INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center St. • Suite 600 • Berkeley, California 94704-1198 • (510) 666-2900 • FAX (510) 666-2956

Broadcasting Time cannot be Approximated within a Factor of $\frac{57}{56} - \epsilon$

Christian Schindelhauer*

TR-00-02

March 2000

Abstract.

In the beginning the information is available only at some sources of a given network. The aim is to inform all nodes of the given network. Therefore, every node can inform its neighborhood sequentially and newly informed nodes can proceed in parallel within their neighborhoods. The process of informing one node needs one time unit. The broadcasting problem is to compute the minimum length of such a broadcasting schedule.

The computational complexity of broadcasting is investigated and for the first time a constant lower inapproximability bound is stated, i.e. it is $\mathcal{NP}$-hard to distinguish between graphs with broadcasting time smaller than b and larger than $(\frac{57}{56} - \epsilon)b$ for any $\epsilon > 0$. This improves on the lower bounds known for multiple and single source broadcasting, which could only state that it is $\mathcal{NP}$-hard to distinguish between graphs with broadcasting time b and $b+1$, for any $b \geq 3$. This statement is proven by reduction from E3-SAT, the analysis of which needs a carefully designed book-keeping and counting argument.

* Thanks to Andreas Jakoby, Rüdiger Reischuk and Antonio Piccolboni. Parts of this work are supported by a stipend of the "Gemeinsames Hochschulsonderprogramm III von Bund und Länder" through the DAAD. email: `schindel@icsi.berkeley.edu`

1 Introduction

Broadcasting reflects the sequential and parallel aspects of disseminating information in a network. In the beginning the information is available only at some *sources*. The aim is to inform all nodes of the given network. Therefore, every node can inform its neighborhood sequentially and newly informed nodes can proceed in parallel within their neighborhoods. The process of informing one node needs one time unit. Thus, the game is played in rounds (For a formal definition see next section).

The *multiple source broadcasting problem* **(MB)** is the following: Given a graph with multiple sources, compute its minimum broadcasting time. It is known to be NP-hard [GaJo79] and this is even the case for constant broadcasting time, like 3 [JRS96] or 2 [Midd93]. Yet it is not justified to claim that this entails a lower bound for approximating broadcasting within a factor of $\frac{3}{2}$, since no family of graphs is known whose broadcasting time is $2b$ and which cannot be approximated within $3b$ for any $b > 1$. For many other approximation problems it is an easy task to define a *padding* argument that transfers inapproximability bounds to larger instances with larger values. E.g. the approximation problem of satisfiability [Håst97] is satisfy as many clauses as possible. Consider a family of Boolean formulas for which it is $\mathcal{NP}$-hard to distinguish between formulas with s satisfiable clauses and αs satisfiable clauses. Then, it is an easy task to create a formula of with $2s$ satisfiable clauses with this property. Another example for a paddable approximation problem is tree-width. Here, a padding technique replaces every node with a clique and transfers small differences of tree-width of graphs to higher values preserving the lower bound property [BGH91]. Therefore, the lower bound factor of $\frac{3}{2}$ for broadcasting time can be stated, if such a padding technique could be discovered for MB.

In the literature the broadcasting problem has often been considered only for one informed source at the beginning, here called *single source broadcasting problem* **(SB)**. Note that the broadcasting time $\boldsymbol{b(G)}$ is at least $\log_2 |V|$ for a graph $G = (V, E)$, since during each round the number of informed vertices can at most double. The minimal graph providing this lower bound is a *binomial tree* F_n [HHL88]: F_0 consists of a single node and F_{n+1} consists of disjunct subtrees $F_0, \ldots, F_n$ whose roots $r_0, \ldots, r_n$ are connected to the new root r_{n+1}. Also the *hyper-cube* $C_n = \{\{0,1\}^n\}, \{\{w0v, w1v\} \mid w, v \in \{0,1\}^*\}$ has this minimum broadcasting time since binomial trees can be embedded.

The upper bound on $b(G)$ is $|V| - 1$, which is needed for the chain graph representing maximum sequential delay (fig. 1) and the star graph (fig. 2) producing maximum parallel delay. Hence, the topology of the processor network highly influences the broadcasting time. For the broadcasting behavior of these and other graphs, see [LP88,BHLP92]. An overview over broadcasting is given in [HHL88].

