Computing the broadcast time of a graph

Marika Ivanova*, Dag Haugland

Department of Informatics, 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 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

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

*Corresponding author

Email addresses: `Marika.Ivanova@uib.no` (Marika Ivanova), `Dag.Haugland@uib.no` (Dag Haugland)

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. 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 application such as communication among computer processors or telephone networks. Situations where the signals have to cover large distances typically assume sending the signal to one neighbor at the time. Inter-satellite communication networks thus constitute a prominent application area [1]. In particular, 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 Section 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 [16]. For bipartite planar graphs with maximum degree 3, NP-completeness persists even if $t = 2$ or if there is only one source [9]. When $t = 2$, the problem also remains NP-complete for cubic planar graphs [12], grid graphs with maximum degree 3, complete grid graphs, chordal graphs, and for split graphs [9]. The single-source variant of the decision version of MBT is NP-complete for grid graphs with maximum degree 4, and for chordal graphs [9]. The problem is known to be polynomial in trees [16]. Whether the problem is NP-complete for split graphs with a single source was stated as an open questions in [9], 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 [14] introduces a dynamic programming algorithm based on generating all maximum matchings in an induced bipartite graph. Additional contributions of [14] are heuristic approaches for near optimal broadcasting. From more recent works we mention [6], which 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 [1], where a greedy inexact method is proposed together with a mathematical programming model. Examples of additional efficient heuristics can be found e.g. in [7, 8, 17, 18].

Approximation algorithms for MBT are studied in [10]. The authors argue that methods presented in [14] provide no guarantee on the performance, and show that a wheel-graph is an example of an unfavourable instance. They introduce an $\mathcal{O}(\sqrt{n})$-additive approximation algorithm for broadcasting in general graphs with n nodes. They further provide approximation algorithms for several graph classes with small separators with approximation ratio proportional to the separator size times $\log n$. Throughout the text, the symbol log refers to the logarithm of base 2. Another algorithm with $\mathcal{O}\left(\frac{\log n}{\log \log n}\right)$-approximation ratio is given in [3]. Most of the works cited above consider a single source.

A related problem extensively studied in the literature is the minimum broadcast graph problem [5, 11]. 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. MgGarvey et al. [11] 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* $v \in V \setminus S$*,* $\{v, \pi(v)\} \in E$*,*
3. *for all* $k = 1, \ldots, t$ *and all* $v \in V_k$*,* $\pi(v) \in V_{k-1}$*, and*
4. *for all* $u, v \in V_k \setminus V_{k-1}$*,* $\pi(u) = \pi(v)$ *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 received 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 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, the corresponding* broadcast forest *is the digraph* $D = (V, A)$*, where* $A = \{(v, \pi(v)) : v \in V\}$*. 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 $\delta_G(v)$. The set of neighbors of $v \in V$ in G is denoted by $N_G(v)$. For a given subset $U \subseteq V$ of nodes, we define $N_G(U) = \bigcup_{v \in U} N_G(v)$. Likewise, $N^+_{\overrightarrow{G}}(v) = \left\{u \in V : (v, u) \in \overrightarrow{E}\right\}$ and $N^-_{\overrightarrow{G}}(v) = \left\{u \in V : (u, v) \in \overrightarrow{E}\right\}$ denote the neighbor sets of node v in $\overrightarrow{G}$, and $N_{\overrightarrow{G}}(v) = N^+_{\overrightarrow{G}}(v) \cup N^-_{\overrightarrow{G}}(v)$. Finally, we let $\delta^+_{\overrightarrow{G}}(v) = \left|N^+_{\overrightarrow{G}}(v)\right|$ and $\delta^-_{\overrightarrow{G}}(v) = \left|N^-_{\overrightarrow{G}}(v)\right|$ denote, respectively, the out-degree and the in-degree of $\overrightarrow{G}$, and we let $\delta_{\overrightarrow{G}}(v) = \delta^+_{\overrightarrow{G}}(v) + \delta^-_{\overrightarrow{G}}(v)$

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 ($(u, v) \in \overrightarrow{E}$, $k = 1, \ldots, t$)

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

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-variable that appear in the objective function. The 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 values of the decision variable induce 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

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

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

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

$$x \in \{0,1\}^{\vec{E} \times \{1,\dots,t\}}. \tag{2d}$$

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.

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 $(V_0,\dots,V_t,\pi)$ 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,*

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

Consider the m-step Fibonacci numbers $\{f_k^m\}_{k=1,2,\ldots}$ [13], a generalization of the well-known (2-step) Fibonacci numbers, defined by $f_k^m = 0$ for $k \le 0$, $f_1^m = 1$, and other terms according to the linear recurrence relation

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

The generalized Fibonacci numbers are instrumental in the derivation of a lower bound on $\tau(G,S)$, depending on the maximum node degree $d = \max\{\delta_G(v) : v \in V\}$ 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.

