ALGORITHMS AND LP-DUALITY BASED LOWER BOUNDS IN AD-HOC RADIO NETWORKS

BY ROHAN JUDE FERNANDES

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Computer Science

Written under the direction of

Martín Farach-Colton

and approved by

New Brunswick, New Jersey

October, 2007

ABSTRACT OF THE DISSERTATION

Algorithms and LP-Duality Based Lower Bounds in Ad-hoc Radio Networks

by Rohan Jude Fernandes

Dissertation Director: Martín Farach-Colton

An Ad-hoc Radio Network consists of nodes with no knowledge of its neighbors and knowledge of its own ID and n, the size of the network. In this thesis, we present algorithms and lower bounds related to ad-hoc network initialization. Sensor networks are an important type of ad-hoc Radio Network, consisting of sensor nodes - very weak computers. Usually sensor nodes get distributed at random on a surface where they must wake up and initialize into a Radio Network. Due to the strict restrictions on sensor node capabilities, it is difficult to find efficient solutions to even basic problems.

In our first result, we present a formal WEAK SENSOR MODEL that summarizes the literature on sensor node restrictions, taking the most restrictive choices when possible. We show that sensor connectivity graphs have low-degree subgraphs with good *hop-stretch*, as required by the WEAK SENSOR MODEL. We then give a WEAK SENSOR MODEL-compatible protocol for finding such graphs that runs in $O(\log^2 n)$ time with high probability.

We then present new lower bounds for collision-free transmissions in Radio Networks. Our main result is a tight lower bound of $\Omega(\log n \log(1/\epsilon))$ on the ϵ-failure-probability time required by a fair randomized protocol to achieve a clear transmission in a one-hop network. We also prove a new lower bound for the important multi-hop

setting of nodes distributed as a connected Random Geometric Graph. In this setting, we prove a lower bound of $\Omega(\log\log n \log(1/\epsilon))$ on fair protocols for clear transmissions in the well-studied case of sensor nodes distributed uniformly at random with enough nodes to ensure connectivity, and thus for more complicated problems such as MIS.

In the Wake-Up problem, we have a multi-hop, ad-hoc radio network with locally synchronized nodes. Each node either wakes up spontaneously or is activated when it receives a signal from a neighboring node. All nodes have knowledge of n, the number of nodes in the network and the diameter D. We present a new lower bound of $\Omega(D \log n \log(1/\epsilon))$ on the ϵ-failure-probability time taken by a fair protocol to wake up the entire network. Our lower bound is tight for high-probability protocols when $D \in \mathcal{O}(n/\log^2 n)$.

Acknowledgements

I have spent nearly seven years in graduate school and in many ways this has been an exhilirating journey. I began graduate study in the Computer Science department at Stony Brook University after Prof. Steven Skiena was kind enough to respond to an email from a prospective graduate student in India. I owe my introduction and early training in the world of research to him.

I came to Rutgers in the spring semester of 2003. I was directed to Radio Network research by my advisor Martín Farach-Colton. It is impossible to summarize the role that Martín has played as an advisor – part intellectual mentor, part culture guru and part graduate-student life navigator. Our interaction has always benefited from his acerbic wit and ever-present humor. I owe the critical part of my scientific development to him, as also some of my taste for red wine, classical music and international cuisine. His astute advice on all matters, academic, professional and financial has reliably guided me.

A special mention needs to be made of my friend and colleague Miguel Mosteiro. A considerable part of this thesis (Chapters 3 and 5 and Section 2.3) is the result of our long and fruitful collaboration with our advisor and has appeared in [FCFM05] and [FCFM06]. I also benefited greatly from early discussions with Graham Cormode and S. Muthukrishnan that led to the work detailed in Chapter 3. Special thanks are due to Muthu for introducing me to RGGs and sharing a pre-publication version of [MP05]. I would also like to thank the members of my committee, Michael Saks, Michael Bender and S. Muthukrishnan for taking the time to serve on it. Prof. Saks deserves special thanks for reading the thesis carefully and making corrections to the results of Chapter 5.

During the course of graduate study, I have taken many classes and I am glad to say that I learnt something in all of them. To Steven Skiena and Michael Bender goes the credit for firing my interest in algorithm design and analysis. In their divergent styles of teaching the same topics, I experienced a deeper understanding of the subject. I owe a great deal as well to classes taught by Leo Bachmair, Ker-I Ko, Joseph Mitchell, Martín Farach-Colton, Doron Zeilberger, Michael Grigoriadis, Michael Saks, Jeffry Kahn, Paul Feehan and Richard Falk.

One often misses the subtle nuances of a subject until one has to teach it. My understanding of many subjects has improved with my experiences as a teaching assistant. I owe a debt to the kind supervision of Steven Skiena, Michael Bender, Bud Graff, Ahmed Elgammal, Endre Szemerédi, S. Muthukrishnan and Michael Fredman, all of whom I have been proud to work with.

A key part of my intellectual development has been the interaction with my fellow students and colleagues at Rutgers and Stony Brook. I would like to acknowledge my fellow students from Rutgers: Miguel Mosteiro, Sambuddha Roy, Carlos Diuk, Yang Huang, Naomi Miller, Lihong Li, Andre Madeira, Alexander Strehl, Mangesh Gupte, Nikita Lytkin and Devendra Desai, and from Stony Brook: Pavel Sumazin and Vinhthuy Phan, for useful and thought-provoking discussions.

I could not have completed graduate school without the considerable personal support I received from my friends and family. I would like to thank my parents, Gertrude Mabel and John Paul Fernandes and my brothers Errol, Kevin and Joel for their steadfast support. My parents have been a source of inspiration to me and remain so. Special thanks are also due to Vivek Sasikumar, Amit Bhandari and Neelotpal Kundu with whom I share very old friendships.