The computational complexity of single source broadcasting has been studied for a long time and its decision problem (SBD) (decide whether the broadcasting time is less or equal to a given deadline T_0) is $\mathcal{NP}$-complete [GaJo79,SCH81]. Slater et al. also show, for the special case of trees, that a divide-and-conquer strategy leads to a linear algorithm. In [JRS96] this result is generalized for graphs with a small tree-width according to a tree decomposition of the edges. On the other hand, SBD is $\mathcal{NP}$-complete even for the restricted case of ternary planar graphs or ternary graphs with logarithmic depth [JRS96].

The best known polynomial-time approximation algorithm for SB has a factor of $O\left(\frac{\log^2|V|}{\log\log|V|}\right)$ for a graph $G = (V, E)$, and $2B$ for B-bounded-degree graphs [Ravi94]. SB is approximable within $O\left(\frac{\log|V|}{\log\log|V|}\right)$ if $G = (V, E)$ has bounded tree-width with respect to the standard tree decomposition [MRSR95].

All proofs of the hardness of broadcasting known so far could establish only a difference of one time unit between the broadcasting time $b(G)$ and a polynomial time computable upper bound, no matter whether one considers single or multiple sources. For multiple sources a padding technique (as described above) would establish a factor of $\frac{3}{2}$ at once. For a single source even this does not help for a constant lower bound larger than 1. The best lower bound so far is a factor of $1 + \frac{1}{\Theta(\log|V|)}$ [JRS96].

In this paper we do not use a *padding* technique that preserves approximation factors, neither do we use PCP techniques. The proof consists in a sophisticated reduction from E3-SAT to single source broadcasting, such that the inapproximability of satisfiability is inherited by the reduction graph. Of course this lower bound also applies to MB. So, recent developments in the area of *probabilistic checkable proofs* enable this breakthrough [Håst97].

The reduction-graph has a very high degree at the source (and only there). Thus, a good broadcasting strategy has to make most of its choices there and this can be shown to be equivalent to choosing the assignment of an E3-CNF-formula. A careful book-keeping of the broadcasting times of certain nodes representing literals and clauses gives the lower bound of $\frac{57}{56} - \epsilon$.

The paper is organized as follows. After formalizing some relevant notation, the general lower bound is stated and proved. In the last section, conclusion, these results are summarized and possible further developments are discussed.

2 Notation

All problem instances are undirected graphs. However, for the broadcast schedule we use directed edges to indicate the direction of information flow. The undirected versions of these edges form a subset of all edges of the given graph.

Definition 1 (single source broadcasting) *Let $G = (V, E)$ be an undirected graph with a vertex $v_0 \in V$, the* **source**. *The task is to compute the* **broadcasting time $b(G, v_0)$**, *the minimum length T of a* **broadcast schedule** *S, that is a sequence of sets of directed edges $S = (E_1, E_2, \ldots, E_{T-1}, E_T)$. Their nodes are in the sets $V_0 = \{v_0\}$, $V_T = V$, where for $i > 0$ we define $V_i := V_{i-1} \cup \{v \mid (u, v) \in E_i \text{ and } u \in V_{i-1}\}$. A broadcast schedule S fulfills the properties*

1. $E_i \subseteq \{ (u, v) \mid u \in V_{i-1}, \{u, v\} \in E \}$ *and*
2. $\forall u \in V_{i-1} : |E_i \cap (\{u\} \times V)| \leq 1$.

The set of nodes V_i have received the broadcast information by round i. For an optimal schedule with length T_0 the set V_{T_0} is the first to include all nodes of the network. E_i is the set of edges used for sending information at round i. Each processor $u \in V_{i-1}$ can use at most one of its outgoing edges in every round.

Definition 2 *Let S be a broadcast schedule for (G, V_0) where $G = (V, E)$. The broadcasting time of a node $v \in V$ is defined as $b_S(v) = \min\{i \mid v \in V_i\}$. A broadcast schedule S is called* **busy** *if it holds:*

1. $\forall \{v, w\} \in E \ : \ b_S(w) > b_S(v) + 1 \quad \Rightarrow \quad \exists w' \in V \ : \ (v, w') \in E_{b_S(w)-1}$
2. $\forall v \in V \setminus \{v_0\} \ : \ |\bigcup_i E_i \ \cap (V \times \{u\})| = 1$

In a busy broadcasting schedule every processor tries to inform a neighbor in every step starting from the moment it is informed. When this fails it stops. By this time, all its neighbors are informed. Furthermore, every node is informed only once.

