Deterministic broadcasting in unknown radio networks

Bogdan S. Chlebus* Leszek Gąsieniec†§ Alan Gibbons† Andrzej Pelc‡
Wojciech Rytter*†

Abstract

We consider the problem of distributed deterministic broadcasting in radio networks of unknown topology and size. The network is synchronous. If a node u can be reached from two nodes which send messages in the same round, none of the messages is received by u. Such messages block each other and node u either hears the noise of interference of messages, enabling it to detect a *collision*, or does not hear anything at all, depending on the model. We assume that nodes are completely ignorant of the network: they know neither its topology, nor size, nor even their immediate neighborhood. The initial knowledge of every node is limited to its own label. We study the time of deterministic broadcasting under this total ignorance scenario.

Previous research has concentrated on distributed randomized broadcasting algorithms working for unknown networks, and on deterministic off-line broadcasting algorithms assuming full knowledge of the radio network. Ours are the first broadcasting algorithms simultaneously distributed and deterministic, that work for arbitrary totally unknown radio networks.

The results for the model without collision detection: We develop a linear-time broadcasting algorithm for symmetric graphs, which is optimal. For arbitrary n-node graphs, we prove a lower bound $\Omega(D \log n)$, where D is the diameter, and develop an algorithm working in time $O(n^{11/6})$. Next we show that broadcasting with acknowledgement is not possible in this model at all.

For the model with collision detection, we develop efficient algorithms for broadcasting and for acknowledged broadcasting in strongly connected graphs.

*Instytut Informatyki, Uniwersytet Warszawski, Banacha 2, 02–097 Warszawa, Poland, {chlebus,rytter}@mimuw.edu.pl

†Department of Computer Science, University of Liverpool, Peach Street, Chadwick Bldg, Liverpool L69 7ZF, United Kingdom, {leszek,gibbons,rytter}@csc.liv.ac.uk

‡Département d'Informatique, Université du Québec à Hull, Hull, Québec J8X 3X7, Canada, Andrzej_Pelc@uqah.uquebec.ca
Research supported in part by NSERC grant OGP 0008136 and EPSRC grant GR/M37561. This research was done during this author's stay at University of Liverpool.

§Research supported in part by NUF-NAL (The Nuffield Foundation Awards to Newly Appointed Lecturers) award.

1 Introduction

Increasing interest in radio communication networks is due to their expanding applications, such as cellular phones and wireless local area networks. Low cost of infrastrucure and the flexibility of radio networks make them an attractive alternative to other types of communication media.

A radio network is a collection of transmitter-receiver devices (referred to as *nodes*). Every node can *reach* a given subset of other nodes, depending on the power of its transmitter and on topographic characteristics of the surrounding region. A radio network can thus be conveniently modeled by its *reachability graph* in which the existence of a directed edge (uv) means that node v can be reached from u. In this case u is called a *neighbor* of v. If the power of all transmitters is the same then the reachability graph is symmetric.

Nodes send messages in synchronous *rounds* or *time slots*, measured by a global clock which indicates the current round number. In every round every node acts either as a *transmitter* or as a *receiver*. A node acting as a receiver in a given round gets a message iff exactly one of its neighbors transmits in this round. The message received is the one that was transmitted. If at least two neighbors v and v' of u transmit in a given round, none of the messages is received by u in this round. In this case we say that a *conflict* or a *collision* occurred at u.

One of the fundamental tasks in network communication is *broadcasting*. Its goal 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 directed paths in the network. It is assumed that there exists a directed path from the source to any node. In the sequel we restrict attention to such graphs only. One of the most important performance parameters of a broadcasting scheme is the total time, i.e., the number of rounds it uses to inform all the nodes of the network.

1.1 Previous work In most of the research on broadcasting in radio networks [1, 4, 6, 8, 12] the network is modeled as an undirected graph, which corresponds to the assumption that the reachability graph is symmetric. A lot of effort has been devoted to finding good upper and lower bounds on the 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 [8] it was proved that broadcasting can be done in time $O(D + \log^5 n)$, for any n-node network of diameter D. In [11] the authors discussed fault-tolerant broadcasting in radio networks arising from geometric locations of nodes on the line and in the plane.

In the above papers, the topology of the radio networks was known in advance, and broadcasting algorithms were deterministic and centralized. On the other hand, in [4] a randomized protocol was given for arbitrary radio networks where nodes have no topological knowledge of the network, not even about neighbors. This randomized protocol runs in expected time $O(D \log n + \log^2 n)$. In [12] it was shown that for any randomized broadcast protocol and parameters D and n, there exists an n-node network of diameter D requiring expected time $\Omega(D \log(n/D))$ to execute this protocol.

The above results leave unexplored the scenario of broadcasting protocols which are simultaneously distributed and deterministic, and are to operate correctly in radio networks whose nodes have no knowledge of the network, or even of their immediate neighborhood. In [4] the authors established a lower bound $\Omega(n)$ on deterministic broadcasting time in such networks. The first paper to deal with broadcasting protocols in this situation was [7]. However, radio networks considered there were quite restrictive: the authors considered the situation when nodes are situated on a line, and every node can reach all nodes within a certain distance. Deterministic broadcasting in unknown networks for models other than radio communication have been studied in [3, 10].

1.2 Models In many applications, nodes of a radio network are mobile, and hence the topology of the network as well as its size may change over time. Hence it is important to design algorithms that do not assume any knowledge that nodes may have about the network, and which nevertheless work efficiently for arbitrary networks.

In this paper we study execution time of broadcasting algorithms that are deterministic and distributed, and work in arbitrary radio networks of unknown topology and size. Thus the initial situation is that of complete ignorance concerning the network: the *a priori* knowledge of every node is limited to its own label. Labels are distinct integers between 1 and $Z = O(n)$, for n-node networks. (The existence of distinct labels is necessary. If the radio network is anonymous, it is clear that deterministic broadcasting cannot be done even in the 4-cycle.) We study the general case when networks can be arbitrary directed graphs, and also the more restrictive scenario of symmetric graphs, as in [1, 8, 12]. This is the first paper to develop deterministic broadcasting protocols in radio networks under such weak assumptions.

If all the necessary information about the network is available to nodes, time of broadcasting can be known in advance and thus all nodes can be aware of the termination of broadcasting as soon as it is completed. A different situation occurs in our setting. Since even the size of the network is unknown, broadcast can well be finished but no node need be aware of this fact. Consequently, we study the following two communication tasks: In *radio broadcasting* (RB) the goal is simply to communicate the source message to all nodes. In *acknowledged radio broadcasting* (ARB) the goal is to achieve RB and inform the source about it. This may be essential, e.g., when the source has several messages to disseminate, and none of the nodes should learn the next message until all nodes get the previous one.