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

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

- 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 $\pi(v) = u$.

Clearly, such a solution exists, and $(V_0, \dots, V_t, \pi)$ is optimal if π also satisfies condition 2 of Definition 1. We thus have $t \le \tau(G, S)$. It remains to prove that $2\sigma \sum_{k=1}^{t-1} f_k^{d-1} < n \le 2\sigma \sum_{k=1}^{t} f_k^{d-1}$.

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

Consider a period $k \in \{2, \dots, t-1\}$. The assumptions on $(V_0, \dots, V_t, \pi)$ imply that π is a bijection from L_k to $L_{k-1} \cup \dots \cup L_{k-d+1}$. 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 $j \le 0$, we get $|L_k| = 2\sigma f_k^{d-1}$. Further, $|L_t| \le \sum_{j=1}^{d-1} |L_{t-j}| = 2\sigma f_t^{d-1}$. It follows that $2\sigma \sum_{k=1}^{t-1} f_k^{d-1} = \sum_{k=1}^{t-1} |L_k| = |V_{t-1}| < n = |V_t| \le \sum_{k=1}^{t} |L_k| \le 2\sigma \sum_{k=1}^{t} f_k^{d-1}$, which completes the proof. □

4.2. Continuous relaxations of integer programming models

For $t \in \mathbb{Z}_+$, define $\Omega(t) \subseteq [0,1]^{\overrightarrow{E} \times \{1,\dots,t\}}$ 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 $t^* = \min \{t \in \mathbb{Z}_+ : \Omega^{=} \neq \emptyset\}$ 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 $\Omega^{=}(t) \neq \emptyset$, $\zeta(t) \le t^* \le \tau(G, S)$.*

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 $\hat{x} \in [0,1]^{\overrightarrow{E} \times \{1,\dots,t+1\}}$ such that for all $(u, v) \in \overrightarrow{E}$, $\hat{x}_{uv}^{k} = x_{uv}^{k}$ $(k \le t)$ and $\hat{x}_{uv}^{t+1} = 0$. An analogous extension of z to $\hat{z} \in [0,1]^{\{1,\dots,t+1\}}$ 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 $\zeta(t+1) \le \zeta(t)$.

For $t \in \mathbb{Z}_+$ such that $\Omega^=(t) \neq \emptyset$, $t \geq t^*$ thus implies $\zeta(t) \leq \zeta(t^*)$. Since the only lower bounds on z_k in (1a)–(1h) are $\max_{v \in V \setminus S} \sum\limits_{u \in N(v)} x_{uv}^k \leq 1$ and $z_{k-1} \leq 1$, we get $\zeta(t) \leq \zeta(t^*) \leq t^*$. 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 of t 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 applies to a reformulation of* (1a)–(1h)*, where a unique integer variable y replaces $z_1, \ldots, z_t$, and the objective is to minimize y subject to the constraints $y \geq \sum\limits_{k=1}^{t} k \sum\limits_{u \in N(v)} x_{uv}^k$ $(v \in V \setminus S)$,* (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.

Recall from Section 2 that $D = (V, A)$ denotes the broadcast forest corresponding to $(V_0, \ldots, V_t, \pi)$. Conditions 1–4 of Definition 1 imply that

5. for all $v \in V$, $\delta_D^+(v) + \delta_D^-(v) \leq \delta_G(v)$.

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 ($\min_{v \in V} \delta_G(v) < \max_{v \in V} \delta_G(v)$).

Denote the source nodes $S = \{v_1, \dots, v_\sigma\}$ and the non-source nodes $V \setminus S = \{v_{\sigma+1}, \dots, v_n\}$, where $\delta_G(v_{\sigma+1}) \geq \delta_G(v_{\sigma+2}) \geq \cdots \geq \delta_G(v_n)$, and let $d_i = \delta_G(v_i)$ $(i = 1, \dots, n)$. Thus, $\{d_1, \dots, d_n\}$ 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 $(V_0, \dots, V_t, \pi)$ 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 $k = 1, \dots, t-1$, $u \in V_k \setminus V_{k-1}$ and $v \in V_{k+1} \setminus V_k$ implies $\delta_G(u) \geq \delta_G(v)$.

A rigorous proof of this follows next.

