Solving the Connected Dominating Set Problem and Power Dominating Set Problem by Integer Programming

Neng Fan and Jean-Paul Watson

Discrete Math and Complex Systems Department
Sandia National Laboratories, Albuquerque, NM 87185, USA
{nnfan,jwatson}@sandia.gov

Abstract. In this paper, we propose several integer programming approaches with a polynomial number of constraints to formulate and solve the minimum connected dominating set problem. Further, we consider both the power dominating set problem – a special dominating set problem for sensor placement in power systems – and its connected version. We propose formulations and algorithms to solve these integer programs, and report results for several power system graphs.

Keywords: Connected Dominating Set Problem, Power Dominating Set Problem, Integer Programming, Spanning Tree, Connected Subgraph.

1 Introduction

The *minimum dominating set* (MDS) problem is stated as follows: For a graph $G = (V,E)$, a dominating set is a subset D of V such that every vertex not in D is linked to at least one member of D by some edge. The minimum dominating set problem is to find a dominating set with smallest cardinality. The decision version of the MDS is a classical NP-complete problem [1].

The dominating set D of a graph G is a *connected dominating set* if each vertex in D can reach any other vertex in D by a path that traverses only vertices within D. That is, D induces a connected subgraph of G. The *minimum connected dominating set* (MCDS) problem is to find a connecting dominating set with the smallest possible cardinality among all connected dominating sets of G. The concept of a connected dominating set is quite useful in the analysis of wireless networks, social networks, and sensor networks, as studied extensively by Du's group in [2–6]. The MCDS problem was recently studied in disk graphs [7] and unit ball graphs [8]. For an extensive discussion of heuristic algorithms for and applications of the MCDS problem, we refer to [9, 10].

Integer programming (IP) approaches for the MCDS problem have attracted less attention than heuristic methods. In [11, 12], although IP formulations were presented, the algorithms were still based on heuristic and simulation methods. In [13, 14], mixed integer programming (MIP) approaches were used to formulate the MCDS, while [15] introduced a MIP approach with exponential number $O(2^{|V|})$ of constraints based on spanning trees to exactly solve this problem.

In this paper, building on the IP formulation for the MDS problem, we add different kinds of constraints to ensure the connectivity of the subgraph induced by D.

G. Lin (Ed.): COCOA 2012, LNCS 7402, pp. 371–383, 2012.

Considering the fact that a graph is connected if and only if it has a spanning tree, constraints implementing sub-tour elimination, cutset, and other concepts, as reviewed in [16] for the minimum spanning tree problem, can be leveraged for IP formulations of the MCDS problem. However, because of the exponential number of constraints, the computational expense is prohibitive for large graphs. Therefore, we use a polynomial number of constraints to ensure connectivity, leveraging Miller-Tucker-Zemlin constraints, Martin constraints, and commodity flow constraints.

The *power dominating set* (PDS) problem was originally proposed for solving a sensor placement problem in power system graphs, usually referred to as the PMU placement problem [17, 18]. A *power system graph* is an undirected graph $G = (V,E)$, where the vertex set V represents a set of buses, and the edge set E represents a set of transmission lines. Additionally, there is a subset V_Z of V, which represents the set of zero-injection buses that consist of transhipment buses in the system. A power dominating set D is obtained by considering the following two physical laws: (i) if $v \in D$, then v and its neighbors (denoted by $N(v)$) are all covered (Ohm's law); (ii) if $v \in V_Z$, and all vertices within the set $\{v\} \cup N(v)$ except one are covered, then the uncovered vertex in $\{v\} \cup N(v)$ is also covered (Kirchhoff's current law). The PDS problem is to find a subset of vertices D with smallest cardinality that covers all vertices in V. This problem has been widely studied in the power systems literature, as shown in [18], and recently in the area of general combinatorial optimization [19]. The PDS problem can be extended to consider connected vertex sets, yielding the *connected power dominating set* (CPDS) problem.

The reminder of this paper is organized as follows. In Section 2, we introduce IP formulations for the MDS and MCDS problems. In Section 3, we introduce four types of connectivity constraints to ensure the connectivity of the subgraph induced by the dominating set. In Section 4, we introduce the power dominating set problem, connected power dominating set problem, and their associated IP formulations. In Section 5, we test and compare our formulations and algorithms on several power system graphs. Finally, we conclude in Section 6 with a summary of our results.

2 Dominating Set Problem

In a graph $G = (V,E)$ with $V = \{1,2,\cdots,n\}$, let $A = (a_{ij})_{n\times n}$ be the neighborhood matrix such that $a_{ij} = a_{ji} = 1$ if $(i,j) \in E$ or $i = j$, and $a_{ij} = a_{ji} = 0$ otherwise. Without loss of generality, we define the edge set E as follows: $E = \{(i,j) : a_{ij} = 1, \forall i,j \in V \text{ with } i < j\}$.