In and around graduate school, I have made some of my closest friends and the camaraderie has held me in good stead. I would like to acknowledge my friends and colleagues, Carlos Diuk, Miguel Mosteiro, Silvia Romero, Sambuddha Roy, Mangesh Gupte, Devendra Desai, Anupama Reddy, Wei Zhuang, Yihua Wu, Yang Huang, Alexander Strehl, Thomas Walsh, Nikita Lytkin, Lihong Li, Shabbir Dahodwala, Vikas

Desai and Arvind Seth. Yang Huang, Yi Jin and Amit Bhandari deserve special mention for their invaluable help with the last and most important part of graduate school – the job-search. I will always remember my landlady-for-a-year – Cathy Cummings for her friendship and the best breakfast-table conversations I have ever had.

The work in this dissertation was made possible through the generous support of DIMACS, the Computer Science department and the following grants: NSF CCR 00-87022, NSF EIA 02-05116, NSF CCF 06-21425 and The Alfred P. Sloan Foundation 99-10-8.

Last, but by no means the least, Naomi Miller deserves special mention for her support, patience and companionship during the last phase of the program.

Dedication

To my late grandmother, 'Māi' – Mary Fernandes.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The beginning is the chiefest part of any work.

Plato (c. 427-347 B.C.)
Republic

In recent years, vast advances have been made in the area of wireless technology. Radio Transmitters have increasingly become smaller, cheaper and more durable. In fact, advances in technology have made it possible to integrate sensing, processing and communication in a low-cost device, popularly known as a *sensor node*. Sensor nodes are randomly deployed over an area and must self-organize as a radio-communication network called a *Sensor Network* . Even though communication among sensor nodes is through radio broadcast, it is useful to set up explicit links between nodes in order to establish routing paths and prevent flooding. During this initialization process Sensor Networks adhere to the ad-hoc Radio Network model, i.e., each node has no knowledge of its in-neighbors and out-neighbors. Thus any protocol running on such nodes can only use for information, knowledge of a unique ID and n, the number of nodes in the entire network. This thesis deals with algorithms and lower bounds in this unknown topology setting.

A sensor network is capable of achieving large tasks through the coördinated effort of sensor nodes, but individual nodes have severe limitations on memory size, life cycle, range of communication, etc. Any Sensor Network initialization algorithm must be fast and distributed, and must resolve channel contention issues. The network constructed by such an algorithm must be connected and must have low degree and diameter. The limitations on individual sensors nodes make this problem non-trivial, and its adequate resolution is crucial for making sensors useful.

There are two main types of issues in sensor network formation: those relating to geometric properties and those relating to network protocols; and any solution achieved for either must be compatible with an accurate model of sensor nodes. On the one hand, coverage and connectivity in sensor networks are dependent on the distribution of nodes in an area and the range of transmission of each node. Additionally, the density of nodes in an area determines the minimum path length between any two nodes in the induced connectivity graph. The limited range of transmission makes these properties geometric. On the other hand, protocols for sensor network formation are limited by the fact that sensor nodes share a common channel of communication and that they do not typically have access to directional or positional information. Memory limitations in sensor nodes also impose the restriction that a node can only keep track of $O(1)$ neighbors.

Sensor Network initialization research has three parts: (*i*) to specify a comprehensive model that captures all the restrictions present in sensor nodes; (*ii*) given that under those restrictions is not possible to establish all the links of the connectivity graph, to show that there exists a subgraph of the connectivity graph that would make a connected network without asymptotically increasing the cost of delivering messages; and (*iii*) to give a fast distributed protocol that works under the constraints of the specified model.

In Chapter 3, we present a formal WEAK SENSOR MODEL that summarizes the literature on sensor node restrictions, taking the most restrictive choices when possible. Given the WEAK SENSOR MODEL, we argue that a good sensor network must have constant degree and asymptotically optimal *hop-stretch*. We show that any appropriate random geometric graph has such a subgraph. Finally, we give a WEAK SENSOR MODEL-compatible $O(\log^2 n)$ high-probability protocol for finding such a subgraph. Ours is the first network initialization algorithm that is implementable on sensor nodes.

In Chapter 5, we investigate lower bounds in the ad-hoc Radio Network setting. We show new lower bounds for collision-free transmissions in Radio Networks. Our main result is a tight lower bound of $\Omega(\log n \log(1/\epsilon))$ on the ϵ-failure-probability time of a fair randomized protocol for clear transmission in the one-hop setting. This improves on the previous best lower bound of $\Omega(\log n \log(1/\epsilon)/(\log\log n + \log\log(1/\epsilon))$ on the ϵ-failure-probability time of a fair randomized protocol for clear transmission in a one-hop, globally-synchronized Radio Network that was proved in [JS02].

We also prove a new lower bound for the important multi-hop setting of nodes distributed as a connected Random Geometric Graph. In this setting, we prove a lower bound of $\Omega(\log\log n \log(1/\epsilon))$ on fair protocols for clear transmissions in the well-studied case of sensor nodes distributed uniformly at random with enough nodes to ensure connectivity, and thus for more complicated problems such as MIS.

The Broadcast and Wake-up problems in Radio Networks with unknown topology are the subject of Chapter 4, Here, we comprehensively cover randomized algorithms and lower bounds on randomized algorithms for the two closely related problems in both one-hop and multi-hop networks. Finally in Chapter 6, we use the powerful lower-bounding technique developed in Chapter 5 to prove the first problem-specific lower bound on the wake-up problem for multi-hop Radio Networks with unknown topology. Our ϵ-probability lower bound of $\Omega(\min\{D \log n \log(1/\epsilon), D + \frac{n \log(1/\epsilon)}{\log n}\})$ is the best known lower bound for fair protocols, when $D \in \mathcal{O}(n/\log^2 n)$. It is tight for high-probability protocols under the same condition on D.

Chapter 2

Models

A good model can advance fashion by ten years.

Yves Saint Laurent (b. 1936)

In this chapter, we review models used in Radio Networks that are usually found in the literature. It is not always the case in Radio Networks that every node is connected directly to every other node. Furthermore, in many cases these connections are not symmetric. Therefore, a model for the topology of the network needs to be defined. Also, depending on the application, Radio Networks have very different node constraints, e.g. in some networks nodes have ternary feedback but in others the feedback is just binary. Therefore, a detailed model of the constraints present in the nodes forming the network is also needed. We summarize in this section models of topology and node constraints used in the Radio Networks area and in Section 2.3 we focus in Sensor Networks describing in detail our harsh Weak Sensor Model [FCFM05]. More details about Sensor Networks classification and taxonomies can be found in [TAGH02].

