Minimum Power Minimum D-Hop Dominating Sets in Wireless Sensor Networks

Trac N. Nguyen, Dung T. Huynh, and Jason A. Bolla

Department of Computer Science
University of Texas at Dallas
Richardson, Texas 75083-0688
{nguyentn,huynh,jason.bolla}@utdallas.edu

Abstract. Clustering structures are used in wireless ad hoc and sensor networks to provide for efficient communications and control. In addition to communications requirements, another important area of concern is power consumption. With that in mind, we would like to find a good network structure that uses a minimum power. In graph theoretic terminology, this paper considers the problem of clustering to be the problem of assigning powers to a set of nodes in the plane, such that we minimize total power and yield a graph that has a dominating set of a desired size. We first show that this problem is NP-complete for planar geometric graphs. We then propose heuristic solutions to the problem, present simulation data for the heuristics, and discuss the results.

1 Introduction

Wireless sensor networks have been widely used in military and civilian applications. Due to the limited power available in each sensor, an important problem in wireless sensor networks is maximizing the network lifetime. Finding techniques to minimize the total power usage of a network while maintaining certain network properties has been the focus of several recent research papers.

Computing dominating sets in a network is one approach to extend the network lifetime as discussed in [3], [4], [6], [5] and [2]. The work in [3] and [4] addresses the use of connected dominating sets in routing or forming a backbone. There are also some studies on how to enhance the connectivity in a network using dominating sets. One way is to ensure that each sensor node is required to connect to at least k dominating sensors as in [5]. Another technique is to create a wake-up schedule for a collection of disjoint (connected) dominating sets as discussed in [6]. The work in [1] also discusses the use of schedules for disjoint dominating sets to provide better coverage. In [2], the author considers the physical characteristic of the battery to schedule a sleep time for sensors in each (connected) dominating set. [7] discusses the construction of a backbone with different adjustable transmission ranges.

Y. Li et al. (Eds.): WASA 2008, LNCS 5258, pp. 176–187, 2008.

In this paper, we study the problem of assigning minimum total power to sensor nodes to form a connected graph that has a dominating set of a desired size. We prove that this problem is NP-complete for planar geometric graphs. In view of known NP-completeness results for WSNs that hold for general graphs only (which is unrealistic), our result is significant as it is derived for the planar geometric graphs which are among the simplest models of WSNs (other simple models include the unit disk graphs which have been extensively studied in the literature). The main contributions of this paper are:

1. The NP-Completeness of the minimum power minimum d-hop dominating set problem for planar geometric graphs (the proof is fairly technical as presented below).
2. The introduction of four heuristics for the minimum total power minimum dominating set problem as well as some simulation results that show how these heuristics perform and illustrate trade-offs between total power usage and dominating set size.

The rest of this paper is organized as follows. Section 2 provides definitions and explanations of models used in this paper. Section 3 contains the NP-completeness proof. The four heuristics described in Section 4 are Shortening Diameter, Shrinking Dominating Set, Shortening All Paths, and Power Level Search. Section 5 describes the simulation results for these heuristics, and Section 6 contains some concluding remarks.

2 Preliminaries

A wireless sensor network is represented as a graph $G = (V, E)$ where each vertex (node) is a sensor node and each edge is a communication link established when two sensors nodes are in the broadcast range of each other. An undirected edge is the combination of two directed edges between two sensor nodes, which are able to transmit and receive information from each other. In this paper, we assume edges are undirected (bidirectional). There are some restricted graph models in wireless networks such as planar graphs, unit disk graphs and planar geometric graphs. A d-hop dominating set (DS) in a graph $G(V, E)$ is a subset S of nodes such that every node is in S or at most d hops away from a node in S.

In our experimental model each node has a broadcast range. The communication between nodes in a network on the plane does not have noise or obstacles. This model of wireless sensor networks forms a geometric graph. A *geometric* graph is defined as a set of points $p_1, .., p_n$ on the plane where each point is specified by its x and y coordinates together with its transmission radius r_i. An edge exists between two points if they are within the transmission radius of each other. A geometric graph is said to be *planar* if it can be arranged so that no edge crosses another. In our simulation, the points $p_1, .., p_n$ are generated randomly on the plane.

