DIMACS Technical Report 2004-45
September 2004

A Computational Study of the Broadcast Domination Problem

by

Steven B. Horton[1,2] Cláudio N. Meneses[3,4] Arup Mukherjee[5]
M. Erol Uluçakli[6]

[1]This work is a result of DIMACS Reconnect '04, Lafayette College, attended by all four authors.
[2]Department of Mathematical Sciences, United States Military Academy, West Point, NY `steve.horton@usma.edu`
[3]Supported by the Brazilian Federal Agency for Post-Graduate Education (CAPES) - Grant No. 1797-99-9.
[4]Dept. of Industrial and Systems Engineering, University of Florida, Gainesville, FL `claudio@ufl.edu`
[5]Department of Mathematical Sciences, Montclair State University, Montclair, NJ `mukherjeea@mail.montclair.edu`
[6]Department of Mechanical Engineering, Lafayette College, Easton, PA `ulucakli@lafayette.edu`

ABSTRACT

The Broadcast Domination problem seeks a function $f : V \to \{0, 1, 2, \dots\}$ that minimizes $\sum_{v \in V} f(v)$ with the constraint that for every v with $f(v) = 0$, there is a vertex $u \in V$ with $f(u)$ at least the distance from u to v in G. This is a generalization of the standard Dominating Set problem. This note investigates the problem from a computational standpoint.

1 Introduction and motivation

Domination in graphs is a well known and thoroughly studied subject [10, 11]. A *dominating set* S is a subset of the vertices in a graph such that every vertex in the graph either belongs to S or has a neighbor in S. The associated decision problem, DOMINATING SET, occupies a prominent place in the computational complexity literature [9]. This definition also leads naturally to the associated optimization problem which is to find a dominating set of minimum cardinality. Numerous variants of this problem have been studied [1, 10, 11, 13].

In [6], Erwin introduced another variant known as *broadcast domination*. In this version, the vertices in S (known as *dominators*) each have an associated nonnegative integer that can be thought of as *broadcast range*. Then vertices not in S are dominated if they are within the broadcast range of some dominator. More formally, we define a *dominating broadcast* as an integer valued function f on the vertices such that every vertex of the graph is distance at most $f(v)$ from some vertex v that has $f(v) > 0$. A broadcast domination is optimal if it minimizes the sum of the costs of the dominators across all vertices in the graph. These costs are typically taken to be the $f(v)$ values. Some related problems are discussed in [5].

The standard optimal domination problem is NP-hard on arbitrary graphs [9]. It is also NP-hard on several classes of graphs, including planar graphs [9], bipartite graphs [4], and chordal graphs [3]. The problem can be solved in polynomial time on, for example, AT-free graphs [12], permutation graphs [8], interval graphs [7], and trees [9].

If the broadcast domination problem is modified to include a maximum allowable broadcast power as a part of the instance, it is easy to see that the resulting problem is NP-hard since it reduces to DOMINATING SET when the maximum allowable broadcast power is set to 1. The complexity status of broadcast domination (without any broadcast power restrictions) is open. Broadcast domination is solvable in polynomial time on paths, cycles, complete graphs, and grid graphs [5]. It is also efficiently solvable on interval graphs, series-parallel graphs, and trees following work in [2].

This paper is organized in the following way. Necessary graph terminology and background, a formal definition of broadcast domination, and some old results are given in the next section. Section 3 gives integer programming formulations for the problem, along with computational results. Section 4 provides a brief summary with some concluding remarks.

2 Broadcast domination

All graphs in this paper are simple and connected. Let $G = (V, E)$ be a graph with $n = |V|$ and $m = |E|$. For any vertex $v \in V$, the *neighborhood* of v is the set $N(v) = \{u \mid uv \in E\}$ and the *closed neighborhood* is the set $N[v] = N(v) \cup \{v\}$. Similarly, for any set $S \subseteq V$, $N(S) = \cup_{v \in S} N(v) - S$ and $N[S] = N(S) \cup S$. A set S is a *dominating set* if $N[S] = V$. The minimum cardinality of a dominating set of G is denoted by $\gamma(G)$.