For $i \in V$, let $x_i \in \{0,1\}$ be a decision variable such that $x_i = 1$ if vertex i is included in the dominating set; $x_i = 0$ otherwise. An IP formulation of the MDS problem can then be given as follows:

$$\textbf{[MDS]} \qquad \min \sum_i x_i \tag{1a}$$

$$s.t. \ \sum_j a_{ij}x_j \geq 1, x_i \in \{0,1\}, \forall i \in V \tag{1b}$$

Any feasible solution to formulation (1) will form a dominating set D of G by $D = \{v_i : x_i = 1\}$. Let $G_D = (D, E_D)$ be the subgraph induced by the dominating set D, where $E_D = \{(i, j) \in E : a_{ij}x_ix_j = 1, \forall i, j \in V \text{ with } i < j\}$.

To yield an IP formulation for the related MCDS problem, we must additionally include connectivity constraints to ensure that the subgraph G_D is connected. In next section, we study four MIP approaches to model the connectivity constraints in the MCDS for subgraphs G_D.

3 Connectivity Constraints of Subgraphs

Definitionally, a graph G_D is connected if and only if it has a spanning tree. Therefore, some methods for solving the minimum spanning tree problem can be leveraged to formulate efficient MIPs for the MCDS problem.

3.1 Miller-Tucker-Zemlin Constraints

Miller-Tucker-Zemlin constraints were originally proposed for solving the traveling salesman problem in [20], and were used to eliminate sub-tours when solving the k-cardinality tree problem in [21].

Following the method proposed in [21], we let $G_d = (V \cup \{n+1, n+2\}, A)$ be a directed graph based on $G = (V, E)$, where $A = \{(n+1, n+2)\} \cup \left\{\bigcup_{i=1}^{n} \{(n+1, i), (n+2, i)\}\right\} \cup E \cup E'$ and $E' = \{(j, i) : a_{ji} = 1, \forall i, j \in V \text{ with } i > j\}$. That is, we introduce two additional vertices $n+1$ and $n+2$, add directed edges $n+1$ and $n+2$ to every $i \in V$ and $(n+1, n+2)$, and make each edge $(i, j) \in E$ bi-directional.

The idea behind Miller-Tucker-Zemlin constraints is to find a directed spanning tree $T_d = (V \cup \{n+1, n+2\}, E_d)$ of G_d such that $n+1$ is the root connecting to both $n+2$ and those vertices not in the dominating set D, $n+2$ is connected to a vertex v_r within D, and all other vertices are formed a tree with root v_r. As shown in Fig. 1, the directed spanning tree has a connected subgraph (shown within the dashed circle) whose vertices form the connected dominating set.

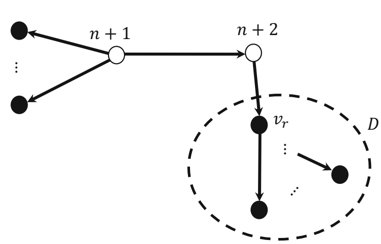

Fig. 1. The idea behind Miller-Tucker-Zemlin constraints

For $(i, j) \in A$, let $y_{ij} \in \{0, 1\}$ be a decision variable such that $y_{ij} = 1$ if (i, j) is selected into the directed tree T_d and $y_{ij} = 0$ otherwise. Additionally, for $i \in V \cup \{n+1, n+2\}$, let u_i be a non-negative decision variable, as introduced in [20] to eliminate sub-tours. The Miller-Tucker-Zemlin (MTZ) constraints to ensure connectivity are formulated as follows:

[MTZ]

$$\sum_{i \in V} y_{n+2,i} = 1 \tag{2a}$$

$$\sum_{i:(i,j) \in A} y_{ij} = 1, \forall j \in V \tag{2b}$$

$$y_{n+1,i} + y_{i,j} \leq 1, \forall (i,j) \in E \cup E' \tag{2c}$$

$$(n+1)y_{ij} + u_i - u_j + (n-1)y_{ji} \leq n, \forall (i,j) \in E \cup E' \tag{2d}$$

$$(n+1)y_{ij} + u_i - u_j \leq n, \forall (i,j) \in A \setminus (E \cup E') \tag{2e}$$

$$y_{n+1,n+2} = 1 \tag{2f}$$

$$u_{n+1} = 0 \tag{2g}$$

$$1 \leq u_i \leq n+1, i \in V \cup \{n+2\} \tag{2h}$$

$$x_i = 1 - y_{n+1,i}, \forall i \in V \tag{2i}$$

In the formulation MTZ, Constraint (2a) identifies one vertex as root of the dominating set D. Constraints (2b) ensure that all vertices within V are connected to some other vertex. Constraints (2c) require that in any feasible solution either $i \in V$ is directly connected to $n+1$ or else it may be connected to other vertices in D. Without the term $(n-1)y_{ji}$, Constraints (2d) and (2e) are the original Miller-Tucker-Zemlin constraints [20] to guarantee the solutions have no sub-tours. The added term was proposed in [22] as an improvement for sub-tour elimination constraints. Constraint (2f) requires that the edge $(n+1, n+2)$ is in T_d. Constraints (2g) and (2h) present the choice of arbitrary non-negative integers for variables u_i. Finally, Constraints (2i) ensure that vertex i is either connected to $n+1$ or a vertex in the dominating set.