3 NP-Completeness of Minimum Total Power Minimum D-Hop Dominating Sets

In this section we prove that the problem of minimizing the total power usage to yield a d-hop dominating set of a bounded size for planar geometric graphs is NP-complete.

Consider a set V of transceivers (nodes) in the plane. Each node u is assigned a power level denoted by $p(u)$. The signal transmitted by node u can only be received by a node v if the distance between u and v, denoted by $d(u, v)$, is $\leq p(u)$. We only consider the bidirectional case in which a communication edge exists between two nodes, u and v, only if both $p(u) \geq d(u, v)$ and $p(v) \geq d(v, u)$. The main problem investigated in this paper is defined as follows. Let $d > 0$ be a fixed integer.

MINIMUM TOTAL POWER MINIMUM D-HOP DOMINATING SETS IN PLANAR GEOMETRIC GRAPHS

Instance: *Given a set of N nodes $V = \{v_1, v_2, ..., v_N\}$ on a plane where each node v_i has a set of power levels $P_i = \{p^i_1, p^i_2, .., p^i_M\}$ at which node v_i can transmit, a positive number Q and a positive integer $K \leq N$.*

Question: *Is there a power assignment to each node that induces a planar geometric graph $G(V, E)$ containing a d-hop dominating set of size $\leq K$ such that the total power usage by the nodes in G is $\leq Q$?*

In the following we show that the problem of assigning minimum power to a set of nodes in the plane in order to obtain a planar geometric graph that has a d-hop dominating set of a desired size is NP-complete.

Theorem 1. The Minimum Total Power Minimum d-hop Dominating Set problem is NP-Complete for planar geometric graphs.

Proof. The Minimum Total Power Minimum d-hop Dominating Set (MTP-MDDS) problem is clearly in NP. Given a set V of nodes v_i in the plane, a set P_i of power levels for v_i, Q, and K, we can nondeterministically assign a power level in P_i to node v_i, nondeterministically select a subset S of nodes, and verify in polynomial time that (1) the power assignment yields a connected and planar geometric graph $G(V, E)$, (2) each node in $G(V, E)$ is either in the set S or d hops away from a node in S (i.e., S is a d-hop DS), (3) S has the size of $\leq K$, and (4) the total power usage of all nodes is $\leq Q$.

To prove the NP-hardness of the MTP-MDDS problem, we construct a polynomial-time reduction from the vertex cover for planar graphs with maximum degree 3 (VC-Deg3) problem, which was proven to be NP-complete in [10]. Given an instance $< G(V, E), K >$ of VC-Deg3, we construct the instance $< V', \{P_1, P_2, ..P_N\}, Q, K' >$ of MTP-MDDS as follows. First, we use Valiant's result [9] to embed the planar graph G into the Euclidian plane:

> *A planar graph with maximum degree 4 can be embedded in the plane using $O(|V|)$ area in such a way that its vertices are at integer coordinates and its edges are drawn so that they are made up of line segments of form $x = i$ or $y = j$, for integers i and j.*

This embedding process can easily be designed to satisfy the additional requirements that each edge must be of length at least 3 units and that every pair of parallel edges in the embedded graph must also be at least 3 units apart.

Let δ be the unit distance in the plane, l_{uv} be the length of the edge connecting two *original* nodes u, v (embedded in the plane), and $d \geq 2$ be the number of hops. Letting $\sigma := \delta/3$, we define three radii r_1, r_2 and r_3 as follows: $r_1 := \sigma/(d+1)$, $r_2 := \sigma + 0.001\delta$ and $r_3 := \sigma * 2$. Every edge (which is a set of line segments) connecting any two *original* nodes is modified by placing additional nodes to create an instance of MTP-MDDS as follows:

