Abstract

We study the shortest-path broadcast problem in graphs and digraphs, where a message has to be transmitted from a source node $s$ to all the nodes along shortest paths, in the classical telephone model. For both graphs and digraphs, we show that the problem is equivalent to the broadcast problem in layered directed graphs. We then prove that this latter problem is NP-hard, and therefore that the shortest-path broadcast problem is NP-hard in graphs as well as in digraphs. Nevertheless, we prove that a simple polynomial-time algorithm, called MDST-broadcast, based on min-degree spanning trees, approximates the optimal broadcast time within a multiplicative factor $\frac{3}{2}$ in 3-layer digraphs, and $O\left(\frac{\log n}{\log \log n}\right)$ in arbitrary multi-layer digraphs. As a consequence, one can approximate the optimal shortest-path broadcast time in polynomial time within a multiplicative factor $\frac{3}{2}$ whenever the source has eccentricity at most 2, and within a multiplicative factor $O\left(\frac{\log n}{\log \log n}\right)$ in the general case, for both graphs and digraphs. The analysis of MDST-broadcast is almost tight, as we prove that this algorithm cannot approximate the optimal broadcast time within a factor smaller than $\Omega\left(\frac{\log n}{(\log \log n)^2}\right)$.

Keywords: Broadcast time, shortest paths, communication networks, layered graphs.

---

1Supported by MIUR PRIN project AMANDA.
2Supported by ANR project DISPLEXITY, and INRIA project GANG.
3Supported by NSERC.
1. Introduction

1.1. The general context

Broadcasting refers to the task in which one message has to be transmitted from one source node to all the other nodes in a network (we always assume that all nodes are reachable from the source). Constructing efficient broadcast protocols, that is, computing an appropriate scheduling for the communications between nodes, has been the source of a huge amount of work whose nature depend highly on the communication model. In this paper, we use the classical telephone model \cite{14}. In this model, the network is modeled as a connected undirected or directed graph\footnote{"Directed graph" is abbreviated to "digraph" in this paper.}, and communications proceed in a sequence of synchronous rounds. At each round, every node which is aware of the message (that is, either the source, or a node that has received the message during some previous round) can forward the message to at most one of its neighbors in the network. In a digraph, the message can only travel in the direction of the edge along which it is sent. The measure of complexity is the number of rounds necessary to complete broadcast. Given a graph or a digraph $G = (V, E)$, and a node $s \in V$, we denote by $b(G, s)$ the minimum number of rounds required to broadcast a message from $s$ to all nodes in $V$ in the telephone model.

The telephone broadcast problem consists in: given a (di)graph $G = (V, E)$, and $s \in V$, computing $b(G, s)$. In the multicast version of the problem, a set $S$ of terminals is additionally specified, and the objective is to compute the minimum number of rounds to inform all nodes in $S$ (the message can of course be relayed by non-terminal nodes). In fact, in both variants of the problem, we are also interested in computing the optimal communication schedule enabling to reach the optimal broadcast or multicast time. Since no nodes need to be informed twice, this schedule can be represented by a tree $T$ rooted at the source, spanning the terminals, with downward edges labeled at each node $u$ by pairwise distinct integers in $[1, \deg(u)]$ where $\deg(u)$ is the number of children of $u$ in $T$, specifying the order in which $u$’s children should be informed. (Observe that w.l.o.g., we are restricting our attention to schedules where transmissions from a node occur at consecutive rounds.)

The broadcast time of many classical networks is known (cf., e.g., \cite{10, 14, 15, 16} and the references therein), and several efficient randomized broadcast protocols have been proposed \cite{12}. However, the broadcast problem (and thus the multicast problem as well) is known to be NP-complete in graphs (and thus in digraphs as well) \cite{11}. In fact, it is even known that it is NP-hard to approximate the broadcast time within a ratio $3 - \epsilon$ for any $\epsilon > 0$ \cite{6}. There have been several attempts to design polynomial-time approximation algorithms for the broadcast and multicast problems \cite{2, 6, 7, 17, 20}, and the best known approximation ratio is $O\left(\frac{\log k}{\log \log k}\right)$ for $k$-terminal multicast (in the case of undirected graphs), due to \cite{7}. In directed graphs, the broadcast problem appears to be even more difficult to approximate: not only is it unlikely that there exists a
polynomial-time approximation scheme for it, but it is even unlikely that it is in APX. Indeed, it has been proved that, unless $\text{NP} \subseteq \text{DTIME}(n^{O(\log n)})$, the broadcast problem in digraphs cannot be approximated within a ratio less than $\Omega(\sqrt{\log n})$ [6]. The best known approximation algorithm for broadcast in digraphs has approximation factor $O(n)$ [6]. The difficulty appears to be even more severe regarding multicast, for which it is known [9] that the $k$-terminal multicast time cannot be approximated within a factor less than $\Omega(\log k)$. The best polynomial-time algorithm known approximates multicast within a multiplicative factor of $O(\log k)$, but with an additive factor of $O(\sqrt{k})$ [8].

