Faster Deterministic Broadcasting in ad hoc Radio Networks ⋆

Dariusz R. Kowalski[1] and Andrzej Pelc[2]

[1] Instytut Informatyki, Uniwersytet Warszawski,
Banacha 2, 02-097 Warszawa, Poland,
darek@mimuw.edu.pl

[2] Département d'informatique, Université du Québec en Outaouais,
Hull, Québec J8X 3X7, Canada,
Andrzej.Pelc@uqo.ca

Abstract. We consider radio networks modeled as directed graphs. In ad hoc radio networks, every node knows only its own label and a linear bound on the size of the network but is unaware of the topology of the network, or even of its own neighborhood. The fastest currently known deterministic broadcasting algorithm working for arbitrary n-node ad hoc radio networks, has running time $\mathcal{O}(n \log^2 n)$. Our main result is a broadcasting algorithm working in time $\mathcal{O}(n \log n \log D)$ for arbitrary n-node ad hoc radio networks of eccentricity D. The best currently known lower bound on broadcasting time in ad hoc radio networks is $\Omega(n \log D)$, hence our algorithm is the first to shrink the gap between bounds on broadcasting time in radio networks of arbitrary eccentricity to a logarithmic factor. We also show a broadcasting algorithm working in time $\mathcal{O}(n \log D)$ for *complete layered* n-node ad hoc radio networks of eccentricity D. The latter complexity is optimal.

1 Introduction

A radio network is a collection of transmitter-receiver stations. It is modeled as a directed graph on the set of these stations, refered to as *nodes*. A directed edge $e = (u, v)$ means that the transmitter of u can reach v. Nodes send messages in synchronous *steps* (time slots). In every step every node acts either as a *transmitter* or as a *receiver*. A node acting as a transmitter sends a message which can potentially reach all of its out-neighbors. A node acting as a receiver in a given step gets a message, if and only if, exactly one of its in-neighbors transmits in this step. The message received in this case is the one that was transmitted. If at least two in-neighbors v and v' of u transmit simultaneously in a given step, none of the messages is received by u in this step. In this case we say that a *collision* occurred at u. It is assumed that the effect at node u of more

⋆ This work was done during the first author's stay at the Research Chair in Distributed Computing of the Université du Québec en Outaouais, as a postdoctoral fellow. The work of the second author was supported in part by NSERC grant OGP 0008136 and by the Research Chair in Distributed Computing of the Université du Québec en Outaouais.

than one of its in-neighbors transmitting, is the same as that of no in-neighbor transmitting, i.e., a node cannot distinguish a collision from silence.

The goal of *broadcasting* is to transmit a message from one node of the network, called the *source*, to all other nodes. Remote nodes get the source message via intermediate nodes, along paths in the network. In order to make broadcasting feasible, we assume that there is a directed path from the source to any node of the network. We study one of the most important and widely investigated performance parameters of a broadcasting algorithm, which is the total time, i.e., the number of steps it uses to inform all the nodes of the network.

We consider deterministic distributed broadcasting in ad hoc radio networks. In such networks, a node does not have any *a priori* knowledge of the topology of the network, its maximum degree, its eccentricity, nor even of its immediate neighborhood: the only *a priori* knowledge of a node is its own label and a linear upper bound r on the number of nodes. Labels of all nodes are distinct integers from the interval $[0, ..., r]$. Broadcasting in ad hoc radio networks was investigated, e.g., in [3, 5–7, 9–13, 18]. We use the same definition of running time of a broadcasting algorithm working for ad hoc radio networks, as e.g., in [11]. We say that the algorithm works in time t for networks of a given class, if t is the smallest integer such that the algorithm informs all nodes of any network of this class in at most t steps. We do not suppose the possibility of spontaneous transmissions, i.e., only nodes which already got the source message, are allowed to send messages. Of course, since we are only concerned with upper bounds on broadcasting time, all our results remain valid if spontaneous transmissions are allowed. Indeed, a broadcasting algorithm without spontaneous transmissions can be considered in the model with spontaneous transmissions allowed: such transmissions are simply never used by the algorithm. The format of all messages is the same: a node transmits the source message and the current step number.

We denote by n the number of nodes in the network, by r a linear upper bound on n ($r = cn$, for some constant c), by D the eccentricity of the network (the maximum length of a shortest directed path from the source to any other node), and by Δ the maximum in-degree of a node in the network. Among these parameters, only r is known to nodes of the network.

1.1 Related Work

In many papers on broadcasting in radio networks (e.g., [1, 2, 14, 17]), the network is modeled as an undirected graph, which is equivalent to the assumption that the directed graph, which models the network in our scenario, is symmetric. A lot of effort has been devoted to finding good upper and lower bounds on deterministic broadcast time in such radio networks, under the assumption that nodes have full knowledge of the network. In [1] the authors proved the existence of a family of n-node networks of radius 2, for which any broadcast requires time $\Omega(\log^2 n)$, while in [14] it was proved that broadcasting can be done in time $O(D + \log^5 n)$, for any n-node network of diameter D. (Note that for symmetric networks, diameter is of the order of the eccentricity).