1. Keep the *original* nodes at the same locations in the plane.
2. On the edge (u, v) of length l_{uv} connecting two *original* nodes u, v, we add a total of $3 * l_{uv} - 1$ consecutive nodes between u and v such that there is an equal distance of $\delta/3$ from one node to the next. These nodes are called *intermediate* nodes. The two intermediate nodes at the two ends, called *control* nodes, that are adjacent with two *original* nodes will have the exact distance of r_2 to the *original* nodes and the next *intermediate* nodes (also called *interfacing* nodes). This can be accomplished by moving the *control* nodes slightly away from the line segments.
3. Perpendicular to each edge connecting two *orignal* nodes, we attach $(d-1)$ nodes on each *intermediate*, *control*, and *interfacing* node. We also attach to each *original* node $(d-2)$ nodes. These nodes are called *auxiliary* nodes. They are added at the distance r_1 from one node to the next starting from the *intermediate*, *control* or *original* node. The *auxiliary* nodes attached on each *interfacing* start at the distance $(2/3) * r_3$ from the *interfacing* node. Moreover, the *auxiliary* nodes added to the *interfacing* nodes surrounding an *original* node are attached so that they do not belong to the same quadrant (defined by the original node and its incident line segments).

Examples of the embedded graphs are shown in Figure 1 for $d = 2$ (left) and $d = 1$ (right). For $d = 2$, A, B, C, D are *original* nodes, whereas c's are *control* nodes and i's are *interfacing* nodes. Nodes between *interfacing* nodes on each edge are *intermediate* nodes. Other nodes are *auxiliary* nodes.

The numbers of additional *intermediate* (I), *control* (C), *interfacing* (I_f) and *auxiliary* (A) nodes added to $G'(E', V')$ can be computed as follows:

$$I = \sum_{(u,v)\in E} (3*l_{uv}-5), \qquad I_f = C = 2*|E|, \qquad A = (I+C+I_f)*(d-1)+|V|*(d-2)$$

Let V' denote the set of all vertices of G embedded in the plane, and $N := |V|$. To complete the construction of the instance of MTP-MDDS, we define the set of power levels for each node in V, the maximum total power Q, and K' as follows:

Each *auxiliary* node attached to an *interfacing* node at the distance $(2/3) * r_3$ is assigned the power levels $\{0, r_1, r_2, r_3\}$. The power levels assigned to all other *auxiliary* nodes are $\{0, r_1\}$. The set of power levels assigned to each *control* and *intermediate* node is $\{0, r_1, r_2\}$, whereas the set of power levels assigned to each *interfacing* and *original* node is $\{0, r_1, r_2, r_3\}$. Q and K are defined by:

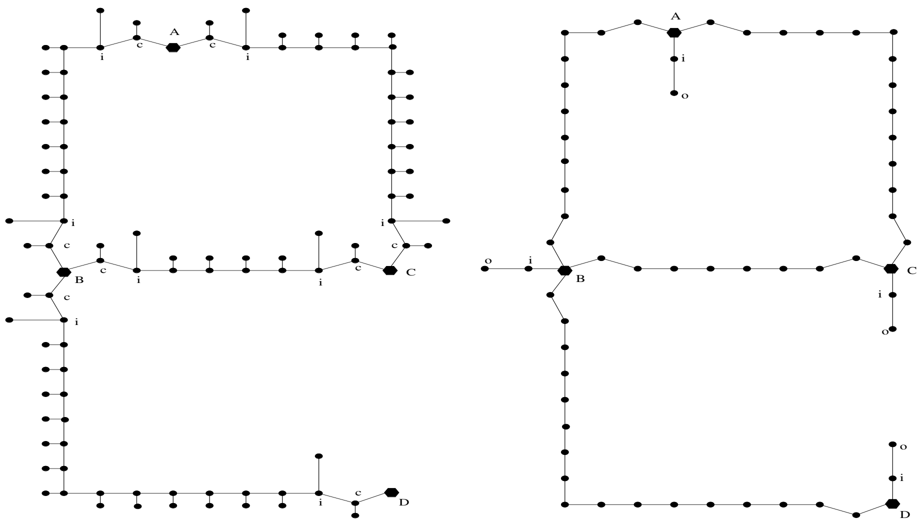

Fig. 1. Construction of $G'(V', E')$ for d=2 (left) and construction of $G'(V', E')$ for d=1 (right)

$$Q = (K + 2 * I_f) * r_3 + ((|V| - K) + (I + C)) * r_2 + (A - I_f) * r_1, \quad K' = K + \sum_{(u,v) \in E} (l_{uv} - 1)$$

where I, I_f, A and C are the numbers of *intermediate*, *interfacing*, *auxiliary* and *control* nodes, respectively, and V is the set of vertices of G. r_1, r_2 and r_3 are the three radii defined earlier.