Lemma 1 *Every broadcasting schedule S can be replaced by a busy schedule S' without delaying any broadcasting time of a node, i.e. $\forall v \in V \ : \ b_{S'}(v) \leq b_S(v)$. Moreover, given S this schedule S' can be computed in polynomial time.*

Proof: The algorithm starts at the source and applies to every node v a local optimization algorithm, called `local-busy-maker`.

```
procedure local-busy-maker(v)
{
 for i = 1 to degree of v do
     if v informs no neighbor in time b_S(v) + i then
         if there is a neighbor w' left with b_S(w') > b_S(v) + i then
             for all v' ∈ V do
                 Remove {v', w'} from all sets E_1, E_2, ...
             od
             Insert {v, w'} in E_{b_S(v)+i}
         fi
     fi
 od
}
```

This local optimization algorithm improves the schedule S without delaying or preventing any information. It affects only nodes in the schedule that have higher broadcasting time than v. After this procedure the node v is busy in the resulting schedule. We apply this algorithm to each node as follows.

```
procedure busy-maker(G, v_0, S)
{
 Remove all edges that inform already informed nodes
 Select among edges that inform a node at the same time
       an arbitrary one and remove the others
 for i = 0 to b_S(G) do
     M := {v | b_S(v) = i}
     for all nodes v in M do
         local-busy-maker(v)
     od
```

```
 od
}
```

∎

From now on, every schedule is considered to be busy.

A chain is defined by $C_n = (\{v_1, \ldots, v_n\}, \{\{v_i, v_{i+1}\}\})$ (fig. 1), and a star by $S_n = (\{v_1, \ldots, v_n\},$
$\{\{v_1, v_i\} \mid i > 1\})$ (fig. 2).

Fact 1 *There is only one busy broadcast strategy that informs a chain with k interior nodes. Let its ends v, w be informed in time $b_w - k \leq b_v \leq b_w$. Then the chain is informed in time $\lceil (b_v + b_w + k)/2 \rceil$ (We assume that the ends have no other obligations for informing other nodes).*

There are $n!$ busy broadcast schedules for the star S_n that describe all permutations of $\{1, \ldots, n\}$ by $(b_S(v_1), \ldots, b_S(v_n))$.

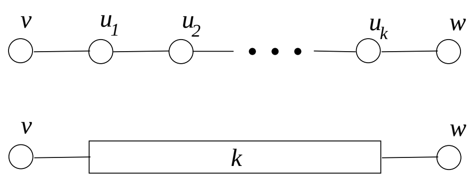

Fig. 1. The chain and its symbol.

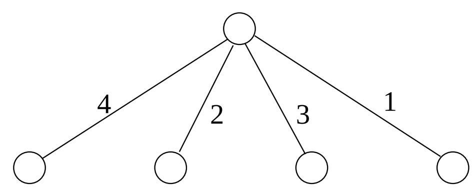

Fig. 2. The star and a busy schedule.

3 The General Lower Bound

In this section we present a reduction from E3-SAT to SB and show how to prove the constant inapproximability factor.

Let F be a 3-CNF with clauses $c_1, \ldots, c_m$ and variables $x_1, \ldots, x_n$. Let $a(i)$ denote the number of occurrences of the positive literal x_i in F. We can assume that every variable occurs as often positive as negated in F, since in the proof of theorem 1 this property is fulfilled. Let $\delta := 2\ell \cdot m$, where $m := \sum_{i=1}^{n} a(i)$ with ℓ being a large enough number to be chosen later on. Note that for the number of clauses M it holds: $M = 2m/3$

We reduce F to an undirected graph $G_{F,\ell}$ with maximum degree δ which occurs at the source v_0. v_0 and its δ neighbors form a star S_δ whose leaves $x_{i,j,k}^b$ for $b \in \{0,1\}$, $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, a(i)\}$, and $k \in \{1, \ldots, \ell\}$ are nodes of ℓ disjunct isomorphic subgraphs $G_1, \ldots, G_\ell$. The leaves of the subgraph S_δ are called **literal nodes**. A subgraph G_k contains literal nodes $x_{i,j,k}^b$, representing the literal x_i^b ($x_i^1 = x_i$, $x_i^0 = \overline{x_i}$).

Now, we give a complete description of a subgraph $G' = G_k$. As a basic tool for the construction of G' we use a chain $C_k(v, w)$ starting at nodes v and ending at w with k interior nodes that are not incident to any other edge of the graph. For the literal nodes corresponding to a variable x_i in G_1 we insert the chains $C_\delta(x_{i,j,k}^0, x_{i,j',k}^1)$ for all $i \in \{1, \ldots, n\}$ and $j, j' \in \{1, \ldots, a(i)\}\}$.

