Computing the broadcast time of a graph

Marika Ivanova*

Department of Theoretical Computer Science and Mathematical Logic, Charles University, Prague, Czech Republic

Dag Haugland

Department of Informatics, University of Bergen, Norway

Bård Hennning Tvedt

Department of Physics, University of Bergen, Norway

Abstract

Given a graph and a subset of its nodes, referred to as source nodes, the minimum broadcast problem asks for the minimum number of steps in which a signal can be transmitted from the sources to all other nodes in the graph. In each step, the sources and the nodes that already have received the signal can forward it to at most one of their neighbor nodes. The problem has previously been proved to be NP-hard. In the current work, we develop a compact integer programming model for the problem. We also devise procedures for computing lower bounds on the minimum number of steps required, along with methods for constructing near-optimal solutions. Computational experiments demonstrate that in a wide range of instances with sufficiently dense graphs, the lower and upper bounds under study collapse. In instances

*Corresponding author

Email addresses: `Marika.Ivanova@uib.no` (Marika Ivanova), `Dag.Haugland@uib.no` (Dag Haugland), `Bard.Tvedt@student.uib.no` (Bård Hennning Tvedt)

Preprint submitted to Discrete Applied Mathematics *Received: date / Accepted: date*

where this is not the case, the integer programming model proves strong capabilities in closing the remaining gap.

Keywords: Broadcasting, Integer Programming, Bounds, Computational Experiments

1. Introduction

Fast and efficient distribution of information gives rise to many optimization problems of growing interest. Information dissemination processes studied in the mathematical and algorithmic literature [8, 11, 16, 18] often fall into one of the categories *gossiping* or *broadcasting*. When all network nodes control each their unique piece of information, and all pieces are to be disseminated to all nodes, the process is called gossiping [2, 3]. Dissemination of the information controlled by one particular source node to all network nodes is referred to as broadcasting [21, 25], and multicasting [1] if a subset of the network nodes are information targets. If the information is to be stored at the source, and assembled by pieces stored at all other nodes, then the information flows in the reverse of the broadcasting direction, and the dissemination process is *accumulation*. Broadcasting and accumulation can both be generalized to processes where only a subset of the nodes need to receive/disseminate information, while the remaining nodes are available as transit units that pass the information on to neighboring nodes.

Information dissemination follows a certain *communication model*. In the *whispering* model, each node sends/receives information to/from at most one other node in its vicinity at the time. The *shouting* model corresponds to the case where nodes communicate with all their neighbor nodes simultaneously. Generalizing whispering and shouting, the communication can also be constrained to neighbor subsets of given cardinality.

In the current work, a problem in the domain of broadcasting is studied. The *minimum broadcast time* (MBT) problem is identified by a graph and a subset of its nodes, referred to as source nodes. Each node in the graph corresponds to a communication unit. The task is to disseminate a signal from the source nodes to all other nodes in a shortest possible time (broadcast time), while abiding by communication rules. A node is said to be *informed* at a given time if it is a source, or it already has received the signal from some other node. Otherwise, the node is said to be *uninformed*. Consequently, the set of informed nodes is initially exactly the set of sources. Reflecting the fact that communication can be established only between pairs of nodes that are located within a sufficiently close vicinity of each other, the edge set of the graph consists of potential communication links along which the signal can be transmitted.

The time is divided into a finite number of steps. Agreeing with the whispering model, every informed node can, in each time step, forward the signal to at most one uninformed neighbor node. Therefore, the number of informed nodes can at most be doubled from one step to the next. This communication protocol appears in various practical applications, such as communication among computer processors or telephone networks. In situations where the signals have to travel large distances, it is typically assumed that the signal is sent to one neighbor at the time. Inter-satellite communication networks thus constitute a prominent application area [4]. Particularly, the MBT problem arises when one or a few satellites need to broadcast data quickly by means of time-division multiplexing.

The current literature on MBT offers some theoretical results, including complexity and approximability theorems. Although inexact solution methods also have been proposed, few attempts seem to be made in order to compute the exact optimum, or to find lower bounds on the minimum broadcast time. The goal of the current text is to fill this gap, and we make the following contributions in that direction:

First, a compact integer programming model is developed. While the model targets the exact minimum in instances of moderate size, its continuous relaxation is suitable for computation of lower bounds in larger instances. Second, we derive lower bounding techniques, both of an analytical nature and in terms of a combinatorial relaxation of MBT, that do not rely on linear programming. Third, we devise an upper bounding algorithm, which in combination with the lower bounds is able to close the optimality gap in a wide range of instances.

The remainder of the paper is organized as follows: Next, we review the current scientific literature on MBT and related problems, and in Section 2, a concise problem definition is provided. The integer program is formulated and discussed in Section 3. Lower and upper bounding methods are derived in Sections 4 and 5, respectively. Computational experiments are reported in Section 6, before the work is concluded by Section 7.

1.1. Literature overview

Deciding whether an instance of MBT has a solution with broadcast time at most t has been shown to be NP-complete [**?** 28]. For bipartite planar graphs with maximum degree 3, NP-completeness persists even if $t = 2$ or if there is only one source [19]. When $t = 2$, the problem also remains NP-complete for cubic planar graphs [22], grid graphs with maximum degree 3, complete grid graphs, chordal graphs, and for split graphs [19]. The single-source variant of the decision version of MBT is NP-complete for grid graphs with maximum degree 4, and for chordal graphs [19]. The problem is known to be polynomial in trees [28]. Whether the problem is NP-complete for split graphs with a single source was stated as an open questions in [19], and has to the best of our knowledge not been answered yet.

A number of inexact methods, for both general and special graph classes, have been proposed in the literature during the last three decades. One of the first works of this category [26] introduces a dynamic programming algorithm that identifies all maximum matchings in an induced bipartite graph. Additional contributions of [26] include heuristic approaches for near optimal broadcasting. Among more recent works, [15] describes a meta heuristic algorithm for MBT, and provides a comparison with other existing methods. The communication model is considered in an existing satellite navigation system in [4], where a greedy inexact method is proposed together with a mathematical programming model. Examples of additional efficient heuristics can be found e.g. in [**?** 13, 12, 29].

Approximation algorithms for MBT are studied in [20]. The authors argue that methods presented in [26] provide no guarantee on the performance, and show that wheel-graphs are examples of unfavorable instances. Another contribution from [20] is an $\mathcal{O}(\sqrt{n})$-additive approximation algorithm for broadcasting in general graphs with n nodes. The same work also provides approximation algorithms for several graph classes with small separators with approximation ratio proportional to the separator size times $\log n$. An algorithm with $\mathcal{O}\left(\frac{\log n}{\log\log n}\right)$-approximation ratio is given in [6]. (Throughout the current text, the symbol log refers to the logarithm with base 2.) Most of the works cited above consider a single source.

A related problem extensively studied in the literature is the minimum broadcast graph problem [10, 21]. A broadcast graph supports a broadcast from any node to all other nodes in optimal time $\lceil \log n \rceil$. For a given integer n, a variant of the problem is to find a broadcast graph of n nodes such that the number of edges in the graph is minimized. In another variant, the maximum node degree rather than the edge cardinality is subject to minimization. McGarvey et al. [21] study integer linear programming (ILP) models for c-broadcast graphs, which is a generalization where signal transmission to at most c neighbours is allowed in a single time step.

Despite a certain resemblance with MBT, the minimum broadcast graph problem is clearly distinguished from our problem, and will consequently not be considered further in the current work.