To prove the correctness of the above polynomial-time reduction, we show that the instance $< G(V, E), K >$ of VC-Deg3 has a vertex cover D of size $\leq K$ if and only if the instance $< V', \{P_1, P_2, .., P_N\}, K', Q >$ of MTP-MDDS has a power assignment that yields a planar geometric graph containing a d-hop dominating set D' of size $\leq K'$, and the total power usage POV is $\leq Q$.

For the only-if direction, suppose that the graph $G(V, E)$ of the VC-Deg3 instance has a dominating set D of size $\leq K$. We define the power assignment for nodes in V' as follows:

- If node $v \in V$ and $v \in D$, assign power level r_3 to the *original* node $v \in V'$.
- If node $v \in V$ and $v \notin D$, assign power level r_2 to the *original* node $v \in V'$.
- Assign the power level r_3 to all *interfacing* nodes and every *auxiliary* node which is at the distance $(2/3) * r_3$ from an *interfacing* node.
- All other auxiliary nodes are assigned the power level r_1.
- Assign the power level r_2 to all other *intermediate* and *control* nodes.

Clearly, the total power assigned to nodes in V' is at most Q and the resulting graph is a planar geometric graph. The d-hop DS D' is constructed as follows:

1. If a node $u \in V$ and $u \in D$, then add the *original* node $u \in V'$ to D'.
2. Given an edge $(u, v) \in E$ with $u \in D$ and $v \notin D$, for every group of 3 consecutive *intermediate* nodes starting from u, we add the *intermediate* node in the middle of the group to D'. The *interfacing* node closest to v is also added to D'.
3. For an edge $(u, v) \in E$ with $u \in D$ and $v \in D$, we add nodes to D' in a manner similar to the previous step.

Clearly, for every set of line segments in E' representing the original edge $(u, v) \in E$, we only add a total of $(l_{uv} - 1)$ non-original nodes to D'. Thus, the total number of nodes in D is $|D'| = |D| + \sum_{(u,v)\in E}(l_{uv} - 1) \leq K' = K + \sum_{(u,v)\in E}(l_{uv} - 1)$.

It is quite straightforward to argue that D' is a d-hop DS. The details are left to the reader. This completes the proof for the *"only-if"* direction.

For the *"if"* direction, suppose that the instance $< V', \{P_1, P_2, ..P_N\}, K', Q >$ has a power assignment with total power $\leq Q$ that yields a connected planar geometric graph $G'(V', E')$ with a d-hop DS D' of size $\leq K'$. Without lost of generality, we may assume that (1) D' is minimal, i.e., D' is not a DS if any node is removed from D', and (2) The power at each node is minimum, i.e., every node uses the least possible power to generate a connected graph. We can construct a vertex cover D with $|D| \leq K$ for $G(V, E)$ based on the following observations:

1. If x is not an original node, there is a unique power level assigned to x to make the resulting graph connected, independent of whether x is in D' or not.
2. For every edge (u, v) in G the number of non-original nodes required to be in D is at least $l_{uv} - 1$ even when one or both of u and v belong to D. From the definition of K, it follows that the number of original nodes in D is $\leq K$. Moreover, the total power usage does not exceed Q even if every original node in D is assigned the power level r_3.
3. Consider an edge (u, v) in G where both of the original nodes u, v do not belong to D. Such an edge (u, v) must have at least l_{uv} non-original nodes in D'. If any two such edges are adjacent and have a common original node, we can add this common *original* node to D', assign to it the power level r_3, and remove one non-original node on each edge from D'. This yields a smaller DS whose size is of course $\leq K'$ and the total power usage is still bounded by Q as pointed out in Observation 2.
4. The *auxiliary* nodes cannot be in D'; otherwise the size of D can be reduced.
5. For each edge $(u, v) \in G'$ with $u \notin D'$ and $v \notin D'$, remove an *intermediate* or *control* node which is closest to u or v, and add u or v to D' with the power level r_3. The size of D' does not change and the total power usage is still $\leq Q$.