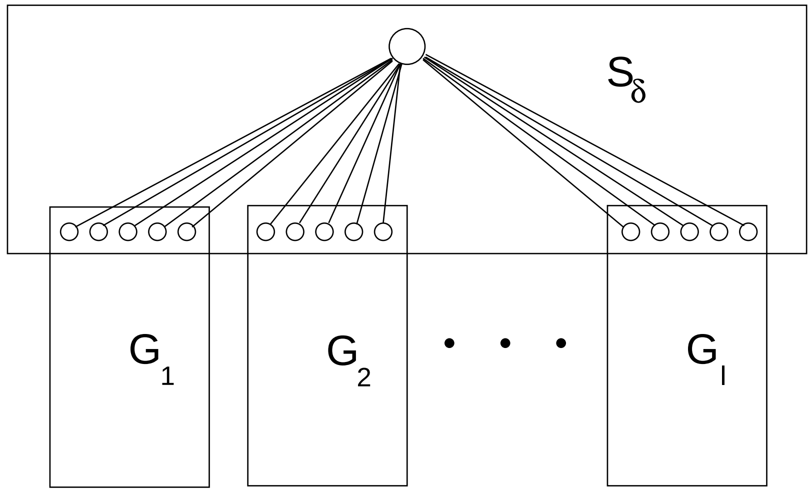

Fig. 3. The reduction graph $G_{F,\ell}$.

For every clause c_ν we introduce a **clause node** $c_{\nu,k}$ which is connected via three chains $C_{\delta/2}(c_{\nu,k}, x^{b_\rho}_{i_\rho,j_\rho,k})$ ($\rho \in \{1,2,3\}$) of length $\delta/2$ to its corresponding literal nodes $x^{b_1}_{i_1,j_1,k}, x^{b_2}_{i_2,j_2,k}, x^{b_3}_{i_3,j_3,k}$ such that every literal node is connected only to one clause node. This completes the construction of G'.

The main idea of the construction is that the assignment of a variable $x_i = \alpha$ roughly shows when a node has to be informed. We call a literal x_i in a subgraph G_k **coherent**, iff it holds: $\exists \alpha \in \{0,1\}$ $\forall j \in \{1,\ldots,a(i)\}$ $b_S(x^{\alpha}_{i,j,k}) \leq \frac{\delta}{2}$ and $b_S(x^{\neg\alpha}_{i,j,k}) > \frac{\delta}{2}$.

Lemma 2 *If F is satisfiable, then $b(G_{F,\ell}, v_0) \leq \delta + 2m + 5$.*

Proof: We describe a broadcasting schedule S for $(G_{F,\ell}, \{v_0\})$, where every literal is coherent.

The busy schedule S informs all literal nodes directly by v_0. Let $\alpha_1,\ldots,\alpha_n$ be a satisfying assignment of F. The literal nodes $x^{\alpha_i}_{i,j,k}$ of graph G_k are informed in the time period $(k-1)m+1,\ldots,km$. The literal nodes $x^{\overline{\alpha_i}}_{i,j,k}$ are informed in the time period $\delta - km + 1,\ldots,\delta-(k-1)m$. m is a trivial upper bound for the degree at a literal node. So, the chains between two literal nodes can be informed in time $\delta + 2m + 1$. A clause can be informed in time $km + \delta/2 + 1$ by an assigned literal node of the first type, which always exists since $\alpha_1,\ldots,\alpha_n$ satisfies F. Note that all literal nodes corresponding to the second type are informed within $\delta - (k-1)m$. So the chains between those and the clause node are informed in time $\delta + m/2 + 1$. ∎

For proving a lower bound for non-satisfying formulas we need the following lemma.

Lemma 3 *Let S be a busy broadcasting schedule for $G_{F,\ell}$. Then,*

1. *every literal node will be informed directly from the source v_0, and*
2. *for $c_i = x^{\alpha_1}_{j_1} \vee x^{\alpha_2}_{j_2} \vee x^{\alpha_3}_{j_3}$: $b_S(c_i) > \frac{\delta}{2} + \min_i\{b_S(x^{\alpha_i}_{j_i})\}$.*

Proof:

1. Every path between two literal nodes that avoids v_0 has at least length $\delta + 1$. By Fact 1 even the first informed literal nodes has no way to inform any other literal node before time point δ, which is the last time a literal node is going to be informed by v_0.
2. follows by 1.