1.2. Our results

In this paper, we are interested in the shortest-path broadcast problem in graphs and digraphs [13] (see also [3]). Shortest path broadcast refers to the broadcast problem in which the message must reach every node $u$ along a shortest path from the given source $s$ to $u$ in the given (di)graph $G$. In other words, the message can only traverse edges of the layered digraphs induced by the edges of the original (di)graph from a node at distance $i$ from $s$ to a node at distance $i+1$ from $s$ in $G$.

We first show that the shortest-path broadcast problem in graphs and digraphs is equivalent to the broadcast problem in layered directed graphs (a.k.a. multi-stage digraphs). Using this equivalence, we then show that the shortest-path broadcast problem is NP-hard in graphs and digraphs. Nevertheless, we prove that the approximation algorithm in [20], based on a minimum-degree spanning tree construction, has approximation ratio $O(\log n/\log \log n)$ in general graphs and digraphs. The bad news is that this bound is essentially tight for this algorithm, as we prove that this approach cannot provide an approximation ratio better than $\Omega(n/\log n \log \log n)$. Finally, in instances in which the source has eccentricity 2, we show that shortest-path broadcast time can be approximated within a factor at most $3/2$ for both graphs and digraphs.

1.3. Structure of the paper

We provide the formal definition of our problems in Section 2, where we also establish the equivalence between the shortest-path broadcast problems and the broadcast problem in layered digraphs. In Section 3, we prove that all our problems are NP-hard. Section 4 is then dedicated to the design and analysis of the approximation algorithm based on the minimum-degree spanning tree construction, while, in Section 5, we analyze this algorithm in the case of instances in which the source has bounded eccentricity. We conclude by some considerations about multicast, and the potential fixed parameter tractability nature of the shortest-path broadcast problem where the parameter is the eccentricity of the source, including some open questions.

2. Definition and preliminary results

In this paper, we focus on shortest-path broadcast, that is, the classical broadcast problem where the message is restricted to travel along shortest paths.
In other words, the message can only be transferred from a node at distance
$i$ from the source $s$ to a node at distance $i + 1$ from $s$, for some $i \geq 0$. As
we mentioned in the introduction, shortest-path broadcast is closely related to
broadcast in layered graphs. In this section, we formalize this statement. For
that purpose, let us define formally the shortest-path problems we are interested
in.

- **SP-BCAST**: given a connected graph $G$, a node $s$ of $G$, and $k \geq 0$, de-
cide whether broadcast from $s$ to all nodes in $G$ can be achieved along
shortest-paths in at most $k$ rounds.

- **MIN-SP-BCAST**: given a connected graph $G$, and a node $s$ of $G$, compute
the minimum number of rounds required to broadcast a message from $s$
to all nodes in $G$ along shortest-paths.

- **SP-DIR-BCAST** and **MIN-SP-DIR-BCAST**: same as the above, respectively,
where $G$ is a directed graph (and all nodes are reachable from $s$).

Now, let us define *layered* digraphs. A layered digraph is a directed graph
where the set of nodes is partitioned into $\ell \geq 1$ disjoint subsets $V_0, V_1, \ldots, V_{\ell-1}$
such that if $(u, v)$ is an edge of the digraph then necessarily $u \in V_i$ and $v \in V_{i+1}$
for some $i$, $0 \leq i < \ell - 1$. To state the equivalence between, on the one
hand, shortest-path broadcast in graphs and digraphs, and, on the other hand,
broadcast in layered digraphs, we define the following:

- **LAY-BCAST**: given a layered digraph $L$, with $V_0 = \{s\}$, and $k \geq 0$, decide
whether broadcast from $s$ to all nodes in $L$ can be achieved in at most $k$
rounds.

- **MIN-LAY-BCAST**: given a layered digraph $L$, with $V_0 = \{s\}$, compute the
minimum number of rounds required to broadcast a message from $s$
to all nodes in $L$.