The *distance*, d_{uv}, between two vertices u and v in G is the smallest number of edges on a path between u and v in G. The *eccentricity*, $e(v)$, of a vertex v is the largest distance from v to any vertex of G. The *radius*, $rad(G)$, of G is the smallest eccentricity in G. The *diameter*, $diam(G)$, of G is the largest eccentricity in G.

A function $f : V \rightarrow \{0, 1, ..., diam(G)\}$ is a *broadcast* if for every vertex $v \in V$, $f(v) \leq e(v)$. The set of *broadcast dominators* defined by f is the set $V_f = \{u \mid f(u) > 0\}$. The set of vertices that a vertex v can *hear* is $H_f(v) = \{u \in V \mid d_{uv} \leq f(u)\}$. We will omit the subscript f when the broadcast function is clear from the context. The *cost* of a broadcast f incurred by a set $S \subseteq V$ is $f(S) = \sum_{v \in S} f(v)$. Thus, $f(V)$ is the total cost incurred by broadcast function f.[1] We say that G has an $f(V)$-*broadcast*. More informally, we say that vertex v has an *originating broadcast of power* q if $f(v) = q \geq 1$. If $f(v) = 0$ we say v has no originating broadcast.

A broadcast is *dominating* if $|H(v)| \geq 1$ for every vertex v of G. The term $\gamma_b(G)$ denotes the minimum cost of a dominating broadcast on G. We will refer to a dominating broadcast of cost $\gamma_b(G)$, as an *optimal broadcast*. Although a broadcast function is allowed to assign values from $\{0, 1, ..., diam(G)\}$, we never need to assign values larger than $rad(G)$ to any vertex in order to achieve an optimal broadcast. If we choose a vertex v of minimum eccentricity and assign $f(v) = rad(G)$ while assigning $f(w) = 0$ to all other vertices w, the resulting broadcast is dominating for G. We will call such a broadcast a *radial broadcast*. For some graph classes, a radial broadcast is also an optimal broadcast, and such graphs are called *radial* [5]. However, for interval graphs, series-parallel graphs, and trees, radial broadcasts are not necessarily optimal, as can be seen from the following simple example. Consider a path on 6 vertices: $v_1, v_2, ..., v_6$. A radial broadcast f requires $f(v_3)$ or $f(v_4)$ to be 3, which is the radius of a path on 6 vertices. However a dominating broadcast f' of cost 2 can be achieved by assigning $f'(v_2) = f'(v_5) = 1$.

A broadcast is *efficient* if every vertex hears exactly one broadcast, that is, for every v, $|H(v)| = 1$. The following interesting result is from [5].

Theorem 2.1 (Dunbar et. al. [5]) *Every graph G has a $\gamma_b(G)$-broadcast that is efficient.*

The proof of this theorem is constructive in the sense that it shows how any broadcast can be transformed into an efficient broadcast with the same total cost.

Before we continue, we also observe that the following more restricted problem is NP-complete.

> RESTRICTED BROADCAST DOMINATION: Given a graph $G = (V, E)$, and positive integers $K, M \leq |V|$, is there a broadcast domination f of G such that $\sum_{v \in V} f(v) \leq K$ and $\max_{v \in V} f(v) \leq M$?

If we set $M = 1$, this problem reduces to DOMINATING SET.

[1] The cost function might also be defined differently, reflecting the possibility that the cost of providing a broadcast $f(v)$ is not equal to $f(v)$.

3 IP Formulations and Computational Results

The broadcast domination problem can be formulated as a mixed integer program (MIP), or a purely binary integer program (BIP). We start with the MIP formulation. We define x_i to be the broadcast range of vertex i. In other words, x_i is the decision variable associated with $f(i)$. Note that the x_is are general integer variables. Then y_{ij} is a binary variable that takes on the value 1 if vertex i provides the domination that satisfies vertex j, and 0 otherwise. Recall d_{ij} is the distance from i to j in G. The MIP formulation follows:

$$
\begin{aligned}
& \min \sum_{i \in V} x_i \\
\text{subject to:} \quad & x_j + \sum_{i \neq j} y_{ij} \geq 1 \quad \forall j \in V \ \text{[Domination]} \\
& x_i \geq y_{ij} d_{ij} \quad \forall i \neq j \text{ with } i, j \in V \ \text{[Linking]} \\
& y_{ij} \in \{0, 1\} \text{ with } y_{ii} = 0, i, j \in V \\
& x_i \geq 0 \text{ integer, with } i \in V
\end{aligned}
$$

