Journal of Computer and System Sciences 79 (2013) 1164–1180

Contents lists available at SciVerse ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Leader election in ad hoc radio networks: A keen ear helps ☆

Dariusz R. Kowalski [a,*], Andrzej Pelc [b]

[a] Department of Computer Science, The University of Liverpool, Liverpool L69 3BX, United Kingdom
[b] Département d'informatique, Université du Québec en Outaouais, Gatineau, Québec J8X 3X7, Canada

ARTICLE INFO

Article history:
Received 9 November 2011
Received in revised form 24 March 2013
Accepted 17 April 2013
Available online 6 May 2013

Keywords:
Collision detection
Deterministic algorithms
Distributed algorithms
Leader election
Radio networks

ABSTRACT

We address the fundamental distributed problem of leader election in ad hoc radio networks modeled as undirected graphs. A signal from a transmitting node reaches all neighbors but a message is received successfully by a node, if and only if exactly one of its neighbors transmits in this round. If two neighbors of a node transmit simultaneously in a given round, we say that a collision occurred at this node. Collision detection is the ability of nodes to distinguish a collision from silence. We show that collision detection speeds up leader election in arbitrary radio networks. Our main result is a deterministic leader election algorithm working in time $O(n)$ in all n-node networks, if collision detection is available, while it is known that deterministic leader election requires time $\Omega(n \log n)$, even for complete networks, if there is no collision detection.

1. Introduction

The background and the problem. A radio network is modeled as an n-node undirected connected graph whose nodes are stations having distinct labels. Labels are integers from an interval $\{1, \dots, N\}$, where $N = \Theta(n^\gamma)$, for a constant $\gamma > 1$. We consider *ad hoc* networks in which each node knows only its own label and an upper bound N on all labels, but does not know the topology of the network or its size. Nodes do not even know their immediate neighborhood or their degree. Communication proceeds in synchronous rounds.[1] In each round each node acts either as a transmitter or as a receiver. A signal from a transmitting node reaches all neighbors. A message is *heard* (received successfully) by a node, if and only if it acts as a receiver and exactly one of its neighbors transmits in this round. If at least two neighbors of a node u transmit simultaneously in a given round, none of the messages is heard by this node and we say that a *collision* occurred at u.

An important capability of nodes of a radio network is *collision detection*: the ability of nodes to distinguish a collision from "silence", which is in fact the background noise occurring when no neighbor transmits. (This ability is the "keen ear" of nodes: a collision slightly increases the level of noise always existing in the channel and detecting this difference requires a more sensitive receiving device.) Algorithmic aspects of radio communication have been studied both assuming collision detection and without supposing this capability. Some algorithmic techniques crucially depend on collision detection: even

☆ A preliminary version of this paper appeared in the Proceedings of the 36th International Colloquium on Automata, Languages and Programming (ICALP 2009), Part II, in: Lecture Notes in Comput. Sci., vol. 5556, 2009, pp. 521–533. The work of D. Kowalski was supported by the Engineering and Physical Sciences Research Council [grant number EP/G023018/1]. The work of A. Pelc was partially supported by NSERC discovery grant and by the Research Chair in Distributed Computing at the Université du Québec en Outaouais.

* Corresponding author. Fax: + (44 151) 795 4235.
E-mail address: D.Kowalski@liverpool.ac.uk (D.R. Kowalski).

[1] In Section 4 we argue why this assumption is needed for leader election.

0022-0000/$ – see front matter
http://dx.doi.org/10.1016/j.jcss.2013.04.003

in the case when no message can be heard due to collisions, nodes can engage in "conversations" and transmit important control information using noise and silence as bits. This information can help to prepare future successful message transmissions. On the other hand, it has been shown that in some cases collision detection can be simulated in networks whose nodes do not have this capability; see, e.g., the procedure Echo designed in [1].

Hence it is natural to ask how crucial is collision detection for the efficiency of performing tasks in radio networks. More precisely, can the availability of collision detection speed up the execution of a computational task in ad hoc radio networks? It has been long known that in the special case of *single-hop* radio networks (also known as a *multiple-access channel*), i.e., those with the topology of a complete graph, deterministic leader election can be done in time $\Theta(\log n)$ with collision detection [2–4]. On the other hand, our lower bound $\Omega(n \log n)$ for networks without collision detection (see Theorem 2.2) holds even for complete graphs. Hence the answer to the above question is positive for the special case of single-hop networks. Leader election is a task involving symmetry breaking: initially all nodes have the same status *non-leader* and the goal is for all nodes but one to keep this status and for the remaining single node to get the status *leader*. All nodes must learn the leader's identity.