2. Network Model and Definitions

The communication network is represented by a connected graph $G = (V, E)$ and a subset $S \subseteq V$ referred to as the set of sources. We denote the number of nodes and the number of sources by $n = |V|$ and $\sigma = |S|$, respectively. The digraph with nodes V and arcs (u, v) and (v, u) for each $\{u, v\} \in E$ is denoted $\overrightarrow{G} = (V, \overrightarrow{E})$.

Definition 1. *The* broadcast time $\tau(G, S)$ *of a node set* $S \subseteq V$ *in* G *is defined as the smallest integer* $t \geq 0$ *for which there exist a sequence* $V_0 \subseteq \cdots \subseteq V_t$ *of node sets and a function* $\pi : V \setminus S \to V$*, such that:*

1. $V_0 = S$ *and* $V_t = V$,

2. *for all* $v \in V \setminus S$, $\{\pi(v), v\} \in E$,

3. *for all* $k = 1, \ldots, t$ *and all* $v \in V_k$, $\pi(v) \in V_{k-1}$, *and*

4. *for all* $k = 1, \ldots, t$ *and all* $u, v \in V_k \setminus V_{k-1}$, $\pi(u) = \pi(v)$ *only if* $u = v$.

Referring to Section 1, the node set V_k is the set of nodes that are informed in time step k. Initially, only the sources are informed ($V_0 = S$), whereas all nodes are informed after t time steps ($V_t = V$), and the set of informed nodes is monotonously non-decreasing ($V_{k-1} \subseteq V_k$ for $k = 1, \ldots, t$). The parent function π maps each node to the node from which it receives the signal. Conditions 2–3 of Definition 1 thus reflect that the sender is a neighbor node in G, and that it is informed at an earlier time step than the recipient node. Because each node can send to at most one neighbor node in each time step, condition 4 states that π maps the set of nodes becoming informed in step k to distinct parent nodes. The preimage of v under π, that is, the set of child nodes of v, is denoted $\pi^{-1}(v)$.

The optimization problem in question is formulated as follows:

Problem 1 (Minimum Broadcast Time)**.** *Given* $G = (V, E)$ *and* $S \subseteq V$, *find* $\tau(G, S)$.

Definition 2. *For any* $V_0, \ldots, V_t$ *and* π *satisfying the conditions of Definition 1, possibly with the exception of* t *being minimum, the corresponding* broadcast forest *is the digraph* $D = (V, A)$, *where* $A = \{(\pi(v), v) : v \in V\}$. *If* t *is minimum,* D *is referred to as a* minimum *broadcast forest. Each connected component of* D *is a* communication tree.

It is easily verified that the communication trees are indeed arborescences, rooted at distinct sources, with arcs pointing away from the source. Let $T(s) = (V(s), A(s))$ denote the communication tree in D rooted at source $s \in S$, and let $T_k(s)$ be the subtree of $T(s)$ induced by $V(s) \cap V_k$. Analogously, let D_k be the directed subgraph of D induced by node set V_k. For the sake of notational simplicity, the dependence on $(V_0, \ldots, V_t, \pi)$ is suppressed when referring to the directed graphs introduced here.

The degree of node v in graph G is denoted by $\deg_G(v)$. For a given subset $U \subseteq V$ of nodes, we define $G[U]$ as the subgraph of G induced by U. We let $\deg^+_{\overrightarrow{G}}(v)$ and $\deg^-_{\overrightarrow{G}}(v)$ denote, respectively, the out-degree and the in-degree of node v in $\overrightarrow{G}$, and we let $\deg_{\overrightarrow{G}}(v) = \deg^+_{\overrightarrow{G}}(v) + \deg^-_{\overrightarrow{G}}(v)$. When p is a logical proposition, $\delta_p = 1$ if p is true, and $\delta_p = 0$, otherwise.

3. Exact methods

In this section, we formulate an ILP model for Problem 1, and discuss possible solution strategies.

3.1. Broadcast time model

Given an integer $t \geq \tau(G, S)$, define the variables $((u, v) \in \overrightarrow{E},\ k = 1, \ldots, t)$

$$x^k_{uv} = \begin{cases} 1, & \text{if } v \in V_k \setminus V_{k-1} \text{ and } \pi(v) = u, \\ 0, & \text{otherwise,} \end{cases} \qquad z_k = \begin{cases} 1, & \text{if } k \leq \tau(G, S), \\ 0, & \text{otherwise.} \end{cases}$$

Variable x^k_{uv} thus represents the decision whether or not the signal is to be transmitted from node u to node v in period k, while z_k indicates whether transmissions take place as late as in period k.

An upper bound t on the broadcast time $\tau(G, S)$ is easily available. Because G is connected, the cut between any set V_i of informed nodes and its complement is non-empty, and therefore at least one more node can be informed in each period. It follows that $\tau(G, S) \leq n - \sigma$. The bound is tight in the worst case instance where $S = \{v_1\}$, and G is a path with v_1 as one of its end nodes. Problem 1 is then formulated as follows:

$$
\begin{aligned}
& \min \sum_{k=1}^{t} z_k && && \text{(1a)} \\
\text{s. t.} \quad & \sum_{k=1}^{t} \sum_{v \in N(u)} x_{vu}^k = 1 && u \in V \setminus S, && \text{(1b)} \\
& \sum_{v \in N(u)} x_{uv}^k \leq \sum_{\ell=1}^{k-1} \sum_{w \in N(u)} x_{wu}^\ell && u \in V \setminus S, k = 2, \ldots, t, && \text{(1c)} \\
& \sum_{v \in N(u)} x_{uv}^k \leq z_k && u \in V, k = 1, \ldots, t, && \text{(1d)} \\
& z_k \leq z_{k-1} && k = 2, \ldots, t, && \text{(1e)} \\
& x_{uv}^1 = 0 && (u, v) \in \overrightarrow{E}, u \in V \setminus S, && \text{(1f)} \\
& x_{uv}^k = 0 && (u, v) \in \overrightarrow{E}, v \in S, k = 1, \ldots, t && \text{(1g)} \\
& x \in \{0, 1\}^{\overrightarrow{E} \times \{1, \ldots, t\}}, z \in \{0, 1\}^{\{1, \ldots, t\}}. && && \text{(1h)}
\end{aligned}
$$

By (1b), every non-source node u receives the signal from exactly one adjacent node v in some time step k. The requirement that a non-source node u informs a neighbor v in the k-th time step only if u is informed by some adjacent node w in an earlier time step is modeled by (1c). Constraints (1d) enforce that each node $u \in V$ forwards the signal to at most one adjacent node v in each time step. It also sets correct values to the z-variables that appear in the objective function. The (redundant) valid inequalities (1e) reflect that $k \leq \tau(G,S)$ only if $k-1 \leq \tau(G,S)$. Lastly, constraints (1f) and (1g) state, respectively, that non-source nodes do not transmit in the first time step, and that sources never receive the signal.