As for broadcasting in ad hoc symmetric radio networks, an $\mathcal{O}(n)$ algorithm assuming spontaneous transmissions was constructed in [9] and a lower bound $\Omega(D \log n)$ was shown in [5], in the case when spontaneous transmissions are precluded.

Deterministic broadcasting in arbitrary directed radio networks was studied, e.g., in [6–13, 18]. In [8], a $\mathcal{O}(D \log^2 n)$-time broadcasting algorithm was given for all n-node networks of eccentricity D, assuming that nodes know the topology of the network. Other above cited papers studied broadcasting time in ad hoc directed radio networks. The best known lower bound on this time is $\Omega(n \log D)$, proved in [12]. As for the upper bounds, a series of papers presented increasingly faster algorithms, starting with time $O(n^{11/6})$, in [9], then $O(n^{5/3} \log^{1/3} n)$ in [13], $O(n^{3/2}\sqrt{\log n})$ in [18], $O(n^{3/2})$ in [10], and finally, $O(n \log^2 n)$ in [11], which corresponds to the fastest currently known algorithm, working for ad hoc networks of arbitrary maximum degree. In another approach, broadcasting time is studied for ad hoc radio networks of maximum degree Δ. This work was initiated in [6], where the authors constructed a broadcasting scheme working in time $O(D \frac{\Delta^2}{\log^2 \Delta} \log^2 n)$, for arbitrary n-node networks with eccentricity D and maximum degree Δ. (While the result was stated only for undirected graphs, it is clear that it holds for arbitrary directed graphs, not just symmetric). This result was further investigated, both theoretically and using simulations, in [7, 4]. On the other hand, a protocol working in time $O(D\Delta \log^{\log \Delta} n)$ was constructed in [3]. Finally, an $O(D\Delta \log n \log(n/\Delta))$ protocol was described in [12] (for the case when nodes know n but not Δ). If n is also unknown, the algorithm from [12] works in time $O(D\Delta \log^a n \log(n/\Delta))$, for any $a > 1$.

Finally, randomized broadcasting in ad hoc radio networks was studied, e.g., in [2, 17]. In [2], the authors give a simple randomized protocol running in expected time $O(D \log n + \log^2 n)$. In [17] it was shown that for any randomized broadcast protocol and parameters D and n, there exists an n-node network of eccentricity D, requiring expected time $\Omega(D \log(n/D))$ to execute this protocol.

1.2 Our Results

The main result of this paper is a deterministic broadcasting algorithm working in time $\mathcal{O}(n \log n \log D)$ for arbitrary n-node ad hoc radio networks of eccentricity D. This improves the best currently known broadcasting time $\mathcal{O}(n \log^2 n)$ from [11], e.g., for networks of eccentricity polylogarithmic in size. Also, for $D\Delta \in \omega(n)$, this improves the upper bound $O(D\Delta \log n \log(n/\Delta))$ from [12]. The best currently known lower bound on broadcasting time in ad hoc radio networks, is $\Omega(n \log D)$ [12], hence our algorithm is the first to shrink the gap between bounds on deterministic broadcasting time for radio networks of arbitrary eccentricity, to a logarithmic factor. Our algorithm is non-constructive, in the same sense as that from [11]. Using the probabilistic method we prove the existence of a combinatorial object, which all nodes use in the execution of the deterministic broadcasting algorithm. (Since we do not count local computations in our time measure, such an object — the same for all nodes — could be found by exhaustive search performed locally by all nodes, without changing our result). It should be noted that a constructive broadcasting protocol working in time $O(n^{1+o(n)})$ was obtained in [15].

We also show a broadcasting algorithm working in time $\mathcal{O}(n \log D)$ for *complete layered* n-node ad hoc radio networks of eccentricity D. The latter complexity is optimal, due to the matching lower bound $\Omega(n \log D)$, proved in [12] for this class of networks, even assuming that nodes know parameters n and D. The best previous upper bound on broadcasting time in complete layered n-node ad hoc radio networks of eccentricity D was $\mathcal{O}(n \log n)$ [11]. Hence we obtain a gain for the same range of values of D as before.

If nodes do not know any upper bound on the size of the network, the upper bound $\mathcal{O}(n \log^2 n)$ from [11] remains valid, using a simple doubling technique, which probes possible values of n. In our case, the doubling technique cannot be used directly, since we deal with two unknown parameters, n and D. However, we can modify our algorithm in this case, obtaining running time $\mathcal{O}(n \log n \log\log n \log D)$, which still beats the time from [11], e.g., for networks of eccentricity polylogarithmic in size. For $D\Delta \in \Omega(n)$, this also improves the upper bound $O(D\Delta \log^a n \log(n/\Delta))$, for any $a > 1$, proved in [12] for the case of unknown n and Δ.