The constraints and variables in formulation MTZ represent a portion of a mixed-linear program. By solving MDS in conjunction with MTZ, any feasible solution x will imply a dominating set D, and form a directed spanning tree T_d of G_d. The induced subtree of T_d by D has a root, which is connected to $n+2$. Therefore, the connectivity of the subgraph G_D by D can be guaranteed. In MTZ, there are $(|V|+2) + (2|E| + 2|V|+1) = O(|E|+|V|)$ decision variables and $1+|V|+2|E|+2|E|+(2|V|+1)+1+1+|V| = O(|E|+|V|)$ constraints.

3.2 Martin Constraints

In [23], Martin presented a reformulation for solving the minimum spanning tree problem with a polynomial number of constraints instead of an exponential number of constraints. This method was also used in [24] by Yannakakis, and was recently referenced in [25, 26]. The objective is still to find a (undirected) spanning tree $T_D = (D, E_T)$ of $G_D = (D, E_D)$.

For $i, j \in V$, let $y_{ij} \in \{0,1\}$ be a decision variable such that $y_{ij} = 1$ if edge (i, j) is selected into the tree T_D and $y_{ij} = 0$ otherwise. For $i, j, k \in V$, let $z_{ij}^k \in \{0,1\}$ be a decision variable such that $z_{ij}^k = 1$ if edge (i,j) is in the tree T_D of G_D and vertex k is on side of j (i.e., vertex k is within the resulted component containing j after removal edge (i,j) from T_D), and $z_{ij}^k = 0$ if edge (i,j) is in the tree and k is not on side of j, or if edge (i,j) is not in the tree or the pair (i,j) is not an edge.

The Martin constraints to ensure connectivity of G_D are formulated as follows:

[MARTIN]

$$\sum_{(i,j)\in E} y_{ij} = \sum_{i\in V} x_i - 1 \tag{3a}$$

$$y_{ij} \le x_i, y_{ij} \le x_j, \forall (i,j) \in E \tag{3b}$$

$$z_{ij}^k \le y_{ij}, z_{ij}^k \le x_k, \forall (i,j) \in E, k \in V \tag{3c}$$

$$z_{ji}^k \le y_{ij}, z_{ji}^k \le x_k, \forall (i,j) \in E, k \in V \tag{3d}$$

$$y_{ij} - M(3 - x_i - x_j - x_k) \le z_{ij}^k + z_{ji}^k \le y_{ij} + M(3 - x_i - x_j - x_k), \forall i,j,k \in V \tag{3e}$$

$$1 - M(2 - x_i - x_j) \le \sum_{k'\in V\setminus\{i,j\}} z_{ik'}^j + y_{ij} \le 1 + M(2 - x_i - x_j), \forall i,j \in V \tag{3f}$$

$$y_{ij}, z_{ij}^k \in \{0,1\}, \forall (i,j) \in E, k \in V,\ y_{ij} = 0, z_{ij}^k = 0, \forall i,j,k \in V, (i,j) \notin E \tag{3g}$$

Constraint (3a) ensures that the number of edges within the tree T_D is one less than the number of vertices within T_D, and Constraint (3b) ensures that the selection of edges within E_D relies on the selection of its two ends.

If any one, two, or three vertices of $i,j,k \in V$ are not part of the tree of G_D (i.e., one, two, or three of x_i, x_j, x_k become 0), $z_{ij}^k = z_{ji}^k = 0$ by Constraints (3c)-(3d), and the Constraints (3e) become non-binding constraints and have no influence on the results as M is a large positive constant. Similarly, if any one or two of vertices $i,j \in V$ are not part of G_D, Constraint (3f) become non-binding.

If vertices $i,j,k \in V$ are within the proposed tree of G_D (i.e., $i,j,k \in D$ and $x_i = x_j = x_k = 1$), $z_{ij}^k, z_{ji}^k \in \{0,1\}$ by Constraints (3c)-(3d), and Constraints (3e)-(3f) become

$$z_{ij}^k + z_{ji}^k = y_{ij}, \sum_{k\in D\setminus\{i,j\}} z_{ik}^j + y_{ij} = 1, \forall i,j,k \in D.$$

This represents the original formulation of Martin's constraints, as discussed in [26].

The constraint $z_{ij}^k + z_{ji}^k = y_{ij}$ implies that (i) if $(i,j) \in E_T$ (i.e., $y_{ij} = 1$), vertex k is either on the side of j ($z_{ij}^k = 1$) or on the side of i ($z_{ji}^k = 1$); (ii) if $(i,j) \notin E_T$ (i.e., $y_{ij} = 0$), k is between i,j ($z_{ij}^k = 0, z_{ji}^k = 0$).

The constraint $\sum_{k\in D\setminus\{i,j\}} z_{ik}^j + y_{ij} = 1$ means that (i) if $(i,j) \in E_T$ (i.e., $y_{ij} = 1$), edges (i,k) who connect i are on the side of i ($z_{ik}^j = z_{ij}^k = 0$ and $z_{ij}^k = 1$); (ii) if $(i,j) \notin E_T$ (i.e., $y_{ij} = 0$), there must be an edge (i,k) such that j is on the side k ($z_{ik}^j = 1$ for some k).