3.1.1. Decision version

The nature of MBT suggests another modelling approach derived from model (1). For a given positive integer t, we maximize the number of nodes v that receive a signal from some neighbor u within t time steps. Hence, $\tau(G,S)$ is the smallest value of t for which the maximum attains the value $n-\sigma$. If the optimal objective function value is $n-\sigma$, the corresponding value of x induces a broadcast forest with broadcast time t. Otherwise, if the optimal objective function value is smaller than $n-\sigma$, $t < \tau(G,S)$, and model (2) is solved again with t replaced by $t+1$. The computational efficiency of a search where t is increased from some lower bound on $\tau(G,S)$ is largely dependent of the tightness of available upper and lower bounds on $\tau(G,S)$. If an upper bound $\bar{t} \geq \tau(G,S)$ is known, and it is revealed that for $t = \bar{t}-1$ fewer than $n-\sigma$ can receive the signal in time t, it is concluded that $\tau(G,S) = \bar{t}$.

The decision version of a model for MINIMUM BROADCAST TIME takes the form

$$\max \sum_{v\in V\setminus S} \sum_{u\in N(v)} \sum_{k=1}^{t} x_{uv}^{k} \tag{2a}$$

$$\text{s. t.} \quad \sum_{k=1}^{t} \sum_{v\in N(u)} x_{vu}^{k} \leq 1 \qquad u \in V \setminus S, \tag{2b}$$

$$\sum_{v\in N(u)} x_{uv}^{k} \leq 1 \qquad u \in S, k = 1, \ldots, t, \tag{2c}$$

$$(1c), (1f), (1g), \tag{2d}$$

$$x \in \{0,1\}^{\overrightarrow{E}\times\{1,\ldots,t\}}. \tag{2e}$$

In the transition from the optimization model (1), constraint (1b) is replaced by (2b). The former is an inequality in the decision version, because not all nodes are necessarily reached within the given time limit. To pursue the constraint that sources inform at most one neighbor node in each period, (1d) is replaced by (**??**).

4. Lower bounds

Strong lower bounds on the minimum objective function value are of vital importance to combinatorial optimization algorithms. In this section, we study three types of lower bounds on the broadcast time $\tau(G, S)$.

4.1. Analytical lower bounds

Any solution $(V_0, \ldots, V_t, \pi)$ satisfying conditions 1–4 of Definition 1, also satisfies $|V_{k+1}| \leq 2\,|V_k|$ for all $k \geq 0$. Combined with the observation made in Section 3.1.1, this yields the following bounds:

Observation 1. *For all instances* (G, S) *of Problem 1,*

$$\left\lceil \log \frac{n}{\sigma} \right\rceil \leq \tau(G, S) \leq n - \sigma. \tag{3}$$

Consider the m-step Fibonacci numbers $\{f_k^m\}_{k=1,2,\ldots}$ [23], a generalization of the well-known (2-step) Fibonacci numbers, defined by $f_k^m = 0$ for $k \leq 0$, $f_1^m = 1$, and other terms according to the linear recurrence relation

$$f_k^m = \sum_{j=1}^{m} f_{k-j}^m, \qquad \text{for } k \geq 2.$$

The generalized Fibonacci numbers are instrumental in the derivation of a lower bound on $\tau(G, S)$, depending on the maximum node degree $d = \max\{\deg_G(v) : v \in V\}$ in G. The idea behind the bound is that the broadcast time can be no shorter than what is achieved if the following ideal, but not necessarily feasible, criteria are met: Every source transmits the signal to a neighbor node in each of the periods $1, \ldots, d$, and every node $u \in V \setminus S$ transmits the signal to a neighbor node in each of the first $d - 1$ periods following the period when u gets informed. An exception possibly occurs in the last period, as there may be fewer nodes left to be informed than there are nodes available to inform them.

Proposition 2.

$$\tau(G, S) \geq \min \left\{ t : 2\sigma \sum_{j=1}^{t} f_j^{d-1} \geq n \right\}.$$

Proof. Consider a solution $(V_0, \ldots, V_t, \pi)$ with associated broadcast graph D, such that $V_0 \neq V_1 \neq \cdots \neq V_{t-1} \neq V_t$,

- conditions 1 and 3–4 of Definition 1 are satisfied,
- for each source $u \in S$ and each $j = 1, \ldots, \min\{d, t-1\}$, there exists a node $v \in V_j \setminus V_{j-1}$ such that $\pi(v) = u$, and
- for each $k \in \{1, \ldots, t-2\}$, each node $u \in V_k \setminus V_{k-1}$, and each $j = k+1, \ldots, \min\{k+d-1, t-1\}$, there exists a node $v \in V_j \setminus V_{j-1}$ such that $\pi(v) = u$.

That is, all sources send the signal to some uninformed node (not necessarily a neighbor node) in all periods up to $\min\{d, t-1\}$. All nodes that received the signal in period k, forward it to some uninformed node in all periods up to $\min\{d-1, t-1\}$, and all nodes are informed in period t. Because condition 2 of Definition 1, stating that the flow of information follows E, is not imposed, such a solution $(V_0, \ldots, V_t, \pi)$ exists for an appropriate choice of t. Since the solution implies that every node is actively receiving or sending for up to d consecutive periods, until the signal is broadcasted in period t, it follows that $\tau(G, S) \geq t$. It remains to prove that the chosen t is the smallest value satisfying $2\sigma \sum_{k=1}^{t} f_k^{d-1} \geq n$, i.e., that $2\sigma \sum_{k=1}^{t-1} f_k^{d-1} < n \leq 2\sigma \sum_{k=1}^{t} f_k^{d-1}$.

For $k = 1, \ldots, t$, let $L_k = \{v \in V_k : \deg_{D_k}(v) = 1\}$ denote the set of nodes with exactly one out- or in-neighbor in D_k, and let $L_k = \emptyset$ for $k \leq 0$. That is, for $k > 1$, L_k is the set of nodes that receive the signal in period k, whereas L_1 consists of all nodes informed in period 1, including the sources S. Hence, $L_1, \ldots, L_{t-1}$ are disjoint sets (but L_t may intersect L_{t-1}), and $V_k = L_1 \cup \cdots \cup L_k$ for all $k = 1, \ldots, t$.

Consider a period $k \in \{2, \ldots, t-1\}$. The assumptions on $(V_0, \ldots, V_t, \pi)$ imply that π is a bijection from L_k to $L_{k-1} \cup \cdots \cup L_{k-d+1}$. Thus, $|L_k| =$

$\sum_{j=1}^{d-1} |L_{k-j}|$. Since also $|L_1| = 2\sigma = 2\sigma f_1^{d-1}$ and $|L_j| = f_j^{d-1} = 0$ for $j \leq 0$, we get $|L_k| = 2\sigma f_k^{d-1}$. Further, $|L_t| \leq \sum_{j=1}^{d-1} |L_{t-j}| = 2\sigma f_t^{d-1}$. It follows that $2\sigma \sum_{k=1}^{t-1} f_k^{d-1} = \sum_{k=1}^{t-1} |L_k| = |V_{t-1}| < n = |V_t| \leq \sum_{k=1}^{t} |L_k| \leq 2\sigma \sum_{k=1}^{t} f_k^{d-1}$, which completes the proof. □

4.2. Continuous relaxations of integer programming models

For $t \in \mathbb{Z}_+$, define $\Omega(t) \subseteq [0,1]^{\overrightarrow{E} \times \{1,\ldots,t\}}$ as the set of feasible solutions to the continuous relaxation of (2a)–(2d), and let

$$\Omega^{=}(t) = \left\{ x \in \Omega(t) : \sum_{u \in N(v)} \sum_{k=1}^{t} x_{uv}^k = 1 \quad (v \in V \setminus S) \right\}.$$

Let $t^* = \min \{t \in \mathbb{Z}_+ : \Omega^{=}(t) \neq \emptyset\}$ be the smallest value of t for which the optimal objective function value in the relaxation equals $n - \sigma$. Existence of t^* follows directly from $\Omega^{=}(\tau(G,S)) \neq \emptyset$.