2 The Broadcasting Algorithm

In this section we show a deterministic broadcasting algorithm working in time $\mathcal{O}(n \log n \log D)$ for arbitrary n-node ad hoc radio networks of eccentricity D. Let r be the linear bound on the size of the ad hoc network. Taking $2^{\lceil r \rceil}$ instead of r, we can assume that $\log r$ is a positive integer. The parameter r is known to all nodes. We first show our upper bound under the additional assumption that D is known to all nodes. At the end of this section we show how this assumption can be removed without changing the result. Given a 0-1 matrix $T = [T_i(v)]_{i \le t; v \le r}$, where, $t = 3600 \cdot (1 + \log r) \cdot r \log D$, we define the following procedure

Procedure Fast-Broadcasting(T)

After receiving the source message and the current step number $a(1 + \log r) + b$, for some parameters a, b such that $0 \le b \le \log r$ and $0 < a(1 + \log r) + b \le t$, node v waits until step $t_v = (a + 1)(1 + \log r) - 1$.
for $i = t_v + 1, \ldots, t$ **do**
  Substep A. if $T_i(v) = 1$ **then** v transmits in step i.
  Substep B. if $i \equiv v \mod r$ **then** v transmits in step i.

We now define the following random 0-1 matrix $\hat{T} = [\hat{T}_i(v)]_{i \le t; v \le r}$. For all parameters a, b such that $0 \le b \le \log r$ and $0 < a(1 + \log r) + b \le t$, we have $\mathsf{Pr}\left[T_{a(1+\log r)+b}(v) = 1\right] = 1/2^b$, and all events $\hat{T}_i(v) = 1$ are independent. The period consisting of steps $a(1 + \log r), \ldots, (a + 1)(1 + \log r) - 1$ is called *stage* a. Algorithm Fast-Broadcasting is the execution of Procedure Fast-Broadcasting($\hat{T}$), for the above defined random matrix $\hat{T}$.

In what follows, v_0 denotes the source. In order to analyze Algorithm Fast-Broadcasting, we define the following classes of directed graphs. A *path-graph* consists of a directed path $v_0, \ldots, v_k$, where $k \leq D$, possibly with some additional edges $(v_l, v_{l'})$, for $l > l'$, and with some additional nodes v, whose only out-neighbors are among $v_1, \ldots, v_k$. The path $v_0, \ldots, v_k$ is called the *main path.* A *simple-path-graph* consists of a directed path $v_0, \ldots, v_k$, where $k \leq D$, with some additional nodes v, each of which has exactly one out-neighbor and this out-neighbor is among $v_1, \ldots, v_k$. For any path-graph $G = (V, E)$, the graph $\bar{G}$ is the subgraph of G on the same set of nodes, containing the main path and satisfying the condition that for every node v outside the main path, v has exactly one neighbor v_l in $\bar{G}$, where $l = \max\{l' : (v, v_{l'}) \in E\}$. By definition, for every path-graph G, the graph $\bar{G}$ is a simple-path-graph.

In general, path-graphs do not satisfy our assumption that there is a directed path from the source v_0 to any other node. Hence, for path-graphs, we modify our model of broadcasting as follows, and call it the *adversarial wake-up model.* We assume that nodes outside the main path also get the source message, but are woken up by an adaptive adversary in various time-steps not exceeding t. More precisely, the adversary can wake up any node v outside the main path in any step $t_v \leq t$, providing v with the source message and step number t_v. The adversary acts according to some *wake-up pattern* $\mathcal{W}_m$, from the family $\{\mathcal{W}_m\}_{m \leq M}$ of all possible wake-up patterns. A wake-up pattern is a function assigning to every vertex v outside the main path, a step number $t_v \leq t$. For every wake-up pattern $\mathcal{W}_m$ and stage a of Procedure Fast-Broadcasting(T), we define an integer $f_a(m)$ as follows. If node v_k has the source message after stage a, we fix $f_a(m) = k$. Otherwise, $f_a(m) = l$, where v_{l+1} is the last node on the main path which does not have the source message after stage a, but has an in-neighbor having the source message after stage a, assuming that the adversary uses pattern $\mathcal{W}_m$.

We now describe a high-level outline of the proof of our upper bound. First, using the probabilistic method, we show the existence of a matrix T, such that Procedure Fast-Broadcasting(T), applied to any simple-path-graph and to any wake-up pattern, delivers the source message to the last node of the main path, in time $\mathcal{O}(n \log n \log D)$. Next, we show that the same is true for any path-graph. Finally, we show that Procedure Fast-Broadcasting(T) completes broadcasting in all graphs in time $\mathcal{O}(n \log n \log D)$.