From the above observations, we may assume that every edge (u, v) in G must have at least an *original* node and $(l_{uv} - 1)$ non-original nodes in D'. The vertex cover set D is constructed from D as follows: If an *original* node $u \in V'$ belongs to D', add $u \in V$ to D.

Clearly, we only add to D the *original* nodes in G' representing nodes in D'. For each set of line segments in G' representing an edge in G, we remove at least $(l_{uv} - 1)$ nodes from D'. The total number of *original* nodes from D' included in D is at most:

$$|D| \leq |D'| - \sum_{(u,v)\in E} (l_{uv} - 1) \leq K' - \sum_{(u,v)\in E} (l_{uv} - 1) = K$$

From the construction of D, it is clear that every edge in G is covered by a node in D. Thus, D is a vertex cover in G. This completes the proof of Theorem 1 for the case $d \geq 2$.

For $d = 1$, we define the radii r_1 and r_2 by $r_1 := \sigma + 0.001$ and $r_2 := \sigma * 2$, where $\sigma := \delta/3$, and construct the instance of MTP-MDDS as follows:

1. Keep the *original* nodes at the same locations in the plane.
2. On the edge (u, v) of length l_{uv} connecting two *original* nodes u, v, we place a total of $3 * l_{uv} - 1$ consecutive nodes such that there is an equal distance of $\delta/3$ from one node to the next. These nodes are called *intermediate* nodes. The two *intermediate* nodes that are adjacent with two *original* nodes will have the exact distance of r_1 to the *original* and to the next *intermediate* node. This can be accomplished by placing these two *intermediate* nodes slightly away from the line segments. These nodes are called *control* nodes, and their adjacent intermediate nodes are called *interfacing* nodes.
3. We attach to each *original* node two more nodes. The first node is placed at distance r_1 from the *original* node. This node is called the *support* node. The second node, called the *auxiliary* node, is placed at distance r_2 from the *support* node.

The total number of *intermediate* (I), *auxiliary* (A) and *support* nodes (S) can be computed as follows:

$$A = S = |V|, \qquad I = \sum_{(u,v)\in E} (3 * l_{uv} - 1)$$

To complete the construction of the instance of MTP-MDDS for $d = 1$, we define the set of power levels for each node, and the numbers Q and K'. Each *original*, *support* and *control* node has the set of power levels $\{0, r_1, r_2\}$. The interfacing nodes are assigned the power levels $\{0, r_2\}$, whereas all other *intermediate* nodes including the *control* nodes are assigned the power levels $\{0, r_1\}$. Furthermore,

$$Q = (K + 2 * |E| + 2 * |V|) * r_2 + (I - 2 * |E|) * r_1, \quad K' = K + |V| + \sum_{(u,v)\in E} (l_{uv} - 1)$$

An example of an instance of MTP-MDDS for $d = 1$ can be found in Fig. 1 (right). Nodes A, B, C, D are *original* nodes, and the i' and o' nodes are *support* and *auxiliary* nodes, respectively.

Observe that each edge (u, v) represented by a set of line segments in G' must have at least $(l_{uv} - 1)$ non-original nodes in D'. Moreover, to dominate each pair of *auxiliary* and *support* nodes, at least one of them has to be in D. Hence, $|D'| = |D| + |V| + \sum_{(u,v)\in E}(l_{uv} - 1)$. The correctness proof for the case $d = 1$ is similar to the case $d \geq 2$. This concludes the proof of Theorem 1.

4 Heuristics

In this section, we describe four heuristics for the minimum total power minimum DS problem along with some supporting algorithms to find graph diameter and compute dominating sets. To compute dominating sets, we use the Progressive Maximum Degree D-Hop Dominating Set (Minimum Dominating Set) algorithm introduced in [8]. The heuristics presented are: Shortening Diameter, Shrinking Dominating Set, Shortening All Paths, and Power Level Search. The Finding Diameter algorithm is a supporting algorithm used by the Shortening Diameter heuristic to find the diameter of a graph.