2.1 Topology Models

Regarding the topology of a network, a well known specification is given by a *directed graph*. A directed graph is a pair of sets $\{V, E\}$, where V is a set of *points* or *nodes* and E is a set of ordered pairs of distinct points taken from V. Any such pair is called an *arc* or an *edge*. In our context, the points model the nodes of the network and the arcs represent the ability to send messages directly (in one hop) from one node to another. If the communication in a network is achieved through wires, an edge AB in the graph represents the link that facilitates the communication from A to B. If on the other hand the communication in a network is wireless, an edge AB in the graph implies that B is in the range of transmission of A. Whenever this relation is symmetric, an *undirected graph* can be used as a model. For example, in a wireless network where all nodes have the same range of transmission, an undirected graph is a suitable model because if a node B is reachable from a node A, A is also reachable from B.

2.1.1 Topology in Radio Networks

The connectivity model widely used in Radio Networks where all nodes have the same range of transmission is the *Geometric Graph* (GG). The specification of a GG includes a pair of sets $\{V, E\}$ and a number $r \in \mathbb{R}^+$. The set of nodes are points in $\mathbb{R}^2$ and an edge $AB \in E$ if and only if A and B are separated by an Euclidean distance of at most r. As mentioned before, the graph is undirected because the range of transmission of all nodes is assumed to be the same. If this is not the case, more sophisticated models are needed.

There are also some variations of a GG in the literature. When the distance r, modeling the range of transmission is normalized to 1, the graph is called *Unit Disk Graph* (UDG). For cases in which the connectivity beyond some distance $r \in (0, 1]$ is uncertain, there is a generalization of a GG called in the literature *Quasi-Unit Disk Graph* (QUDG). The later model can be extended with a distribution on the probability of being connected when the separation distance is bigger than the uncertainty threshold. Also, the uncertainty threshold can be defined as a function of the angle with respect to some direction of reference for cases where directional antennae are used.

Of course, any of these models can be also extended with node sets in higher dimensional spaces and with threshold distances under different metrics. The particular extension depending on the setting we are modelling. A common, simple extension is to consider the points in $\mathbb{R}^3$ to model the deployment of the network in the real world. Another possible extension is to consider a distribution on the probability of two nodes being connected. Such an extension would imply a combination of the classical random graph model [ER59] with a GG. A more appropriate application of randomness to the

GG model in the specific area of Sensor Networks is explained in the next section.

2.1.2 Topology in Sensor Networks

In addition to a comprehensive model for the various constraints present in Sensor Networks, a formal model of the potential connectivity of the network needs to be defined. In the past, computer networks have been modeled by means of classical random graphs. Starting in 1959 with a paper by Erdös and Renyi [ER59], the field of random graphs has been widely explored. The classical *Bernoulli* random graph model is denoted as $\mathcal{G}_{n,p}$ where n is the number of nodes and p is the probability of existence of each edge. Random graph models have been used for instance to model the web-graph [ACGL02, KKR+99] where the structure of the random graph gives insight into the behavior of the web-graph.

However, the classical random graph model is not adequate for the Sensor Network setting because the probability of having an edge AB is either 0 or 1 depending on the Euclidean distance between A and B. For example, if the pairs AB and BC are connected in the Bernoulli random graph model, the probability of AC being connected is still p. On the other hand, having the same knowledge about AB and BC in the sensor network setting increases the likelihood that AC are connected because connectivity is a function of proximity.

Regarding the deployment of nodes in a Sensor Network, *deterministic deployment*, i.e., the placement of nodes at specific locations, is only possible for small networks in a friendly environment. However, this scenario is not realistic for most of the intended applications of Sensor Networks where a large area is expected to be covered and the environment is expected to be either hostile or remote. Two models of *random deployment* of nodes are used. In one model, n nodes are assumed to be distributed uniformly at random so that each node is equally likely to fall in any location of the area of interest, independently of the other nodes. The other model is a stationary Poisson point process with intensity n where the number of nodes in disjoint regions is Poisson distributed and mutually independent.

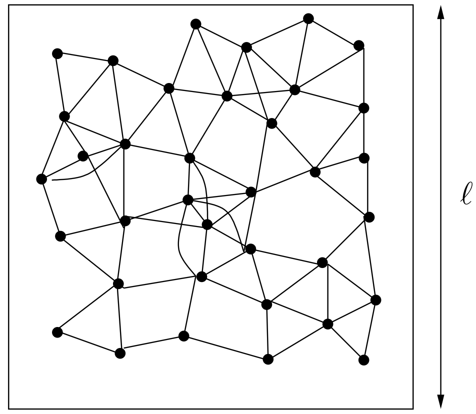


Figure 2.1: A Random Geometric Graph.

Thus, Sensor Networks are best modeled by *Random Geometric Graphs* (RGG) in $\mathbb{R}^2$ [Pen03]. In the Random Geometric Graph Model $\mathcal{G}_{n,r,\ell}$, n nodes are distributed uniformly at random in $[0,\ell]^2$, and nodes are connected by an edge if and only if they are at an Euclidean distance of at most $r \leq \ell$, the *connectivity radius* (Figure 2.1). The node density depends on the relative values of n,r and ℓ. A specific instance of $\mathcal{G}_{n,r,\ell}$ is a *Random Geometric Graph (RGG)*, also referred to as $G(n,r,\ell)$. A popular instance of this model is $\mathcal{G}_{n,r,1}$ or simply $\mathcal{G}_{n,r}$. Of course, sometimes, a two dimensional model may be inadequate when the terrain in which the sensors are positioned is uneven. In this case an extension to three dimensional random geometric graphs may be needed.