Fix an integer a. Suppose that $\hat{T}_i(v)$ are fixed for all $i < (a+1)(1+\log r)$. Our first goal is to show that for a fixed simple-path-graph G, the set $\{m : f_a(m) < f_{a+1}(m) \leq k\}$ contains a constant fraction of values $\{m : f_a(m) < k\}$, for any stage a, with high probability. More precisely, we call stage $a+1$ *successfull*, if either $\{m : f_a(m) < k\} = \emptyset$ or $|\{m : f_a(m) < f_{a+1}(m) \leq k\}| \geq \frac{1}{36} \cdot |\{m : f_a(m) < k\}|$ in the execution of Algorithm Fast-Broadcasting.

Lemma 1. *With probability at least* 0.1*, stage* $a+1$ *is successfull.*

Proof. The proof will appear in the full version of the paper.

The next lemma implies that the last node of the main path of a simple-path-graph gets the source message by step $\mathcal{O}(n \log n \log D)$ of Algorithm Fast-Broadcasting, with probability at least $1-(0.45)^{n\log D}$.

Lemma 2. *Fix a simple-path-graph G and a wake-up pattern $\mathcal{W}_m$ in the adversarial wake-up model. By stage $3600 \cdot n\log D$, the number of successful stages is at least $72n\log D$ with probability at least $1-(0.45)^{n\log D}$. This number of successful stages is sufficient to deliver the source message to node v_k.*

Proof. We consider only substeps of type A. Consider $3600 \cdot n\log D$ consecutive stages, since the beginning of Algorithm Fast-Broadcasting. From Lemma 1, stage a is successful with probability at least 0.1, moreover this probability is at least 0.1 independently of successes in another stages.

The probability, that among $3600n\log D$ consecutive stages at most $72D\log n$ are successful is at most

$$\sum_{k=0}^{72D\log n} \binom{3600n\log D}{k} \cdot \left(\frac{9}{10}\right)^{3600n\log D-k} \leq$$
$$\leq \binom{3600n\log D}{0} \cdot \left(\frac{9}{10}\right)^{3600n\log D} + \binom{3600n\log D}{1} \cdot \left(\frac{9}{10}\right)^{3600n\log D-1} +$$
$$+ \sum_{k=2}^{72D\log n} \frac{(3600n\log D)^{3600n\log D+1}}{k^k(3600n\log D-k)^{3600n\log D-k}} \cdot \left(\frac{9}{10}\right)^{3600n\log D-k}$$
$$\leq 4001n\log D \cdot \left(\frac{9}{10}\right)^{3600n\log D} +$$
$$+ \sum_{k=2}^{72D\log n} 3600n\log D \cdot \left(\frac{3600n\log D}{k}\right)^k \left[\frac{9}{10}\left(1+\frac{k}{3600n\log D-k}\right)\right]^{3600n\log D-k}$$
$$\leq 4001n\log D \cdot \left(\frac{9}{10}\right)^{3600n\log D} +$$
$$+72D\log n \cdot 3600n\log D \cdot \left(\frac{3600n\log D}{72n\log D}\right)^{72n\log D} \left[\frac{9}{10}\cdot\frac{50}{49}\right]^{49\cdot 72n\log D}$$
$$\leq 4001n\log D \cdot \left(\frac{9}{10}\right)^{3600n\log D} + 72D\log n \cdot 3600n\log D \cdot \left(50^{72}\cdot(0.92)^{49\cdot 72}\right)^{n\log D},$$

which is at most $(0.45)^{n\log D}$, for sufficiently large n. We used inequalities $\frac{b^b}{e^b} \leq b! \leq \frac{b^{b+1}}{e^b}$, for $b \geq 2$, and the fact that function $\left(\frac{C}{x}\right)^x$ is increasing for $0 < x \leq \frac{C}{e}$, for positive constant C. We also used the inequality $\frac{D}{\log D} \leq \frac{n}{\log n}$.

In order to complete the proof, we show that, if there are at least $72D\log n$ successfull stages by stage a, then $\{m : f_a(m) < k\} = \emptyset$. Let a_x, for $x = 1,\ldots,\log n$, be the first stage after $72D\cdot x$ successfull stages. It is enough to prove, by induction on x, that $|\{m : f_{a_x}(m) < k\}| < n/2^x$. For $x=1$ this is straightforward. Assume that this inequality is true for x. We show it for $x+1$. Suppose the contrary: $|\{m : f_{a_{x+1}}(m) < k\}| \geq n/2^{x+1}$. It follows that during each successfull stage b, where $a_x \leq b < a_{x+1}$, at least $\frac{n/2^{x+1}}{36}$ values $f_b(m)$ are smaller than $f_{b+1}(m)$. Since in stage a_x there were fewer than $n/2^x$ such values, and each of them may increase at most D times, we obtain that in stage a_{x+1}, the set $\{m : f_{a_{x+1}}(m) < k\}$ would have fewer than

$$\frac{\frac{n}{2^x} \cdot D}{\frac{n}{36 \cdot 2^{x+1}} \cdot 72D} = 1$$

elements, which contradicts the assumption that $|\{m : f_{a_{x+1}}(m) < k\}| > n/2^{x+1}$.