The constraints and variables in formulation MARTIN represent a portion of a mixed-linear program. The number of new decision variables is $|V|^2 + |V|^3 = O(|V|^3)$, while the number of constraints to ensure connectivity is $1 + 2|E| + 4|E||V| + 2|V|^3 + 2|V|^2 = O(|V|^3)$.

3.3 Single-Commodity Flow Constraints

For $i \in V$, let $r_i \in \{0,1\}$ be a decision variable such that $r_i = 1$ if vertex i is chosen to be the root v_r of G_D for "sending" $\sum_{i \in V} x_i - 1$ unit flow to other vertices within the dominating D, and $r_i = 0$ otherwise. If each vertex in D except v_r consumes exactly one unit, and the vertices outside D consume none, the connectivity of G_D is guaranteed. This method was used to ensure subgraph connectivity in [27] for solving problems in wildlife conservation.

For each edge $(i,j) \in E \cup E'$ (see Section 3.1 for the definition of E'), let f_{ij} denote the amount of flow from vertex i to vertex j. The constraints enforcing single-commodity flow (SCF) can then be formulated as follows:

[SCF]

$$\sum_{i \in V} r_i = 1 \tag{4a}$$

$$r_i \le x_i, \forall i \in V \tag{4b}$$

$$f_{ij} \ge 0, \forall (i,j) \in E \cup E' \tag{4c}$$

$$f_{ij} \le x_i \sum_{k \in V} x_k,\ f_{ij} \le x_j \sum_{k \in V} x_k, \forall (i,j) \in E \cup E' \tag{4d}$$

$$\sum_j f_{ji} \le n(1 - r_i), \forall i \in V \tag{4e}$$

$$\sum_j f_{ji} - \sum_j f_{ij} = x_i - r_i \sum_{j \in V} x_j, \forall i \in V \tag{4f}$$

$$r_i \in \{0,1\}, \forall i \in V \tag{4g}$$

Constraints (4a) and (4b) select one vertex from the dominating set as the root to transmit the single-commodity flow. Constraints (4c) ensure the non-negativity of the flow, while Constraints (4d) require that the flow of edge (i,j) is 0 if either end of (i,j) is not selected into the dominating set. Constraints (4e) ensure that the inflow of the selected root is 0. Finally, Constraints (4f) ensure the balance of flows on each vertex. If vertex i is the selected root (i.e., $r_i = 1, x_i = 1$), the outflow of i is equal to $\sum_{j \in V} x_j - 1$, i.e., one unit is transmitted to each selected vertex. If vertex i is in the dominating set D but is not the root (i.e., $x_i = 1, r_i = 0$), the difference between the inflow and outflow will equal one, implying that vertex i consumes one unit; otherwise, vertex i is not in D (i.e., $x_i = 0, r_i = 0$), and all inflows and outflows will be 0.

Any feasible solution to the MDS problem with SCF constraints will guarantee that every vertex within the dominating set D except the selected root will consume one unit of flow transmitted from the root, and the connectivity of the subgraph induced by D will be ensured.

The quadratic terms $r_i x_j$ can be easily linearized by introducing $w_{ij} = r_i x_j$ with constraints $w_{ij} \le r_i$, $w_{ij} \le x_j$, $w_{ij} \ge r_i + x_j - 1$, and $w_{ij} \ge 0$. Similarly, the quadratic terms $x_i x_k$ can be linearized by introducing $w'_{ik} = x_i x_k$.

Additionally, the following constraints can be added such that the first appearance of $x_i = 1$ implies $r_i = 1$:

$$r_i \leq (n+1-\sum_{i'=1}^{i} x_{i'})/n, \forall i \in V. \tag{5}$$

Such constraints can reduce the degeneracy of the choice of root vertex within the dominating set. Without loss of generality, assume that i_a is first vertex with $x_{i_a} = 1$ (i.e., $x_i = 0$ for $i < i_a$) and i_b is the second one with $x_{i_b} = 1$ (i.e., $x_i = 0$ for $i_a < i < i_b$). Therefore, by (5), there are four cases: (i) for $i < i_a$, $r_i \leq (n+1-0)/n = 1+1/n$ and by (4b), $r_i = 0$; (ii) for $i = i_a$, $r_{i_a} \leq 1$; (iii) for $i_a < i < i_b$, $r_i \leq 1$ and from (4b), $r_i = 0$; (iv) for $i \geq i_b$, $r_i \leq (n+1-2)/n = (n-1)/n$ and from (4g), $r_i = 0$. Thus, by (4a), $r_{i_a} = 1$.

There are $|V| + 2|E| = O(|E| + |V|)$ decision variables and $1 + |V| + 2|E| + 4|E| + |V| + |V| = O(|E| + |V|)$ constraints in the MDS problem with SCF constraints.