Finally, in order to precisely state the equivalence between the above prob-
lems, we refer to a quite strong notion of approximation-preserving reduction,
that is, the *strict reduction* [4] (S-reduction), which strengthens the *linear re-
duction* (L-reduction) defined in [19]. Intuitively, an optimization problem $A$
is S-reducible to another optimization problem $B$ if (1) any instance of $A$ can
be transformed into an instance of $B$ with the same optimal value, and (2)
any solution for $B$ can be transformed into a solution for $A$ with the same
cost. More formally, let $A$ and $B$ be optimization problems, and $c_A$ and $c_B$
(respectively, $\text{OPT}_A$ and $\text{OPT}_B$) their cost functions (respectively, their optimal
value functions). A pair of functions $f$ and $g$ is an S-reduction from $A$ to
$B$ if all of the following conditions are met: (1) functions $f$ and $g$ are com-
putable in polynomial time, (2) if $x$ is an instance of problem $A$, then $f(x)$ is
an instance of problem $B$, (3) if $y$ is a solution to $f(x)$, then $g(x, y)$ is a solu-
tion to $x$, (4) $\text{OPT}_B(f(x)) = \text{OPT}_A(x)$, and (5) for every solution $y$ to $f(x)$,
$c_A(x, g(x, y)) = c_B(f(x), y)$. The two problems $A$ and $B$ are S-equivalent if
there exists an S-reduction from $A$ to $B$, and an S-reduction from $B$ to $A$. 
Lemma 1. \textsc{min-sp-bcast}, \textsc{min-sp-dir-bcast}, and \textsc{min-lay-bcast} are all S-equivalent.

Proof. First, let us show that \textsc{min-sp-bcast} and \textsc{min-sp-dir-bcast} are both S-reducible to \textsc{min-lay-bcast}. To see why, given a (di)graph $G$, and given a source node $s \in V(G)$, let $f_1(G, s) = (G_s, s)$, where $G_s$ is the directed graph on the same set of nodes as $G$, where the edge $(u, v)$ belongs to $E(G_s)$ if and only if the directed edge $(u, v)$ or the edge $\{u, v\}$ in $G$ belongs to at least one shortest path starting from $s$. Note that $G_s$ is directed and layered, with $V_0 = \{s\}$. By construction, any broadcast protocol from $s$ in $G_s$ is a shortest-path broadcast protocol from $s$ in $G$, and vice-versa, with the same execution times. That is, by choosing $g_1(x, y) = y$, we get that $(f_1, g_1)$ is an S-reduction.

Conversely, let us show that \textsc{min-lay-bcast} is reducible to both \textsc{min-sp-bcast} and \textsc{min-sp-dir-bcast}. The S-reduction to \textsc{min-sp-dir-bcast} is trivial since the instance graph of \textsc{min-lay-bcast} is directed, i.e., choose $f_2$ as the identity function, and $g_2 = g_1$. The S-reduction to \textsc{min-sp-bcast} is defined by choosing, for any layered digraph $L$ with $V_0 = \{s\}$, $f_3(L) = (U, s)$ where $U$ is the graph obtained from $L$ by removing the orientation of the edges, and $g_3 = g_1$. Indeed, we then have:

$$
\text{OPT}_{\textsc{min-sp-bcast}}(f_3(L)) = \text{OPT}_{\textsc{min-sp-bcast}}(U, s) = \text{OPT}_{\textsc{min-lay-bcast}}(f_1(U, s)) = \text{OPT}_{\textsc{min-lay-bcast}}(U_s) = \text{OPT}_{\textsc{min-lay-bcast}}(L)
$$

where the first equality is by definition of $f_3$, the second equality by the fact that \textsc{min-sp-bcast} is S-reducible to \textsc{min-lay-bcast} by $(f_1, g_1)$, the third equality is by definition of $f_1$, and the last equality follows from the fact that, given a layered digraph $L$ with $V_0 = \{s\}$, we have $U_s = L$. Similarly, regarding the cost function, for any solution $y$ of \textsc{min-sp-bcast} for the instance $(U, s)$, we have:

$$
\text{c}_{\textsc{min-lay-bcast}}(L, g_3(f_3(L), y)) = \text{c}_{\textsc{min-lay-bcast}}(L, y) = \text{c}_{\textsc{min-lay-bcast}}(U_s, y) = \text{c}_{\textsc{min-sp-bcast}}((U, s), g_1((U, s), y)) = \text{c}_{\textsc{min-sp-bcast}}(f_3(L), y)
$$

where the first equality is by definition of $g_3$, the second equality follows from the fact that $U_s = L$, the third equality follows from the fact that \textsc{min-sp-bcast} is S-reducible to \textsc{min-lay-bcast} by $(f_1, g_1)$, and the last equality follows from the definitions of $f_3$ and $g_3$.