Some properties commonly studied for random geometric graphs within the context of sensor networks are

Physical Coverage For the region in question, what fraction of the region is covered by balls of radius r, centered on the points thrown randomly into the region with uniform distribution? More specifically we are interested in the number of nodes we must throw such that the fraction of the region covered is $1 - o(1)$.

Graph Connectivity What is the relation among n, r and ℓ when a graph $G(n,r,\ell)$ becomes connected? In keeping with the random nature of the model we say that $G(n,r,\ell)$ is connected when it is connected *with high probability*.

Route Stretch Given two nodes u, v in a graph $G(n,r,\ell)$, *stretch*(u,v) is defined as the ratio of the shortest distance between u and v in the graph to the normed distance between the two points in the plane. The *stretch* of $G(n, r, \ell)$ is the maximum of the *stretch* over all pairs of points (u, v) in $G(n, r, \ell)$.

The theory of random geometric graphs is a key tool to study some of the underlying properties in Sensor Networks such as connectivity or coverage. However, the results obtained in this field can not be directly applied to Sensor Networks due to the additional constraints present in them.

2.2 Node Constraints Models

Radio Networks is a vast area and there are myriad applications of such a technology, e.g., cellular phones, wireless computer networks, ad-hoc networks, etc. Depending on the specific application the nodes forming the network have very different constraints on their processing and communication capabilities, i.e., range of transmission, life cycle, storage size, etc. In addition to formal models of the topology or the potential connectivity among nodes, an appropriate model of the constraints of the nodes in the network has to be defined in order to properly design and analyze protocols. We summarize here some of the models used in Radio Networks and Sensor Networks.

2.2.1 Radio Networks

In a seminal paper [BYGI92], Bar-Yehuda, Goldreich and Itai presented a formal model of a radio network that specifies many of the important restrictions on sensor nodes, including, e.g., limits on contention resolution, but they make no mention of computational limits, such as small memory. More precisely, the model consists of an arbitrary multi-hop undirected network. The nodes are assumed to be locally synchronous, i.e., they all have the same clock frequency but perhaps different starting times. Each node either receives or transmits within each time slot, but not both. A node receives a message successfully in a time slot if exactly one of its neighboring nodes transmits in that time slot. If more than one neighboring node transmits in the same time slot, the messages are garbled and the node receives noise. It is not possible to detect collisions, hence, a node can not distinguish the case in which no neighboring node transmits from the case in which more than one transmit in the same time slot. The topology of the network is not known a priori. The main difficulty in this model, as well as in most of the models in Radio Networks, is the possibility of message collision, therefore, any protocol for this model has to include contention resolution in order to be useful.

After this model was introduced, some papers [NO00], [KMPS04] have added more restrictions, although often such restrictions are implicit in the text or algorithms rather than fully specified. In the following section we elucidate a complete and comprehensive model for Sensor Networks.

2.3 The Weak Sensor Model

As explained before, nodes in Sensor Networks are designed with the goal of obtaining a device as small as possible and at a very low cost. Therefore, sensor nodes have very harsh constraints in each of its main capabilities, processing, communication and sensing. These strong constraints are the main reason why problems in Sensor Networks are challenging, because the typical solutions utilized in computer networks are not suitable in such a harsh scenario. Therefore, in order to approach any problem in Sensor Networks, and in addition of formal models of the connectivity of the network, a formal model of the various sensor node constraints has to be defined.

Given the various limitations of sensor nodes and the absence of a reliable communication structure after deployment, any sensor network protocol must work under difficult conditions. In this section, we specify the formal Weak Sensor Model that summarizes the literature on sensor node restrictions, taking the most restrictive choices when possible. The protocol for Sensor Network Bootstrapping, described in Chapter 3 is described for the model described here. The lower bound results of Chapters 5 and 6 are proved for the more general Radio Network model.

- Memory size: Sensor nodes may have limited memory size. In fact, asymptotically speaking, if we assume that the memory size is any function in $\omega(1)$ we would be assuming that nodes can have a memory of infinite size. Therefore, in the Weak Sensor Model sensor nodes may store only a constant number of $O(\log n)$ bit words.

- Short transmission range: Due to costs and size restrictions, sensor nodes may not have a large range of transmission. Consequently, not all nodes are reachable from a given node leading to the well known hidden-terminal problem. This limitation has an impact on the density of sensor node deployment.

- Discrete Transmission Range: Some of the extant literature [SWLF04] assumes that nodes can vary their power of transmission. However, assuming that any number of levels can be reached is unrealistic–in particular to analyze the asymptotic behavior of the algorithm. In this model, sensor nodes can adjust their power of transmission to only a $O(1)$ number of levels.

- One channel of communication: although it is assumed in some papers that $\omega(1)$ channels are available in order to avoid collisions, this assumption is unrealistic–specially in order to analyze the asymptotic behavior of protocols. We constrain the number of channels of communication to exactly one.

- Locality: Sensor nodes are distributed over a large area and may not be reachable by a central controller. Hence, each sensor node must be capable of configuring itself automatically.

- Low-information channel contention:

  - Shared channel of communication: Given that this is a Radio Network and that there is only one channel available, the communication with neighboring nodes is through broadcast in a shared channel.

  - Contention-resolution mechanism: If more than one message is placed on a multiple-access communication channel at the same time, a collision occurs and no message is delivered. Hence, sensor nodes have to implement a contention-resolution mechanism to access the channel.

- No initial infrastructure: Right after deployment, the nodes of a Sensor Network have no communication infrastructure available (MAC layer). Therefore, before any information exchange can be carried out, nodes have to self-organize a medium access scheme bringing structure to the network.

- No collision detection: Although in many Radio Networks it is possible to detect a collision, it has been also argued that a collision can not be detected in the presence of noisy channels [BYGI92]. In this model, only two channel states are feasible, single transmission and silence/collision. This scenario is popularly known as *binary channel* and nodes are said to have *binary feedback*.

- Non-simultaneous reception and transmission: A sensor node may not be able to receive while transmitting because, in its vicinity, its own signal overwhelms any signal transmitted by other nodes. Therefore, transmitters also cannot detect collisions.