The continuous relaxation of (1a)–(1h) is feasible for sufficiently large t. We denote its optimal objective function value by $\zeta(t)$.

Proposition 3. *For all $t \in \mathbb{Z}_+$ such that $\Omega^{=}(t) \neq \emptyset$, $\zeta(t) \leq t^* \leq \tau(G,S)$.*

Proof. We first prove that $\zeta(t)$ is non-increasing with increasing t: Let $\Gamma(t)$ denote the set of feasible solutions to the continuous relaxation of (1a)–(1h), and assume $(x,z) \in \Gamma(t)$. Define $\hat{x} \in [0,1]^{\overrightarrow{E} \times \{1,\ldots,t+1\}}$ such that for all $(u,v) \in \overrightarrow{E}$, $\hat{x}_{uv}^k = x_{uv}^k$ $(k \leq t)$ and $\hat{x}_{uv}^{t+1} = 0$. An analogous extension of z to $\hat{z} \in [0,1]^{\{1,\ldots,t+1\}}$ yields $(\hat{x},\hat{z}) \in \Gamma(t+1)$, and $\sum_{k=1}^{t+1} \hat{z}_k = \sum_{k=1}^{t} z_k$ proves that $\zeta(t+1) \leq \zeta(t)$.

For $t \in \mathbb{Z}_+$ such that $\Omega^{=}(t) \neq \emptyset$, $t \geq t^*$ thus implies $\zeta(t) \leq \zeta(t^*) \leq t^*$, where the latter inequality follows from (1a) and $z_1, \ldots, z_{t^*} \leq 1$. The proof is complete by observing that $t^* \leq \tau(G,S)$ follows from $\Omega^{=}(\tau(G,S)) \neq \emptyset$. □

Remark 4. *To compute a lower bound on $\tau(G, S)$, Proposition 3 suggests to solve a sequence of instances of the continuous relaxation of problem* (2a)–(2d)*, and stop by the first value t^* of $t = 1, 2, \dots$ for which the optimal objective function value is $n - \sigma$. Such an approach yields a lower bound (t^*) on $\tau(G, S)$, which is no weaker than the bound achieved by solving the continuous relaxation of* (1a)–(1h)*.*

Remark 5. *Remark 4 also applies to a reformulation of* (1a)–(1h)*, where a unique integer variable y replaces $z_1, \dots, z_t$, and the objective is to minimize y subject to the constraints $y \geq \sum_{k=1}^{t} k \sum_{u \in N(v)} x_{uv}^k$ $(v \in V \setminus S)$,* (1b)–(1c)*, and* (1f)–(1h)*.*

4.3. Combinatorial relaxations

Lower bounds on the broadcast time $\tau(G, S)$ are obtained by omitting one or more of the conditions imposed in Definition 1. For the purpose of strongest possible bounds, the relaxations thus constructed can be supplied with conditions that are redundant in the problem definition. Conditions 1–4 of Definition 1 imply that

5. for all $v \in V$, $|\pi^{-1}(v)| \leq \deg_G(v) - \delta_{v \in V \setminus S}$.

A lower bound on $\tau(G, S)$ is then given by the solution to:

Problem 2 (Node Degree Relaxation)**.** *Find the smallest integer $t \geq 0$ for which there exist a sequence $V_0 \subseteq \cdots \subseteq V_t$ of node sets and a function $\pi : V \setminus S \to V$, satisfying conditions 1 and 3–5.*

Observe that the bound given in Proposition 2 is obtained by exploiting the lower-bounding capabilities of the Node Degree Relaxation. By considering the degree of all nodes $v \in V$, rather than just the maximum degree, stronger bounds may be achieved in instances where G is not regular ($\min_{v \in V} \deg_G(v) < \max_{v \in V} \deg_G(v)$).

Denote the source nodes $S = \{v_1, \ldots, v_\sigma\}$ and the non-source nodes $V \setminus S = \{v_{\sigma+1}, \ldots, v_n\}$, where $\deg_G(v_{\sigma+1}) \geq \deg_G(v_{\sigma+2}) \geq \cdots \geq \deg_G(v_n)$, and let $d_i = \deg_G(v_i)$ ($i = 1, \ldots, n$). Thus, $\{d_1, \ldots, d_n\}$ resembles the conventional definition of a non-increasing degree sequence of G, with the difference that only the subsequence consisting of the final $n - \sigma$ degrees is required to be non-increasing.

For a given $t \in \mathbb{Z}_+$, consider the problem of finding $(V_0, \ldots, V_t, \pi)$ such that $V_0 = S$, conditions 3–5 are satisfied, and $|V_t|$ is maximized. The smallest value of t for which the maximum equals n is obviously the solution to Problem 2.

The algorithm for Problem 2, to follow later in the section, utilizes that the maximum value of $|V_t|$ is achieved by transmitting the signal to nodes in non-increasing order of their degrees. Observe that, contrary to the case of Problem 1, transmissions to non-neighbors are allowed in the relaxed problem. Any instance of Problem 2 thus has an optimal solution where, for $k = 1, \ldots, t-1$, $u \in V_k \setminus V_{k-1}$ and $v \in V_{k+1} \setminus V_k$ implies $\deg_G(u) \geq \deg_G(v)$.

A rigorous proof of this follows next.

Lemma 6. *The maximum value of $|V_t|$ over all $(V_0, \ldots, V_t, \pi)$ satisfying $V_0 = S$ and conditions 3–5, is attained by some $(V_0, \ldots, V_t, \pi)$ where* $\min\{i : v_i \in V_k \setminus V_{k-1}\} > \max\{i : v_i \in V_{k-1}\}$ *($k = 1, \ldots, t$).*

Proof. Consider an arbitrary optimal solution $(V_0, \ldots, V_t, \pi)$, and assume that $v_i \in V_p \setminus V_{p-1}$, $v_j \in V_q \setminus V_{q-1}$, $i < j$, and $1 \leq q < p \leq t$. We prove that the solution obtained by swapping nodes v_i and v_j is also optimal. Let $\bar{V}_k = V_k$ for $k = 0, \ldots, q-1, p, p+1, \ldots, t$, and $\bar{V}_k = (V_k \setminus \{v_j\}) \cup \{v_i\}$ for $k = q, \ldots, p-1$. Because $\left|\bar{V}_t\right| = |V_t|$, we only need to show that $\left(\bar{V}_0, \ldots, \bar{V}_t, \bar{\pi}\right)$ is feasible for some $\bar{\pi}$. In the following, we demonstrate that a valid parent function $\bar{\pi}$ can be obtained by swapping $\pi(v_i)$ and $\pi(v_j)$, along with a simple adjustment ensuring that $|\bar{\pi}^{-1}(v_j)| \leq |\pi^{-1}(v_j)|$.

Define $m = \max\{0, |\pi^{-1}(v_i)| - |\pi^{-1}(v_j)|\}$. Consider the case where $m > 0$. Because v_i has at most one child in each $V_k \setminus V_{k-1}$ ($k = p+1, \ldots, t$), there exist integers $p_1 > \cdots > p_m > p$, and nodes $u_r \in V_{p_r} \setminus V_{p_r-1}$ ($r = 1, \ldots, m$) such that $\pi(u_r) = v_i$, whereas v_j has no child in $\bigcup_{r=1}^{m} (V_{p_r} \setminus V_{p_r-1})$. Let $U = \{u_1, \ldots, u_m\}$, and let $U = \emptyset$ if $m = 0$.