A first important consequence of this lemma is that is any of the three decision problems \textsc{sp-bcast}, \textsc{sp-dir-bcast}, and \textsc{lay-bcast} is NP-complete, then all of them are NP-complete. A second important consequence is that any approximation results, either positive or negative, for any of the three optimization problems \textsc{min-sp-bcast}, \textsc{min-sp-dir-bcast}, and \textsc{min-lay-bcast} immediately
applies to all of them. In any case, note that the broadcast problem in layered digraphs may have also its interest on its own because of the practical importance of these graphs for communication networks, from tightly coupled parallel computers [18] to modern data centers [1].

3. Shortest-path broadcast is hard

Our first result is showing that the broadcast problem remains hard to be solved, even in the case of the shortest paths case.

Theorem 1. Lay-bcast is NP-complete

Proof. We adapt the reduction presented in [5], which was used to show that the broadcast problem is NP-complete even for bounded-degree graphs. We reduce 3-SAT to MBT. To this aim we will make use of the gadget $G_n$ depicted on Figure 1. In this figure, $s = L_{0,0}$. Observe that $b(G_n, s) = n + 1$. Indeed, it is clear that $b(G_n, s) \geq n + 1$, since there are $n + 2$ levels. On the other hand, the unique optimal broadcast scheduling is the one in which each node $L_{i,0}$, for $0 \leq i \leq n$, first serves its left child and then serves its right child, while all other nodes serve their right child only (in the figure, the broadcast times of this scheduling are shown in gray). Note also that, according to this optimal scheduling, all nodes at level $i$, with $1 \leq i \leq n$, receive the message at time $i + 1$, apart from the node $L_{1,0}$, which receives the message at time $i$. Let $\varphi$ be a CNF Boolean formula on variables $\{v_1, \ldots, v_n\}$ with clauses $\{c_1, \ldots, c_m\}$, where $|c_i| = 3$ for $1 \leq i \leq m$. We then construct a layered graph $G_\varphi$ as follows (see Figure 2). A copy of the gadget $G_n$ is connected to $2n$ copies of the gadget $G_m$ (whose sources are denoted by $F_1, \ldots, F_n$, and $T_1, \ldots, T_n$) and to a path of $m + 3$ nodes (denoted by $p_1, p_2, \ldots, p_{m+3}$) by the following edges: $(L_{n,i}, F_i)$ and $(L_{n,i}, T_i)$, for $1 \leq i \leq n$, and $(L_{m+1,0}, p_1)$. For any $i$, $1 \leq i \leq n$, the node $L_{m+1,0}$ of the gadget rooted at $F_i$ (respectively, $T_i$) is connected to a node $f_i$ (respectively, $t_i$). Finally, for each clause $c_j$ there is a node $x_j$. An edge is connecting the node $L_{n,j}$ of the gadget rooted at $F_j$ (respectively, $T_j$) to node $x_j$ if and only if the literal $\neg v_i$ (respectively, $v_i$) belongs to $c_j$. Observe that there are exactly $n + m + 5$ levels in $G_\varphi$ ($n + 2$ levels in $G_n$, and $m + 3$ levels in the path $p_1, \ldots, p_{m+3}$).

We have that $b(G_\varphi, s) \geq n + m + 4$ because the broadcast time cannot be less than the number of levels minus 1. We now prove that $\varphi$ is satisfiable if and only if $b(G_\varphi, s) = n + m + 4$.

If $\varphi$ is satisfiable, then let $\tau$ be a satisfying truth-assignment. We can construct an optimal broadcast scheduling as follows. We apply the optimal broadcast scheduling in the gadget $G_n$. The node $L_{m+1,0}$ will inform the node $p_1$ at time $n + 2$. The message can propagate through the path starting from this latter node and arrive at node $p_{m+3}$ at time $n + m + 4$. For any $i$, $1 \leq i \leq n$, if $\tau(v_i) = 0$ (respectively, 1), then the node $L_{n,i}$ of $G_n$ will inform the node $F_i$ (respectively, $T_i$) at time $n + 2$, and the node $T_i$ (respectively, $F_i$) at time $n + 3$.