• Asynchronicity: No global clock or other synchronizing mechanism is assumed, but all sensor nodes have the same clock frequency. We assume that time is divided into *slots*. The use of a slotted scenario instead of a more realistic unslotted one was justified in [Rob75], where it was shown that they differ only by a factor of 2 because a packet can interfere in no more than 2 time-slots. This type of synchronicity is usually called *local synchronization*.

• Limited life cycle: Sensor nodes may be powered by sources such as solar energy. These sensors may go down from time to time to recharge. This necessitates simpler and fast computations and energy-efficient protocols.

• No position information: Due to cost and size restrictions, sensor nodes may not have position information obtained using a global or local positioning system, directional antenna or other specialized hardware.

• Adversarial node wake-up schedule: Given that the sensor nodes are deployed over large areas and given the lack of a centralized controller, we can not expect all sensor nodes to start the execution of protocols in the same time slot. Therefore, in order to analyze these protocols in a worst case scenario, we assume the existence of an adversary that determines the wake-up schedule.

- Unreliability: In addition to the lack of guarantees of a constant power supply, due to low cost, sensor nodes are unreliable. Hence, sensor network protocols have to be designed to be robust in the case of failures of one or more sensors.

In chapter 3, our node constraints model is the Weak Sensor Model and the potential connectivity of the nodes is modeled by a random geometric graph. As explained in Section 2.1.2, the deployment of nodes in a random geometric graph can also be interpreted as a Poisson process in the plane where the number of points in $[0, \ell]^2$ is given by the Poisson distribution with mean n. In our proofs, we assume the uniform deployment, i.e., each of the sensors is equally likely to fall at any location in $[0, \ell^2]$ independently of the other sensors, although the results hold for the Poisson model as well with almost no change in the proof techniques.

Chapter 3

Bootstrapping a Hop-optimal Network in the Weak Sensor Model

> The weak are more likely to make the strong weak than the strong are likely to make the weak strong.
>
> Marlene Dietrich (1904-1992)

A sensor network is capable of achieving large tasks through the coördinated effort of sensor nodes, but individual nodes have severe limitations on memory size, life cycle, range of communication, etc. Any sensor network initialization algorithm must be fast and distributed, and must resolve channel contention issues. The network constructed by such an algorithm must be connected and must have low degree and diameter. The limitations on individual sensor nodes make this problem non-trivial, and its adequate resolution is crucial for making sensors useful.

There are two main types of issues in sensor network formation: those relating to geometric properties and those relating to network protocols; and any solution achieved for either must be compatible with an accurate model of sensor nodes. On the one hand, coverage and connectivity in sensor networks are dependent on the distribution of nodes in an area and the range of transmission of each node. Additionally, the density of nodes in an area determines the minimum path length between any two nodes in the induced connectivity graph. The limited range of transmission makes these properties geometric. On the other hand, protocols for sensor network formation are limited by the fact that sensor nodes share a common channel of communication and that they do not typically have access to directional or positional information. Memory limitations in sensor nodes also impose the restriction that a node can only keep track of $O(1)$ neighbors.

Until recently, the existing literature on sensor network initialization did not sufficiently handle all aspects of the problem. All random geometric graph results related to ad-hoc wireless networks require $\omega(1)$ degree (see e.g. [MP05]). All proposed protocols for sensor network formation include some inappropriate hardware assumptions. For example, the sensor network formation protocol in [SWLF04] builds a constant-degree network, but relies on positional information hardware. The protocol proposed in [BLRS03] also builds a constant degree network, but relies on the preëxistence of a scheme for channel-contention resolution. The different models implicit in such results are inadequate and poorly reflect the various limitations under which sensor nodes operate, and indeed, there seems to be considerable confusion in the literature as to what are or are not reasonable assumptions about the capabilities of sensor nodes.

Sensor Network initialization research has three parts: (*i*) to specify a comprehensive model that captures all the restrictions present in sensor nodes; (*ii*) given that under those restrictions it is not possible to establish all the links of the connectivity graph, to show that there exists a subgraph of the connectivity graph that would connect the network without asymptotically increasing the cost of delivering messages; and (*iii*) to give a fast distributed protocol that works under the constraints of the specified model.

In Chapter 2, we presented a formal Weak Sensor Model that summarizes the literature on sensor node restrictions, taking the most restrictive choices when possible. Given the Weak Sensor Model, we argue in this chapter that a good sensor network must have constant degree and low *hop-stretch* (defined in Section 3.2). We show that any appropriate random geometric graph has such a subgraph. Finally, we give a Weak Sensor Model-compatible protocol for finding such a subgraph. Ours is the first network initialization algorithm that is implementable on sensor nodes.

3.1 Related Work

The Sensor Networks research area is very active and includes a vast body of theoretical and empirical research work that is impossible to completely cover here. We give a broad overview of some of this work in this section. A more complete description of related work in Sensor Network initialization can be found in [Mos07, Chapter 5]

3.1.1 Threshold properties in $\mathcal{G}_{n,r}$ and $\mathcal{G}_{n,r,\ell}$

Gupta and Kumar [GK98], in a seminal paper in the field of random geometric graphs computed the minimum radius needed to obtain a large connected component with high probability. This and other results [Pen03] give us a critical radius such that each node will have many neighbors. Of course, sometimes, a two dimensional model may be inadequate when the terrain in which the sensors are positioned is uneven. In this case an extension to three dimensional random geometric graphs may be needed.

In the $\mathcal{G}_{n,r,\ell}$ model, tight thresholds for connectivity, coverage and route stretch, were shown by Muthukrishnan and Pandurangan in [MP05] using an overlapping dissection technique called bin-covering. More recently, Goel, Rai and Krishnamachari showed in [GKR04] that in fact all monotone graph properties have a sharp threshold for random geometric graphs. Other properties of random geometric graphs such as vertex degree or k-connectivity were studied in [AR97a, AR97b, Pen99].