Lemma 3. *There exists a matrix T of format $r \times 3600r(1 + \log r)\log D$ such that, for any n-node simple-path-graph G of eccentricity at most D, and any wake-up pattern, Procedure Fast-Broadcasting(T) delivers the source message to the last node of the main path of G in $3600n \log D$ stages, for sufficiently large n.*

Proof. First consider D such that $\log D > 20(1+c)$. (Recall that $r = cn$. Knowledge of c is not necessary). In this case we consider only substeps of type A. There are at most

$$\sum_{k=1}^{D} \binom{r}{n} \cdot \binom{n}{k} \cdot k! \cdot k^{n-k} \leq D^{\alpha n}$$

different simple-path-graphs G with n nodes and eccentricity at most D, for some positive constant $\alpha < 1.1$. This is because

$$\binom{r}{n} \cdot \binom{n}{k} \cdot k! \cdot k^{n-k} \leq 2^r \cdot 2^n \cdot k^n = 2^{n(1+c)+n\log k} \leq 2^{n(1+c)+n\log D} < D^{1.05 \cdot n} ,$$

for any $k \leq D$, and consequently

$$\sum_{k=1}^{D} \binom{r}{n} \cdot \binom{n}{k} \cdot k! \cdot k^{n-k} \leq D \cdot D^{1.05 \cdot n} < D^{1.1 \cdot n}$$

for sufficiently large n ($n > 20$).

Let $\hat{T}$ be the random matrix defined previously. By Lemma 2, the probability that Algorithm Fast-Broadcasting working on $\hat{T}$ delivers the source message to the last node of the main path of G by stage $3600n \log D$, for all simple-path-graphs G and all wake-up patterns $\mathcal{W}_m$, is at least

$$1 - \sum_{G} (0.45)^{n \log D} \geq 1 - D^{\alpha n} \cdot (0.45)^{n \log D} > 1 - D^{1.1n} \cdot (0.45)^{n \log D} > 0 ,$$

where the sum is taken over all simple-path-graphs G with n nodes chosen from r labels, and of eccentricity at most D. Using the probabilistic method we obtain, that there is a matrix T satisfying the lemma.

Next, consider D such that $\log D \leq 20(1 + c)$. In this case we consider only substeps of type B. Since D is constant, by step $D \cdot r \in \mathcal{O}(n)$, every node of the main path will receive the source message by the round-robin argument. This concludes the proof.

For every wake-up pattern $\mathcal{W}_m$ and step i of Procedure Fast-Broadcasting(T), we define an integer $f'_i(m)$ as follows, by analogy to $f_a(m)$. If node v_k has the source message after step i, we fix $f'_i(m) = k$. Otherwise, $f'_i(m) = l$, where v_{l+1} is the last node on the main path which does not have the source message after step i, but has a neighbor having the source message after step i, assuming that the adversary uses pattern $\mathcal{W}_m$. Obviously $f_a(m) = f'_{(a+1)(1+\log r)-1}(m)$.

Lemma 4. *Fix a path-graph G and a matrix T. For any wake-up pattern $\mathcal{W}_m$ and any step i, values $f'_i(m)$ are the same for G and for $\bar{G}$. Consequently, for any stage a, values $f_a(m)$ are the same for G and for $\bar{G}$.*

Proof. The proof will appear in the full version of the paper.

Theorem 1. *There is a matrix T of format $r \times 3600r(1 + \log r)\log D$ such that for every n-node graph G of eccentricity D, Procedure Fast-Broadcasting(T) performs broadcasting on G in time $\mathcal{O}(n \log n \log D)$.*

Proof. Take the matrix T from Lemma 3. Suppose the contrary: after step $3600n(1+\log r)\log D$, there is a node v without the source message. Consider the subgraph H of G, which contains a shortest directed path $v_0, \ldots, v_k = v$ from the source to node v, with all induced edges between nodes of this path, and all those in-neighbors v' of nodes $v_1, \ldots, v_k$ which received the source message by step $3600n(1 + \log r)\log D$, together with corresponding arcs (v', v_l). By definition, H is a path-graph, and hence $\bar{H}$ is a simple-path-graph. By Lemma 3 we obtain that v received the source message in $\bar{H}$, by step $3600n(1 + \log r)\log D$. (We need to apply Lemma 3 to the wake-up pattern "generated" by Procedure Fast-Broadcasting(T) working on G: every node of $\bar{H}$ outside the main path is woken up in the adversarial wake-up model in the time step in which it gets the source message for the first time when Procedure Fast-Broadcasting(T) is executed on G). From Lemma 4 we obtain, that the same is true in H. Since the considered wake-up pattern is generated by Procedure Fast-Broadcasting(T) working on G, we conclude that v received the source message by step $3600n(1 + \log r)\log D$, when Procedure Fast-Broadcasting(T) is executed on G. This is a contradiction which concludes the proof.