We then apply the optimal broadcast scheduling to each gadget $G_m$, so that if
\( \tau(v_i) = 0 \) (respectively, \( \tau(v_i) = 1 \)), then all nodes \( L_{m,j} \) of the gadget rooted at \( F_i \) (respectively, \( T_i \)), \( 1 \leq j \leq m \), will receive the message at time \( n + m + 3 \), and all nodes \( L_{m,j} \) of the gadget rooted at \( T_i \) (respectively, \( F_i \)), will receive the message at time \( n + m + 4 \). Since \( \tau \) satisfies \( \varphi \), we get that, for each node \( x_j \), there must exist one of its parents that has received the message at time \( n + m + 3 \). Hence, \( x_j \) will receive the message at time \( n + m + 4 \). Finally, all nodes \( L_{n+1,0} \) of the gadgets \( G_m \) will receive the message at time \( n + m + 3 \), and can inform the corresponding \( f_j \) or \( t_j \) nodes at time \( n + m + 4 \).

Conversely, assume that \( b(G_\varphi, s) = n+m+4 \). This implies that the broadcast scheduling applied to the gadget \( G_n \) must be optimal. Otherwise, either \( L_{n+1,0} \) or \( L_{n,i} \), for some \( i \), \( 1 \leq i \leq n \), has received the message at time at least \( n + 2 \). In the first case, the overall broadcast time would be greater than \( n + m + 4 \), since there are \( m + 3 \) levels below \( L_{n+1,0} \), while, in the other case, either \( F_i \) or \( T_i \) has received the message at time at least \( n + 4 \) and the overall broadcast time would be greater than \( n + m + 4 \), since there are \( m + 1 \) levels below these two nodes. Hence, all terminal nodes of the gadget \( G_n \) must receive the message at time \( n + 1 \). For each variable \( v_i \), we set \( \tau(v_i) = 0 \) (respectively, 1) if the node \( F_i \) (respectively, \( T_i \)) has received the message at time \( n + 2 \). This yields a truth-assignment (since either \( F_i \) or \( T_i \) has received the message at time \( n + 2 \).

Figure 1: The gadget \( G_n \) of the NP-completeness proof
and either $F_i$ or $T_i$ has received the message at time $n + 3).$ Since all clause nodes $x_j$ have received the message at time $n + m + 4,$ this implies that they have received the message from a node belonging to a gadget $G_m$ whose source node $S_i$ has received the message at time $n + 2,$ that is, the literal corresponding to $x_j$ included in the clause corresponding to $S_i$ has been assigned the value 1.

In other words, all clauses are satisfied by $\tau$. \hfill \Box

As a direct consequence of Lemma 1, we get the following result.

**Corollary 1.** Both sp-bcast and sp-dir-bcast are NP-complete.

4. The approximation algorithm MDST-broadcast

A spanning tree $T$ rooted at a node $u$ in a given (di)graph $G$ is a minimum-degree spanning tree (MDST) of $G$ rooted at $u$ if no spanning trees of $G$ rooted at $u$ can have a maximum degree smaller than the maximum degree of $T$. (Note that, if $G$ is non directed, then the root actually plays no role). We analyze the following 2-stage algorithm, called MDST-broadcast, for solving the broadcast problem in layered digraphs:
1. Compute a MDST $T$ rooted at the source $s$;
2. Compute an optimal broadcast schedule from $s$ in $T$ (i.e., using only the edges of $T$).

Note that both stages can be computed in polynomial time, thanks to [22] and [21], respectively. (In fact, in layered digraphs, computing a MDST can also be done by solving a series of flow problems between each pair of consecutive levels, but this solution is less efficient than the one in [22]).

**Theorem 2.** The algorithm MDST-broadcast approximates MIN-LAY-BCAST within multiplicative factor $O\left(\frac{\log n}{\log \log n}\right)$.

**Proof.** Ravi [20] has defined the **poise** of a graph, and has shown that it is closely related to the broadcast time of the graph. The poise $p(G)$ of a graph $G$ is defined as the minimum $\Delta_T + D_T$ taken over all spanning trees of $G$, where $\Delta_T$ and $D_T$ denote the maximum degree and the depth of $T$, respectively. This definition extends directly to our setting, where the trees are bounded to be rooted at $s$. Hence, by applying the same arguments as in [20], we obtain that

$$\frac{1}{2} p(G) \leq b(G, s) \leq O\left( p(G) \frac{\log n}{\log \log n} \right)$$

where the upper bound is obtained by computing a greedy broadcast protocol in a spanning tree with poise $p(G)$, completing in this many rounds.