The domination constraints force each vertex to either be a dominator ($x_j \geq 1$), or be dominated by at least one other vertex. The linking constraints allow y_{ij} to be 1 only when x_i is large enough to provide domination out to vertex j.

There are n domination constraints and $n^2 - n$ linking constraints for a total of n^2. There are $n^2 - n$ binary variables, and n general integer variables. Note that although no upper bound on the x_is is needed in the formulation, computational performance is generally improved by including an upper bound. The radius of the graph is a natural choice for this bound since a dominator of that broadcast power placed at the center of the graph would dominate the entire graph.

The problem can also be formulated as a binary integer program. We define x_{ik} to be 1 if vertex i has an originating broadcast of power k, and 0 otherwise. Define the set of possible k values, except 0, as $K = \{1, 2, \ldots, rad(G)\}$. Note that $x_{i0} = 1$ means vertex i has no originating broadcast. The y_{ij}s are defined exactly as they are in the MIP formulation. Here is the BIP formulation:

$$\begin{aligned}
&\min \sum_{i\in V}\sum_{k\in K} k\cdot x_{ik} \\
\text{subject to:}\quad &\sum_{k\in K} x_{jk} + \sum_{i\neq j} y_{ij} \geq 1 \quad \forall j\in V \text{ [Domination]} \\
&\sum_{k\in K} k\cdot x_{ik} \geq y_{ij}d_{ij} \quad \forall i,j\in V\times V : i\neq j \text{ [Linking]} \\
&\sum_{k\in K\cup\{0\}} x_{ik} = 1 \quad \forall i\in V \text{ [Broadcast]} \\
&x_{ik}\in\{0,1\} \quad \forall i\in V, k\in K\cup\{0\} \\
&y_{ij}\in\{0,1\} \text{ with } y_{ii}=0, i,j\in V
\end{aligned}$$

The domination and linking constraints play the same role they did in the MIP formulation, and they do so in a very similar fashion. The broadcast constraints force each vertex to have exactly one associated $f(v)$ value. Note that k sometimes ranges over K, and sometimes over $K \cup \{0\}$. In the case of the domination constraints, we want the former since an originating broadcast of power 0 provides no domination. In the case of the broadcast constraints, we want the latter since $f(v) = 0$ is a valid choice. In the case of the objective function and the linking constraints, it makes no difference which range is used since the $k = 0$ case makes no contribution to $\sum_{k\in K} k \cdot x_{ik}$.

There are n domination constraints, $n^2 - n$ linking constraints, and n broadcast constraints for a total of $n^2 + n$. There are $n \cdot (rad(G) + 1)$ x_{ik} variables and $n^2 - n$ y_{ij} variables, all of which are binary. Since $rad(G) \leq \lfloor \frac{n}{2} \rfloor$, the total number of variables cannot exceed $(\lfloor \frac{n}{2} \rfloor + 1)n + n^2 - n = n\lfloor \frac{n}{2} \rfloor + n^2 \leq \frac{3}{2}n^2$.

We solved a collection of instances of the broadcast domination problem, with different types and sizes of graphs. The graph class named gt-n contains random labeled trees on n vertices. The graph classes gs-n and gvs-n are random graphs on n vertices using the standard Erdös-Renyi model where each of the $\binom{n}{2}$ possible edges appears independently with fixed probablity p. gs-40, gs-70, and gs-100 were created using edge probabilities 0.14, 0.07, and 0.06, respectively. gvs-40, gvs-70, and gvs-100 were created using edge probabilities 0.09, 0.05, and 0.04, respectively. Note that the gvs-n class is "very sparse" in the sense that the value of p used was just high enough to generate some connected instances without too many trials. Only connected instances were studied, since broadcast domination on an unconnected graph can be solved on its connected components. The graph class gtp-n starts with a random tree on n vertices as defined above. Then a much smaller random tree (on 9, 11, 14 , and 15 vertices for gtp-40, gtp-70, gtp-100, and gtp-150, respectively) was created and the instance is formed using the union of the edges of the two graphs. The resulting graphs are not trees, but have some important properties that distinguish them from the other classes under study. More about our motivation behind the definition of the gtp-n class appears in Section 4. Finally, we also considered simple paths on n vertices.