In the Shortening Diameter heuristic we successively shorten the longest of all shortest paths by increasing the power levels of the nodes along that path, whereas Shortening All Paths increases the power level of all nodes during each

```
MINIMUM DOMINATING SET (G(V,E))
 Initialize all nodes as uncovered nodes;.
 Set Dominating Set S to be empty;
Do {
        Pick the uncovered node x with the largest number of uncovered d-hop neighbors;
        Set node x to be covered and add node x to the dominating set S;
        Set all uncovered d-hop neighbors of node x to be covered;  }
 Until (all nodes are covered);
 Return Dominating Set S;
```

Fig. 2. Minimum Dominating Set Algorithm

```
SHORTENING ALL PATHS
Input:   Distances of every pair of nodes in the plane and the hop number D

Call Minimum Spanning Graph algorithm;
Do {
        Call the  Minimum Dominating Set algorithm;
        Calculate the total number of nodes in the dominating set S;
        If (|S| > 1)
                For (Every node x in G) {
                        Find all 2-hop neighbors of x;
                        Increase the power of x to reach its farthest 2-hop away neighbor;  }
        For (Every node x in G)
                Reduce the power level of x to reach its farthest 1-hop neighbor;
        Calculate the total power usage;    }
Until (|S| == 1)
```

Fig. 3. Shortening All Paths Algorithm

```
FINDING DIAMETER
Input:    A graph G(V,E)
Output   The longest of all shortest paths.
    largest_level = 0;
    For every node x in G(V,E) {
        Initialize order level of every node 0 and node x with level 1;
        d = 1;
        While (there is still a node with order level = 0) {
            Find all d hops away neighbors of x;
            For (Each neighbor y of x)
                If (y has the order level = 0)
                    Set order level of y = d;
            d++;  }
        If (d > largest_level) {
            Set the largest_level = d;
            Store nodes on the path from x to a neighbor that has the order level d;} }
    Output the longest path;
```

Fig. 4. Finding Diameter Algorithm

```
POWER LEVEL SEARCH
Input:    Distances of every pair of nodes in the plane and hop number D

    Innitialize  upper_level = the farthest distance of any pair of nodes;
    Innitialize curr_level = 0 and lower_level = 0;
    While (upper_level > lower_level) {
        Using binary search to find the next good power level which makes the graph connected;
        Initialize Level of every node to be 0 and initialize Mutihome of every node to be 0;
        If (upper_level > lower_level) {
            Call Minimum Dominating Set algorithm and set Level of node in dominating set S to be 0;
            For (every node x that is in G but not in S) {
                Set Level of x to be the number of hops from x to the closest node in S;
                Set Multihome of x to be the total number of node in S that are d hops away from x;}
            For (int  u = 1;  u <= D;   u++)
                For (Every node x in G) {
                    If ( Multihome of x == 1   &&   x is not in S)
                        Set the power level of x to reach the closest 1-hop neighbor y that has
                          the Level lower that x;
                    Else If (Multihome of x  > 1   &&    x is not in S    &&   Level of x == 1)
                        Set the power level of x to reach the farthest 1-hop neighbor y that has
                          the Level value of 0 or D-1;
                    Else If (Multihome of x > 1   &&     x is not in S    &&   Level of x > 1)
                        Set the power level of x to reach the farthest 1-hop neighbor y that has
                          the Level lower than x; }
            For (Every node in G)
                Reduce the power level of x to reach the farthest 1 hop neighbor;
            Calculate the total power usage and the total number of nodes in the dominating set S; }}
```