An algorithm accomplishes RB in t rounds, if all nodes know the source message after round t, and no messages are sent after round t. An algorithm accomplishes ARB in t rounds, if it accomplishes RB in t rounds, and if, after round t, the source knows that all nodes know the source message. (The algorithm starts in round 1 and the current round number is indicated by the global clock.) It should be noted that if ARB is accomplished after $O(t)$ rounds then a more demanding goal of additionally informing all nodes that RB is accomplished can also be achieved in $O(t)$ rounds: the source can simply broadcast the information *done* after learning it.

Since nodes are initially ignorant of the network, it is not surprising that efficiency of broadcasting depends on how fast they can acquire some knowledge of it. Our algorithms are *adaptive*, i.e., a node can schedule its future actions on the basis of its local history. A node can obviously gain knowledge from previously obtained messages. There is, however, another potential way of learning useful information. The availability of this method depends on what exactly happens during a conflict, i.e., when u acts as a receiver and two or more neighbors of u transmit simultaneously. As mentioned above, u does not get any of the messages in this case. However, two scenarios are possible. Node u may either hear nothing (except for the background noise), or it may receive *interference noise* different from any message received properly but also different from background noise. These two scenarios are often referred to as the absence (resp. availability) of *collision detection* (cf., e.g., [4]). Which of the two scenarios occurs in a particular situation may depend on technological characteristics of the transmitter/receiver devices used by the nodes. A discussion justifying both scenarios can be found in [9, 4]. We will see that efficiency and even feasibility of a particular communication task are significantly influenced by the choice between these scenarios.

1.3 Overview of results

A Model without collision detection Our main results are in this model. We first give an algorithm which performs radio broadcasting for all symmetric graphs in time $O(n)$, where n is the number of nodes in the graph. While it is not difficult to design such an algorithm assuming that nodes know their neighborhood, it should be recalled that we do not assume this knowledge and we do not even assume the knowledge of any bound on network size. In coping with this total ignorance of the network, the main problem becomes assuring that every node has learned its complete neighborhood. Since the lower bound $\Omega(n)$ from [4] holds for the same model, our algorithm is asymptotically optimal.

For arbitrary directed graphs we give a broadcasting algorithm running in time $O(n^{11/6})$. It is based on a combinatorial idea using Vishkin's deterministic sample [15]. We also prove a lower bound $\Omega(D \log n)$. More precisely, for any parameters D and n, with $D < 2n/3$, and any deterministic broadcasting algorithm, we show an n-node graph of diameter D on which this algorithm requires time $\Omega(D \log n)$. Since D can be of order $\Theta(n)$, this lower bound shows that the assumption about graph symmetry, used to design the linear time algorithm, is essential. On the other hand, our lower bound should be compared to the weaker lower bound $\Omega(D \log(n/D))$ obtained in [12] for expected running time of randomized broadcasting algorithms.

Our last result in this model is an impossibility result. We show that no deterministic algorithm can perform acknowledged radio broadcasting, even for symmetric graphs.

B Model with collision detection For the class of strongly connected graphs we show a radio broadcasting algorithm working in time $O(n \cdot ecc)$, where ecc is the eccentricity of the source in the graph, i.e., the largest distance from the source to any other node. We also observe how the collision detection capability can be used to code messages. We show a simple scheme that broadcasts a message of size l in time $O(l \cdot ecc)$, for arbitrary graphs. Hence it is asymptotically optimal for messages of size $O(1)$.

The impact of collision detection is even stronger for acknowledged radio broadcasting. While this is an impossible task without this capability, availability of it permits to perform ARB rather fast. For symmetric graphs we show an algorithm working in time $O(n)$ for n-node graphs. If the graph is non-symmetric, in order to make ARB possible, it must be at least strongly connected. For such graphs we show an algorithm for acknowledged radio broadcasting working in time $O(n \cdot ecc)$.

2 The model without collision detection

In this section we consider broadcasting in the model without collision detection, similarly as, e.g., in [4]. Thus, if at least two neighbors of u transmit in a given round, the effect at u is identical as if none of its neighbors transmitted.

2.1 Radio broadcasting in symmetric graphs

Algorithm EXPLORE-AND-EXPAND: The algorithm works for every symmetric graph in time $O(n)$, where n is the number of nodes. In every round, at most one node acts as a transmitter and all other nodes act as receivers, hence collisions are avoided. The algorithm works in phases, numbered by consecutive positive integers. Every phase starts in the round following the end of the previous phase. Phase k lasts $7 \cdot 2^{k-1}$ rounds divided into 3 stages. Stage A consists of 2^{k-1} rounds, stage B consists of 2^k rounds, and stage C consists of 2^{k+1} rounds. Let L_k be the set of nodes with labels 1,..., 2^k, and let G_k be the connected component of the network containing the source and induced by L_k, if the label of the source is at most 2^k. Otherwise, let G_k be the empty graph. The following invariant Inv will be maintained after phase k, for any positive integer k.

- A spanning tree T_k of G_k and an Eulerian cycle C_k on this tree are constructed.
- Nodes on the cycle are labeled by consecutive positive integers and each node knows the number(s) assigned to it in this enumeration. These numbers are called *k-tags* of the node.
- Every node in G_k knows the labels of all its neighbors in G_k.
- Every node in G_k knows the source message.

Assume that the invariant is maintained after phase $k-1$ of Algorithm EXPLORE-AND-EXPAND. The following description of phase k shows that the invariant is maintained after it as well.

Stage A Rounds in stage A of phase k are numbered by integers $2^{k-1}+1$,..., $2^{k-1}+2^{k-1}$. In round number j of stage A the node with label j acts as a transmitter and sends a message containing its label.

Stage B Rounds of this stage are numbered by integers 1,..., 2^k. A node in the cycle C_{k-1} is called *active* in a given round of stage B if it has heard a message in stage A or in an earlier round of stage B. An active node acts as a transmitter in every round of stage B whose number corresponds to its k-tag. In such a round the node sends a *contact message* which is a fixed single-bit signal.

Stage C If the source becomes active during stage B, it initiates stage C. At the beginning of this stage every node v knows its neighborhood N_v in G_k. In this stage a token initially situated in the source explores G_k in a DFS manner, and constructs a spanning tree T_k of this subgraph and an Eulerian cycle C_k on this tree. The token carries the source message and a counter, initially set to 1, which is incremented by one after each move. We use an idea similar to that from [2] in order to limit the token's moves only to edges of the spanning tree.

Every node v maintains a list Q_v containing the set of its neighbors in G_k which were not yet visited by the token. Q_v is initialized to N_v. Whenever v gets the message `visited` from some neighbor w, or whenever v gets the token from w, node v removes w from the list Q_v.

