, consider a subclass of MTCPCDS defined over the grid graph such that 1) $p_{max}(v) = 1$ for all $v \in V$. In such grid graph, the subclass of MTCPCDS is equivalent to MCDS since the power level of each node in an optimal solution of the subclass has to be either 0 or 1. (the power level of a node is 0 means the node is not in the CDS. Otherwise, it is in the CDS.) By [18], MCDS is still NP-hard even in such grid graph. Therefore, the subclass of MTCPCDS is also NP-hard. As a result, MTCPCDS without the constraint on the maximum power level of each node is NP-hard in general UDGs.

## 3 A MST Based Approximation Algorithm for MTCPCDS (MST-MTCPCDS)

Now, we introduce MST-MTCPCDS. Given $\langle V, P_{max} \rangle$, the algorithm performs the following steps in a sequential order.

1. Constructs an edge-weighted auxiliary graph $G_{aux}^{EW} = (V_{aux}^{EW}, E_{aux}^{EW})$ such that for any two node pair $u$ and $v$ in $V$, $(u, v)$ is in $E_{aux}^{EW}$ if and only if $d^{\alpha}(u, v) \leq \min\{p_{max}(v_i), p_{max}(v_j)\}$. Also, $w_E(u, v) = d^{\alpha}(u, v)$ is assigned as the edge weight of $(u, v)$. Note that such construction can be done in a fully distributed (localized) manner by letting each node exchange a "hello" message with its neighbors.

2. Finds an MST $T_{mst}$ of $G_{aux}^{EW}$ using an existing distributed MST algorithm such as Kruskal's algorithm. Suppose $D$ is the set of non-leaf nodes in $T_{mst}$. Clearly, $D$ is a CDS of $G_{aux}^{EW}$ since $G_{aux}^{EW}[D]$ is connected and all nodes in $V \setminus D$ is adjacent to at least one node in $D$.
3. Assign the communication power of each node as follows: i) For each $v \in D$, we set $p(v)$ to the maximum edge weight between $v$ and any $u$ such that $v$ and $u$ are adjacent in $T_{mst}$, and ii) for each node $w \in V \setminus D$, we need to adjust $w$'s power properly so that it can send a message to at least one node in $D$.

**Theorem 2.** *The running time of MST-MTCPCDS is $O(n^2)$.*

*Proof.* The first step takes $O(n^2)$ time to construct $G_{aux}^{EW}$ and assign a weight on each edge of it. The second step takes $O(n^2)$ time to compute an MST $T_{mst}$ using Kruskal's algorithm and find a set $D$ of non-leaf nodes of $T_{mst}$. The last step takes $O(|D| \cdot \Delta)$ time to determine the communication power level of each node in $D$ by observing its neighbors, where $\Delta$ is the maximum degree of $G_{aux}^{EW}$. As a result, the running time of MST-MTCPCDS is $O(n^2)$.

**Theorem 3.** *The approximation ratio of MST-MTCPCDS is 2 $\Delta$ for the MTCPCDS problem.*

*Proof.* Suppose $T$ is any spanning tree in $G_{aux}^{EW}$. Let $NL(T)$ be the set of non-leaf nodes in $T$ and $E(T)$ be the edges in $T$. We denote the weight of an edge $e$ and the communication power level of node $v$ by $w_E(e)$ and $p(v)$, respectively. Since each edge is connecting two end points, $w_E(e)$ can be included in $\sum_{v \in NL(T)} p(v)$ at most two times. Therefore, we have $\sum_{v \in NL(T)} p(v) \leq 2 \sum_{e \in E(T)} w_E(e)$, and $\sum_{e \in E(T)} w_E(e) \leq \Delta \sum_{v \in NL(T)} \max_{\{(v,u) \in T | \forall u \in V\}} d^{\alpha}(v,u) = \Delta \sum_{v \in NL(T)} p(v)$, where $\Delta$ is the maximum degree of $G_{aux}^{EW}$.