Fig. 5. Power Level Search Algorithm

iteration. In the Shrinking Dominating Set heuristic we increase the power of the nodes of the current DS and obtain a new DS of smaller size. The Power Level Search algorithm simply searches for all power levels that yield DSs of different sizes. All four heuristics and supporting algorithms are described in Figures 2, 3, 4, 5, 6 and 7.

```
SHORTENING DIAMETER
Input:  Distances of every pair of nodes in the plane and the hop number D

Call Minimum Spanning Graph algorithm;
Do {
        Call the  Minimum D ominating Set algorithm;
        Calculate the total number of nodes in the dominating set S;.
        If (|S| > 1) {
                Call the Finding Diameter algorithm;
                For (Every node x of the longest path) {
                        Increase the power of x to reach the farthest 2 hops away neighbor y which
                        is also a member of the longest path; } }.
        For (Every node x in G)
                Reduce the power level of x to reach its farthest 1 hop neighbor;
        Calculate the total power usage; }
Until (|S| == 1);
```

Fig. 6. Shortening Diameter Algorithm

```
SHRINKING DOMINATING SET
Input:  Distances of every pair of nodes in the plane and hop number D

Initialize total_power = 0 and prev_total_power = -1;
Initialize |S| = 0 and prev_ds = -1;
Call Minimum Spanning Graph algorithm ;
While (total_power != prev_total_power   ||   |S| != prev_ds) {
        Call the  Minimum Dominating Set algorithm;
        Calculate the total number of nodes in the dominating set S;
        If (|S| !=  prev_ds   ||   total_power != prev_total_power)
                For (Every node x in S)
                        Increase the power level of x to reach the closest node y which is also a
                         member of S;
        Calculate the total power usage and set it to total_power;
        Set prev_total_power = total_power;
        Set prev_ds = |S|;
        For (Every node x in G)
                Reduce the power of x to reach its farthest 1 hop neighbor;  }
```

Fig. 7. Shrinking Dominating Set Algorithm

5 Experimental Results

To perform our experiment we randomly generate 5 different sets of 300 nodes on an area of size 30x30 units. For each set of nodes generated each heuristic is run several rounds, and each round provides a total power usage and a new minimal dominating set. We calculate the total power usage and the dominating set size for each round. The results for 2-hop DSs are presented in Figures 8 and 9.

In terms of the DS size, the Shortening Diameter heuristic gives the greatest number of choices for DS size and total power usage. The Shortening Diameter heuristic increases the power of the nodes along the diameter of the current graph to connect nodes on this diameter that are two hops apart from each other. As the total power usage increases, the DS size gradually decreases until it is equal to 1. On the other hand, the Shrinking Dominating Set heuristic

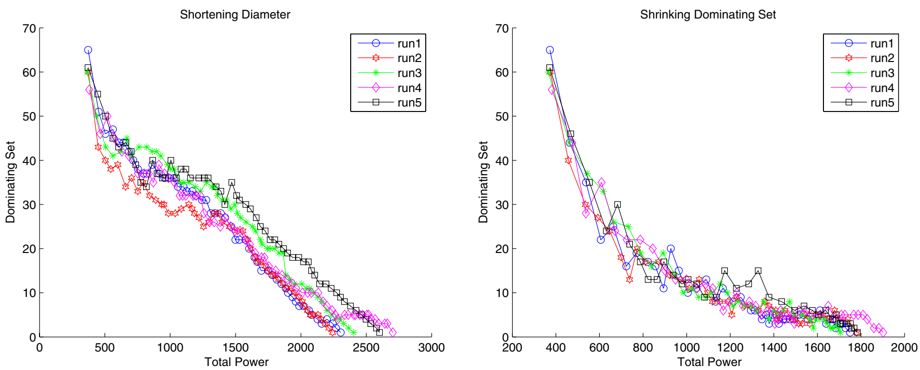

Fig. 8. Power Level w.r.t DS for Shortening Diameter (left) and Shrinking Dominating Set (right)