Lemma 6. *For a positive integer t and node sets $V_0 \subseteq V_1 \subseteq \cdots \subseteq V_t$, where $V_0 = S$, there exists a $\pi : V_t \setminus S \mapsto V_t$ such that $(V_0, \dots, V_t, \pi)$ satisfies conditions 3–5 if and only if, for all $k = 1, \dots, t$,*

$$|V_k \setminus S| \leq \sum_{v_i \in S} \min\{d_i, k\} + \sum_{\ell=1}^{k-1} \sum_{v_i \in V_\ell \setminus V_{\ell-1}} \min\{d_i - 1, k - \ell\}. \qquad (4)$$

Proof. Consider the broadcast forest D_k corresponding to some feasible $(V_0, \dots, V_k, \pi)$. Condition 4 implies that every node $v_i \in V_\ell \setminus V_{\ell-1}$ $(1 \leq \ell < k)$ has at most one child node in each of the node sets $V_{\ell+1}, \dots, V_k$, and thus no more than $k - \ell$ child nodes in D_k. Condition 5 implies that v_i has at most $d_i - 1$ child nodes in D_k. The corresponding bounds for a source $v_i \in V_0$ are k and d_i, respectively. Conversely, the conditions 4–5 are satisfied for some π if both bounds are respected at all nodes. The proof is complete by observing that condition 3 is satisfied if and only if the number $|V_k \setminus S|$ of non-root nodes in D_k is no larger than the total child node capacity $\sum_{v_i \in S} \min\{d_i, k\} + \sum_{\ell=1}^{k-1} \sum_{v_i \in V_\ell \setminus V_{\ell-1}} \min\{d_i - 1, k - \ell\}$ in D_k. □

Lemma 7. *The maximum value of $|V_t|$ over all $(V_0, \dots, V_t, \pi)$ satisfying $V_0 = S$ and conditions 3–5, is attained by some $(V_0, \dots, V_t, \pi)$ where $\min\{i : v_i \in V_k \setminus V_{k-1}\} > \max\{i : v_i \in V_{k-1}\}$ $(k = 1, \dots, t)$.*

Proof. Consider an arbitrary optimal solution $(V_0, \ldots, V_t, \pi)$, and assume that $v_i \in V_p \backslash V_{p-1}$, $v_j \in V_q \backslash V_{q-1}$, $i < j$, 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 $V'_k = V_k$ for $k = 0, \ldots, q-1, p, p+1, \ldots, t$, and $V'_k = (V_k \setminus \{v_j\}) \cup \{v_i\}$ for $k = q, \ldots, p-1$. Because $|V'_t| = |V_t|$, we only need to show that $(V'_0, \ldots, V'_t, \pi')$ is feasible for some π'.

By feasibility of $(V_0, \ldots, V_t, \pi)$, the only-if part of Lemma 6 implies that (4) holds for $k = 1, \ldots, t$. The if part of the same lemma shows that if (4) applies to $(V'_0, \ldots, V'_t)$, too, it is also feasible.

Let α_k and α'_k denote the right hand side of (4) corresponding to $(V_0, \ldots, V_t)$ and $(V'_0, \ldots, V'_t)$, respectively. Then $\alpha'_k = \alpha_k$ for $k = 1, \ldots, q-1$. For $k = q+1, \ldots, p$, we have $\alpha'_k = \alpha_k + \min\{d_i - 1, k-q\} - \min\{d_j - 1, k-q\} \geq \alpha_k$, since $d_i \geq d_j$. For $k = p+1, \ldots, t$, we have $\alpha'_k = \alpha_k + \min\{d_i - 1, k-q\} + \min\{d_j - 1, k-p\} - \min\{d_j - 1, k-q\} - \min\{d_i - 1, k-p\}$. It thus remains to show that $K_1 + L_1 \geq K_2 + L_2$, where

$$K_1 = \min\{d_i - 1, k-q\}, \quad L_1 = \min\{d_j - 1, k-p\},$$

and

$$K_2 = \min\{d_i - 1, k-p\}, \quad L_2 = \min\{d_j - 1, k-q\}.$$

We now investigate what values the difference

$$(K_1 + L_1) - (K_2 + L_2) \tag{5}$$

can attain. Observe that $K_1 \geq L_2$ and $K_2 \geq L_1$. If $K_2 > L_1$, there are two distinct cases: If $K_2 = d_i - 1 = K_1$, then also $L_2 = d_j - 1 = L_1$. Otherwise, $K_2 = k - p \leq K_1$, implying $L_1 = d_j - 1$, and since $d_j - 1 \leq k - p \leq k - q$, we have $L_2 = d_j - 1 = L_1$. We conclude that (5) always takes a non-negative value, which completes the proof. $\square$

Alg. 1 operates with the set $I(t)$ that consists of indices of nodes informed within the first t time steps. The function $a_i(t)$ determines the number of nodes informed by node v_i within t time steps, and finally, the set $F(t)$ consists of nodes that in period t can inform other nodes. That is, $v \in F(t)$ if v informed less than $\delta_G(v)$ other nodes, and v itself is also informed.