Now, suppose $D^*$ is an optimal solution of the MTCPCDS problem. Then, there should be a spanning tree $T^*$ of $D^*$ on $G_{aux}^{EW}$. Also, suppose $D$ is an output of our algorithm given an input $G_{aux}^{EW}$, and $T$ is a corresponding spanning tree of $D$. Then, we can observe 1) in MST-MTCPCDS, $T$ is an MST of $G_{aux}^{EW}$. Since $T^*$ is a spanning tree, we have $\sum_{e \in E(T)} w_E(e) \leq \sum_{e \in E(T^*)} w_E(e)$, and 2) $D^*$ has to be a set of non-leaf nodes of $T^*$. Otherwise, we can remove a leaf node from $D^*$ which contradicts to our assumption that $D^*$ is optimal. Therefore, we have $\sum_{v \in NL(T^*)} p(v) = \sum_{v \in D^*} p(v)$. As a result, $\sum_{v \in D} p(v) \leq 2 \sum_{e \in E(T)} w_E(e) \leq 2 \sum_{e \in E(T^*)} w_E(e) \leq 2\Delta \sum_{v \in NL(T^*)} p(v) = 2\Delta \sum_{v \in D^*} p(v)$, and the theorem holds true.

## 4 GREEDY-MTCPCDS: A New Greedy Heuristic Algorithm for MTCPCDS

GREEDY-MTCPCDS consists of two distinct phases. In the first phase, given a MTCPCDS problem instance, the algorithm computes a $G_{aux}^{EW}$ in a distributed manner as MST-MTCPCDS does. Therefore, no node needs to keep the global information of $G_{aux}^{EW}$. In the second phase, it applies a distributed greedy strategy to $G_{aux}^{EW}$. At the beginning of the second phase, the color of each node is white, but later becomes gray or black. At the end, the set of black nodes forms a CDS.

Given a node $v_i \in V$ and its current communication power $p(v_i)$, the cost to increase its communication power to $p_{new}(v_i)$ is defined as

$$Cost(p(v_i), p_{new}(v_i)) = (p_{new}(v_i) - p(v_i))/(|N[p_{new}(v_i)]|),$$

where $N[p_{new}(v_i)]$ is the set of white nodes in $G_{aux}^{EW}$ dominated by $v_i$ using the new communication power $p_{new}(v_i)$. In case that $|N[p_{new}(v_i)]| = 0$, which implies that $v_i$ cannot reach any white neighbor even using its maximum communication power, $Cost(p(v_i), p_{new}(v_i))$ returns $-1$. Intuitively, this cost function is representing the cost-efficiency of increasing the communication power of $v_i$ from $p(v_i)$ to $p_{new}(v_i)$.

The second phase consists of multiple rounds. Each round is initiated by a current root $r_c$. The very first round is started by electing a new current root $r_c$ with minimum $Cost_{best}(r_c)$, where

$$Cost_{best}(r_c) = \min_{p(r_c) < p_{new}(r_c) \leq p_{max}(r_c)} \{Cost(p(r_c), p_{new}(r_c))\}.$$

Once elected, the $r_c$ increases its communication power to $Cost_{best}(r_c)$. Note that for any $r_c$, the effective number of choices for $p_{new}(r_c)$ is bounded by $\Delta$, which is the maximum degree of $G_{aux}^{EW}$. Then, $r_c$ becomes a black node and each (white) node to which $r_c$ can send a signal using the new communication power $p_{new}(r_c)$ becomes gray. Note that those gray nodes are some of the neighbors of $r_c$ in $G_{aux}^{EW}$. Next, $r_c$ constructs a node set $X$ of the gray nodes, selects the node $v_i \in X$ with minimum $Cost_{best}(v_i)$ value, and sends an invitation to $v_i$ with $X$ and $W = \bigcup_{v_j \in X} Cost_{best}(v_j)$.

On receiving the invitation, $v_i$ becomes a new $r_c$ and repeats the round. Generally speaking, after a new $r_c$ is elected, followings are performed in a sequential order.

1. $r_c$ becomes a black node, adjusts its communication power to $p_{new}(r_c)$ such that $Cost_{best}(r_c)$ can be achieved. All of white neighbors reachable from $r_c$ using the new communication power become gray. Then, $r_c$ calculates $X'$ and $W'$, where $X'$ is the set of gray nodes at most two hops far from $r_c$ and $W'$ is the set of $Cost_{best}(v_i)$ for each $v_i \in X'$. Then, merges those with old $X$ and $W$ which inherited from the previous root (i.e. $X \leftarrow X \bigcup X'$ and $W \leftarrow W \bigcup W'$). While merging, any new information overwrites its old version. To optimize the size of $X$ and $W$, for each $v_j \in X$, if $Cost_{best}(v_j) = -1$, we can remove $v_j$ from $X$ and $Cost_{best}(v_j)$ from $W$ since $v_j$ does not have any reachable white node anymore.
2. Once the merged, the new $r_c$ picks a node $v \in X$ with the minimum $Cost_{best}(v)$ value (which can be found from $W$ within a linear time) as the next $r_c$. If $X$ is empty, then all nodes should be either black or gray, and thus $r_c$ terminates this phase. Otherwise, $r_c$ sends an invitation message to another node $v \in X$ with minimum $Cost_{best}(v)$ value.