∎

If we assume that only one clause per Boolean formula is not satisfied, this lemma only implies that if F is not satisfiable, then $b(G_{F,\ell}, \{v_0\}) > \delta + \ell$. This already improves the best known lower bounds for approximability. But we can do much better if we take into account that every non-satisfied clause of an assignment causes an in-coherent literal, and therefore the number of nodes informed too late increases proportionally to the number of unsatisfied clauses.

$$\mathrm{F} = \underbrace{(x_1 \vee \overline{x}_2 \vee x_3)}_{C_1} \wedge \underbrace{(x_2 \vee \overline{x}_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x}_1 \vee x_3 \vee \overline{x}_4)}_{C_3} \wedge \circ \circ \circ$$

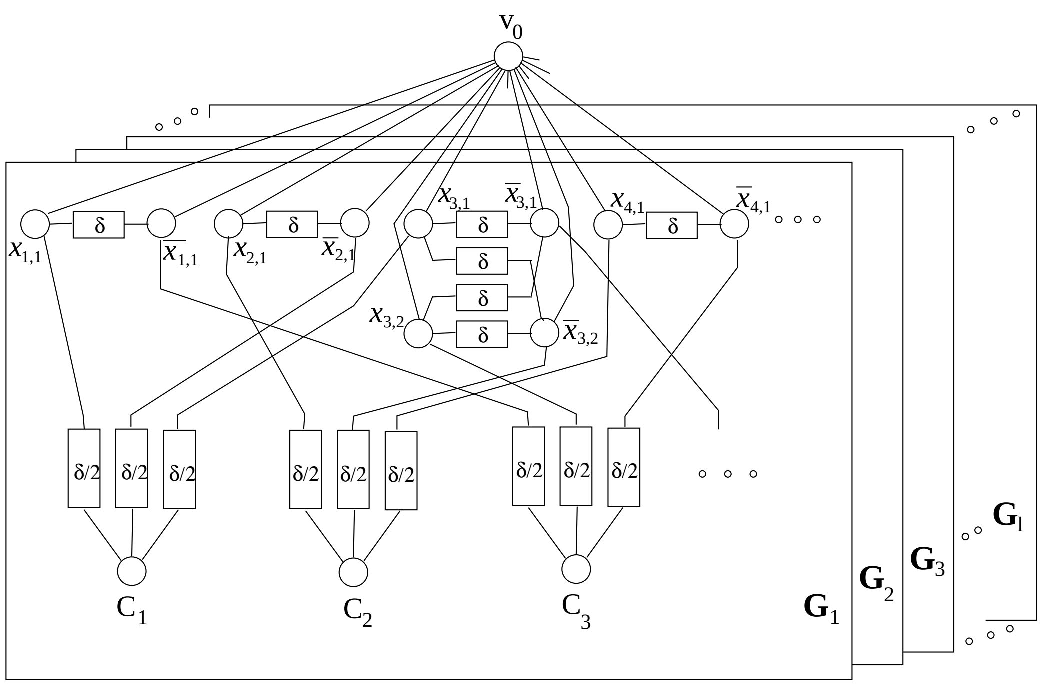

Fig. 4. Transforming F into $G_{F,\ell}$

E3-SAT denotes the satisfiability problem of Boolean CNF-formulas where in each clause exactly three literals appear.

Theorem 1 *[Håst97] For any $\epsilon > 0$ it is $\mathcal{NP}$-hard to distinguish satisfiable E3-SAT formulas from E3-SAT formulas for which only a fraction $7/8 + \epsilon$ of the clauses can be satisfied, unless $\mathcal{P} = \mathcal{NP}$.*

To transfer this result to broadcasting, we consider a busy schedule S for graph $G_{F,\ell}$ and define a corresponding assignment for F. Then, we categorize every literal node as *high*, *low* or *neutral*, depending on whether the literal node is coherently assigned and whether it is delayed (later informed than $\delta/2$). Furthermore we classify some clause nodes as *high*. Every unsatisfied clause of the E3-SAT-formula F increases the number of high literals. Besides this, high and low literal nodes come in pairs, but possibly in different subgraphs G_i and G_j. The overall number of the high nodes will be larger than those of the low nodes. A large difference will result in a good lower bound.

Theorem 2 *For every $\epsilon > 0$ there exist graphs with broadcasting time at most b such that it is $\mathcal{NP}$-hard to distinguish those from graphs with broadcasting time at least $(\frac{57}{56} - \epsilon)b$.*

Proof: Consider an unsatisfiable E3-SAT-formula F, the above described graph $G_{F,\ell}$ and a busy broadcasting schedule S on it. The schedule defines for each subgraph G_k an assignment $x_{1,k}, \ldots, x_{n,k}$
$\in \{0,1\}^n$ as follows. We assign the variable $x_{i,k} = \alpha$ if the number of literal nodes with $b_S(x^{\alpha}_{i,j,k}) \leq \delta/2$ is larger than those with $b_S(x^{\overline{\alpha}}_{i,j,k}) \leq \delta/2$. If both numbers are equal, wlog let $x_{i,k} = 0$.