In the case of $\ell$-layer digraphs, the poise is simply equal to $\Delta_{\text{min}} + \ell - 1$ where $\Delta_{\text{min}}$ denotes the maximum degree of any minimum-degree spanning tree (rooted at $s$), because the depth of any spanning tree of an $\ell$-layer digraph is equal to $\ell - 1$. Hence $p(G)$, and a spanning tree $T$ with poise $p(G)$, are computable in polynomial time in layered digraphs (because, as we observed before, a spanning tree with maximum degree equal to $\Delta_{\text{min}}$ is computable in polynomial time in these graphs [22]).

Therefore, a broadcast protocol completing in $O(p(G) \frac{\log n}{\log \log n})$ rounds is computable in polynomial time (e.g., by computing an optimal broadcast protocol in $T$ with the algorithm in [21]). If follows that MDST-broadcast returns a broadcast protocol whose completion time is $O\left( \frac{\log n}{\log \log n} \right)$ times the optimal broadcast time, since $p(G)$ is at most twice $b(G, s)$. □

The following result shows that the above analysis is almost tight.

**Theorem 3.** The approximation ratio of algorithm MDST-broadcast is at least $\Omega\left(\frac{\log n}{\log \log n}\right)$ for MIN-LAY-BCAST.

**Proof.** We present an instance $(G, s)$ of the broadcast problem in layered digraphs (see Figure 3), for which MDST-broadcast approximates $b(G, s)$ within ratio $\Theta\left(\frac{\log n}{\log \log n}\right)$.

To describe the instance, let $d \geq 2$ be a power of two and $h$ be a number. The source $s$ is the root of a complete binary tree $B$ with $d^h$ leaves (hence of
Figure 3: The instance \((G, s)\) in the proof of Theorem 3

depth \(h \log_2 d\). Every leaf \(u_i\), with \(i = 1, \ldots, d^h\), of \(B\) is the root of a path \(P_i\) of length \(d\). Node \(u_1\) is also the root of a complete \(d\)-ary tree \(D\) with \(d^h\) leaves (hence of depth \(d\)), whose nodes belong to the \(P_i\)s. More formally, the root \(u_1\) of \(D\) is the first node of \(P_1\), the \(d\) children of \(u_1\) in \(D\) are the second nodes of \(P_1, \ldots, P_d\), the \(d^2\) grand-children of \(u_1\) in \(D\) are the third nodes in \(P_1, \ldots, P_d\), and, in general, the \(d^k\) nodes at level \(k\) of \(D\) are the \((k+1)\)th nodes of \(P_1, \ldots, P_d\). In particular, the leaves of \(D\) correspond to the last nodes of the \(P_i\)s. The connections between the nodes in \(D\) (i.e., which nodes of level \(\ell\) of \(D\) are connected to which nodes of level \(\ell + 1\)) are arbitrary, apart from the fact that every node \(x\) in \(D\) which is the successor in some path \(P_i\) of a node \(y\) in \(D\) must be a child of \(y\) in \(D\).

For every pair \(v, w\) of adjacent nodes in some \(P_i\), such that \(v \notin D\) and \(w \in D\), we connect \(v\) to \(d\) new leaves. Let \(U\) denote the set of all these new leaves. Note that \(|U| = d(d^h - 1)\) because each of the \(d^h\) paths \(P_i\), apart from \(P_1\), has exactly one node \(w_i\) belonging to \(D\) whose predecessor \(v_i\) in the path is not in \(D\). Note that, according to how the connections in \(D\) have been constructed, the sub-path of \(P_i\) starting from \(w_i\) is included in \(D\).

Overall, the layered digraph has \(n = \Theta(d^{h+1})\) nodes, as there are \(\Theta(d^h)\) nodes in \(B\) and in \(D\), \(\Theta(d^{h+1})\) in the collection of paths \(P_i\), \(i = 1, \ldots, d^h\), and \(\Theta(d^{h+1})\) in \(U\).

On the one hand, there exists a spanning tree \(T\) of \(G\) with maximum out-
degree $d$. For instance $T$ could consist of the tree $B$, the tree $D$ and the collection of $d^h - 1$ “brooms” formed by the sub-path of $P_i$ from $u_i$ to $v_i$ and the $d$ children of $v_i$ in $U$, for $i = 2, \ldots, d^h$. The existence of a spanning tree $T$ of $G$ with maximum out-degree $d$ implies that the broadcast time returned by MDST-broadcast is $O(d \cdot h)$. Indeed, MDST-broadcast must return a spanning tree of maximum out-degree at most $d$. Such a spanning tree must include $B$ because, in fact, any spanning tree of $G$ must include $B$. Broadcasting in the binary tree $B$ takes $2h \log d$ rounds. Hence, after this many rounds, broadcasting takes at most $O(dh)$ additional rounds because broadcasting in a tree with maximum degree $d$ and depth $d$ requires at most $O(dh)$ rounds.