The instances were created using the Combinatorica package for Mathematica, using the commands `RandomTree[n]` and `RandomGraph[n,p]`. `RandomTree[n]` generates a random string of $n-2$ integers from $\{1, 2, ..., n\}$ (known as a Prűfer code) and then uses the bijection between such strings and labeled trees on n vertices [14]. The specific graphs are available from the authors on request.

We implemented our formulations and solved the instances described above using ILOG CPLEX 9.0 with AMPL, using a Dell Latitude laptop with a 1794Mhz processor and 648MB of available virtual memory. Table 1 summarizes the results.

Table 1: Computational Results

Graph	$\|V\|$	$\|E\|$	$\gamma_b(G)$	MIP		BIP	
				Time (sec)	Iterations	Time (sec)	Iterations
gt-40	40	39	8	32.4	21855	15.8	12632
gt-70	70	69	13	2369	436308	421	83210
gt-100	100	99	15	4401	486078	1955	212507
gs-40	40	120	3	20.2	14337	2.1	4673
gs-70	70	180	4	77.0	24759	46.2	25413
gs-100	100	287	4	128	32772	369	64420
gvs-40	40	67	4	6.09	10136	4.87	7195
gvs-70	70	110	5	98.7	44274	49.7	26516
gvs-100	100	200	5	171.0	39086	484.7	98228
gtp-40	40	48	7	75.2	37769	15.4	13788
gtp-70	70	80	8	1461	412479	145.6	47850
gtp-100	100	113	10	1262	269258	1720	222112
gtp-150	150	163	10	2759	354514	7222	457047
P_{39}	39	38	13	3554	2734106	35.5	12129
P_{40}	40	39	14	21342	10654323	73.6	12126

For most of the graphs tested, the BIP formulation outperformed the MIP formulation, in some cases significantly. But this is deceptive. The time savings was usually most pronounced for graphs with less than 100 vertices. For graphs of 100 or more vertices, the MIP formulation was typically better. This suggests that the computational effort required to solve instances of broadcast domination is less sensitive to instance size in the case of the MIP formulation, as opposed to the BIP formulation.

For our MIP implementation, we used $\lfloor \frac{n}{2} \rfloor$ instead of $rad(G)$ as the upper bound for each x_i, since the former is easy to compute inside of the AMPL model. Similarly, we allowed the index set of originating broadcasts to be $K = \{1, 2, \ldots, \lfloor \frac{n}{2} \rfloor\}$ for the BIP implementation. Somewhat surprisingly, tests performed on some of these graphs using $rad(G)$ instead of $\lfloor \frac{n}{2} \rfloor$ showed little, if any, computational effort saved. This was true for both MIP and BIP formulations. In the case of BIP formulations, CPLEX's presolve phase typically eliminated most of the excess x_{ik} variables that result from using $\lfloor \frac{n}{2} \rfloor$ as the upper bound on K. It is less clear why the lower upper bound on x_i in the MIP formulation didn't help CPLEX find a solution faster.

4 Summary

The primary results in this paper are MIP and BIP formulations for the broadcast domination problem, some computational results, and a comparison of these formulations from a computational standpoint. It is interesting to note that for the instance classes we considered, CPLEX seemed to take more time and require more iterations for graphs that were relatively sparse, with trees being the worst in this sense. The interest here comes from the fact that broadcast domination can be efficiently solved on trees using dynamic programming following work in [2].