3.4 Multi-commodity Flow Constraints

In Section 3.3, the connectivity of $G_D = (D, E_D)$ is enforced through a single commodity flow. In the following, the connectivity of a selected subset D is guaranteed by associating a separate commodity with each vertex. Assume that v_r is the selected root within D, such that there will be one unit of flow from v_r to each selected vertices of its own commodity type. This method was used to ensure subgraph connectivity in [27] for solving problems in wildlife conservation.

For each edge $(i, j) \in E \cup E'$ (see Section 3.1) and $k \in V \setminus \{v_r\}$, let f_{ij}^k be a decision variable such that $f_{ij}^k != 0$ if edge (i, j) carries flow of type k, and 0 otherwise. The flow outside of the dominating set should be 0, the flow of type k equals 0 if k is outside D, and the flow of type v_r should be 0, i.e.,

$$f_{ij}^k \leq x_i,\ f_{ij}^k \leq x_j,\ f_{ij}^k \leq x_k,\ f_{ij}^k \leq 1 - r_k,\ f_{ij}^k \geq 0, \forall (i, j) \in E \cup E', \forall k \in V.$$

For the root v_r, there is no inflow, and the outflow of type k is x_k, i.e.,

$$\sum_{j:(j,v_r)\in E\cup E'} f_{jv_r}^k = 0, \quad \sum_{j:(v_r,j)\in E\cup E'} f_{v_r j}^k = x_k, \forall k \in V \setminus \{v_r\}.$$

For vertex $i \in V \setminus \{v_r\}$, the inflow of type i is x_i and the outflow of type i is 0, i.e.,

$$\sum_{j:(j,i)\in E\cup E'} f_{ji}^i = x_i, \quad \sum_{j:(i,j)\in E\cup E'} f_{ij}^i = 0, \forall i \in V \setminus \{v_r\}.$$

For vertex $i \in V \setminus \{v_r\}$, the flow of type $k (k \neq i)$ should be balanced at i, i.e.,

$$\sum_{j:(j,i)\in E\cup E'} f_{ji}^k = \sum_{j:(i,j)\in E\cup E'} f_{ij}^k, \forall i \in V \setminus \{v_r\}, \forall k, k \neq i.$$

For $i \in V$, let $r_i \in \{0, 1\}$ be a decision variable such that $r_i = 1$ if vertex i is chosen to be the root of G_D, and $r_i = 0$ otherwise. The above constraints by multi-commodity flow (MCF) to ensure connectivity of G_D can be equivalently formulated as follows:

[MCF]

$$f_{ij}^k \leq x_i,\ f_{ij}^k \leq x_j,\ f_{ij}^k \leq x_k,\ f_{ij}^k \leq 1 - r_k,\ f_{ij}^k \geq 0, \forall (i,j) \in E \cup E', \forall k \in V \tag{6a}$$

$$\sum_{j:(j,i) \in E \cup E'} f_{ji}^k \leq M(1 - r_i), \forall i, k \in V \tag{6b}$$

$$x_k - r_k - M(1 - r_i) \leq \sum_{j:(i,j) \in E \cup E'} f_{ij}^k \leq x_k - r_k + M(1 - r_i), \forall i, k \in V \tag{6c}$$

$$\sum_{j:(j,i) \in E \cup E'} f_{ji}^i = x_i - r_i, \quad \sum_{j:(i,j) \in E \cup E'} f_{ij}^i = 0, \forall i \in V \tag{6d}$$

$$\sum_{j:(i,j) \in E \cup E'} f_{ij}^k - Mr_i \leq \sum_{j:(j,i) \in E \cup E'} f_{ji}^k \leq \sum_{j:(i,j) \in E \cup E'} f_{ij}^k + Mr_i, \forall i, k \in V, k \neq i \tag{6e}$$

$$\sum_{i \in V} r_i = 1,\ r_i \leq x_i, \forall i \in V \tag{6f}$$

where M is a sufficiently large positive constant.

Any feasible solution to the MDS problem with MCF constraints will guarantee that every vertex i within the dominating set D excluding the selected root will consume one unit of type i flow transmitted from the root, and the connectivity of the subgraph induced by D will be ensured. There are $|V| + 2|E||V| = O(|E||V|)$ decision variables and $8|E||V| + 2|V|^2 + |V| + |V|^2 + 1 + |V| = O(|E||V|)$ constraints in formulation (6). Similarly, Constraints (5) can be added to reduce the degeneracy of the selection for the root.

4 Power Dominating Set Problem and Connected Power Dominating Set Problem

For a power graph $G = (V, E)$, there is a given subset $V_Z \subset V$ of zero-injection vertices. As explained in [19], a power dominating set D is obtained by leveraging two physical laws: (1) if $v \in D$, then v and its neighbors (denoted by $N(v)$) are all covered (Ohm's law); (2) if $v \in V_Z$, and all vertices within the set $\{v\} \cup N(v)$ except one are covered, then the uncovered vertex in $\{v\} \cup N(v)$ is also covered (Kirchhoff's current law). The *power dominating set* (PDS) problem is to find a subset D with smallest cardinality that covers all vertices in V. The first law applies similarly as that for the dominating set problem (i.e., a selected vertex covers all neighbors of itself), while the second law can significantly reduce the dominating number for a given graph. Let the set of zero-injection vertices be denoted by $V_Z = \{v_i \in V : Z_i = 1\}$, where the parameter $Z_i = 1$ indicates that v_i is a zero-injection vertex; $Z_i = 0$ otherwise.