When v gets the token with counter c then:

• v assigns itself k-tag $c+1$; v sends the message $\langle$ `label(v), visited` $\rangle$;

• If $Q_v = \emptyset$ then

v sends the token with counter $c+1$ to the node w from which it got it for the first time (by

sending a message $\langle$ `token, c+1, label(w)`$\rangle$)

else

v sends the token with counter $c+1$ to the node w with the smallest label in the list Q_v. ◀

THEOREM 2.1. *Algorithm* EXPLORE-AND-EXPAND *performs RB in time $O(n)$, in any n-node graph.*

Although Algorithm EXPLORE-AND-EXPAND accomplishes radio broacasting after $O(n)$ rounds, none of the nodes is aware that the process is finished, i.e., ARB is not achieved. This is not accidental: we will prove later that this task is impossible to achieve without collision detection.

2.2 Radio broadcasting in arbitrary graphs

Lower bound We now prove that the assumption of graph symmetry is essential for the existence of a radio broadcasting algorithm working in linear time. For arbitrary graphs we prove the lower bound $\Omega(D \log n)$ on the time of radio broadcasting, where D is the diameter of the network. More precisely, for any $D < 2n/3$, and any deterministic broadcasting algorithm, we show an n-node graph of diameter D on which this algorithm requires time $\Omega(D \log n)$. Thus there exist graphs on which radio broadcasting requires time $\Omega(n \log n)$.

In order to prove this we first have to define precisely a radio broadcasting protocol. We want to capture the intuition that the behavior of any node in any given round depends only of its local history, this history consisting of the node's label and all messages previously received by it. The decision made by a node, given a particular local history, amounts to the choice: act as a receiver or act as a transmitter and transmit a message m (which is coded as a finite binary sequence).

We first define an *input* in round t. This is either a sequence $(t, (t_1, m_1), ..., (t_k, m_k))$, or a sequence $(t, \sigma, (t_1, m_1), ..., (t_k, m_k))$ where $t_1 < \cdots < t_k < t$ are positive integers, σ is an arbitrary finite binary string and m_i are arbitrary messages (coded as finite binary strings). The intuitive meaning of an input is the following. The input of a given node $v \neq s$ in round t is the entire information v obtained from its neighbors before this round, during the execution of the protocol, together with the number of the current round, read on the global clock. Integers t_i are rounds preceding t in which exactly one of v's neighbors acted as a transmitter (in other rounds v could not hear anything), and m_i is the message transmitted by this unique neighbor in the respective round. The input of the source (in any round) contains, moreover, the source message σ. Hence the total knowledge of a node in a given round consists of its input in this round, and of its label.

Let $\mathcal{I}$ denote the set of all inputs and $\mathcal{N}$ the set of all positive integers. Let ρ, τ, θ and ϕ be special symbols whose meaning will be specified below. Let M be the set of pairs (τ, m) and let K be the set of pairs (ϕ, m), where m is an arbitrary finite binary string.

An RB-protocol is a function $\mathcal{P} : \mathcal{I} \times \mathcal{N} \rightarrow (\{\rho\} \cup M) \times (\{\theta\} \cup K)$. $\mathcal{P}(I, l)$ is the output of the protocol $\mathcal{P}$ in round t at a node with label l, if the input of this node in round t is I. This output is a pair $(\mathcal{P}_1(I, l), \mathcal{P}_2(I, l))$, where $\mathcal{P}_1(I, l) \in \{\rho\} \cup M$ and $\mathcal{P}_2(I, l) \in \{\theta\} \cup K$. $\mathcal{P}_1(I, l)$ represents the decision of the node how to act: it is either ρ (act as a receiver) or (τ, m) (act as a transmitter and transmit message m). $\mathcal{P}_2(I, l)$ represents the state of knowledge of the node: it is either θ (I am yet uninformed), or (ϕ, m) (I know the source message and I think it is m). The protocol $\mathcal{P}$ achieves RB by round t, if $\mathcal{P}_2(I, l) = (\phi, \sigma)$ in round t, for all nodes of the network, where σ is the source message (included in the input of the source), i.e, if, in round t every node knows the (correct) source message.

Thus an RB-protocol tells a node with a given label and given input in a given round, what it should do: act as a receiver or transmit a particular message. It also specifies if the node should consider that it knows the source message. We say that a radio network executes an RB-protocol $\mathcal{P}$ if all nodes make decisions according to the function $\mathcal{P}$.

The following lower bound holds even if every node knows the size n of the network.

THEOREM 2.2. *Let D and n be positive integers such that $D < 2n/3$. For any RB-protocol $\mathcal{P}$ there exists an n-node graph of diameter D with source s, such that $\mathcal{P}$ works in time $\Omega(D \log n)$ on G.*

Proof. Fix D, n and $\mathcal{P}$ as in the formulation of the theorem. Assume that D is even and let $D = 2k$. We consider only the case $n = 3k + 1$. (It is easy to modify the argument for odd D or larger n.) The graph G is defined as follows. Its set of nodes is $V = \{s, a_i, b_i, c_i : i = 1, ..., k\}$ and its set of directed edges is $E = \{(s, a_1), (s, b_1)\} \cup \{(a_i, c_i), (b_i, c_i) : i = 1, ..., k\} \cup \{(c_i, a_{i+1}), (c_i, b_{i+1}) : i = 1, ..., k-1\}$. The node s is the source. Clearly $D = 2k$ is the diameter of G. Note that the topology of the graph does not depend on the protocol $\mathcal{P}$. However, the protocol will determine labels of nodes of G. The essence of the proof is to assign labels in such a way as to force $\mathcal{P}$ to use time $\Omega(D \log n)$ on G.

We define the following integers $t_i : i = 1, ..., k$: $t_i = \lfloor \frac{1}{2} \log(n - k - 1 - 2(i-1)) \rfloor - 1$. Assign label 1 to s and label $i+1$ to c_i, for $i = 1, ..., k$. We will assign labels to a_i and b_i by induction in such a way that node c_i will be unable to learn correctly the source message before time $t_1 + \cdots + t_i$. This will follow from the fact that nodes a_i and b_i (the only neighbors of c_i) will either both act as transmitters or both as receivers in rounds $t_1 + \cdots + t_{i-1} + 1, ..., t_1 + \cdots + t_i$, and consequently c_i will not get any message in these rounds.

Suppose that labels of nodes a_j and b_j, for $j < i$, are already assigned. The input of c_i in round t depends on t and on the source message σ (and, of course, on the protocol $\mathcal{P}$ fixed in advance). Denote this input by $J(t, i, \sigma)$. Likewise, the inputs of nodes a_i and b_i in round t depend on t and on the source message σ. Note that, in every round t and for every σ, the inputs of nodes a_i and b_i are identical. Denote this common input by $I(t, i, \sigma)$.