Recall that a graph is radial if the broadcast function $f(v) = rad(G)$ and $f(w) = 0$ for all $w \in V - v$ for some vertex v of minimum eccentricity satisfies $f(v) = \gamma_b(G)$. Observe that $rad(G)$ is simply $\min_i \max_j d_{ij}$, and that a row i of the d_{ij} matrix where $\max_j d_{ij} = rad(G)$ corresponds to a vertex of minimum eccentricity. These measures are easy to compute. Since $rad(G) \geq \gamma_b(G)$, this suggests a very quick way to find an upper bound on $\gamma_b(G)$. It turns out that this bound is satisfied at equality for several of the graphs we studied. All of the gs-n and gvs-n graphs from our test instances have this property. CPLEX seems to handle these cases relatively quickly. Our motivation behind defining the structure of the gtp-n class of graphs was to investigate a class of graphs that were not radial, yet were not known to submit to efficient algorithms using combinatorial methods.

By far the worst performing instances were paths, where optimal dominating broadcast functions can be found by inspection. It is easy to see that paths are most cheaply dominated by dominators of power 1, so for a path with vertices labeled in increasing order, we simply set $f(v) = 1$ for $v \equiv 2 \pmod 3$ and $f(v) = 0$ otherwise. This leads quickly to $\gamma_b(P_n) = \lceil \frac{n}{3} \rceil$. Both the MIP and BIP formulations take an exceptionally large amount of computational effort to solve the problem on paths, given the instance size. The performance of the former was particularly bad. The graphs P_n have the property that can be crudely described as "symmetry" which accounts for some of the computational issues. In addition, it seems that instances with multiple optimal solutions (as is the case with a path on 40 vertices) take significantly more time than those with only one (as with a path on 39 vertices).

This note leaves some important questions unanswered. Central among these is the application of integer programming techniques to solve instances of broadcast domination arising in practice. In addition, all of our testing used the default settings for CPLEX. It may be possible to reduce the computational effort required by adjusting optimization options. Finally, it would be interesting to formally establish a result like the one suggested in the second paragraph of this section. Given a random graph with parameters n and p, can some statement be made about the likelihood that such an instance is radial? To us it appears that sufficiently dense random graphs are radial with very high probability.

References

[1] C. Berge, *Theory of Graphs and its Applications*, no. 2 in Collection Universitaire de Mathematiques, Dunod, Paris, 1958.

[2] J. R. S. Blair, P. Heggernes, S. B. Horton, and F. Manne, *Broadcast domination algorithms for interval graphs, series-parallel graphs, and trees*, Congressus Numerantium, (2004). Accepted.

[3] K. S. Booth and J. H. Johnson, *Dominating sets in chordal graphs*, SIAM J. Comput., 11 (1982), pp. 191–199.

[4] A. K. Dewdney, *Fast Turing reductions between problems in NP*, Tech. Rep. 71, Dept. of Computer Science, University of West Ontario, 1981.

[5] J. E. Dunbar, D. J. Erwin, T. W. Haynes, S. M. Hedetniemi, and S. T. Hedetniemi, *Broadcasts in graphs*, (2002). Submitted.

[6] D. J. Erwin, *Dominating broadcasts in graphs*, Bulletin of the Institute of Combinatorics and its Applications, 42 (2004), pp. 89–105.

[7] M. Farber, *Domination, independent domination, and duality in strongly chordal graphs*, Disc. Appl. Math., 7 (1984), pp. 115–130.

[8] M. Farber and J. M. Keil, *Domination in permutation graphs*, J. Algorithms, 6 (1985), pp. 309–321.

[9] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman and Co., 1978.

[10] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater, *Domination in Graphs: Advanced Topics*, Marcel Dekker, New York, 1998.

[11] ——, *Fundamentals of Domination in Graphs*, Marcel Dekker, New York, 1998.

[12] D. KRATSCH, *Domination and total domination in asteroidal triple-free graphs*, Tech. Rep. Math/Inf/96/25, F.-Schiller-Universität, Jena, 1996.

[13] O. ORE, *Theory of Graphs*, no. 38 in American Mathematical Society Publications, AMS, Providence, 1962.

[14] S. SKIENA AND S. PEMMARAJU, *Computational Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*, Cambridge University Press, Cambridge, UK, 2003.