However, the more general question whether any task in *arbitrary* radio networks can be performed faster with collision detection than without it, remained open. In other words, does there exist an important computational task such that every algorithm solving it for all networks without collision detection is slower than some algorithm solving it for all networks with collision detection? This paper gives a positive answer to this question.

Our results. We show that deterministic leader election in arbitrary ad hoc radio networks is faster with collision detection than without it. More precisely, we show a deterministic leader election algorithm working in time $O(n)$ in all n-node networks, if collision detection is available. This complexity is optimal. On the other hand, we prove formally that deterministic leader election requires time $\Omega(n \log n)$ even for complete networks, if collision detection is not available in the system. Our linear time algorithm is based on several novel techniques introduced to handle communication in radio networks with collision detection: remote token elimination, fuzzy-separating families, and distributed fuzzy-degree clustering.

Related work. Algorithmic aspects of radio communication in networks of arbitrary topology have been intensely studied in the last two decades, starting with the seminal paper [5]. A lot of attention has been devoted to efficient algorithms for such tasks as broadcasting, in particular in ad hoc radio networks, both in the deterministic [6–8] and in the randomized [9,7,10,11] setting. The above papers do not assume collision detection. The model with collision detection has been less studied in the context of radio broadcasting: cf. [12] for arbitrary networks and [13] for geometric networks.

Leader election, which is the task considered in the present paper, is a classic topic in distributed computing, and has been widely studied in the early history of this domain (cf. [14]). Most of the results on leader election in the radio model concern *single-hop* networks of known size n. Some of these results were originally obtained for other distributed problems but have corollaries for leader election. For the time of deterministic leader election without collision detection, the complexity $O(n \log n)$ follows from [6]. A constructive upper bound $O(n \operatorname{polylog}(n))$ follows from [15]. For the time of deterministic algorithms with collision detection, matching bounds are also known: $\Omega(\log n)$ follows from [16], and $O(\log n)$ follows from [2–4]. For the expected time of randomized algorithms without collision detection, the same matching bounds are known: $\Omega(\log n)$ follows from [11] and $O(\log n)$ from [17]. Finally, randomized leader election with collision detection can be done faster: matching bounds $\Omega(\log \log n)$ (for fair protocols) and $O(\log \log n)$ on the expected time were proved in [18]. For more references and a detailed study of classes of randomized protocols and energy issues for leader election in single-hop networks, see [19,20].

For leader election in *arbitrary radio networks* results are much less complete. The best upper bound on the time of deterministic algorithms without collision detection is $O(n \log^3 n)$ [21]. To the best of our knowledge, no results are published for deterministic leader election with collision detection, but the lower bound $\Omega(n)$ and the upper bound $O(n \log n)$ are folklore (see, respectively, Proposition 2.1 and the procedure `L_Elect` (X, k) taken in the case $k = n + 1$). Note that the linear lower bound also holds for the more restrictive model with collision detection.

Structure of the paper. In Section 2 we give a lower bound $\Omega(n \log n)$ on the time of leader election in the model without collision detection, and state the folklore result $\Omega(n)$ for the model with collision detection. We also introduce the combinatorial tools used in the paper, such as (strongly-)selective families and new structures called fuzzy-separating families, accompanied by known and new upper and lower bounds on the size of these combinatorial objects. In Section 3 we develop the final Algorithm `Ext_Ma_LE` in four steps: first we describe basic subroutines to be used in the algorithm, then we give an auxiliary algorithm under the additional assumption that the size of the network and the neighborhood are known to each node (Algorithm `Known_Neighb_LE`), then we remove the assumption of known neighborhood (Algorithm `Ma_LE`), and finally we remove the assumption of known size of the network (Algorithm `Ext_Ma_LE`). Each of these algorithms is followed by its correctness and complexity analysis. In Section 4 we argue why the assumption about synchrony of communication, made in this paper, cannot be removed in the study of leader election. Open problems are proposed in Section 5.