Fix two distinct source messages σ' and σ''. We prove the following Claim by induction on i:

Claim For any $t \le t_1 + \cdots + t_i$, $J(t, i, \sigma') = J(t, i, \sigma'')$.

Suppose the Claim is true for $j < i$. Thus, for any $t \le t_1 + \cdots + t_{i-1}$, we have $J(t, i-1, \sigma') = J(t, i-1, \sigma'')$. Hence, for any $t \le t_1 + \cdots + t_{i-1}$, we have $I(t, i, \sigma') = I(t, i, \sigma'')$. It follows that, regardless of the labels assigned to a_i and b_i, we have $J(t, i, \sigma') = J(t, i, \sigma'')$, for any $t \le t_1 + \cdots + t_{i-1}$. In particular, $J(t^*, i, \sigma') = J(t^*, i, \sigma'')$, for $t^* = t_1 + \cdots + t_{i-1}$.

Now consider any round t such that $t_1 + \cdots + t_{i-1} < t \le t_1 + \cdots + t_{i-1} + t_i$. Denote by T the set of these rounds. Let L be the set of labels assigned to nodes a_j and b_j, for $j < i$. The set $R = \{k+2, k+3, ..., n\} \setminus L$ of still available labels has size $n - k - 1 - 2(i-1)$. Since there are $t_i = \lfloor (\log(n - k - 1 - 2(i-1)))/2 \rfloor - 1$ rounds in T, it follows that there are at least $\lceil \sqrt{n - k - 1 - 2(i-1)} \rceil$ labels l in R such that $\mathcal{P}_1(I(t, i, \sigma'), l)$ are all equal to ρ or are all in M, for any $t \in T$. Denote the set of those labels by R'. By the same argument, there are at least two labels l_1 and l_2 in R', such that $\mathcal{P}_1(I(t, i, \sigma''), l_1)$ and $\mathcal{P}_1(I(t, i, \sigma''), l_2)$ are both equal to ρ or are both in M, for any $t \in T$. Assign labels l_1 and l_2 to nodes a_i and b_i, respectively. If the source message is σ' or σ'' and t is a round in T, both nodes a_i and b_i act as transmitters or both act as receivers. In any case, node c_i does not get any message in these rounds. More precisely, we have $J(t, i, \sigma') = J(t^*, i, \sigma')$ and $J(t, i, \sigma'') = J(t^*, i, \sigma'')$, for any $t \in T$. In view of $J(t^*, i, \sigma') = J(t^*, i, \sigma'')$ we get $J(t, i, \sigma') = J(t, i, \sigma'')$, for any $t \in T$. We already showed the same equality for any $t \le t_1 + \cdots + t_{i-1}$. This implies the equality for all $t \le t_1 + \cdots + t_i$, which proves the claim by induction.

We can now conclude the proof of the theorem. Suppose that $\mathcal{P}$ is a protocol achieving RB by round $f = t_1 + \cdots + t_k$. Consider the node c_k (which has label $k+1$). Since $J(f, k, \sigma') = J(f, k, \sigma'')$, we have $\mathcal{P}(J(f, k, \sigma'), k+1) = \mathcal{P}(J(f, k, \sigma''), k+1)$ and hence $\mathcal{P}_2(J(f, k, \sigma'), k+1) = \mathcal{P}_2(J(f, k, \sigma''), k+1)$. If this value is θ we have a contradiction, as node c_k is not yet informed. If this value belongs to K, we have $(\phi, \sigma') = \mathcal{P}_2(J(f, k, \sigma'), k+1) = \mathcal{P}_2(J(f, k, \sigma''), k+1) = (\phi, \sigma'')$, thus implying $\sigma' = \sigma''$, again a contradiction.

Hence $\mathcal{P}$ uses time at least $t_1 + \cdots + t_k$. By the definition of t_i, this is $\Omega(D \log n)$. $\square$

We do not know what is the optimum complexity of a radio broadcasting algorithm working for arbitrary graphs. Below we give a simple algorithm, working in time $O(n^2)$ for any n-node graph, and a more subtle algorithm with complexity $O(n^{11/6})$.

Algorithm SIMPLE-SEQUENCING: The algorithm works in consecutive phases numbered by positive integers. Phase k consists of 2^k segments, each consisting of 2^k rounds. Rounds in each segment are numbered by integers $1, ..., 2^k$. (This number should not be confused with the global number of the round.) The first segment of phase 1 starts in round 1.

Every node acts as a transmitter in exactly one round (after getting the source message), in all other rounds it acts as a receiver. A node acting as a transmitter sends the source message. In every round at most one node acts as a transmitter, hence collisions are avoided. The node with label i that got the source message in round r waits till the first round $r' > r$ whose number in its segment is i and acts as a transmitter in this round. ◀

THEOREM 2.3. *Algorithm* SIMPLE-SEQUENCING *performs RB in time $O(n^2)$, in any n-node graph.*

Our next algorithm improves on the time complexity of Algorithm SIMPLE-SEQUENCING by using two essential tools: one is the notion of *k-selective* families, and the other is that of *deterministic sample*, due to Vishkin [15] and used successfully in parallel string-matching.

Let $[a] = \{1, 2, \ldots, a\}$. A family $\mathcal{F}$ of subsets of U is said to be *k-selective* for the set U iff for any $X \subseteq U$, $|X| \le k$, there is a set $Y \in \mathcal{F}$ satisfying $|X \cap Y| = 1$.

We will construct a small k_m-selective family for the set $[2^m]$, with large k_m. The crucial tool is that of a *deterministic sample.* Denote by $\mathcal{P}_r(m)$ the family of all subsets of cardinality r of $[m]$. A deterministic sample with parameters r, m is a member of the set

$$DetSamples(r,m) = \{(D,\alpha) : D \in \mathcal{P}_r(m), \alpha \in \{0,1\}^r\}.$$

For $\Delta = (D,\alpha) \in DetSamples(r,m)$ define $STRINGS(\Delta)$ as the set of all binary strings of length m whose sequence of bits on positions in D, read in increasing order of indices, equals α.

LEMMA 2.1. *[deterministic-samples] [15]. The family* $\{STRINGS(\Delta)\} : \Delta \in DetSamples(r,m)\}$ *is* 2^r-*selective for the set* $\{0,1\}^m$.

LEMMA 2.2. $|DetSamples(\lceil m/6 \rceil, m)|$ *is* $O(2^{5m/6})$.