For $i \in V$, let $x_i \in \{0, 1\}$ be a decision variable such that $x_i = 1$ if vertex i is selected into the power dominating set and $x_i = 0$ otherwise. For $i, j \in V$, let $p_{ij} \in \{0, 1\}$ be a decision variable such that $p_{ij} = 1$ if Kirchhoff's current law applied to zero-injection vertex i can provide a coverage for vertex j and $p_{ij} = 0$ otherwise. Following the method in [18], the PDS problem to find a smallest dominating subset can be formulated as follows:

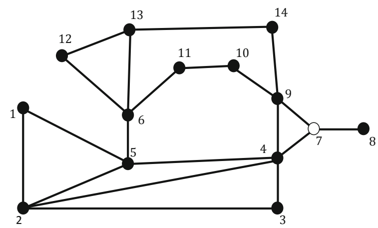

Fig. 2. An example graph with 14 vertices

$$\textbf{[PDS]} \quad \min_{x,p} \sum_i x_i \tag{7a}$$

$$s.t. \sum_j a_{ij} x_j + \sum_j a_{ij} Z_j p_{ji} \geq 1, \forall i \in V \tag{7b}$$

$$\sum_j a_{ij} p_{ij} = Z_i, \forall i \in V \tag{7c}$$

$$p_{ij} = 0, \forall i,j \text{ with } a_{ij} = 0 \text{ or } i \notin V_Z \tag{7d}$$

$$x_i, p_{ij} \in \{0,1\}, \forall i,j \in V \tag{7e}$$

The objective (7a) is to minimize the cardinality of the power dominating set. For each vertex $i \in V$, the first part of Constraint (7b) follows Ohm's law while the second part of (7b) follows Kirchhoff's current law with possible coverage from its neighbors. In the PDS problem, all vertices will be covered, and Constraints (7c) denote that every zero-injection vertex i provides coverage for itself or one of its neighbors. Constraint (7d) ensures that p_{ij} equals 0 if the pair (i,j) is not an edge or i is not a zero-injection vertex, and Constrains (7e) ensure the binary choices of the x_i and p_{ij} variables.

The PDS formulation is an integer linear program. Similarly, any feasible solution to the PDS will form a dominating set D of G by $D = \{v_i : x_i = 1\}$. Let $G_D = (D, E_D)$ be the subgraph induced by the power dominating set D, where $E_D = \{(i,j) \in E : a_{ij} x_i x_j = 1, \forall i,j \in V \text{ with } i < j\}$. For the *connected power dominating set* (CPDS) problem, we have to add connectivity constraints to ensure that the subgraph G_D is connected, following the methods introduced in Section 3.

5 Numerical Experiments

All MIP formulations are implemented in C++ and solved using CPLEX 12.1 via IBM's Concert Technology library, version 2.9. All experiments were performed on a Linux workstation with 4 Intel(R) Xeon(TM) CPU 3.60GHz processors and 8 GB RAM. The optimality gap was set to be 1%.

First, we consider an illustrative example using a graph with 14 vertices and 20 edges (as shown in Fig. 2). By solving the MDS problem formulated in (1), a minimum

Table 1. Minimum objective function values for power graphs

Graph				Dominating Set Problems		Connected Dominating Set Problems	
Name	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert V_Z\rvert$	MDS	PDS	MCDS	CPDS
IEEE-14-Bus	14	20	1	4	3	5	4
IEEE-30-Bus	30	41	6	10	7	11	9
IEEE-57-Bus	57	78	15	17	11	31	24
RTS-96	73	108	22	20	14	32	28
IEEE-118-Bus	118	179	10	32	28	43	39
IEEE-300-Bus	300	409	65	87	68	129	112

Note: The column $|V_Z|$ denotes the number of zero-injection vertices.

Table 2. Comparison of formulation sizes

Number of	MDS(1)	PDS(7)	MTZ(2)	MARTIN(3)	SCF(4)	MCF(6)
decision var.	$\lvert V\rvert$	$2\lvert E\rvert+\lvert V\rvert$	$O(\lvert E\rvert+\lvert V\rvert)$	$O(\lvert V\rvert^3)$	$O(\lvert E\rvert+\lvert V\rvert)$	$O(\lvert E\rvert\lvert V\rvert)$
constraints	$\lvert V\rvert$	$2\lvert V\rvert$	$O(\lvert E\rvert+\lvert V\rvert)$	$O(\lvert V\rvert^3)$	$O(\lvert E\rvert+\lvert V\rvert)$	$O(\lvert E\rvert\lvert V\rvert)$

dominating set is $\{2,6,7,9\}$, with cardinality 4. By solving the MCDS problem as formulated in (1) coupled with any one type of connectivity constraints (2), (3), (4), or (6), a minimum connected dominating set is $\{4,5,6,7,9\}$ with cardinality 5.