We conclude this section by observing that the assumption that eccentricity D is known to all nodes can be removed without changing our result. It is enough to apply Algorithm Fast-Broadcasting for parameter r and for eccentricities 2^{2^i}, for $i = 1, ..., \lceil \log \log r \rceil$. Broadcasting will be completed after the execution of Algorithm Fast-Broadcasting for $i = \lceil \log \log D \rceil$. Total time will be at most 4 times larger than running time of Algorithm Fast-Broadcasting when D is known.

Observe that using only substeps of type B in Procedure Fast-Broadcasting(T) we can trivially get the estimate $\mathcal{O}(nD)$ on broadcasting time. Hence the upper bound from Theorem 1 can be refined to $\mathcal{O}(n \cdot \min\{\log n \log D, D\})$.

3 Broadcasting with Unknown Bound on Network Size

In this section we show how our estimate of broadcasting time changes if nodes do not know any parameters of the network: neither its eccentricity D nor any upper bound r on the number of nodes. Denote by $AFB(x,y)$ the execution of Algorithm Fast-Broadcasting for the upper bound x on the size of the network and for eccentricity y, running in time $3600x(1+\log x)\log y$ (it exists by Theorem 1). We construct the following

Algorithm Modified-Fast-Broadcasting

$i := 1$
repeat
  $i := i+1$, $l := 1$
  while $2^{2^l} < 2^{i-l}$ **do**
    $AFB(2^{i-l}, 2^{2^l})$ (1)
    $l := l+1$
  $AFB(2^{i-l}, 2^{i-l})$ (2)

The following observations hold:

1. For every n-node graph G with parameters r and D, broadcasting on G is completed by the time when algorithm $AFB(2^{\lceil \log r \rceil}, \min\{2^{\lceil \log r \rceil}, 2^{2^{\lceil \log\log D \rceil}}\})$ is executed. This happens for $i = \lceil \log r \rceil + \lceil \log\log D \rceil$ and $l = \lceil \log\log D \rceil$, either in (1) if $\lceil \log r \rceil > 2^{\lceil \log\log D \rceil}$, or in (2) otherwise.

2. $ABF(2^{i-l}, 2^{2^l})$ performs broadcasting in $3600 \cdot 2^{i-l} \cdot (i-l+1) \cdot 2^l$ steps.
$ABF(2^{i-l}, 2^{i-l})$ performs broadcasting in $3600 \cdot 2^{i-l} \cdot (i-l+1) \cdot (i-l)$ steps.

3. Fix i. Let l_0 be the largest index l for which $AFB(2^{i-l}, 2^{2^l})$ is executed in (1). The execution of loop "while" lasts at most

$$\sum_{l=1}^{l_0} 3600 \cdot 2^{i-l} \cdot (i-l+1) \cdot 2^l \le \sum_{l=1}^{\lceil \log i \rceil} 3600 \cdot 2^{i-l} \cdot (i-l+1) \cdot 2^l \le 3600 \cdot \lceil \log i \rceil \cdot 2^i \cdot i$$

steps. The execution of (2) lasts

$$3600 \cdot 2^{i-l_0-1} \cdot (i-l_0-1+1) \cdot (i-l_0-1) \le 3600 \cdot 2^{i-l_0-1} \cdot (i-l_0+1) \cdot 2^{l_0+1}$$

steps. The latter inequality follows from the condition $2^{2^{l_0}} \ge 2^{i-l_0-1}$. We further have

$$3600 \cdot 2^{i-l_0-1} \cdot (i-l_0+1) \cdot 2^{l_0+1} = 3600 \cdot 2^{i-l_0}(i-l_0+1) \cdot 2^{l_0} \le 3600 \lceil \log i \rceil \cdot 2^j \cdot i \ .$$

4. The total number of steps till the execution of "repeat" for $i = i_0$, is at most

$$\sum_{i=2}^{i_0} 2 \cdot 3600 \cdot \lceil \log i \rceil \cdot 2^i \cdot i \le 7200 \cdot 2^{i_0+1} \cdot i_0 \cdot \lceil \log i_0 \rceil \ ,$$

since (2) lasts at most the same time as the last preceding execution of (1).

5. For any r and D, the total time of Algorithm Modified-Fast-Broadcasting is at most

$$7200 \cdot 2^{\lceil \log r \rceil + \lceil \log \log D \rceil + 1} \cdot (\lceil \log r \rceil + \lceil \log \log D \rceil) \cdot \lceil \log(\lceil \log r \rceil + \lceil \log \log D \rceil) \rceil \ ,$$

which is in $\mathcal{O}(r \log r \log \log r \log D) \subseteq \mathcal{O}(n \log n \log \log n \log D)$. This proves the following theorem.