1. A literal node is **coherently** assigned, iff $b_S(c^{\alpha}_{i,j,k}) \leq \delta/2 \;\Leftrightarrow\; x_{i,k} = \alpha$. All coherently assigned literal nodes are **neutral**.
2. A literal node $x^{\alpha}_{i,j,k}$ is **high** if it is not coherently assigned and delayed, i.e. $x_{i,j,k} = \alpha$ and $b_S(x^{\alpha}_{i,j,k}) = \delta/2 + \epsilon_1$ for $\epsilon_1 > 0$.
   Every high literal node can be matched to a neutral delayed literal node $x^{\overline{\alpha}}_{i,j',k}$, i.e. $b_S(^{\overline{\alpha}}_{i,j',k}) = \delta/2 + \epsilon_2$ for $\epsilon_2 > 0$. Then Fact 1 show that the chain between them can be informed in time $\delta + \frac{\epsilon_1+\epsilon_2}{2}$ at the earliest.
3. A literal node $x^{\alpha}_{i,j}$ is **low** if it is not coherently assigned and not delayed, i.e. $x_{i,j,k} = \overline{\alpha}$ and $b_S(x^{\alpha}_{i,j,k}) \leq \delta/2$.
4. A clause node c is **high**, if all its three connected literal nodes are coherent and delayed, i.e. $\forall i \in \{1,2,3\}$ $b_S(x^{\alpha_i}_{j_i}) = \delta/2 + \epsilon_i$, for $e_i > 0$. Since Lemma 3, this clause node will be informed not earlier than $\delta + \min\{\epsilon_1, \epsilon_2, \epsilon_3\}$ and the chain to the most delayed literal node will be informed at $\delta + (\min\{\epsilon_1, \epsilon_2, \epsilon_3\} + \max\{\epsilon_1, \epsilon_2, \epsilon_3\})/2$ at the earliest.
5. All other clause nodes are **neutral**.

Suppose all literal nodes are coherent, then the number of high clauses equals ℓ times the number of unsatisfied clauses of F. This is the easy case. In general we have to face a "very" non-coherent schedule and take high and low literal nodes into account.

Lemma 4 *Let q be the number of low literal nodes, p the number of high literal nodes, and p' the number of high clause nodes. Then the following holds*

1. $p = q$,
2. $b_S(G_{F,\ell}, v_0) \;\geq\; \delta + p$,

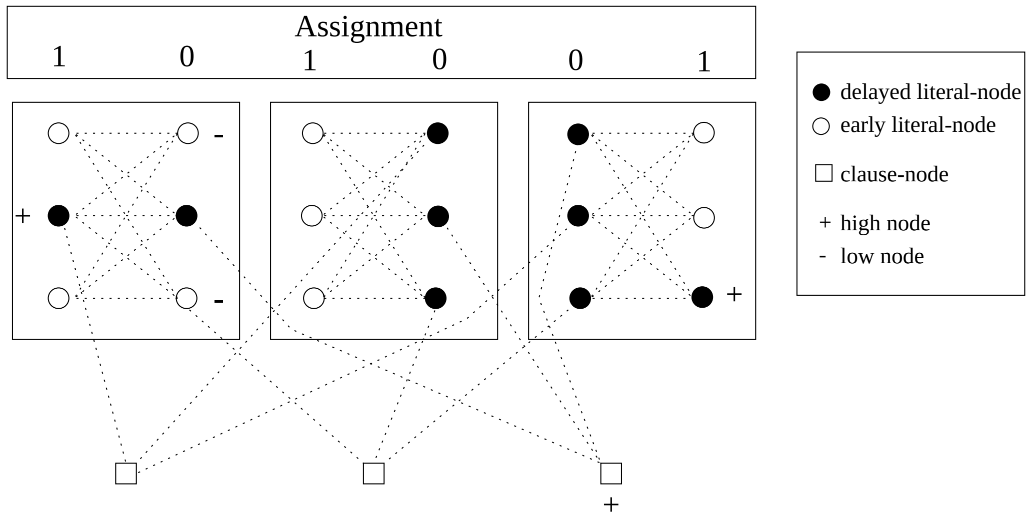

Fig. 5. High and low literal nodes. The middle literal is the only coherent one.