Proof. We estimate the number of deterministic samples $\Delta = (D,\alpha)$. There are $2^{\lceil m/6 \rceil}$ possible strings α and $\binom{m}{\lceil m/6 \rceil}$ possible subsets D. By the Stirling formula we have, for some constant c:
$\binom{m}{\lceil \frac{m}{6} \rceil} \le c \cdot \frac{m^m}{(\frac{m}{6})^{\frac{m}{6}} \cdot (m(1-\frac{1}{6}))^{m(1-\frac{1}{6})}} = c \cdot (6^{\frac{1}{6}} \cdot (\frac{6}{6-1})^{\frac{6-1}{6}})^m.$
Since $6^{\frac{1}{6}} \cdot (\frac{6}{5})^{\frac{5}{6}} \le 2^{2/3}$, we get $\binom{m}{\lceil \frac{m}{6} \rceil} = O(2^{\frac{2}{3}m})$, which proves the lemma. □

Lemma 2.1 for $r = \lceil m/6 \rceil$ and Lemma 2.2 yield the construction of a $2^{\lceil m/6 \rceil}$-selective family of size $O(2^{5m/6})$ on the set $[2^m]$, for any integer m, in view of the natural bijection between $\{0,1\}^m$ and $[2^m]$. For any positive integer m, fix such a family $\mathcal{F}_m$ and let $f_m = |\mathcal{F}_m|$ and $k_m = 2^{\lceil m/6 \rceil}$.

Procedure m-SEGMENT: Let $F_1, \ldots, F_{f_m}$ be an arbitrary enumeration of $\mathcal{F}_m$. The procedure works in $k_m f_m + 2^m$ rounds and consists of two parts: the *head* composed of k_m consecutive sequences, called *windows*, each of f_m rounds corresponding to the appropriate sets F_i, followed by the *tail* composed of 2^m rounds. In any round of the head, corresponding to set F_i, every *active* node with label $l \in F_i$ which knows the source message, acts as a transmitter and sends the source message; all other nodes act as receivers. In the lth round of the tail, the node v with label l acts as a transmitter, sends the source message and becomes *inactive*, provided that it is *active* and knows the source message; all other nodes act as receivers.

Algorithm SELECTIVE-BROADCASTING: The algorithm works in consecutive phases numbered by positive integers. At the beginning of the first phase all nodes are active. Phase m consists of $\lceil 2^m/k_m \rceil$ consecutive calls of procedure m-SEGMENT described above. ◀

A node v is called an m-node if there exists a directed path from the source s to v, whose all nodes, including s and v, have labels at most 2^m. The following lemma is proved in the appendix.

LEMMA 2.3. *After phase m of Algorithm* SELECTIVE-BROADCASTING *all m-nodes are inactive.*

Proof. We prove the lemma by induction on m. Assume it is true for all $m' < m$. An m-node is called *waiting* if it knows the source message, and is still active. An m-node is called *m-border* if it is not an $(m-1)$-node but has a neighbor which is an $(m-1)$-node. By the inductive hypothesis, all m-border nodes are waiting at the beginning of phase m. Consider a fixed call S of procedure m-SEGMENT in phase m. We will show that the number of waiting nodes at the beginning of the tail of S is at least $\min(k_m, x)$, where x is the total number of m-nodes that are still active. (This implies that after $\lceil 2^m/k_m \rceil$ calls of the procedure all m-nodes are inactive.) Consider two cases. If at the beginning of some window in S the number of waiting m-nodes is larger than k_m, all these nodes will be waiting at the beginning of the tail of S. Otherwise, in each window the number of waiting nodes increases, in view of k_m-selectivity of $\mathcal{F}_m$. Hence at least k_m nodes are waiting after k_m windows (or all the remaining m-nodes are waiting, if less than k_m m-nodes are still active). □

THEOREM 2.4. *Algorithm* SELECTIVE-BROADCASTING *performs radio broadcasting in time $O(n^{11/6})$, in any n-node graph.*

Proof. Let m^* be such that $2^{m^*-1} < Z \le 2^{m^*}$, where Z is the maximum label. By Lemma 2.3 all nodes know the source message after phase m^* and no messages are sent after this phase. Phase m lasts $(k_m \cdot |\mathcal{F}_m| + 2^m) \cdot \lceil 2^m/k_m \rceil$ rounds. Since $|\mathcal{F}_m| = O(2^{5m/6})$, $k_m = O(2^{m/6})$ and $O(n) = O(Z) = O(2^{m^*})$, the total time of execution of phases 1 to m^* is $O(n^{11/6})$. □

Similarly as in the case of Algorithm EXPLORE-AND-EXPAND for symmetric graphs, in the case of Algorithms SIMPLE-SEQUENCING and SELECTIVE-BROADCASTING, none of the nodes is aware that the process is finished, i.e., acknowledged radio broadcasting is not achieved. In what follows we will show that ARB is in fact impossible to achieve without collision detection.

2.3 Infeasibility of acknowledged radio broadcasting We conclude this section by proving that without collision detection ARB cannot be accomplished, even for symmetric graphs. Again we need to define precisely an ARB-protocol. We use the same notation as in the definition of an RB-protocol. An ARB-protocol is defined in a similar way. It is a function $\mathcal{P} : \mathcal{I} \times \mathcal{N} \to (\{\rho\} \cup M) \times (\{\omega\} \cup \{\theta\} \cup K)$, where ω is an additional special symbol. Outputs not containing the symbol ω have the same meaning as before. The output $\mathcal{P}(I, s) = (\mathcal{P}_1(I, s), \mathcal{P}_2(I, s))$, where s is the label of the source, I is its input in round t, and $\mathcal{P}_2(I, s) = \omega$, means that the source considers RB to be finished by round t.

The protocol $\mathcal{P}$ achieves ARB by round t, if the following two conditions are satisfied:

- $\mathcal{P}_2(I_s, s) = \omega$, where s is the label of the source and I_s is the input of the source in round t.
- for any round t', if $\mathcal{P}_2(J_s, s) = \omega$, where J_s is the input of the source in round t', then $\mathcal{P}_2(J_v, v) = (\phi, \sigma)$, where J_v is the input of the node with label v in round t', for all labels v of nodes other than the source, where σ is the source message (included in the input of the source).

The above formalism captures the property that ARB is accomplished by round t if, in round t, the source considers RB to be accomplished, and if, in any round, such a conviction of the source implies that RB is indeed accomplished by this round.

The following result states the infeasibility of ARB, even in the class of symmetric graphs.

THEOREM 2.5. *For any ARB-protocol $\mathcal{P}$ there exists a symmetric graph on which $\mathcal{P}$ is not correct.*

Proof. Fix an ARB-protocol $\mathcal{P}$ and fix two distinct source messages σ' and σ''. Let t' (resp. t'') be the first round in which $\mathcal{P}$ completes ARB on the 1-node graph consisting only of the source s, if the source message is σ' (resp. σ''). Let $t = \max(t', t'')$. Thus t is the time after which the source realizes that it is the only node of the network, if the source message is σ' or σ''. The idea of the proof is to construct a labeled graph G with at least two nodes, for which

- some nodes do not get the source message after t rounds;
- the input of the source during the first t rounds is the same as in the 1-node graph consisting of the source, i.e., the source does not get any messages in the first t rounds,

assuming that the source message is σ' or σ''.