Theorem 2. *Algorithm Modified-Fast-Broadcasting completes broadcasting on any n-node network of eccentricity D, in time $\mathcal{O}(n \log n \log \log n \log D)$, even when nodes do not know any parameters of the network or any bound on its size.*

Similarly as in Section 2, the above upper bound can be refined to $\mathcal{O}(n \cdot \min\{\log n \log \log n \log D, D\})$.

4 Optimal Broadcasting in Complete Layered Networks

In [12] the authors prove a lower bound $\Omega(n \log D)$ on deterministic broadcasting time on any n-node network of eccentricity D. This is done using complete layered networks. All nodes of such networks can be partitioned into layers L_0, L_1,..., L_D where L_0 consists of the source and the set of directed edges is $\{(v,w) : v \in L_i, w \in L_{i+1}, i = 0, 1, ..., D-1\}$. More precisely, it is shown in [12] that for every deterministic broadcasting algorithm there is a complete layered n-node network of eccentricity D, such that this algorithm requires time $\Omega(n \log D)$ to perform broadcast on this network. This result holds even when n and D are known to all nodes.

In this section we present a deterministic broadcasting algorithm which works on every complete layered n-node network of eccentricity D in time $\mathcal{O}(n \log D)$, and thus it is optimal. Hence, any lower bound sharper than $\Omega(n \log D)$, on broadcasting time in arbitrary radio networks, would have to be established for graphs more complicated than complete layered networks. Our result is also an improvement of the upper bound $\mathcal{O}(n \log n)$, proved in [11] for n-node complete layered networks.

We use the following definition of an (r,k)-selective family. A family $\mathcal{F}$ of subsets of R is called (r,k)-*selective*, for $k \le r$, if for every subset Z of $\{1, \ldots, r\}$, such that $|Z| \le k$, there is a set $F \in \mathcal{F}$ and element $z \in Z$, such that $Z \cap F = \{z\}$.

Lemma 5. [**12**] *For every $r \ge 2$ and $k \le r$, there exists an (r,k)-selective family $\mathcal{F}$ of size $\mathcal{O}(k \log((r+1)/k))$.*

Let $\mathcal{F}_i$ denote a $(r, 2^i)$-selective family, for $i = 1, \ldots, \log r$. By Lemma 5, we can assume, that $f_i = |\mathcal{F}_i| \le \alpha 2^i \log((r+1)/2^i)$, for some constant $\alpha > 0$, and for all $i = 1, \ldots, \log r$. Let $\mathcal{F}_i = \{F_i(1), \ldots, F_i(f_i)\}$.

Algorithm Complete-Layered
```
for i = 1, ..., log r do
  for j = 1, ..., f_i do
      if v ∈ F_i(j) then v transmits
for j = 1, ..., r do
  if v = j then v transmits
```

Theorem 3. *Algorithm Complete-Layered completes broadcasting in $\mathcal{O}(n \log D)$ time, for any n-node complete layered network of eccentricity D.*

Proof. For $D = 1$ the proof is obvious. Assume $D \geq 2$. Fix an n-node complete layered network G of eccentricity D. Let L_l denote the l-th layer of G, and $d_l = |L_l|$, for $l = 0, \ldots, D$. Let t_l denote the step in which all nodes in L_l received the source message for the first time.

Claim. $t_{l+1} - t_l \leq 4\alpha d_l \log\big(2(r+1)/d_l\big)$, for every $l = 0, \ldots, D-1$.

In step $t_l + 1$ all nodes in L_l start transmitting. After at most

$$\begin{aligned}
\sum_{i=1}^{\lceil \log d_l \rceil} f_i &\leq \alpha \sum_{i=1}^{\lceil \log d_l \rceil} 2^i \log((r+1)/2^i) \\
&\leq \alpha \Big[\sum_{i=1}^{\lceil \log d_l \rceil} 2^i \log(r+1) - \sum_{i=1}^{\lceil \log d_l \rceil} i \cdot 2^i \Big] \\
&\leq \alpha \Big[2^{\lceil \log d_l \rceil + 1} \log(r+1) - (2^{\lceil \log d_l \rceil + 1} \lceil \log d_l \rceil - 2^{\lceil \log d_l \rceil + 1} + 1) \Big] \\
&\leq \alpha 2^{\lceil \log d_l \rceil + 1} \log \frac{2(r+1)}{d_l} \\
&\leq 4\alpha d_l \log \frac{2(r+1)}{d_l}
\end{aligned}$$

steps, all nodes in L_l complete transmissions according to the selective family $\mathcal{F}_{\lceil \log d_l \rceil}$. By definition of $\mathcal{F}_{\lceil \log d_l \rceil}$, there is a step among $t_l + 1, \ldots, t_l + \lfloor 4\alpha d_l \log \frac{2(r+1)}{d_l} \rfloor$ such that exactly one node in L_l transmits in this step. Consequently all nodes in L_{l+1} get the source message by step $t_l + \lfloor 4\alpha d_l \log \frac{2(r+1)}{d_l} \rfloor$. This completes the proof of the Claim.