3. $b_S(G_{F,\ell}, v_0) \ \geq \ \delta + (p + 3p')/2$.

Proof:

1. Consider the set of nodes $x^{\alpha}_{i,j,k}$, for $j \in \{1, \ldots, a(i)\}$ and $\alpha \in \{0, 1\}$. For this set let $p_{i,k}$ be the number of high nodes, $q_{i,k}$ the number of low nodes and $r_{i,k}$ the number of nodes with time greater than $\delta/2$. By the definition of high and low nodes the following holds for all $i \in \{1, \ldots, n\}$, $k \in \{1, \ldots, \ell\}$:

$$r_{i,k} - p_{i,k} + q_{i,k} = a(i) \ .$$

From Fact 1 and Lemma 3 we know that half of the literal nodes are informed within $\delta/2$ and the rest later on:

$$\sum_{i,k} r_{i,k} = \delta/2 = \sum_{i,k} a(i) \ ,$$

Then, it follows that:

$$q - p = \sum_{i,k} r_{i,k} - p_{i,k} + q_{i,k} - a(i) = 0 \ .$$

2. Note that each of the p high (delayed) literal node $x^{\alpha}_{i,j,k}$ can be matched to a coherent delayed literal node $x^{\overline{\alpha}}_{i,j',k}$. Furthermore, a chain of length δ has to be informed by these nodes. If the latest of the high nodes and its partners is informed at time $\delta/2 + \epsilon$, then fact 1 shows that the chain cannot be informed earlier than $\delta + \epsilon/2$.
The broadcasting time of all literal nodes is different. Therefore it holds $\epsilon \geq 2p$, proving $b_S(G_{F,\ell}, v_0) \ \geq \ \delta + p$.
3. Every clause node is connected to three neutral delayed literal nodes. The task to inform all chains to the three literal nodes is done at time $\delta + \epsilon'/2$ at the earliest, if $\delta/2 + \epsilon'$ was the broadcasting time of the latest literal node. For p' high clause nodes, there are

$3p'$ corresponding delayed neutral literal nodes and, besides that, there are p delayed high literal nodes (whose matched partners may intersect with the $3p'$ neutral literal nodes). Nevertheless, the latest high literal node with broadcasting time $\delta/2+\epsilon''$ causes a broadcast time on the chain to a neutral delayed literal-node of at least $\delta+\epsilon''/2$.

From both groups consider the most delayed literal node $v_{\max}$. Since every literal node has a different broadcasting time it holds $\epsilon'' \geq 3p'+p$, and thus $b_S(v_{\max}) \geq \delta+(3p'+p)/2$. ∎

Suppose all clauses are satisfiable. Then Lemma 2 gives an upper bound for the optimal broadcasting time of $b(G_{F,\ell}, v_0) \;\leq\; \delta + 2m + 5$.

Let M be the number of clauses in F and assume that at least κM clauses are unsatisfied for every assignment. Consider a clause node that represents an unsatisfied clause with respect to the assignment which is induced by the broadcast schedule. Then at least one of the following cases can be observed:

- The clause node is high, i.e. its three literal nodes are coherently assigned.
- The clause node is neutral and one of its three literal nodes is low.
- The clause node is neutral and one of its three literal nodes is high.

Since each literal node is chained to one clause node only, this implies

$$\kappa \cdot \ell \cdot M \;\leq\; p' + p + q \;=\; p' + 2p \;.$$

Assume $p \geq 3p'$, then it follows that $p \geq \frac{3}{7}(2p+p')$. Then it holds for the broadcasting time of any busy schedule S

$$b_S(G_{F,\ell}, v_0) \;\geq\; \delta + p \;\geq\; \delta + \frac{3}{7}(p'+2p) \;.$$

On the other hand, if $p < 3p'$, then $\frac{1}{2}(p+3p') \geq \frac{3}{7}(2p+p')$ and

$$b_S(G_{F,\ell}, v_0) \;\geq\; \delta + \frac{1}{2}(p+3p') \;\geq\; \delta + \frac{3}{7}(p'+2p) \;.$$

Note that $2m = 3M$ and that $\delta = 2m \cdot \ell$. Combining both cases we get

$$b_S(G_{F,\ell}, v_0) \;\geq\; \delta + \frac{3}{7}\kappa \cdot \ell \cdot M \;=\; \delta \cdot \left(1 + \frac{1}{7} \cdot \kappa\right) \;.$$