Let $\bar{\pi}(v) = v_i$ for all $v \in U$, and $\bar{\pi}(v) = v_j$ for all $v \in \pi^{-1}(v_i) \setminus U$. Also, let $\bar{\pi}(v) = v_i$ for all $v \in \pi^{-1}(v_j) \setminus \{v_i\}$. If $\pi(v_i) = v_j$, let $\bar{\pi}(v_j) = v_i$, otherwise let $\bar{\pi}(v_j) = \pi(v_i)$. Let $\bar{\pi}(v_i) = \pi(v_j)$. For all other non-source nodes, that is, all $v \in V \setminus S$ for which $v_i \neq \pi(v) \neq v_j$, let $\bar{\pi}(v) = \pi(v)$.

If $m > 0$, $|\bar{\pi}^{-1}(v_i)| = |\pi^{-1}(v_i)| \leq \deg_G(v_i) - 1$ and $|\bar{\pi}^{-1}(v_j)| = |\pi^{-1}(v_j)| \leq \deg_G(v_j) - 1$. Otherwise, $|\bar{\pi}^{-1}(v_i)| = |\pi^{-1}(v_j)| \leq \deg_G(v_j) - 1 \leq \deg_G(v_i) - 1$, and $|\bar{\pi}^{-1}(v_j)| = |\pi^{-1}(v_i)| \leq |\pi^{-1}(v_j)| \leq \deg_G(v_j) - 1$. For $v_i \neq v \neq v_j$, $|\bar{\pi}^{-1}(v)| = |\pi^{-1}(v)|$, and thus $\left(\bar{V}_0, \ldots, \bar{V}_t, \bar{\pi}\right)$ satisfies condition 5. It is straightforward to show that $\left(\bar{V}_0, \ldots, \bar{V}_t, \bar{\pi}\right)$ also satisfies conditions 3–4. $\square$

Algorithm 1 takes as input the number σ of sources and the number n of nodes, along with the vertex degrees $d_1, \ldots, d_n$, where $d_{\sigma+1} \geq \cdots \geq d_n$. It operates with a counter ν of informed nodes, initiated to σ. Thus, nodes $v_1, \ldots, v_\nu$ are informed, whereas $v_{\nu+1}, \ldots, v_n$ are not. A counter denoted a_i ($i = 1, \ldots, n$) keeps track of the number of nodes informed by node v_i. The set F consists of indices i of informed nodes that still have not sent the signal to $d_i - 1$ nodes (d_i nodes if $i \leq \sigma$). In each iteration of the outer loop of the algorithm, all nodes v_i for which $i \in F$, informs some currently uninformed node, and all counters are updated accordingly. The process stops when all n nodes are informed, and the number of performed iterations is returned.

Data: $\sigma, n, d_1, \ldots, d_n \in \mathbb{Z}_+$

$a_1 \leftarrow \cdots \leftarrow a_n \leftarrow 0$, $\nu \leftarrow \sigma$

for $t = 1, 2, \ldots$ **do**

  $F \leftarrow \{i = 1, \ldots, \nu : a_i < d_i - \delta_{i>\sigma}\}$

  for $i \in F$ **do** $a_i \leftarrow a_i + 1$

  $\nu \leftarrow \nu + |F|$

  if $\nu \geq n$ **return** t

end

Algorithm 1: Lower bound exploiting the degree distribution

Proposition 7. *Algorithm 1 returns a lower bound on $\tau(G, S)$.*

Proof. Follows from Lemma 6 and the subsequent discussion. □

5. Upper bounds

Access to an upper bound $\bar{t} \geq \tau(G, S)$ affects the number of variables in the models studied in Section 3.1. Particularly in the decision version (2), the iterative approach can be terminated once the solution is found to be infeasible for broadcast time limit $\bar{t} - 1$. Algorithms that output feasible, or even near-optimal solutions, are instrumental in the computation of upper bounds. Further, such methods are required in sufficiently large instances, where exact approaches fail to terminate in practical time.

5.1. Existing heuristic methods

Building on earlier works [13, 14], Harutyunyan and Jimborean [12] study a heuristic (considering $\sigma = 1$) departing from a shortest-path tree of G. A sequence of local improvements is performed in the bottom-up direction in the tree, starting by the leafs and terminating at the root node. Rearrangements of the parent assignments are made in order to reduce the broadcast time needed in subtrees. The heuristic has running time $\mathcal{O}(|E| \log n)$.

5.2. A construction method

Consider an integer $t' \geq 0$, node sets $S = V_0 \subseteq V_1 \subseteq \cdots \subseteq V_{t'} \neq V$ and a function $\pi : V \setminus S \mapsto V$, where $\{\pi(v), v\} \in E$ for all $v \in V_{t'} \setminus S$, and conditions 3–4 of Definition 1 are satisfied for $t = t'$. That is, $(V_0, \ldots, V_{t'}, \pi)$ defines a broadcast forest corresponding to the instance $(G[V_{t'}], S)$, but the forest does not cover V. In particular, if $t' = 0$, the broadcast forest is a null graph on S, while it is a matching from S to $V_1 \setminus S$ if $t' = 1$.

This section addresses the problem of extending the partial solution $(V_0, \ldots, V_{t'}, \pi)$ by another node set $V_{t'+1}$, such that the conditions above also are met for $t = t' + 1$. With $t' = 0$ as departure point, a sequence of extensions results in a broadcast forest corresponding to instance (G, S). Each extension identifies a matching from $V_{t'}$ to $V \setminus V_{t'}$, and all matched nodes in the latter set are included in $V_{t'+1}$. A key issue is how to determine the matching.

Since the goal is to minimize the time (number of extensions) needed to cover V, a *maximum cardinality* matching between $V_{t'}$ and $V \setminus V_{t'}$ is a natural choice. Lack of consideration of the matched nodes' capabilities to inform other nodes is however an unfavorable property. Each iteration of Alg. 2 rather sees $k \geq 1$ time periods ahead, and maximizes the total number of nodes in $V \setminus V_{t'}$ that can be informed in periods $t'+1, \ldots, t'+k$. Commitment is made for only one period, and the matched nodes are those that are informed in period $t'+1$ from some node in $V_{t'}$. The maximization problem in question is exactly the one addressed by model (2), where $V_{t'}$ is considered as sources, k the upper bound on the broadcast time, and the graph is G with all edges within $V_{t'}$ removed. Choosing $k = 1$ corresponds to the maximum matching option, whereas large values of k (e.g., $k = n - \sigma$) makes Alg. 2 an exact method.

Data: $G = (V, E), S \subseteq V, k \in \{1, \ldots, n - \sigma\}$

1 **for** $t' = 0, 1, \ldots$ **do**

2   **if** $S = V$ **return** t'

3     $x \leftarrow$ optimal solution to the instance of model (2) with $t = k$

4   **for** $\{u, v\} \in E$ *such that* $u \in S$, $v \in V \setminus S$, *and* $x^1_{uv} = 1$ **do**

5         $S \leftarrow S \cup \{v\}$

6   **end**

7 **end**

Algorithm 2: Construction of near-optimal solutions through sequences of matchings

In many instances, there are multiple options for selecting an optimal solution x (line 3). Two different sequences of solutions generated during the course of Alg. 2 may result in different broadcast times. Let us take a hypercube C_j as an example which has $\tau(C_j, S) = j$ for $\sigma = 1$ regardless of selection of the source node. Alg. 2 with $k = 1$ always finds $\tau(C_j, S)$ for $j = 1 \dots 3$. It also finds the optimum for $j \geq 4$ provided that the subgraph induced by the current S at the beginning of each iteration t' forms a hypercube $C_{t'}$. If this tie-breaking rule is violated, the algorithm may fail to find the optimum as demonstrated in C_4 (TODO: image or description in the text?)