3.1.2 Bluetooth

Bluetooth [BS01, Blu, MB00], which also limits the local connectivity of nodes, is a local area wireless technology designed to enable voice and data communication between various electronic devices. In these networks the nodes have less restrictive constraints (like power supply, range of transmission, memory capacity, etc.) than in sensor networks. In Bluetooth, a group of devices sharing a common channel is called a piconet. Each piconet has a master unit that selects a frequency hopping sequence for the piconet and controls the access to the channel. Other participants of the group known as slave units are synchronized to the hopping sequence of the piconet master. The maximum number of slaves that can simultaneously be active in a piconet is seven. A slave in one piconet can be a master or slave in another piconet. Piconets can also be interconnected via bridge nodes to form a bigger ad hoc network known as a scatternet.

There has been considerable work on schemes for the formation of scatternets.

Barrière et al. [BFNO03] proposed a distributed construction technique for Bluetooth scatternets of low degree and fixed diameter. This technique is useful even in the dynamic case where nodes are assumed to come alive and drop dead from time to time. However this technique is restricted to networks where all nodes are within transmission range of each other and hence is unrealistic for the purpose of sensor network formation. Salonidis et al. [SBTL01] earlier proposed an algorithm for constructing scatternets, but this technique suffers from the same limitations as above and further is restricted to 32 nodes and static layout. Schemes proposed for scatternet formation in [LS01, SBTL01, WTH02, ZBC01, FMPC04] are designed to work in the more general case where all nodes may not be within transmission range of each other. Techniques proposed in these are strictly heuristic or do not fit in the Weak Sensor Model.

3.1.3 Cellular Systems

There are various reasons why medium access control protocols used in cellular systems can not be used in Sensor Networks. In a cellular system, mobile nodes are a single hop away from distinguished nodes called *base stations* and the base stations form a wired backbone. The primary goal of a medium access scheme in a cellular system is to guarantee quality of service and efficient bandwidth use, but power efficiency has a secondary role given that the base stations have constant power supply and the users can replenish the batteries of the mobile nodes. In Sensor Networks there is no central control such as a base station and power efficiency dominates the life cycle of the network, therefore existing solutions for cellular systems can not be applied.

3.2 Geometric Analysis of Sensor Networks

Recall that sensor nodes may only set up links with a constant number of neighbors, a consequence of the memory size limitation in the Weak Sensor Model, and since sensor nodes are distributed uniformly at random, the potential connectivity relation defines a *Random Geometric Graph* (RGG). Hence, any protocol for network formation must set up links defining a constant-degree spanning subgraph of the RGG. However, ignoring potential links may result in an increase in path lengths in the subgraph. This increase in path length can be measured in two ways: in terms of increase in the number of *hops* or increase in route stretch.

In applications where the propagation delay is significant, route stretch is an appropriate measure of optimality. However, sensor networks have small inter-node distances, and propagation delay is low. One of our primary concerns in the WEAK SENSOR MODEL is that we should minimize energy consumption at each node so as to maximize the life cycle. Thus, a Sensor Network is optimal when it minimizes the number of transmissions, which is to say, minimizes the number of hops in each path, rather than the weighted path length. Note that schemes have been proposed that attempt to minimize energy consumption [SWLF04], and these favor many short hops over a few long ones. However, any such scheme requires an $\Omega(1)$ number of transmission power levels and, furthermore, ignores the contention resolution overhead of the extra hops. A formal definition of stretch in terms of hops follows.

Let the *length* of a path connecting two nodes in a given graph be the number of edges of such a path. Let $d_{min}(u,v)$ be the shortest path between two nodes u and v in the RGG $G(n,r,\ell)$. Let $D(u,v)$ be the Euclidean distance between u and v in the plane. Note that in $G(n,r,\ell)$, $\lceil D(u,v)/r \rceil$ is a lower bound on $d_{min}(u,v)$. Call this lower bound, $d_{opt}(u,v)$. The *hop-stretch* of (u,v) is defined as the ratio $d_{min}(u,v)/d_{opt}(u,v)$. The *hop-stretch* of $G(n,r,\ell)$ is the maximum of the *hop-stretch* of (u,v) over all pairs of points (u,v) in $G(n,r,\ell)$. In the rest of this section we will outline a scheme to obtain a constant degree hop-optimal subgraph from a sufficiently dense random geometric graph.

3.2.1 Disk Covering Scheme for Sensor Network Formation

The Disk Covering Scheme presented in this section shows the existence of a bounded degree, bounded stretch subgraph of an RGG. The description and analysis of a distributed algorithm is presented in later sections 3.3.1, 3.3.2 and 3.3.3. Before describing the scheme, we introduce some necessary terminology.

Definition 1. *A* Random Geometric Graph *or $G(n, r, \ell)$ is an instance of $\mathcal{G}_{n,r,\ell}$ where r is the connectivity radius.*

Given a sufficiently dense $G(n, r, \ell) = \langle V, E \rangle$, the goal of the disk covering scheme is to produce as output a spanning subgraph $\langle V', E' \rangle$ such that $V' = V$, $E' \subset E$, the maximum degree is bounded by a constant and the path length is asymptotically optimal. The precise nature of the path length optimality is given in the proof of Theorem 11.

Definition 2. *The graph obtained as a result of the disk covering scheme is called the* Constant-degree Hop-optimal Spanning Graph *(CHSG)*

The following definitions will be relevant here and their meaning will be clear after the disk covering scheme is fully described.

Definition 3. *All nodes covered by the same disk at the end of the disk covering scheme are connected to each other in the RGG and will be referred to as a* **disk-clique**.

Definition 4. *Some (possibly all) of the nodes covered by the same disk at the end of the disk covering scheme are connected to each other by a spanner in the CHSG and will be referred to as a* **disk-spanner**.

Definition 5. *A* **bridge** *is a node, lying at the center of a disk, that is designated to communicate between two or more disk-cliques.*

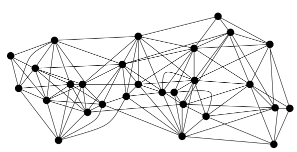
The pseudocode in Algorithm 1 summarizes the Disk Covering Scheme, where a and b are tunable parameters that affect the maximum degree and hop-stretch of the CHSG. Figure 3.1 illustrates this protocol.