For the power dominating set of the graph in Fig. 2, assume that the set of zero-injection vertices is $V_Z = \{7\}$. A minimum power dominating set is $\{2,6,9\}$ with cardinality 3, obtained by solving the formulation (7). In this dominating set, vertices $2,1,3,4,5$ are covered by vertex 2; vertices $6,5,11,12,13$ are covered by vertex 6; and vertices $9,4,7,10,14$ are covered by vertex 9. By Kirchhoff's current law, vertex 8 is covered because vertices in $\{7\}\cup N(7) = \{7,4,9,8\}$ are all covered with the exception of vertex 8. Similarly, the minimum connected power dominating set $\{4,5,6,9\}$ with cardinality 4 can be computed using formulation (7) with any one of the constraints (2), (3), (4), or (6).

Next, we test our models on the six power graphs considered in [28]. First, we removed all parallel edges in these graphs. The objective values for the minimum dominating set, minimum connected dominating set, minimum power dominating set, and minimum connected power dominating set problems are shown in Table 1, while the wall clock run-times (in seconds) are reported in Table 3. In Table 1, we also present statistical information for the test instances, including the number of vertices, edges, and the number of zero-injection vertices in the case of the power dominating set problem.

From Table 1, we observe that the cardinality of the minimum power dominating set is less than the cardinality of minimum dominating set for a given graph. Application of Kirchhoff's current law to zero-injection vertices can reduce the dominating number of a graph. Additionally, minimum connected dominating sets have larger cardinality than their non-connected counterparts.

In Table 2, we present the number of decision variables and constraints for each formulation. The four types of constraints we used to ensure set connectivity have at most $O(|V|^3)$ decision variables, and at most $O(|V|^3)$ constraints. In contrast to the formulation for sub-tour elimination in [15], which has exponential number $O(2^{|V|})$ number of constraints, our proposed methods should yield more tractable computation for even larger graphs.

Table 3. Solution times for formulations with different connectivity constraints

Graph Name	Dominating Set Problems		Minimum Connected Dominating Set Problems				Minimum Connected Power Dominating Set Problems			
	MDS	PDS	MTZ	MARTIN	SCF	MCF	MTZ	MARTIN	SCF	MCF
IEEE-14-Bus	0	0	0.02	0.14	0.15	0.14	0.04	0.12	0.49	0.15
IEEE-30-Bus	0	0	0.22	2.39	299.39	0.89	0.32	1.72	265.10	1.17
IEEE-57-Bus	0.01	0.01	200.59	14671.70	64641.60	6738.05	60.81	5309.74	12448.10	2579.66
RTS-96	0.01	0.03	445.69	>24h	>24h	47236.40	55266.10	>24h	>24h	53752.20
IEEE-118-Bus	0.01	0.04	699.83	85455.70	>24h	36263.10	50.67	>24h	>24h	78715.40
IEEE-300-Bus	0.01	0.27	5033.97	>24h	>24h	>24h	72437.40	>24h	>24h	>24h

From Table 3, we observe that it is quite fast to compute optimal solutions to the dominating set problem without connectivity constraints. In contrast, the imposition of connectivity constraints significantly impacts computational tractability. The MTZ constraints (2) yield the best performance. Comparing the two methods with the same number $O(|E|+|V|)$ of decision variables, MTZ and SCF, MTZ (with fewer constraints) yields significantly better performance. The other connectivity formulations are quite slow requiring more than 24 hours for solving problems arising in large graphs. For example, there are $|V| = 73$ vertices and $|E| = 108$ edges in the RTS-96 graph, yielding more than 73^3 binary variables in formulation (3).

6 Conclusions

We presented four optimization models to ensure the connectivity of the subgraph induced by the dominating set of a graph. All models are formulated as mixed integer programs with a polynomial number of constraints, and were tested on many representative graphs. Among these models, the one with Miller-Tucker-Zemlin constraints to ensure connectivity has the best performance. We further note that the MIP formulations we examine here can be easily extended to solve the minimum spanning tree problem, the maximum leaf spanning tree problem, the k-cardinality tree problem, and the Steiner tree problem.

Future research directions include using efficient branch-and-cut methods to further reduce the computational complexity, and comparing the results obtained by formulations considering an exponential number of constraints. To improve the efficiency of the methods described in this paper, more valid inequalities should be further studied and high-performance computing methods should be leveraged. For some graphs, for example $1 \times n$ grid graphs, the dominating set problem can in theory be solved in polynomial time and tests should be performed on these cases to verify computational complexity results.