On the other hand, every minimum-degree spanning tree of $G$ must include the $d$-ary tree $D$. This is because the nodes $v_i$ on the paths $P_i$ need to “cover” their $d$ children in the set $U$, and hence they cannot cover their only child $w_i$ in $D$. This implies that the edge $(v_i, w_i)$ cannot be part of the spanning tree. Hence all the nodes in $D$ can only be spanned by using the edges in $D$. The fact that $D$ is included in every minimum-degree spanning tree of $G$ implies that the protocol computed by MDST-broadcast completes in at least $\Omega(dh)$ rounds because broadcasting in $D$ takes at least these many rounds (by induction on $k$, informing all nodes at level $k$ requires at least $kd$ rounds).

To sum up, the protocol returned by MDST-broadcast completes in

$$t_{\text{MDST-broadcast}} = \Theta(dh)$$

rounds. Let’s now focus on the optimal protocol. We have $b(G, s) = \Theta(h \log d + d)$, since there are $\Theta(h \log d)$ layers and $d$ additional steps suffice to transmit to all the leaves; this broadcast time is achieved by broadcasting in $B$ and by then using the paths $P_i$s in parallel.

We now set $h = \lceil d / \log d \rceil$, or $d = \Theta(h \log h)$. Then, $n = \Theta(d^{h+1}) = 2^{\Theta(d)}$, or $d = \Theta(\log n)$ and $h = \Theta(\log n / \log \log n)$. Also, $b(G, s) = \Theta(d) = \Theta(h \log h)$, so the performance ratio is $\Omega(h) = \Omega(\log n / \log \log n)$.

Again, as a direct consequence of Lemma 1, Theorems 2 and 3 imply the following result.

**Corollary 2.** The algorithm MDST-broadcast approximates MIN-SP-BCAST and MIN-SP-DIR-BCAST within multiplicative factor $\Theta(\frac{\log n}{\log \log n})$.

### 5. Sources with Bounded Eccentricities

We have seen that the algorithm MDST-broadcast enables to approximate the shortest-path broadcast problem with a multiplicative factor $O(\frac{\log n}{\log \log n})$. In this section, we study the performances of MDST-broadcast as a function of the eccentricity of the source. According to the reduction from the shortest-path broadcast problem to the broadcast problem in layered digraph stated in Lemma 1, this corresponds to analyzing the performances of MDST-broadcast as a function of the number of layers $\ell$ in a layered digraph.
Theorem 4. The algorithm MDST-broadcast approximates MIN-LAY-BCAST within multiplicative factor \( \ell - 1 \) in \( \ell \)-layered digraphs.

Proof. Let \( \Delta_{\text{min}} \) be the maximum degree of a the minimum-degree spanning tree \( T \) computed in the first step of MDST-broadcast. Broadcasting in \( T \) takes at most \( (\ell - 1)\Delta_{\text{min}} \) rounds (by induction on \( \ell \)). On the other hand, the spanning tree corresponding to an optimal broadcast protocol has degree at least \( \Delta_{\text{min}} \), and thus \( b(G, s) \geq \Delta_{\text{min}} \).

We now provide a better bound on the approximation ratio of MDST-broadcast, in the case \( \ell = 3 \).

Theorem 5. Algorithm MDST-broadcast approximates MIN-LAY-BCAST within multiplicative factor \( \frac{3}{2} \) in 3-layer digraphs.

Proof. Let \( G = (\{s\} \cup V_1 \cup V_2, E) \) be a 3-layer digraphs. Let us denote by \( t_{\text{opt}} = b(G, s) \) the broadcast time of an optimal protocol \( \text{opt} \), and by \( t_{\text{alg}} \) the broadcast time of the protocol \( \text{alg} \) computed by our algorithm MDST-broadcast.

For \( i = 1, \ldots, |V_1| \), let \( v_i \in V_1 \) be the \( i \)th node informed by \( s \) in \( \text{alg} \), and let \( B_i \subset V_2 \) be the set of nodes informed by \( v_i \) in \( \text{alg} \). We can assume, w.l.o.g., that \( |B_1| \geq |B_2| \geq \cdots \geq |B_\kappa| \geq 0 \).