3.2.2 Analysis of the Disk Covering Scheme

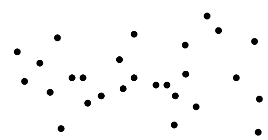
In this section the Disk Covering Scheme described in Section 3.2.1 is proved to produce a CHSG with asymptotically optimal path length. In Section 3.2.2 we establish a bound on the maximum degree of a node in the CHSG. In Section 3.2.2 two useful results for a connected $G(n, r, \ell)$ are established: A bound on hop-stretch and bounds on the node

Algorithm 1: Disc Covering Scheme

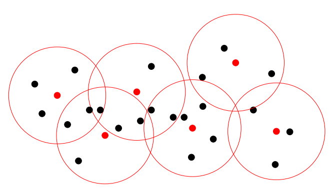
1. Add all nodes from the RGG to the CHSG.
2. Lay down *small* disks of radius $ar/2, 0 < a < 1$ centered on nodes, such that no central node is covered by more than one small disk and no node is left uncovered. We call each central node a *bridge*. Note that the bridges form a Maximal Independent Set (MIS) of the spanning subgraph $G(n, ar/2, \ell) \subseteq G(n, r, \ell)$.
3. Add to the CHSG all edges from the RGG that connect bridges.
4. Expand the small disks into *big* disks of radius $br/2, a < b \leq 1$.
5. Add to the CHSG the necessary edges to form a *disk-spanner* of constant degree among nodes covered by the same big disk.



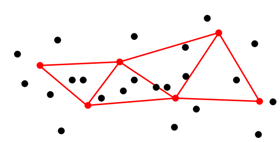
(a) The connectivity graph



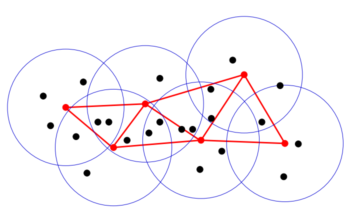
(b) Step 1



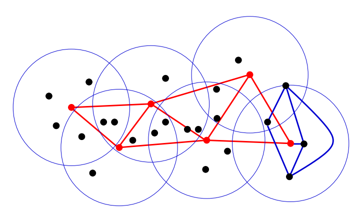
(c) Step 2



(d) Step 3



(e) Step 4



(f) Step 5 (partially)

Figure 3.1: Illustration of the Disk Covering Scheme

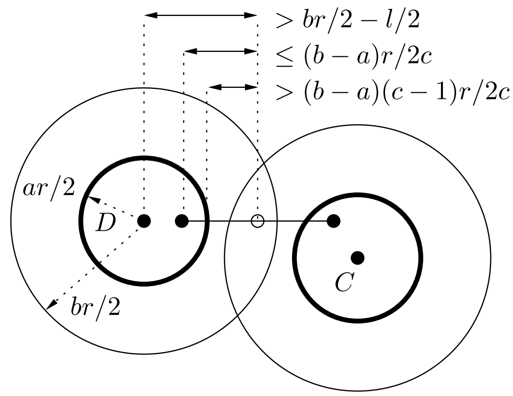


Figure 3.2: Illustration for Lemma 6

density. Finally, in Section 3.2.2 we prove a theorem on the hop-optimality of the CHSG.

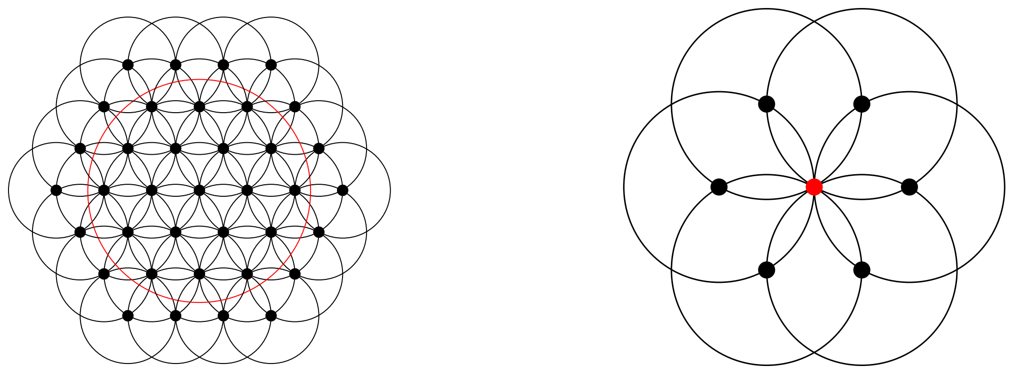
Degree Bound

Lemma 6. *At the end of the Disk Covering Scheme, run on a $G(n, r, \ell)$ that satisfies the conditions of Lemma 10, each edge of length at most $(b-a)r/c$ has both endpoints within a single big disk w.h.p, for any constant $c > 1$.*

Proof. For the sake of contradiction, assume there exists such an edge e of length $l \leq (b-a)r/c$ not covered completely by one big disk. All nodes are covered by small disks. Each endpoint of e has to be covered by a different big disk, otherwise e is already covered. Call C the center of e. Call D the center of any big disk partially covering e. Since e has at least one point outside the big disk, the distance $d(D, C) > br/2 - l/2$ as shown in figure 3.2.

Therefore, all centers of big disks that partially cover e are located outside a circle of radius $(r-l)/2$ centered on C. Then, the corresponding small disks leave an uncovered area bigger than the area of a circle of radius $r' > br/2 - l/2 - ar/2 \geq (b-a)(c-1)r/2c$. Since there is no small disk in this area, there is no node in this area, otherwise it would be a disk center. But, as proved in Lemma 9, in any circle of radius $\Theta(r)$ there are $\Theta(\log l)$ nodes w.h.p. This is a contradiction. $\square$

Lemma 7. *The degree of any node in the CHSG is in $O(1)$.*



(a) Bridge nodes

(b) Non-bridge nodes

Figure 3.3: Illustration of the upper bound on the degree.