After the second phase, the set of black nodes and its corresponding power assignments will be a CDS of $G_{aux}^{EW}$. Now, we analyze the time and message complexities of GREEDY-MTCPCDS.

**Theorem 4.** *Both the time and message complexities of GREEDY-MTCPCDS are* $O(|V|\Delta^2)$.

*Proof.* It take $O(n^2)$ time to obtain $G_{aux}^{EW}$ in the first phase. Now, we discuss about the second phase. In each round, $r_c$ needs to collect the cost information from every black and gray node within two hops. In detail, $r_c$ first sends the query message to its direct neighbors using one broadcasting message and this incurs $O(1)$ time and takes $O(1)$ messages. For each direct neighbor $v_i$ of $r_c$, $v_i$ needs to spend $O(1)$ time and incur $O(1)$ messages to broadcast the query to its direct neighbors. Also, $v_i$ will take $O(\Delta)$ time and incur $O(\Delta)$ messages to collect the cost information from its direct neighbors. To send this to the $r_c$, it will take $O(1)$ time and generate $O(1)$ messages. Therefore, both the time and message complexities of one round is $O(1) + O(\Delta) \cdot O(1 + \Delta + 1) = O(\Delta^2)$. Since we can have at most $|V|$ rounds, the time and message complexities of GREEDY-MTCPCDS are $O(|V|\Delta^2)$.

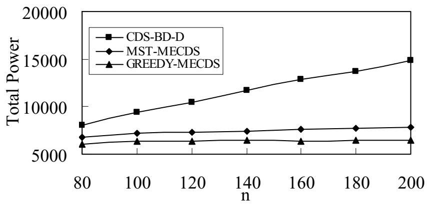
## 5 Simulation Results and Analysis

To the best of our knowledge, there is no CDS computation algorithm adjusting the communication power of each node. Therefore, we compare the average performance of our algorithms with CDS-BD-D in [7], a typical MCDS algorithm. The simulations are conducted over a $100 \times 100$ 2-D space. We compare the total communication power and the size of CDSs generated by the three algorithms under different parameter settings. For each parameter setting, we obtain an averaged result from 100 trials. In each trial, we randomly place $n$ nodes over the terrain. If the induced $G_{aux}^{EW}$ by the nodes is disconnected, we simply discard it and generate a new one.

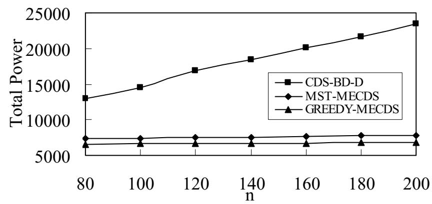
As we mentioned, the power model to send a message over a distance $d$ is $E = \beta \cdot d^{\alpha}$. For simplicity, we normalize $\beta = 1$. We assume that the signal is moving in the air and set $\alpha$ to 2. Then, the remaining tunable parameters in the simulations are as follows:

1. The number of nodes $n$. We vary $n$ from 80 to 200 to check the scalability of the algorithms.
2. The interval of maximum power of each node $[a, b]$. In the simulation, the maximum power of each node is generated from a normal distribution with mean equal to $\frac{a+b}{2}$ and standard deviation equal to $\frac{b-a}{4}$.

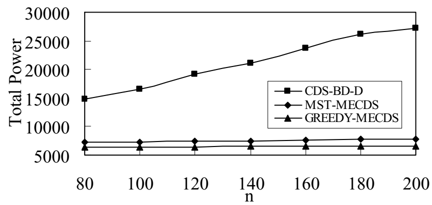
In Figure 1, we compare the averaged total communication power of CDSs generated by the three approaches. In Figure 1(a), 1(b), and 1(c), the maximum communication power of each node is from $[100, 400]$, $[100, 900]$, and $[400, 900]$, respectively. For each interval, we vary the number of nodes from 80 to 200. From this simulation results, we can clearly see both of MST-MTCPCDS and




(a) The maximum power of each node is randomly chosen between 100 to 400.




(b) The maximum power of each node is randomly chosen between 100 to 900.