Proposition 8. *Alg. 1 returns a lower bound on $\tau(G, S)$.*

Proof. Follows from Lemma 7 and the subsequent discussion. $\square$

Data: $I(0) = \{1, \dots, \sigma\}, a_1(0) = \dots = a_n(0) = 0.$

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

  $F(t) = \{i \in I(t-1) : a_i(t-1) < d_i\},$

  $I(t) = \{1, \dots, |I(t-1)| + |F(t)|\},$

  $a_i(t) = a_i(t-1) + \begin{cases} 1, i \in F(t) \cup (I(t) \setminus I(t-1)) \\ 0, \text{ otherwise.} \end{cases}$

  if $|I(t)| = n$ **then**

    return t

  end

end

Algorithm 1: Lower bound exploiting distribution of degrees

5. Upper bounds

A knowledge of an upper bound $\bar{t}$ affects the number of variables in the studied model. Particularly in the decision version, the iterative approach can be terminated once the solution is found to be infeasible for broadcast time limit $\bar{t}-1$. The algorithm presented in this section iteratively constructs broadcast forest $T = (V_T, A_T)$, where in the last iteration $V_T = V$.

5.1. Restricted broadcast tree method

The following idea is based on the observation that at every time step, the maximum number of nodes that can be informed equals the size of maximum matching between already informed nodes and the rest. In the first iteration, the only informed nodes are the sources. Once a maximum matching is found, the set of informed nodes is extended by the endpoints of the matching that were not yet informed. This process is repeated until all nodes become informed. The number of iteration necessary to inform all nodes is then the upper bound on the broadcast time.

A maximum matching can be determined by an exact polynomial algorithm, or by solving an integer program. Even though the second option is not a polynomial method, the solution time is negligible for the considered instance sizes.

A maximum matching can be found with the help of the integer program (2) presented earlier with maximum time step t set to 1. This model can be conveniently employed for an extension of this approach by increasing the maximum time step. A solution in each iteration gives a maximum number of newly informed nodes within the imposed time limit by finding a set of node disjoint broadcast trees rooted at nodes informed in previous iterations. In this way we use the principle of rolling horizon method known from planning and scheduling. For the next iteration, only some nodes are selected for extending the set of informed nodes, typically only the ones reachable in a single time step, thus a matching. The steps are expressed by a pseudocode in Alg. 2.

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

$V_T \leftarrow S, A_T \leftarrow \emptyset$;
$\bar{t} \leftarrow 0$;
while $V_T \neq V$ **do**
  $S \leftarrow V_T$;
  $x \leftarrow$ optimal solution to model (2);
  $V_T \leftarrow V_T \cup \{v \in V \setminus V_T : x^1_{uv} = 1, u \in V_T\}$;
  $A_T \leftarrow A_T \cup \{(u, v) \in V_T \times V \setminus V_T : x^1_{uv} = 1\}$;
  $\bar{t} \leftarrow \bar{t} + 1$;
end
return $\bar{t}$;

Algorithm 2: A method for determining an upper bound based on iterative search for trees

For calculating an upper bound, it is not necessary to store node and arcs sets V_T and A_T in Alg 2, but it is essential when knowledge of the actual broadcast tree is desirable.

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 some of the experimental settings we use randomly generated instances. The generating procedure takes a number of nodes and a parameter p as an input. First, it generates a random tree with the given number of nodes. It then iterates over all pairs of nodes that are not yet connected by an edge, and places an edge between them with the probability p.

Apart from randomly generated instances, we also evaluate data sets of various sizes and densities available online [15]. These existing collections of instances are used as benchmarks for algorithms for the minimum Steiner tree problem. We evaluate datasets named “I160”, “I320” and “I640” with selected edge cardinalities stated in the tables with results. The only source node in these instances is always the node with id 1. In each instance category, 20 graphs are available.

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 explain. 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 iter-

Instance		Lower bounds				Upper bounds			
n	p	fib	deg	LP	OPT	UB-4	UB-3	UB-2	UB-1
125	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
	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
250	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
	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
500	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
	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
1000	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
	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 $\sigma = 1$

ative 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.

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 $\sigma = 1$

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 $n = 1000$ 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 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

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 $\sigma = 1$

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.

[1] 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)

[2] Cormen, T. H., Leiserson, C. E., Rivest, R. L, Introduction to Algorithms, MIT Press, 401 – 402, 1990.

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

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

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

[6] 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)

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

[8] 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)

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

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

[11] 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)

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

[13] 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.

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

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

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

[17] 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)

[18] Jimborean, C., New Heuristics for Message Broadcasting in Arbitrary Networks, Master thesis, Concordia University, Montréal, Québec, Retrieved from https://spectrum.library.concordia.ca/977717/1/Jimborean_MCompSc_F2013.pdf (2013)