Intuitively, the broadcast time found by Alg. 2 is likely to be closer to the optimum for larger k. This intuition is also confirmed by a vast majority of instances in the experimental section. Nevertheless, there are instances for which Alg. 2 with smaller k finds a better solution. One such example is a tree T on 17 nodes with one source s and $\deg(s) = 2$. The first neighbor n_1 of s is a root of a binomial tree B_2 of degree two. One of the leafs of B_2 is an endpoint of a path of length 3. The second neighbor n_2 of s is adjacent to a root of binomial tree B_3. Alg. 2 with $k = 3$ decides that $n_1 \in V_1$, $n_2 \in V_2$ and eventually finds $\tau T, \{s\}$. With $k = 5$, $n_1 \in V_2$, $n_2 \in V_1$, which results in a suboptimal solution. (TODO: is such an example useful? Should there be an illustration?)

Remark 8. *If $k = 1$, then the running time of Alg. 2 is $\mathcal{O}\left(n^{\frac{3}{2}}|E|\right)$, because the number of iterations is no more than n, and the maximum cardinality matching is found in $\mathcal{O}\left(\sqrt{n}|E|\right)$ time [17]. For fixed $k = 2$, the problem solved in each iteration is NP-hard [19].*

Remark 9. *If $k = 1$, no attempt is made to favor the nodes with many uninformed neighbors. To that end, maximum cardinality matching can be replaced by maximum* vertex-weight *matching (MVM), where each node $v \in V \setminus S$ is assigned the weight $1 + |\{u \in V \setminus S : \{u, v\} \in E\}|$. Other weights* *reflecting the capability of node u to inform other nodes in later periods could also be considered. The running time of Alg. 2 increases to* $\mathcal{O}(n^2|E|)$*, as MVM is solved in* $\mathcal{O}(n|E|)$ *time [5]. An approximate MVM-solution within* $\frac{2}{3}$ *of optimality is found in* $\mathcal{O}(|E| + n \log n)$ *time [5].*

6. Experimental Results

The aim of our experiments is to evaluate computational abilities of B&B applied to the studied model, and determine its usability on different instance types. Also, we assess the strength of upper and lower bounding methods discussed above, as well as continuous relaxations of the models.

In our experimental settings we consider. various types of instances, some of them are generated with randomness, others are standard graph classes from the literature (e.g., [9]) often used in cited works. The following paragraphs describe all of them.

Geometric graph on unit square. A geometric graph is a graph with vertices embedded in an n-dimensional Euclidean space. Two vertices are connected by an edge if the distance between them is smaller or equal to a given radius. The geometric graph instances, used in this paper, have vertices placed on the unite sphere. These vertices, or points on the unit square, have been created by normalizing 3 random numbers drawn from a Gaussian distribution. Graph tool's [24] geometric graph has been used to add edges between vertices that are closer to each other than a Euclidean radius r. The final step is ensuring the graph is connected. Initially, all vertices are marked as un-visited. Starting at an arbitrary vertex removing itself from un-visited and adding the vertices connected to its edges to-be-visited. Continue the process until the list of to-be-visited vertices are empty. If there are more un-visited vertices a new edge is added between the last visited vertex and the first vertex in the un-visited array. These edges are not subject to the radius constraint initially used. The process continues until all vertices are visited. The result is a graph where:

- if the radius is large enough, a grid is formed across the unit sphere. This mimics a satellite network, where the edges represent line-of-sight.
- for medium sized radiuses, the arising graphs often contains local clusters simulating a network.
- with yet smaller distances the graph would be a spanning tree.

Hypercube. The hypercube graph Q_n is the graph formed from the nodes and edges of a hypercube which is an n-dimensional generalization of a square ($n = 2$) and cube ($n = 3$). Every hypercube graph Q_n is a bipartite graph with 2^n nodes and $2^{n-1}n$ edges.

Cube-connected cycle. When each node in a hypercube Q_n is replaced by a cycle on n nodes, we obtain a cube connected cycle of order n. If a nodes is indexed by a pair of numbers (x, y), where $0 \leq x < 2^n$ and $0 \leq y < n$, each of the $n2^n$ nodes is connected to n neighbors $(x, (y + 1) \mod n)$, $(x, (y - 1) \mod n)$, and $(x \oplus 2y, y)$, where $\oplus$ denotes the exclusive or operation on the two binary operands.

Harary graph. The Harary graph $H_{k,n}$ is an example of a k-connected graph with n nodes and the smallest possible number of edges.

Debruijn graph. Each vertex of an n-dimensional Debruijn graph represents a binary number of length n. There is an oriented edge (b_1, b_2) if and only if $b_2 = b_1 << 1$ or $b_2 = b_1 << 1 + 1$. Note that we ignore the edge orientation and loops in order to obtain a valid MBT instance. This definition of Debuijn graphs on binary numbers is in fact a special case fitting our purposes. Generally, the underlying set can be any set of m symbols.

Shuffle exchange. Like in the case of Debruijn graph, vertices of a shuffle exchange graph represent binary string of length n. There is an edge between b_1 and b_2 if b_2 differs from b_1 in its last bit or if b_2 is obtained from b_1 by left or right cyclic shift.

Lattice. Lattice is a 4-connected grid graph of $m \times n$ vertices

We consider instances with 1 or 2 sources. Some of the instances are regular graphs, so the selection of a single source does not matter. One source is always the node with ID 0 except for lattices, where the ID is selected to correspond to instances in cited articles. In case of two sources, the second source has ID $\lfloor n/2 \rfloor$.

6.1. Comparison of upper and lower bounds

In the first set of experiments we study how does the strength of different methods depend on the parameter p. We generate random single and double-source instances of sizes 125, 250, 500 and 1000 with increasing p. Tabs. 1 and 2 summarize the results of randomly generated instances. Each entry is calculated as an average from 100 instances for a given n, p and a selected method. Results obtained for existing instances are stated in Tab. 3 where 20 instances are used for obtaining average objective function value for each instance category. Instead of the parameter p, the number of edges is given in the second column.

The column 'fib' consists of Fibonacci bounds. The values are rarely higher than the trivial logarithmic bound. Lower bound obtained from degree sequence in the column 'deg' is slightly better, but in majority of cases is far weaker than the LP bound, which is very tight in all considered experimental settings. Upper bounds UB-t, $t = 1, \ldots 4$ are obtained using Alg. 2, which takes t as one of its input parameters. In general, we observe that the higher t, the tighter upper bound is calculated. There are however some instances where it does not hold, particularly for instances with larger p, but the differences is very small, and can be explained as a coincidence.

We further observe that the span of bounds decreases with increasing p, within one instance size, and also increases for a constant p with increasing instance size. The experiments were not pursued for higher values of p, because it is very common that upper and lower bounds 'deg' and 'UB-4' coincide. This behavior is easy to There are more possibilities how to relay a signal in dense graphs as compared to sparser graphs. It is therefore likely that denser graphs have optimal broadcast time close to the lower bounds. The decreasing span of bounds is also noticeable with increasing instance size.