Then, the protocol $\mathcal{P}$ working on the graph G will induce the source to falsely conclude that ARB is accomplished after t rounds.

The graph G has $2^{2t} + 4t + 2$ nodes and is defined as follows. The set of nodes is $V = \{a, b\} \cup X$, where X has size $2^{2t} + 4t$ and contains the source s. The set of directed edges is $E = \{(a, x), (x, a), (b, x), (x, b) : x \in X\}$, i.e., the graph is symmetric. Choose any node $s \in X$ as the source. Labeling will be defined below and depends on the protocol $\mathcal{P}$. Intuitively, we want to label nodes in such a way that a and b, the only neighbors of s, either both act as transmitters or both act as receivers in the first t rounds of protocol execution. The difficulty is aggravated by the fact that this property has to be satisfied in spite of the possibility that nodes in X, including the source, may spontaneously send some messages that may be received by nodes a and b, in these early rounds.

Assign label 1 to s. Let L denote the set of integers $\{2, ..., 2^{2t} + 4t + 2\}$, let T^* denote the set of integers $\{1, ..., t\}$, and let ϵ denote the empty sequence. For any round $i \in T^*$, let I_i denote the input $I(i, \epsilon)$ (of nodes other than the source), and let $J_{i,\sigma}$ denote the input $I(i, \sigma, \epsilon)$ (of the source), where σ is a source message. Hence I_i (resp. $J_{i,\sigma}$) are inputs in round i corresponding to the situation when a node other than the source (resp. the source, given source message σ) did not get any messages until round i.

Let T' be the set of those rounds $r \le t$ for which $\mathcal{P}_1(J_{r,\sigma'}, 1) \in M$, i.e., those rounds in which the source acts as a transmitter if it has not received any message yet, and if the source message is σ'. Let T'_1 be the set of those rounds $r \in T'$ for which there exists a label $c'_r \in L$ such that $\mathcal{P}_1(I_r, c'_r) \in M$. I.e., T'_1 is the set of those rounds in which the source and some other node whose label is in L act as transmitters if they have not received any message yet, and if the source message is σ'. Let $A' = \{c'_r : r \in T'_1\}$.

Let T'_2 be the set of those rounds $r \in T^* \setminus T'$ for which there exists a unique label $d'_r \in L$ such that $\mathcal{P}_1(I_r, d'_r) \in M$. I.e., T'_2 is the set of those rounds in which a unique node acts as a transmitter if it has not received any message before, and if the source message is σ'. Let $B' = \{d'_r : r \in T'_2\}$.

Finally, let T'_3 be the set of those rounds $r \in T^* \setminus T'$ for which there exist at least two labels $x_r, y_r \in L$ such that $\mathcal{P}_1(I_r, x_r) \in M$ and $\mathcal{P}_1(I_r, y_r) \in M$. I.e., T'_3 is the set of those rounds outside of T' in which at least two nodes with labels in L act as transmitters if they have not received any message before. Let $C' = \{x_r, y_r : r \in T'_3\}$.

The set $Y' = A' \cup B' \cup C'$ has size at most $2t$. Assign labels from Y' to arbitrarily chosen nodes from $X \setminus \{s\}$. Assume that the source message is σ'. This assignment implies that, regardless of how other labels from the set L are assigned to other nodes from X, the only rounds in which nodes a and b can get a message by round t are the rounds from the set $T' \setminus T'_1 \cup T'_2$.

Next, define sets T'', T''_1, T''_2, T''_3, A'', B'', C'', and Y'', similarly as T', T'_1, T'_2, T'_3, A', B', C', and Y', with the exception that source message σ' is replaced by source message σ''. Again Y'' has size at most $2t$. Assign labels from Y'' to arbitrarily chosen, yet unlabeled nodes from $X \setminus \{s\}$. Assume that the source message is σ''. This assignment implies that, regardless of how other labels from the set L are assigned to other nodes from X, the only rounds in which nodes a and b can get a message by round t are the rounds from the set $T'' \setminus T_1'' \cup T_2''$.

Let r be any round in $T' \setminus T_1'$ and let $\mathcal{P}_1(J_{r,\sigma'}, 1) = (\tau, m_r')$. Let u be any round in T_2' and let $\mathcal{P}_1(I_u, d_u') = (\tau, m_u')$. Hence, if the source message is σ', the only messages that nodes a and b can get by round t are messages m_r', for $r \in T' \setminus T_1'$, and m_u', for $u \in T_2'$. More precisely, let $r_1', ..., r_q'$ be an increasing ordering of rounds in $T' \setminus T_1' \cup T_2'$. For any round $p \leq t$ let $j(p)'$ be the largest index j such that $r_j' < p$. Hence, if the source message is σ', the inputs of nodes a and b in round p are identical and equal to $\Gamma_p' = (p, (r_1', m'_{r_1'}), ..., (r'_{j(p)'}, m'_{r'_{j(p)'}}))$.

We define a decreasing sequence of sets L_i', for $i = 0, ..., t$, by induction on i. Let $L_0' = L \setminus Y' \setminus Y''$. Thus L_0' has size at least $2^{2t} + 1$. Suppose that L_{p-1}' is already defined. Consider the following sets of labels:

$$H_1 = \{l \in L_{p-1}' : \mathcal{P}_1(\Gamma_p', l) = \rho\},$$

$$H_2 = \{l \in L_{p-1}' : \mathcal{P}_1(\Gamma_p', l) \in M\}.$$

Thus H_1 (resp. H_2) is the set of those labels in L_{p-1}' which cause a node to act as a receiver (resp. as a transmitter), given input Γ_p' in round p, if the source message is σ'. Let H be the set of maximum size among H_1 and H_2. We define $L_p' = H$. It follows that L_t' has size at least $2^t + 1$.

Now define rounds r_i'', messages $m''_{r_i''}$, and input Γ_p'' similarly as r_i', $m'_{r_i'}$, and Γ_p' but assuming that the source message is σ'' instead of σ'. Define a decreasing sequence of sets L_i'', for $i = 0, ..., t$, by induction on i. Let $L_0'' = L_t'$. Sets L_i'', for $i > 0$, are defined similarly as L_i', replacing input Γ_p' by Γ_p'' and assuming that the source message is σ''. It follows that L_t'' has size at least 2.

Let l_1 and l_2 be distinct labels in L_t''. We assign those labels to nodes a and b, respectively, and we arbitrarily assign the remaining labels in L_0' to yet unlabeled nodes in the set X.