Since $\sum_{l=0}^{D} d_l = n$ and $t_0 = 0$, we have

$$\begin{aligned}
t_D = \sum_{l=0}^{D-1} (t_{l+1} - t_l) \quad &\leq \quad 4\alpha \sum_{l=0}^{D-1} d_l \log \frac{2(r+1)}{d_l} \\
= 4\alpha \sum_{l=0}^{D-1} \left(\log \frac{(r+1)^{d_l}}{d_l^{d_l}} + d_l \right) \quad &\leq \quad 4\alpha \log \frac{(r+1)^{n-d_D}}{\Pi_{l=0}^{D-1} d_l^{d_l}} + 4\alpha n \\
\leq 4\alpha(n - d_D) \log \frac{r+1}{(n-d_D)/D} + 4\alpha n \; .
\end{aligned}$$

We used the fact, that

$$\Pi_{l=0}^{D-1} d_l^{d_l} \geq \left(\frac{n - d_D}{\sum_{l=0}^{D-1} d_l \cdot \frac{1}{d_l}} \right)^{n-d_D} = \left(\frac{n-d_D}{D} \right)^{n-d_D} ,$$

which follows from the inequality between geometric and harmonic averages.

Since, for $D \geq 2$, the function $x \cdot \log \frac{D(r+1)}{x}$ is increasing for $x \leq r+1$, we have

$$4\alpha(n - d_D) \log \frac{r+1}{(n-d_D)/D} + 4\alpha n \leq 4\alpha n \log \frac{r+1}{n/D} + 4\alpha n \in \mathcal{O}(n \log D) \; .$$

References

1. Alon, N., Bar-Noy, A., Linial, N., Peleg, D.: A lower bound for radio broadcast. Journal of Computer and System Sciences 43 (1991) 290–298
2. Bar-Yehuda, R., Goldreich, O., Itai, A.: On the time complexity of broadcast in radio networks: an exponential gap between determinism and randomization. Journal of Computer and System Sciences 45 (1992) 104–126
3. Basagni, S., Bruschi, D., Chlamtac, I.: A mobility-transparent deterministic broadcast mechanism for ad hoc networks. IEEE/ACM Trans. on Networking 7 (1999) 799–807
4. Basagni, S., Myers A.D., Syrotiuk, V.R.: Mobility-independent flooding for real-time multimedia applications in ad hoc networks. Proc. 1999 IEEE Emerging Technologies Symposium on Wireless Communications & Systems, Richardson, TX
5. Bruschi, D., Del Pinto, M.: Lower bounds for the broadcast problem in mobile radio networks. Distributed Computing 10 (1997) 129–135
6. Chlamtac, I., Faragó, A.: Making transmission schedule immune to topology changes in multi-hop packet radio networks. IEEE/ACM Trans. on Networking 2 (1994) 23–29
7. Chlamtac, I., Faragó, A., Zhang, H.: Time-spread multiple access (TSMA) protocols for multihop mobile radio networks. IEEE/ACM Trans. on Networking 5 (1997) 804–812
8. Chlamtac, I., Weinstein, O.: The wave expansion approach to broadcasting in multihop radio networks. IEEE Trans. on Communications 39 (1991) 426–433
9. Chlebus, B., Gąsieniec, L., Gibbons, A., Pelc, A., Rytter, W.: Deterministic broadcasting in unknown radio networks. Distributed Computing 15 (2002) 27–38
10. Chlebus, B., Gąsieniec, L., Östlin, A., Robson, J.M.: Deterministic radio broadcasting. Proc. 27th Int. Coll. on Automata, Languages and Programming (ICALP'2000), LNCS 1853, 717–728
11. Chrobak, M., Gąsieniec, L., Rytter, W.: Fast broadcasting and gossiping in radio networks. Proc. 41st Symposium on Foundations of Computer Science (FOCS'2000), 575–581
12. Clementi, A., Monti, A., Silvestri, R.: Selective families, superimposed codes, and broadcasting on unknown radio networks. Proc. 12th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'2001), 709–718
13. De Marco, G., Pelc, A.: Faster broadcasting in unknown radio networks. Information Processing Letters 79 (2001) 53–56
14. Gaber, I., Mansour, Y.: Broadcast in radio networks. Proc. 6th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'1995), 577–585
15. Indyk, P.: Explicit constructions of selectors and related combinatorial structures with applications. Proc. 13th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'2002), 697–704
16. Kowalski, D., Pelc, A.: Deterministic broadcasting time in radio networks of unknown topology. Proc. 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'2002), 63-72.
17. Kushilevitz, E., Mansour, Y.: An $\Omega(D \log(N/D))$ lower bound for broadcast in radio networks. SIAM J. on Computing 27 (1998) 702–712
18. Peleg, D.: Deterministic radio broadcast with no topological knowledge. Manuscript (2000)