Note that due to the computational time restriction, results in columns 'OPT' are in fact lower bounds, whenever an interruption occurs. If the iterative decision procedure is interrupted, it is not guaranteed that the number of iterations performed is the optimal value. This is apparent in Tab. 3 in instances with $n = 640$ and $\sigma = 960$. The LP relaxation of the model provides a tighter lower bound in average than the integer model.

6.2. Solution time

Average solution time of instances from the previous section is reported in Tabs. 4 and 5.. The columns LP and OPT consists of solution times of solving LP relaxation and B&B applied to the ILP model, respectively. The column 'int.' consists of the percentage of instances of which solving was interrupted after 1 hour, before the optimal solution was found. Each interrupted instance counts with 3600 to the resulting average solution time. The values in the last column 'col.' are the numbers of instances in which the upper and lower bounds collapsed. I.e., how many instances have the lower bound 'deg' value equal to the upper bound 'UB-4' value. In these instances, the optimal solution is not pursued, because its objective function value is already known. Values in the second last column are therefore the proportion of interruptions in the total number of instances to which B&B is actually applied (in which the bounds do not collapse).

All investigated instances of size up to 500 nodes are solved to optimality within the imposed 1 hour time limit. The resulting times are not reported for sparse instance sets of size 125 and 250 in Tab 4 because in the worst case it takes 3 seconds to solve an instance to optimality, and interruptions or bound collapses never occur. Computation of B&B on instances of 1000 nodes is frequently interrupted before the optimal solution is found, particularly those with large p. Even thought the objective function value decreases with increasing p, and so the decision procedure needs to perform less iterations, overall solution time often increases. However, there are some exceptions in the monotonous growth of the solution time. Instances with $n = 1000$

Instance		Lower bounds				Upper bounds			
n	p	fib	deg	LP	OPT	UB-4	UB-3	UB-2	UB-1
	0.001	7.23	8.35	16.46	16.49	18.72	20.28	20.77	23.75
	0.002	7.21	8.20	13.97	13.98	15.80	17.18	17.21	19.50
125	0.004	7.03	8.08	11.44	11.49	13.07	14.06	14.30	16.05
	0.008	7.00	7.86	9.09	9.22	10.71	11.25	11.85	12.90
	0.016	7.00	7.59	7.94	8.01	9.11	9.42	9.83	10.58
	0.001	8.01	9.29	16.18	16.28	18.72	20.09	20.49	23.22
	0.002	8.00	9.17	12.87	12.93	15.23	16.19	16.84	18.78
250	0.004	8.00	8.92	10.06	10.33	12.23	12.87	13.46	14.70
	0.008	8.00	8.83	8.96	9.03	10.31	10.73	11.34	12.18
	0.016	8.00	8.24	8.24	8.24	9.40	9.42	9.71	10.51
	0.001	9.01	10.20	14.35	14.54	17.06	18.47	18.92	20.96
	0.002	9.00	9.95	11.19	11.55	13.69	14.54	15.11	16.53
500	0.004	9.00	9.86	10.09	10.12	11.81	12.12	12.69	13.57
	0.008	9.00	9.33	9.42	9.43	10.53	10.71	11.07	11.84
	0.016	9.00	9.00	9.00	9.00	10.00	9.94	10.00	11.06
	0.001	10.00	10.92	12.39	12.83	15.44	16.39	17.08	18.49
	0.002	10.00	10.77	11.06	11.07	12.99	13.38	14.00	14.98
1000	0.004	10.00	10.41	10.54	10.57	11.80	11.93	12.06	13.06
	0.008	10.00	10.01	10.01	10.01	11.02	11.02	11.02	12.04
	0.016	10.00	10.00	10.00	10.00	10.11	10.06	10.13	12.00

Table 1: Objective function values yielded by different methods of randomly generated instances with $\sigma = 1$

Instance		Lower bounds				Upper bounds			
n	p	fib	deg	LP	OPT	UB-4	UB-3	UB-2	UB-1
	0.001	6.26	7.10	13.33	13.35	15.20	16.70	16.88	19.23
	0.002	6.20	7.04	11.96	11.99	13.57	14.67	14.92	17.11
125	0.004	6.04	7.01	9.75	9.77	11.14	12.05	12.36	13.73
	0.008	6.00	6.96	7.78	7.91	9.19	9.71	10.21	11.38
	0.016	6.00	6.78	6.90	6.97	8.10	8.27	8.62	9.40
	0.001	7.03	8.14	14.59	14.63	16.92	18.07	18.40	20.67
	0.002	7.00	8.01	11.46	11.51	13.40	14.32	14.7	16.61
250	0.004	7.00	7.98	8.99	9.16	10.94	11.57	12.11	13.29
	0.008	7.00	7.86	7.98	7.99	9.22	9.61	10.08	10.89
	0.016	7.00	7.26	7.27	7.27	8.33	8.35	8.61	9.44
	0.001	8.00	9.04	12.82	12.96	15.47	16.47	17.05	18.86
	0.002	8.00	8.98	10.12	10.29	12.38	13.24	13.91	15.11
500	0.004	8.00	8.87	8.98	8.99	10.61	11.03	11.47	12.41
	0.008	8.00	8.54	8.56	8.56	9.50	9.67	9.96	10.79
	0.016	8.00	8.09	8.09	8.09	8.98	8.94	8.98	10.05
	0.001	9.00	10.00	11.23	11.44	14.02	14.79	15.56	16.95
	0.002	9.00	9.01	9.02	9.03	9.18	9.14	9.21	11.06
1000	0.004	9.00	9.51	9.59	9.61	10.83	10.97	11.10	12.06
	0.008	9.00	9.03	9.03	9.03	10.00	10.00	10.00	11.01
	0.016	9.00	9.00	9.00	9.00	9.12	9.07	9.15	11.00

Table 2: Objective function values yielded by different methods of randomly generated instances with $\sigma = 2$

Instance		Lower bounds				Upper bounds			
n	p	fib	deg	LP	OPT	UB-4	UB-3	UB-2	UB-1
160	240	8.00	8.00	8.05	8.05	10.10	10.90	11.20	12.55
160	320	8.00	8.00	8.00	8.00	9.30	9.70	10.05	10.85
320	480	9.00	9.00	9.05	9.20	11.60	12.60	12.85	14.75
320	640	9.00	9.00	9.00	9.00	10.40	10.85	11.20	12.25
640	960	10.00	10.00	10.11	10.00	13.06	13.94	14.67	16.39
640	1280	10.00	10.00	10.00	10.00	12.00	12.25	12.85	13.80

Table 3: Objective function values yielded by different methods of existing instances available online with $\sigma = 1$

and $p = 0.001$ in average take more time than instances with $n = 1000$ and $p = 0.002$. Similar behavior is exhibited in existing instances, where solution time tends to be longer for sparser graphs. It is also obvious and in accordance with the intuition that in denser graphs, the upper and lower bounds collapse more often.