By the definition of l_1 and l_2, nodes a and b either both act as receivers or both act as transmitters in every round $i \in T^*$, regardless of whether the source message is σ' or σ''. It follows that in any such round, the input of the source is $J_{i,\sigma'}$, if the source message is σ', and it is $J_{i,\sigma''}$, if the source message is σ''. Also, the input of any other node in X is I_i, in both cases. However, since for the 1-node graph, ARB is accomplished by round t, it means that $\mathcal{P}_2(J_{t,\sigma'}, 1) = \mathcal{P}_2(J_{t,\sigma''}, 1) = \omega$. Suppose that $\mathcal{P}$ achieves ARB in the graph G. By definition this implies that $\mathcal{P}_2(I_t, l)=(\phi, \sigma')=(\phi, \sigma'')$, for any label l of a node in $X\setminus\{s\}$. This implies $\sigma'=\sigma''$, contradiction. □

3 The model with collision detection

In this section we show how our results change when collision detection is available. Clearly, all RB algorithms described in the previous section remain valid in the present setting. We present a new RB algorithm working more efficiently for a large class of graphs. We also show that (as opposed to the scenario without collision detection) ARB is now possible for all strongly connected graphs, and can in fact be performed efficiently.

We will use the following terminology. A node v acting as a receiver in a given round *hears signal* μ if at least one of its neighbors acts as a transmitter in this round, i.e., if v hears any message or if there is a conflict between its neighbors in this round. Otherwise (if none of its neighbors acts as a transmitter in this round), v *hears signal* λ (background noise). A *contact message* is a fixed one-bit signal.

3.1 Radio broadcasting in strongly connected graphs

Algorithm Bound-and-Broadcast. The algorithm works in two stages. The aim of the first stage is that all nodes find an upper bound on the maximum label Z within a factor of 2. In the second stage the actual broadcasting is performed.

Stage 1 is divided into phases numbered by consecutive positive integers. The kth phase consists of $2^k + 1$ rounds. Rounds of the kth phase are numbered by consecutive integers 1,...,$2^k + 1$. A node is said to be *active* after the jth phase, $j > 1$, if it acted as a transmitter in some round of this phase. All nodes are active in the first phase. It will turn out that after each phase either all nodes are active or all are not active. In the first round of the kth phase all active nodes whose labels are larger than 2^k act as transmitters, sending a contact message, and all others act as receivers. In every round $i > 1$ of the kth phase all active nodes that heard signal μ in round $i - 1$ of this phase and have not heard it in any previous round of this phase, act as transmitters, sending a contact message, and all others act as receivers. The first phase after which all nodes are not active, is the last phase of the first stage. Every node can identify this phase as the first phase after which it is not active. Suppose that the number of this phase is k_0. It can be proved that $2^{k_0-1} < Z \leq 2^{k_0}$. Recall that Z is $O(n)$.

Stage 2 starts in the round following the end of Stage 1. It is divided into phases consisting of 2^{k_0} rounds, numbered by consecutive positive integers. In each phase rounds are numbered by consecutive integers 1,...,2^{k_0}. In each round, at most one node acts as a transmitter, thus collision is avoided. Every node acts as a transmitter in exactly one round, after receiving the source message. Let l_s denote the label of the source. In all rounds $i < l_s$ of the first phase all nodes act as receivers. In round l_s of the first phase the source acts as a transmitter sending the source message. Every other node with label l that got the source message in phase j waits till the first round whose number in its phase is l (this can happen in phase j or in phase $j+1$ at the latest) and acts as a transmitter, sending the source message. ◀

THEOREM 3.1. *Algorithm* BOUND-AND-BROADCAST *performs RB in any strongly connected n-node graph with eccentricity of the source ecc, in time $O(n \cdot ecc)$.*

Although Algorithm BOUND-AND-BROADCAST accomplishes RB in time $O(n \cdot ecc)$, no node is aware of this, i.e., ARB is not accomplished. We will show later that a refinement of this algorithm achieves ARB also in time $O(n \cdot ecc)$.

3.2 Radio broadcasting in arbitrary graphs When collision detection is available, this feature can be used to code messages. Any node that acts as a receiver in a given round can distinguish between signals μ and λ which can thus serve as bits for the encoding. This observation yields the following scheme working for arbitrary graphs.

Algorithm ENCODED-BROADCAST. Suppose that σ is the source message and $(a_1, ..., a_r)$ is a binary representation of σ. The algorithm proceeds in phases. The first phase consists of one round, and every other phase contains $2r+4$ rounds. The second phase starts in round 2, and every other phase starts in the round following the end of the previous phase. In each phase some nodes are *active* and all others are *passive*. Passive nodes act as receivers, while nodes active in a given phase transmit the source message during this phase, as described below. In the first phase only the source is active, it acts as a transmitter and sends the message σ. In each subsequent phase k, those nodes that heard signal μ in at least one round of phase $k-1$ and have not heard it in any phase before $k-1$, become active, and all others are passive. We will maintain the invariant that all active nodes know σ. A node active in a given phase divides it into two-round segments $b_0, b_1, ..., b_r, b_{r+1}$. In both rounds of segments b_0 and b_{r+1} every active node acts as a transmitter and sends a contact message. Consider a segment b_i, for $0 < i < r+1$. If $a_i = 0$, active nodes act as receivers in both rounds of segment b_i. If $a_i = 1$, active nodes act as receivers in the first round of segment b_i, and act as transmitters in the second round of this segment, sending a contact message. After phase k, all nodes that become active in phase $k+1$ determine the limits of phase k as the sequence of rounds between two pairs of consecutive rounds in which they heard signal μ. They divide this sequence of rounds into two-round segments, and reconstruct the message $\sigma = (a_1, ..., a_r)$ using the rule: $a_i = 0$, if signal λ was heard in both rounds of segment b_i; $a_i = 1$, if signal λ was heard in the first round of segment b_i, and signal μ was heard in the second round of this segment. ◀

THEOREM 3.2. *Algorithm* ENCODED-BROADCAST *performs RB in an arbitrary graph, in time $O(|\sigma| \cdot ecc)$, where ecc is the eccentricity of the source and $|\sigma|$ is the number of bits in the source message.*

3.3 Acknowledged radio broadcasting in symmetric graphs As opposed to the scenario without collision detection, ARB is feasible in our present setting. We first show that for symmetric graphs it can in fact be done in time $O(n)$, using a modification of Algorithm EXPLORE-AND-EXPAND.

Algorithm EXPLORE-EXPAND-CHECK. The algorithm is a modification of Algorithm EXPLORE-AND-EXPAND. As before, it works in phases, each of them divided into stages A, B, and C. Stage A of phase k has one round more than previously, added at the very end of this stage. In this round, all nodes with labels larger than 2^k act as transmitters, sending a contact message, and all nodes with labels at most 2^k act as receivers. Every node with label at most 2^k that heard signal μ in this additional round, is called *warned*.