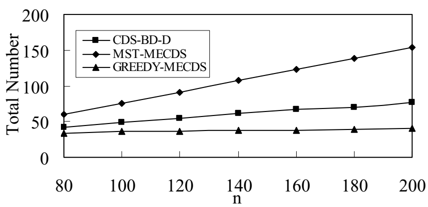

(c) The maximum power of each node is randomly chosen between 400 to 900.

**Fig. 1.** Averaged total communication power of CDSs generated by CDS-BD-D, MST-MTCPCDS, and GREEDY-MTCPCDS

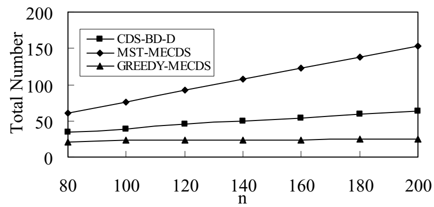
GREEDY-MTCPCDS outperform CDS-BD-D in terms of the averaged total communication power of CDSs. This is natural since CDS-BD-D was designed for MCDS, not for MTCPCDS. We can also observe, while we did not prove the worst case performance of GREEDY-MTCPCDS, GREEDY-MTCPCDS works better than MST-MTCPCDS on average. Therefore, they are in a trade-off relationship.

In Figure 2, we compare the performance of the three approaches using the average size of CDSs, which is a traditional quality measurement for CDS. From the simulation results, we can see that on average, the size of CDSs computed by GREEDY-MTCPCDS is even better than that of CDSs generated by CDS-BD-D, which is an approximation algorithm for the MCDS problem. Meanwhile, we can observe that CDS-BD-D works better than MST-MTCPCDS, but this is understandable since MST-MTCPCDS is an approximation algorithm for MTCPCDS, not for MCDS.

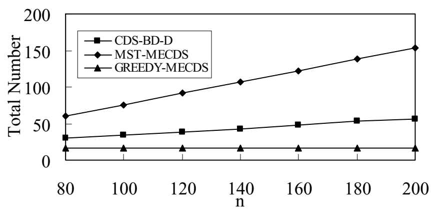
In conclusion, the three algorithms are in a very interesting trade-off relationship. CDS-BD-D is an approximation algorithm for MCDS and has a worst case performance guarantee. MST-MTCPCDS is an approximation algorithm for MTCPCDS and has a worst case performance guarantee. CDS-BD-D is better than MST-MTCPCDS for MCDS, but MCDS is better than CDS-BD-D for MTCPCDS. On the other hand, GREEDY-MTCPCDS is not an approximation algorithm for any of the problems and has no worst case performance guarantee, but on average, it outperforms the other two algorithms in both performance metrics, the size and the total communication power.




(a) The maximum power of each node is randomly chosen between 100 to 400.




(b) The maximum power of each node is randomly chosen between 100 to 900.




(c) The maximum power of each node is randomly chosen between 400 to 900.

**Fig. 2.** Averaged size of CDSs generated by CDS-BD-D, MST-MTCPCDS, and GREEDY-MTCPCDS

# 6 Conclusions and Future Work

In this paper, we proposed the minimum total communication power connected dominating set (MTCPCDS) problem. In detail, given $n$ nodes and their maximum communication powers, MTCPCDS is to determine each node's communication power and to find a CDS of the network. We proved this problem is NP-hard and proposed two distributed approaches. The first approach exploits an existing distributed approximation algorithm for MST to solve MTCPCDS and has a worst case performance guarantee. The second one is a simple greedy algorithm and theoretically runs faster than the first one in a sparse graph. In the extensive simulations, we saw that they are in a very interesting trade-off relationship and produce quality solutions for MTCPCDS.

In [19], the authors studied a problem similar to MTCPCDS, but did not consider the maximum communication power level of each node, and thus is less realistic. In such a case, a constant factor approximation can be easily obtained. However, for MTCPCDS with the maximum communication power level constraint, we were only able to obtain a $O(\Delta)$ approximation. Therefore, obtaining a constant factor approximation of MTCPCDS is still open.

In this paper, we mostly focused on establishing a theoretical foundation of the problem and its solutions. As a future work, we are interested in taking the remaining energy level of each node into the consideration and try to improve our approach so that it can actually help to extend the lifetime of CDS based wireless networks.