7. Concluding Remarks

This work focuses on the minimum broadcast time problem and presents several techniques for determining lower bounds, upper bounds as well as optimal solutions. The main contribution consists in introducing an ILP model and a suitable solving method. Its LP relaxation provides a strong lower bound which often coincides with the optimum, and is stronger than other lower bounding methods presented in this work in vast majority of test instances. We consider various instance types and sizes both from existing

Instance		$\sigma = 1$				$\sigma = 2$			
n	p	LP	OPT	int.	col.	LP	OPT	int.	col.
125	0.016	0	1	0	4	0	1	0	9
250	0.008	1	4	0	2	0	3	0	1
	0.016	1	17	0	20	1	13	0	34
500	0.001	3	7	0	0	2	5	0	0
	0.002	2	26	0	0	2	22	0	0
	0.004	3	53	0	0	2	26	0	0
	0.008	6	214	0	13	4	101	0	0
	0.016	11	390	0	10	6	212	0	11
1000	0.001	12	1366	30	0	9	866	17	0
	0.002	20	814	17	0	24	378	0	0
	0.004	56	1637	33	10	69	1647	30	8
	0.008	101	3399	75	6	107	3378	74	30
	0.016	172	3600	100	89	235	3600	100	88

Table 4: Solution time in seconds of LP relaxation and B&B of randomly generated instances

n	$\lvert E\rvert$	LP	OPT	int.	col.
160	240	0	2	0	0
160	320	0	1	0	0
320	480	1	227	0	0
320	640	1	17	0	0
640	960	3	3155	85	0
640	1280	5	236	0	0

Table 5: Solution time in seconds of LP relaxation and B&B of existing instances available online for $\sigma = 1$

datasets and randomly generated.

We also develop and test a heuristic method that provides an upper bound. Large instances that are too time consuming for B&B can often be solved to optimality by comparing objective function values yielded by LP relaxation and upper bounding methods. The results of numerical experiments indicate that instances with many edges are more time consuming to solve by B&B. At the same time, however, the denser graph, the more likely are the upper and lower bounds to coincide.

There is a potential for the future research in developing stronger upper bounding algorithms and improving the existing ILP model. The problem definition assumes that a node transmits a signal to at most one neighbor in each time step. A further direction of the research considers a generalization of the problem, where nodes can transmit a signal to up to a certain number of neighbors at the same time.

References

[1] Bar-Noy, A., Guha, S., Naor, J., Schieber, B., Multicasting in heterogeneous networks, SIAM Journal on Computing, 30, 2, 347–358, 2000

[2] Bermond, J., Gargano, L., Perennes, S., Optimal sequential gossiping by short messages, Discrete Applied Mathematics 86(1998) 145–155

[3] Bermond, J., Gargano, L., Rescigno, A.A., Vaccaro, U., Fast gossiping by short messages, Lecture Notes in Computer Science 944(1995) 135–146.

[4] Chu, X., Chen, Y., Time division inter-satellite link topology generation problem: Modeling and solution, International Journal of Satellite Communications and Networking, 194 – 206, 36 (2017)

[5] Dobrian, F., Halappanavar, M., Pothen, A., Al–Herz, A., A 2/3-approximation algorithm for vertex weighted matching in bipartite graphs, SIAM Journal of Scientific Computing, 41(1)(2019) A566–A591.

[6] Elkin, M., Kortsarz, G., Sublogarithmic approximation for telephone multicast: path out of jungle, Symposium on Discrete Algorithms, 76 – 85 (2003)

[7] Farley, A. M., Proskurowski, A., Broadcasting in Trees with Multiple Originators, SIAM Journal on Algebraic Discrete Methods 2(4) (1981) 381–386.

[8] Fraigniaud, P., Lazard, E., Methods and problems of communication in usual networks, Discrete Applied Mathematics, 53, 1–3, 79–133, 1994.

[9] Niall Graham, Frank Harary, Hypercubes, shuffle-exchange graphs and de Bruijn digraphs, Mathematical and Computer Modelling, Volume 17, Issue 11, 1993, Pages 69-74,

[10] Grigni, M., Peleg, D., Tight bounds on minimum broadcast networks Networks, 207-222, 4 (1991)

[11] Harutyunyan, H.A., Liestman, A.L., Peters, J.G., Richards, D., Broadcasting and gossiping, In Thulasiraman, K., Arumugam, S., Brandstädt, A., Nishizeki, T. (eds): Handbook of Graph Theory, pp. 1477–1494. Chapman and Hall, 2013.

[12] Harutyunyan, H.A., Jimborean, C., New Heuristic for Message Broadcasting in Network, IEEE 28th International Conference on Advanced Information Networking and Application, 517 – 524, (2014)

[13] Harutyunyan, H.A., Shao, B., An efficient heuristic for broadcasting in networks, Journal of Parallel and Distributed Computing, 68 – 76, 66, 1 (2006)

[14] Harutyunyan, H.A., Wang, W, Broadcasting Algorithm Via Shortest Paths, 16th International Conference on Parallel and Distributed Systems (ICPADS), pp. 299–305, 2010, doi: 10.1109/ICPADS.2010.110.

[15] Hasson, Y., Sipper, M., A Novel Ant Algorithm for Solving the Minimum Broadcast Time Problem, International Conference on Parallel Problem Solving from Nature, 775 – 780 (2004)

[16] Hedetniemi, S.M., Hedetniemi, S.T., Liestman, A.L., A survey of gossiping and broadcasting in communication networks, Networks, 18(4):319–349, 1988.

[17] Hopcroft, J.E., Karp, R.M., An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs, SIAM Journal on Computing, 2(4)(1973) 225–23.

[18] Hromkovič, J., Klasing, R., Monien, B, Peine, R., Dissemination of Information in Interconnection Networks (Broadcasting & Gossiping), In: Du DZ., Hsu D.F. (eds) Combinatorial Network Theory, Applied Optimization, vol 1, pp. 125–212. Springer, Boston, MA, 1996.

[19] Jansen, K., Müller, H., The minimum broadcast time problem for several processor networks, Theoretical Computer Science, 69 – 85, 147 (1995)

[20] Kortsarz, G., Peleg, D., Approximation algorithms for minimum-time broadcast SIAM Journal on Discrete Mathematics, 401 – 427, 8, 3 (1995)

[21] McGarvey, R. G., Rieksts, B. Q., Ventura, J. A., Ahn, N., Binary linear programming models for robust broadcasting in communication networks, Discrete Applied Mathematics, 173 – 84, 204, (2016)

[22] Middendorf, M., Minimum broadcast time is NP-complete for 3-regular planar graphs and deadline 2, Information Processing Letters, 281 – 287, 46 (1993)

[23] Noe, T. D., Post, J. V., Primes in Fibonacci n-step and Lucas n-Step Sequences, J. Integer Seq. 8, Article 05.4.4, 2005.

[24] Peixoto, T. P., The graph-tool python library, 10.6084/M9.FIGSHARE.1164194.V13, (2014)

[25] Ravi, R., Rapid rumor ramification: Approximating the minimum broadcast time, Proceedings 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 1994, pp. 202-213, doi: 10.1109/SFCS.1994.365693.

[26] Scheuermann, P., Wu, G., Heuristic Algorithms for Broadcasting in Point-to-Point Computer Networks, IEEE Transactions on Computers, 804 – 811, 33, 9 (1984)

[27] http://steinlib.zib.de/download.php

[28] Slater, P. J., Cockayne, E. J., Hedetniemi, S.T., Information dissemination in Trees, SIAM Journal on Computing, 692 – 701, 10, 4 (1981)

[29] Wang, W., Heuristics for Message Broadcasting in Arbitrary Networks, Master thesis, Concordia University, Montréal, Québec, Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.633.5827&rep=rep1&type=pdf (2010)