Stage B proceeds exactly as in Algorithm EXPLORE-AND-EXPAND. Stage C is modified as follows. When a node that is warned sends the token to a neighbor, it appends a *warning message* to the token, and the neighbor getting the token becomes warned. If, at the end of phase k, the source is not warned, it knows that RB has been completed. ◀

THEOREM 3.3. *Algorithm* EXPLORE-EXPAND-CHECK *performs ARB in time $O(n)$, in any n-node symmetric graph.*

3.4 Acknowledged radio broadcasting in strongly connected graphs We now show how to modify Algorithm BOUND-AND-BROADCAST in order to accomplish ARB in time $O(n \cdot ecc)$ in any strongly connected n-node graph for which *ecc* is the eccentricity of the source. Note that strong connectivity is a necessary condition for feasibility of ARB.

Algorithm BOUND-BROADCAST-CHECK. The algorithm is a refinement of Algorithm BOUND-AND-BROADCAST. It works in three stages. Stage 1 is identical as in Algorithm Bound-and-Broadcast. Similarly as before, after its completion all nodes know an upper bound on the maximum label Z within a factor of 2. Let k_0 have the same meaning as before. We have $2^{k_0-1} < M \leq 2^{k_0}$.

Stage 2 lasts 2^{k_0} rounds, numbered 1,..., 2^{k_0}. Every node acts as a transmitter in at most one round of this stage, and sends a contact message in this round. In round 1 only the source acts as a transmitter. In round $i > 1$ those nodes which received signal μ in round $i-1$ for the first time, act as transmitters. Every node knows its distance from the source after stage 2.

The aim of Stage 3 is to broadcast the source message to all nodes, and simultaneously learn an upper bound on the eccentricity *ecc* of the source. This will make possible the termination of the algorithm after $O(n \cdot ecc)$ rounds, and informing the source (and in fact all nodes) that RB has been accomplished. Stage 3 starts in the round following the end of stage 2 and is divided into phases numbered by consecutive positive integers. Each phase consists of part 1 and part 2. Each of these parts lasts 2^{k_0} rounds, numbered 1,..., 2^{k_0}, separately for each part.

Part 1 of each phase of Stage 3 is devoted to broadcasting the source message. This is done similarly as in Stage 2 of Algorithm Bound-and-Broadcast.

Part 2 of each phase of Stage 3 is devoted to finding an upper bound on the eccentricity *ecc* of the source. This is done similarly as finding an upper bound on Z in Stage 1. The difference is that instead of using labels of nodes we use their distances from the source, learned in Stage 2, and that now phases have equal length. A node is said to be *active* after the jth phase of Stage 3, $j > 1$, if it acted as a transmitter in some round of part 2 of this phase. All nodes are active in the first phase of Stage 3. Similarly as in Algorithm Bound-and-Broadcast, after each phase either all nodes are active or all are not active. In the first round of part 2 of the kth phase all active nodes whose distances from the source are larger than k act as a transmitter, sending a contact message, and all others act as receivers. In every round $i > 1$ of part 2 of the kth phase all active nodes that heard signal μ in round $i-1$ of part 2 of this phase and did not hear it in any previous round of part 2 of this phase, act as transmitters, sending a contact message, and all others act as receivers. The first phase after which all nodes are not active terminates Stage 3 and also the entire algorithm. Every node can identify this phase as the first phase after which it is not active. ◀

Theorem 3.4. *Algorithm* Bound-Broadcast-Check *performs ARB in any strongly connected n-node graph with eccentricity of the source ecc, in time $O(n \cdot ecc)$.*

Acknowledgements. We would like to thank Marek Chrobak and Mike Robson for interesting discussions on the subject of this paper.

References

[1] N. Alon, A. Bar-Noy, N. Linial and D. Peleg, A Lower Bound for Radio Broadcast, *Journal of Computer and System Sciences* 43 (1991), 290-298.

[2] B. Awerbuch, A new distributed depth-first-search algorithm, *Information Processing Letters* 20, (1985), 147-150.

[3] B. Awerbuch, O. Goldreich, D. Peleg and R. Vainish, A Tradeoff Between Information and Communication in Broadcast Protocols, *Journal of the ACM* 37, (1990), 238-256.

[4] R. Bar-Yehuda, O. Goldreich, and A. Itai, On the time complexity of broadcast in radio networks: An exponential gap between determinism and randomization, Proc. *6th ACM Symposium on Principles of Distributed Computing* (1987), 98 - 108.

[5] R. Bar-Yehuda, A. Israeli, and A. Itai, Multiple Communication in Multi-Hop Radio Networks, Proc. *8th ACM Symposium on Principles of Distributed Computing* (1989), 329 - 338.

[6] I. Chlamtac and S. Kutten, On broadcasting in radio networks - problem analysis and protocol design, *IEEE Transactions on Communications* 33 (1985).

[7] K. Diks, E. Kranakis, D. Krizanc, A. Pelc, The impact of knowledge on broadcasting time in radio networks, Proc. *7th Annual European Symposium on Algorithms*, ESA'99, Prague, Czech Republic, July 1999, LNCS 1643, 41-52.

[8] I. Gaber and Y. Mansour, Broadcast in Radio Networks, Proc. *6th Ann. ACM-SIAM Symp. on Discrete Algorithms*, SODA'95, 577-585.

[9] R. Gallager, A Perspective on Multiaccess Channels, *IEEE Trans. on Information Theory* 31 (1985), 124-142.

[10] L. Gargano, A. Pelc, S. Pérennes and U. Vaccaro, Efficient communication in unknown networks, *Technical Report RR-3609*, January 1999, INRIA, France.

[11] E. Kranakis, D. Krizanc and A. Pelc, Fault-Tolerant Broadcasting in Radio Networks, Proc. *6th European Symposium on Algorithms*, ESA'98, Venice, Italy, August 1998, LNCS 1461, 283-294.

[12] E. Kushilevitz and Y. Mansour, An $\Omega(D\log(N/D))$ Lower Bound for Broadcast in Radio Networks, Proc. *12th Ann. ACM Symp. on Principles of Distributed Computing* (1993), 65-73.

[13] E. Kushilevitz and Y. Mansour, Computation in Noisy Radio Networks, Proc. *9th Ann. ACM-SIAM Symp. on Discrete Algorithms*, SODA'98, 236-243.

[14] K. Pahlavan and A. Levesque, *Wireless Information Networks*, Wiley-Interscience, New York, 1995.

[15] U. Vishkin, Deterministic sampling - A new technique for fast pattern matching, *SIAM J. on Computing* 20 (1991), 22-40.