Observe that \( t_{\text{alg}} = \max_k (k + |B_k|) = \kappa + |B_\kappa| \),

where \( \kappa \) is the value of \( k \) that maximizes \( k + |B_k| \). Observe that \( t_{\text{opt}} \geq |V_1| \geq \kappa \); thus if \( |B_\kappa| \leq \kappa/2 \), we are done. Also observe that \( t_{\text{opt}} \geq |B_1| \geq |B_\kappa| \), since MDST-broadcast is based on a minimum degree spanning tree. Thus, if \( \kappa \leq |B_\kappa|/2 \), we are also done.

Let us define \( q = |B_\kappa|/\kappa \). From the preceding discussion, the theorem holds if either \( q < 1/2 \) or \( q > 2 \). Thus, we assume from now on that \( q \in [1/2, 2] \). We can rewrite (1) as

\[
t_{\text{alg}} = (1 + q)\kappa .
\]

Each set in \( B_1, B_2, \ldots, B_\kappa \) has at least \( |B_\kappa| \) nodes, which implies that

\[
|V_2| \geq \kappa \cdot |B_\kappa| = q\kappa^2 .
\]

Observe that in \( t \) rounds, no protocol can inform more than \( \left( \frac{t}{2} \right)^2 < t^2/2 \) nodes in \( V_2 \). Thus,

\[
t_{\text{opt}} \geq \sqrt{2|V_2|} \geq \sqrt{2q\kappa} .
\]

The performance ratio of the algorithm is therefore bounded by

\[
\frac{t_{\text{alg}}}{t_{\text{opt}}} \leq \frac{(1 + q)\kappa}{\sqrt{2q\kappa}} = \frac{1 + q}{\sqrt{2q}} .
\]

Examining the function \( f(x) = \frac{1 + x}{\sqrt{2x}} \) in the range \( x \in [0.5, 2] \), we find that the only maxima are at the two endpoints, \( x = 1/2 \) and \( x = 2 \). The theorem then follows.
Corollary 3. Algorithm MDST-broadcast approximates MIN-SP-BCAST and MIN-SP-DIR-BCAST within multiplicative factor $\ell - 1$ (respectively, $\frac{3}{2}$) whenever the source has eccentricity at most $\ell$ (respectively, 2).

Note that this bound is tight for the algorithm. The instance where $(V_1, V_2)$ form a complete $K$ by $K^2/2$ bipartite graph has optimal broadcast time of $K+1$, but the minimum-degree MST where each node in $V_1$ has degree $K/2$ results in a broadcast time of $3K/2$.

6. Conclusion

In this paper, we have analyzed the shortest-path broadcast problem in graphs and digraphs. In particular, after having proved the NP-hardness of this problem, we have shown that Algorithm MDST-broadcast approximates the shortest-path broadcast time within a ratio $O\left(\frac{\log n}{\log \log n}\right)$ in both graphs and digraphs. In the case of digraphs, this ratio is smaller than the best known approximation ratio for broadcasting without the shortest path constraint. Moreover, for instances where the source has eccentricity 2, we have shown that shortest-path broadcast time can be approximated within a factor $\frac{3}{2}$.

It is known [20] that the broadcast problem is as hard in (di)graphs with bounded diameter as in general (di)graphs. This may not be the case when we restrict ourselves to shortest-path broadcast. In particular, it is not clear whether the problem is NP-hard for instances where the source has bounded eccentricity, or even eccentricity 2. If yes, an intriguing open problem is whether the shortest-path broadcast problem is fixed parameter tractable (FPT) when parameterized by the eccentricity of the source. (The reduction from SAT in the proof of Theorem 1 uses instances where the eccentricity of the source is unbounded). In fact, up to our knowledge, no results are known on the FPT nature of the broadcast problem even in the case without the shortest-path constraint.

Another direction of research consists to extend the study of shortest-path communication to the multicast problem. One can show that multicast remains hard to approximate in layered digraphs. Indeed, by a reduction similar to the one in Theorem 3 of [9], it is possible to show that, even for 3-layer digraphs, the multicast time cannot be approximated within a ratio smaller than $\Omega(\log n)$. Note that this bound is tight, as the multicast time can be approximated within a ratio $O(\log n)$ in 3-layer digraphs. (One way to achieve this bound is using a greedy algorithm based on flow, for solving a minimum-degree set-cover problem).

Acknowledgment. The authors are thankful to the referees for their helpful detailed comments.