## References


1. Guha, S., Khuller, S.: Approximation Algorithms for Connected Dominating Sets. Algorithmica 20, 374–387 (1996)
2. Funke, S., Kesselman, A., Meyer, U., Segal, M.: A Simple Improved Distributed Algorithm for Minimum CDS in Unit Disk Graphs. ACM Transactions on Sensor Networks (TOSN) 2(3), 444–453 (2006)
3. Li, X., Gao, X., Wu, W.: A Better Theoretical Bound to Approximate Connected Dominating Set in Unit Disk Graph. In: Li, Y., Huynh, D.T., Das, S.K., Du, D.-Z. (eds.) WASA 2008. LNCS, vol. 5258, pp. 162–175. Springer, Heidelberg (2008)
4. Cheng, X., Huang, X., Li, D., Wu, W., Du, D.-Z.: A Polynomial-Time Approximation Scheme for the Minimum-Connected Dominating Set in Ad Hoc Wireless Networks. Networks 42(4), 202–208 (2003)
5. Wan, P.-J., Alzoubi, K.M., Frieder, O.: Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks. ACM Journal on Mobile Networks and Applications (MONET) 9(2), 141–149 (2004)
6. Wu, J., Dai, F., Gao, M., Stojmenovic, I.: On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks. IEEE/KICS Journal of Communications and Networks 4, 59–70 (2002)
7. Kim, D., Wu, Y., Li, Y., Zou, F., Du, D.-Z.: Constructing Minimum Connected Dominating Sets with Bounded Diameters in Wireless Networks. IEEE Transactions on Parallel and Distributed Systems (TPDS) 20(2), 147–157 (2009)
8. Kim, D., Zhang, Z., Li, X., Wang, W., Wu, W., Du, D.-Z.: A Better Approximation Algorithm For Computing Connected Dominating Sets in Unit Ball Graphs. IEEE Transactions on Mobile Computing (TMC) 9(8), 1108–1118 (2010)
9. Thai, M.T., Wang, F., Liu, D., Zhu, S., Du, D.-Z.: Connected Dominating Sets in Wireless Networks with Different Transmission Ranges. IEEE Transactions on Mobile Computing (TMC) 6(7), 721–730 (2007)
10. Guha, S., Khuller, S.: Improved Methods for Approximating Node Weighted Steiner Trees and Connected Dominating Sets. Information and Computation 150(1), 57–74 (1999)
11. Gao, X., Huang, Y., Zhang, Z., Wu, W.: $(6 + \epsilon)$-Approximation for Minimum Weight Dominating Set in Unit Disk Graphs. In: Hu, X., Wang, J. (eds.) COCOON 2008. LNCS, vol. 5092, pp. 551–557. Springer, Heidelberg (2008)
12. Huang, Y., Gao, X., Zhang, Z., Wu, W.: A Better Constant-Factor Approximation for Weighted Dominating Set in Unit Disk Graph. Journal of Combinatorial Optimization (JOCO) 18(2), 179–194 (2009)
13. Dai, D., Yu, C.: A $(5+\epsilon)$-Approximation Algorithm for Minimum Weighted Dominating Set in Unit Disk Graph. Theoretical Computer Science (TCS) 41(8-10), 756–765 (2009)
14. Clementi, A., Crescenzi, P., Penna, P., Rossi, G., Vocca, P.: On the Complexity of Computing Minimum Energy Consumption Broadcast Subgraphs. In: Ferreira, A., Reichel, H. (eds.) STACS 2001. LNCS, vol. 2010, pp. 121–131. Springer, Heidelberg (2001)
15. Wan, P.J., Calinescu, G., Li, X.Y., Frieder, O.: Minimum-energy Broadcasting in Static Ad Hoc Wireless Networks. Wireless Networks 8(6), 607–617 (2002)
16. Ambühl, C.: An Optimal Bound for the MST Algorithm to Compute Energy Efficient Broadcast Trees in Wireless Networks. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 1139–1150. Springer, Heidelberg (2005)

17. Flammini, M., Klasing, R., Navarra, A., Perennes, S.: Improved Approximation Results for the Minimum Energy Broadcasting Problem. Algorithmica 49(4), 318–336 (2007)
18. Clark, B.N., Colbourn, C.J., Johnson, D.S.: Unit Disk Graphs. Discrete Mathematics 86, 165–177 (1990)
19. Carmi, P., Katz, M.J., Segal, M., Shpungin, H.: Fault-Tolerant Power Assignment and Backbone in Wireless Networks. Ad Hoc & Sensor Wireless Networks 4(4), 355–366 (2007)