Theoretical Computer Science 608 (2015) 135–145

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

On the complexity of neighbourhood learning in radio networks ☆

Avery Miller

Department of Computer Science, University of Toronto, Toronto, Canada

ARTICLE INFO

Article history:
Received 5 March 2014
Received in revised form 18 April 2015
Accepted 23 April 2015
Available online 28 April 2015

Keywords:
Radio networks
Lower bounds
Deterministic algorithms
Neighbourhood learning task
Set families

ABSTRACT

Consider a synchronous radio network of n stationary nodes represented by an undirected graph with maximum degree Δ. Suppose that each node has a unique ID from $\{1, \ldots, U\}$, where $U \gg n$. In the neighbourhood learning task, each node must produce a list of the IDs of its neighbours in the network. We prove new lower bounds on the number of slots needed by certain classes of deterministic algorithms that solve this task. First, we show that $O(U)$-slot round-robin algorithms are optimal for the class of collision-free algorithms. Then, we consider algorithms where each node fixes its entire transmission schedule at the start. For such algorithms, we prove a $\Omega(\frac{\Delta^2}{\log \Delta} \log U)$-slot lower bound on schedule length that holds in very general models, e.g., when nodes possess collision detectors, messages can be of arbitrary size, and nodes know the schedules being followed by all other nodes. We also prove a similar result for the SINR model of radio networks. To prove these results, we consider a generalization of cover-free families of sets. We also show a separation between the class of fixed-schedule algorithms and the class of algorithms where nodes can choose to leave out some transmissions from their schedule.

1. Introduction

1.1. Motivation

Neighbourhood learning is an important step in radio network initialization and in algorithms for tasks such as routing [2,17], medium-access control [3], and gossiping [15]. Further, in the study of local computation in distributed computing [23], the standard model assumes that information about nearby nodes can be efficiently collected. Ad hoc radio networks are often temporary networks that are set up in arbitrary configurations for a relatively short amount of time, so it may be necessary to perform neighbourhood learning frequently. If it is not known how to collect neighbourhood information in an efficient way, or, if we are able to prove a strong lower bound for neighbourhood learning, then the actual running time of a solution (that assumes that this information is known) can be significantly worse than what is suggested by the analysis of its running time.

We restrict our attention to deterministic solutions, which guarantee full neighbourhood discovery within a bounded amount of time that is known in advance. Our motivation stems from the fact that neighbourhood learning is often performed as an initialization step before executing algorithms that perform more complex tasks. Randomized solutions that

☆ A preliminary version of this paper appeared in the proceedings of ALGOSENSORS 2013.
E-mail address: avery@averymiller.ca.

http://dx.doi.org/10.1016/j.tcs.2015.04.027

always terminate quickly but allow small amounts of error may cause subsequent algorithms that depend on exact neighbourhood information to fail.

1.2. Model

A static radio network consists of n nodes at arbitrary, fixed locations. Each node executes an algorithm that specifies when it will transmit and what is contained in each of its transmitted messages. When a node is not transmitting a message, we say that it is *listening*. A node can only receive a message if it is listening. We assume that each node has a unique identifier (ID) from some range $\{1, \dots, U\}$. As a consequence of assigning every radio-equipped device a unique identifier, the range of node identifiers that appear in a given network can be significantly larger than the size of the network, so we assume that $U \gg n$. We assume that each node p knows the value of U as well as its own ID, denoted by $ID(p)$.

We assume that time is partitioned into slots, where each slot is an interval of fixed length equal to the length of one transmission. We will only be considering networks where the nodes have synchronized clocks. This means that, for all $i \geq 1$, all nodes execute the ith slot of their algorithm at the same time.

In our work, we consider two popular models of radio networks: the graph model and the SINR model. Additionally, to facilitate the proof of our lower bound in the SINR model, we also introduce a new SMAX model.