Acknowledgements. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed

Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

1. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman & Co., New York (1979)
2. 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)
3. Li, Y., Thai, M.T., Wang, F., Yi, C.W., Wan, P.J., Du, D.-Z.: On greedy construction of connected dominating sets in wireless networks. Wirel. Commun. Mob. Comp. 5, 927–932 (2005)
4. Zhu, X., Yu, J., Lee, W., Kim, D., Shan, S., Du, D.-Z.: New dominating sets in social networks. J. Global Optim. 48(4), 633–642 (2010)
5. Wu, W., Gao, X., Pardalos, P.M., Du, D.-Z.: Wireless networking, dominating and packing. Optim. Lett. 4(3), 347–358 (2010)
6. Ding, L., Gao, X., Wu, W., Lee, W., Zhu, X., Du, D.-Z.: An exact algorithm for minimum CDS with shortest path constraint in wireless networks. Optim. Lett. 5(2), 297–306 (2011)
7. Thai, M.T., Du, D.-Z.: Connected dominating sets in disk graphs with bidirectional links. IEEE Commun. Lett. 10(3), 138–140 (2006)
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 Trans. Mob. Comp. 9(8), 1108–1118 (2010)
9. Blum, J., Ding, M., Thaeler, A., Cheng, X.: Connected dominating set in sensor networks and MANET. In: Du, D.-Z., Pardalos, P. (eds.) Handbook of Combinatorial Optimization, pp. 329–369 (2004)
10. Liu, Z., Wang, B., Guo, L.: A Survey on connected dominating set construction algorithm for wireless sensor networks. Informa. Technol. J. 9, 1081–1092 (2010)
11. Mnif, K., Rong, B., Kadoch, M.: Virtual backbone based on mcds for topology control in wireless ad hoc networks. In: Proceedings of the 2nd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, Quebec, Canada (2005)
12. Yuan, D.: Energy-efficient broadcasting in wireless ad hoc networks: performance benchmarking and distributed algorithms based on network connectivity characterization. In: Proceedings of MSWiM, Quebec, Canada (2005)
13. Morgan, M.J., Grout, V.: Finding optimal solutions to backbone minimisation problems using mixed integer programming. In: Proceedings of the Seventh International Network Conference (INC 2008), Boston, MA, pp. 53–63 (2008)
14. Wightman, P.M., Fabregasy, A., Labradorz, M.A.: An optimal solution to the MCDS problem for topology construction in wireless sensor networks. In: 2010 IEEE Latin-American Conference on Communications (LATINCOM), Belem, Brazil (2010)
15. Simonetti, L., da Cunha, A.S., Lucena, A.: The Minimum Connected Dominating Set Problem: Formulation, Valid Inequalities and a Branch-and-Cut Algorithm. In: Pahl, J., Reiners, T., Voß, S. (eds.) INOC 2011. LNCS, vol. 6701, pp. 162–169. Springer, Heidelberg (2011)
16. Pop, P.C.: A survey of different integer programming formulations of the generalized minimum spanning tree problem. Carpathian J. Mathematics 25(1), 104–118 (2009)
17. Haynes, T.W., Hedetniemi, S.M., Hedetniemi, S.T., Henning, M.A.: Domination in graphs applied to electric power networks. SIAM J. Disc. Math. 15, 519–529 (2002)

18. Aminifar, F., Khodaei, A., Fotuhi-Firuzabad, M., Shahidehpour, M.: Contingency-constrained PMU placement in power networks. IEEE Trans. Power Syst. 25, 516–523 (2010)
19. Aazami, A.: Domination in graphs with bounded progagation: algorithms, formulations and hardness results. J. Comb. Optim. 19, 429–456 (2010)
20. Miller, C.E., Tucker, A.W., Zemlin, R.A.: Integer programming formulation of traveling salesman problems. J. Assoc. Comp. Mach. 7, 326–329 (1960)
21. Quintao, F.R., da Cunha, A.S., Mateus, G.R., Lucena, A.: The *k*-cardinality tree problem: reformulations and lagrangian relaxation. Disc. Appl. Math. 158, 1305–1314 (2010)
22. Desrochers, M., Gilbert, L.: Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. Oper. Res. Lett. 10, 27–36 (1991)
23. Martin, R.K.: Using separation algorithms to generate mixed integer model reformulations. Oper. Res. Lett. 10, 119–128 (1991)
24. Yannakakis, M.: Expressing combinatorial optimization problems by linear programs. J. Comp. Syst. Sci. 43(3), 441–466 (1991)
25. Conforti, M., Cornuéjols, G., Zambelli, G.: Extended formulations in combinatorial optimization. 4OR (8), 1–48 (2010)
26. Kaibel, V., Pashkovich, K., Theis, D.O.: Symmetry Matters for the Sizes of Extended Formulations. In: Eisenbrand, F., Shepherd, F.B. (eds.) IPCO 2010. LNCS, vol. 6080, pp. 135–148. Springer, Heidelberg (2010)
27. Dilkina, B., Gomes, C.P.: Solving Connected Subgraph Problems in Wildlife Conservation. In: Lodi, A., Milano, M., Toth, P. (eds.) CPAIOR 2010. LNCS, vol. 6140, pp. 102–116. Springer, Heidelberg (2010)
28. IEEE reliability test data (2012), `http://www.ee.washington.edu/research/pstca/`