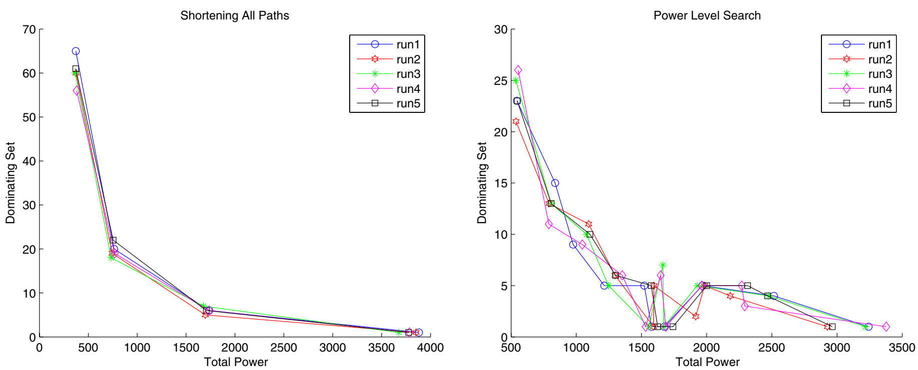

Fig. 9. Power Level w.r.t DS for Shortening All Paths (left) and Power Level Search (right)

concentrates only on the current DS. This heuristic increases the power of nodes in the previous DS to reduce its size. The whole process stops and exits when increasing the power level of previous DS nodes does not change the size or the nodes in the DS.

The Shrinking Dominating Set heuristic provides a few more choices of DS size and total power usage than the Shortening All Paths and Power Level Search algorithms. However, this is still far less than the number of choices provided by the Shortening Diameter heuristic. The Shortening All Paths heuristic increases the power of all nodes to connect every pair of 2 hop neighbors. This reduces the size of the DS rapidly and hence gives the fewest number of choices for the DS size.

The Power Level Search heuristic uses binary search to find a new power level every round. The power level is either decreased or increased each time until a connected graph having a new DS of different size is found. Therefore the Power Level Search gives fewer choices for DS size and total power than the Shrinking Dominating Set heuristic.

The Shortening Diameter heuristic provides significantly more choices for DS size and total power usage than all the others. This is apparently due to the fact that the other heuristics increase the power levels of the nodes in a global fashion whereas Shortening Diameter focuses on nodes along the longest of the shortest paths only. When having more choices is an important factor, it appears that Shortening Diameter is the better choice among the four heuristics.

6 Conclusions

In this paper we show that the Minimum Total Power Minimum D-Hop Dominating Set problem is NP-complete for planar geometric graphs. We also introduce four heuristics and study the trade-offs between DS size and total power usage. From the simulation results for all four heuristics, the Shortening Diameter approach provides significantly more options to find different dominating sets of varying size where total power usage is comparable to the other approaches. The simulation result for Shortening Diameter also provides us an interesting trade-off between total power usage and DS size.

References

1. Cardei, M., MacCallum, D., Cheng, M.X., Min, M., Jia, X., Li, D., Du, D.-Z.: Wireless Sensor Networks with Energy Efficient Organization. Journal of Interconnection Networks 3, 213–229 (2002)
2. Ma, C., Yang, Y., Zhang, Z.: Constructing Battery-Aware Virtual Backbones in Sensor Networks. In: Proceedings of the 2005 International Conference on Parallel Processing, June 2005, pp. 203–210 (2005)
3. Thai, M.T., Du, D.-Z.: Connected Dominating Sets in Disk Graph with Bidirectional Links. IEEE Communication Letters 10(3), 138–140 (2006)
4. 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 6(7) (July 2007)
5. Tang, J., Hao, B., Sen, A.: Relay Node Placement in Large Scale Wireless Sensor Networks. Computer Communications 29, 490–501 (2006)
6. Carle, J., Gallais, A., Simplot-Ryl, D.: Preserving area coverage in wireless sensor networks by using surface coverage relay dominating sets. In: Proceedings. 10th IEEE Symposium on Computers and Communications Issue 27-30, June 2005, pp. 347–352 (2005)
7. Wu, J., Dai, F.: Virtual Backbone Construction in MANETs Using Adjustable Transmission Ranges. IEEE Trans. on Mobile Computing 5(9), 1188–1200 (2006)
8. Vuong, T., Huynh, D.: Connected D-Hops Dominating Sets in Wireless Ad Hoc Networks. SCI 20002/ISAS 2002, Vol IV
9. Valiant, L.: Universality Considerations in VLSI Circuits. IEEE Transactions on Computers C-30, 135–140 (1981)
10. Garey, M.S., Johnson, D.S.: The rectilinear Steiner tree problem is NP-complete. SIAM Journal on Applied Mathematics 32, 826–834