In the graph model, the network is represented by a graph in which two nodes are connected by an undirected edge if and only if they can communicate with one another. A transmission during slot t by a node q is received by a listening node p if and only if p is a neighbour of q and no neighbour of p other than v transmits during slot t. We say that a *transmission collision* occurs at node p during slot t if two or more neighbours of p transmit during slot t.

In the basic graph model, a listening node p receives nothing during a slot t in which a transmission collision occurs at p, i.e., p cannot distinguish between the case where no neighbours transmit and the case where two or more neighbours transmit. In some variants of the graph model, nodes are equipped with *collision detectors*. If a node p possesses a *strong* collision detector, it can detect whether or not it is adjacent to at least one transmitting node (even if p is transmitting). If a node p possesses a *weak* collision detector, then p can detect whether or not it is adjacent to at least one transmitting node when it is listening, but cannot do so when it is transmitting.

In this paper, we say that a network consists of *weak* nodes if nodes do not possess collision detectors, each node can only send its own ID in each message, and no node has any knowledge about the transmission schedules of other nodes. A network consists of *strong* nodes if nodes possess strong collision detectors, nodes can send arbitrary messages, and each node initially knows the schedule associated with each node ID.

In the SINR model of radio networks [16], a listening node p receives a message from node q if q's signal is sufficiently stronger than the sum of all other signals received at p (plus some constant amount of background noise). At p's physical location, the strength of q's signal is calculated as $P/(d(q,p)^\alpha)$, where P is q's transmission power, $d(q,p)$ is the Euclidean distance between nodes q and p, and α is the *path-loss exponent* (usually taken to be greater than 2, so that a signal degrades at least quadratically with respect to distance). Assuming that all nodes transmit with the same power P (known as a uniform power assignment), the sum of all other signals received at p is calculated as $\sum_{q' \neq q}(P/d(q',p)^\alpha)$ for all transmitting nodes q' other than q. Formally, if S is a set of transmitting nodes, then node p receives a message from node q if $\frac{P/d(q,p)^\alpha}{\mathcal{N}+\sum_{q' \in S-\{q\}}(P/d(q',p)^\alpha)} \geq \beta$, where $\mathcal{N}$ represents the constant amount of background noise, and β is a parameter known as the *minimum signal to interference ratio*. We define q to be a *neighbour* of p if a transmission by q alone is received by p, namely, if $P/(\mathcal{N} d(q,p)^\alpha) \geq \beta$.

In the SMAX model, there is a parameter, s_{max}, that specifies the maximum number of simultaneous transmissions that can occur in a single slot such that at least one node receives a message. In particular, no nodes receive a message if greater than s_{max} nodes transmit, and, otherwise, each node receives the message contained in the strongest signal it receives. In this model, we assume that each node can distinguish between a collision (i.e., when greater than s_{max} nodes transmit) and the case where no nodes transmit.

1.3. Problem description

The goal of the neighbourhood learning task is for each node to output a list consisting of the IDs of all nodes with whom it can communicate. More formally, we say that a node v is a *neighbour* of node u if, when u is the only transmitting node, v receives u's message. In this paper, it is assumed that v is a neighbour of u if and only if u is a neighbour of v. We define the *neighbour graph* G_{nbr} on the network nodes by the edge set $\{\{u,v\}|$ if u is the only transmitting node, v receives u's message$\}$. For any network node v, we define v's *neighbour set* as $NBRS(v) = \{ID(u) \mid \{u,v\} \in E(G_{nbr})\}$. The *neighbourhood learning task* requires that each network node v determines $NBRS(v)$. We denote by Δ the maximum degree of G_{nbr}, namely, $\Delta = \max_v |NBRS(v)|$. We are interested in completing the neighbourhood learning task in as few slots as possible.

1.4. Classes of algorithms

A T-slot solution to the neighbourhood learning task is a deterministic algorithm executed by each node $p \in \mathcal{V}$ such that, in each time slot t, p either transmits a message (encoded as a finite binary string) or stays silent, and, at the end of slot