For any $\epsilon > 0$ this gives, for sufficient large ℓ

$$\frac{b_S(G_{F,\ell}, v_0)}{b(G_{F,\ell}, v_0)} \;\geq\; \frac{1+\frac{1}{7}\kappa}{1+\frac{2m+5}{\delta}} \;\geq\; 1 + \frac{1}{7}\kappa - \epsilon$$

Now we choose $\kappa = \frac{1}{8} - \epsilon'$ according to theorem 1 and get the claimed lower bound $\frac{57}{56} - \epsilon''$ for any $\epsilon'' > 0$ ∎

4 Conclusions

The complexity of broadcasting time is a key for understanding the obstacles to efficient communication in networks. This article answers the open question whether broadcasting can be approximated with any constant factor. Since until now the upper bound factor for approximating broadcasting time was known as $O(\frac{\log^2 |V|}{\log\log|V|})$ [Ravi94] and the lower bound was known as one additive time unit. Thus, a lower constant bound of a factor of $\frac{57}{56} - \epsilon$ is a big step forward. Yet it does not seem to be the end of the road.

There are a number of interesting special cases open. For *planar graphs* computing the exact broadcasting time is $\mathcal{NP}$-hard. But there are two reasons why good approximation algorithms seem to be reasonable. The average degree of a planar graph is constant and for bounded-degree graphs there is a constant factor polynomial time approximation algorithm for broadcasting [Ravi94]. Recently, new tree decomposition techniques for planar graphs have been introduced and have resulted in polynomial approximation algorithms. Further, it is known that good tree decompositions help to solve broadcasting more efficiently [JRS96,MRSR95].

There is good hope that the techniques of this paper can be applied as well to ternary graphs. For the moment a lower bound of $1 + 1/O(\log|V|)$ is known for approximating broadcasting time [JRS96]. The upper bound for approximating the broadcasting time of a ternary graph is a constant. So, close upper and lower bounds are still unknown.

However, in practice it is hardly ever possible to determine the ratio between sequential communication overhead at a single node and parallel communication given by the transfer rate. A further complication is that even the network structure may be unknown to all partners because of dynamic unpredictable changes. But even if the network is known in detail and in advance and given the above simple unit-time model, this paper shows that establishing a good broadcasting strategy is a computationally infeasible task.

References

[BHLP92] J.-C. Bermond, P. Hell, A. Liestman, and J. Peters, *Broadcasting in Bounded Degree Graphs,* SIAM J. Disc. Math. 5, 1992, 10-24.

[BGH91] H. Bodlaender, J. Gilbert, H. Hafsteinsson, *Approximating Treewidth, Pathwidth, Frontsize and Shortest Elimination Tree,* Journal of Algorithms, March 1995, 18(2):238-255.

[GaJo79] M. Garey and D. Johnson, *Computers and Intractability, A Guide To the Theory of $\mathcal{NP}$-Completeness*, Freeman 1979.

[Håst97] J. Håstad, *Some optimal inapproximability results,* In Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing, pages 1997, 1-10.

[HHL88] S. Hedetniemi, S. Hedetniemi, and A. Liestman, *A Survey of Gossiping and Broadcasting in Communication Networks*, Networks 18, 1988, 319-349.

[JRS96] A. Jakoby, R. Reischuk, C. Schindelhauer, *The Complexity of Broadcasting in Planar and Decomposable Graphs,* Discrete Applied Mathematics 83 (1998) 179-206.

[LP88] A. Liestman and J. Peters, *Broadcast Networks of Bounded Degree*, SIAM J. Disc. Math. 4, 1988, 531-540.

[Midd93] M. Middendorf, *Minimum Broadcast Time is $\mathcal{NP}$–complete for 3–regular planar graphs and deadline 2,* IPL 46, 1993, 281-287.

[MRSR95] M. V. Marathe, R. Ravi, R. Sundaram, S.S. Ravi, D.J. Rosenkrantz, H.B. Hunt III, *Bicriteria network design problems*, Proc. 22nd Int. Colloquium on Automata, Languages and Programming, Lecture Notes in Comput. Sci. 944, Springer-Verlag, 1995, 487-498.

[Ravi94] R. Ravi, *Rapid rumor ramification: approximating the minimum broadcast time*, Proc. 35th Ann. IEEE Symp. on Foundations of Comput. Sci., IEEE Computer Society, 1994, 202-213.

[SCH81] P. Slater, E. Cockayne, and S. Hedetniemi, *Information Dissemination in Trees*, SIAM J. Comput. 10, 1981, 692-701.