Proof. All bridges are separated by a distance of at least $ar/2$. Connected bridges are at a distance of at most r. In figure 3.3(a) consider the smallest regular hexagon whose side is a multiple of $ar/2$ and covers completely a circle of radius r. Consider a tiling of such hexagon with equilateral triangles of side $ar/2$. As proved by Fejes-Tóth in 1940 [FT40], the hexagonal lattice is indeed the densest of all possible plane packings. Therefore, the number of vertices in such a tiling is an upper bound on the number of bridges that connect to a bridge located in the center of such a hexagon. That number is:

$$3\left\lceil \frac{4}{a\sqrt{3}} \right\rceil \left(\left\lceil \frac{4}{a\sqrt{3}} \right\rceil + 1 \right) \tag{3.1}$$

There is an extra edge that is needed to connect a bridge with its disk-spanner. Since a is any constant such that $0 < a < 1$, the degree of any bridge is in $O(1)$.

Using a simple geometric packing argument, it can be proved that a non-bridge node, is covered by at most $\pi/\arcsin(a/2b)$ big disks. By construction, a non-bridge node is connected to a constant number of neighbors within the same big disk (see figure 3.3(b)). Therefore, the degree of any node is in $O(1)$. $\square$

Hop Stretch and Density in $G(n, r, \ell)$

Theorem 8 demonstrates the existence of a path with an asymptotically optimal hop-stretch. The proof of the theorem uses an overlapping dissection technique, called bin-covering, presented by Muthukrishnan and Pandurangan [MP05]. Prior to presenting the proof of our theorem, we present a theorem from this paper. Through this we will explain some of the parameter conditions used in several theorems.

Theorem [MP05]. *Consider a $G(n, r, \ell)$ and let $r = r(\ell) = \Theta(\ell^\epsilon f(\ell))$, for some $0 \leq \epsilon < 1$, and $f(\ell)$ is a function which grows strictly slower than any function ℓ^γ where $\gamma > 0$. Let $n = \Omega(1)$. Given any two constants $c_1 > 2 - 2\epsilon$ and $c_0 < \frac{1}{2} - \frac{1}{2}\epsilon$,*

- *$G(n, r, \ell)$ is connected w.h.p. if $r^2 n \geq c_1 \ell^2 \ln \ell$, and*
- *$G(n, r, \ell)$ is disconnected w.h.p if $r^2 n \leq c_0 \ell^2 \ln \ell$.*

In the above theorem a threshold property is stated on the density of the RGG. When the number of nodes crosses a critical threshold for a particular radius, then the $G(n, r, \ell)$ is connected *w.h.p.* The above theorem is stated in a slightly different manner than stated in [MP05] in that it is stated *with high probability* rather than *asymptotically almost surely*. Under the same conditions, the proof can be modified to hold under the stronger conditions required by us. We will use a value k appropriately greater than c_1 so that we have the required connectivity *w.h.p.*

Theorem 8. *Given a $G(n, r, \ell)$ where the following conditions are satisfied: $r^2 n = k\ell^2 \ln \ell$, $r = \theta(\ell^\epsilon f(\ell))$, for some $0 \leq \epsilon < 1$, and $f(\ell)$ is a function which grows strictly slower than any function ℓ^γ where $\gamma > 0$, and $0 < \alpha \leq 1$ is a fixed constant. For any constant $k > 5\frac{4+\alpha^2}{\alpha}$, the hop-stretch is $1 + \sqrt{\alpha^2 + 4}$ w.h.p.*

Proof. It is enough to show that for any pair of nodes (u, v), there is a path P defined by a sequence of nodes $\langle u = x_0, x_1, \ldots, x_m = v \rangle$ such that the ratio between the length of P and the number of hops, m is bounded upwards by $r(1 + \sqrt{\alpha^2 + 4})$ *w.h.p.*

For a given pair of nodes (u, v), the bin covering technique is applied as follows. Let r' be the shortest horizontal projection of a segment of length r contained in the strip,

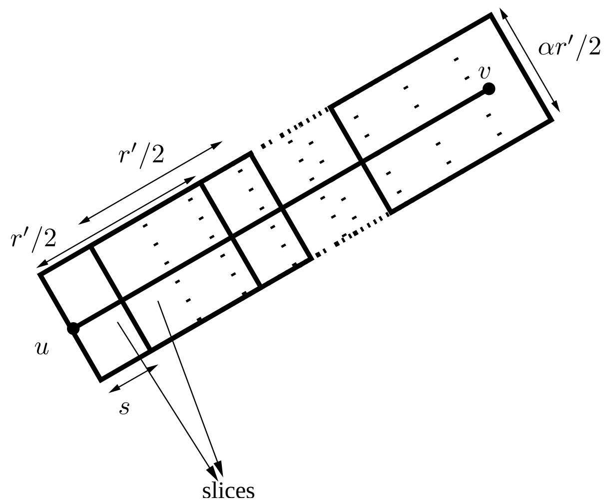


Figure 3.4: Strip between nodes u and v showing bin covering and slices.

i.e. $r' = r/\sqrt{1 + (\alpha/2)^2}$. The line connecting u and v is covered with overlapping bins of dimension $r'/2 \times \alpha r'/2$ with a spacing parameter s, as shown in Figure 3.4. This bin layout will be referred to as a *strip*.

The coordinate system is rotated such that the line segment $\overline{u,v}$ is parallel to the x axis. In what follows all distances are specified within this rotated frame of reference. Let $D_h(x, y)$ and $D_v(x, y)$ be the horizontal and vertical distances respectively between the nodes x and y.

Given a node x_j in the path P the node x_{j+1} is selected using the following criteria:

- The node x_{j+1} lies within the strip.
- $D_h(x_j, x_{j+1}) \leq r'$.
- The horizontal distance $D_h(x_{j+1}, v)$ is minimized.

A *hole* is a rectangle of dimension $r'/2 \times \alpha r'/2$, within a strip, that is devoid of nodes and adjoins a node on the side closest to u.

Consider any 3 consecutive nodes along the path x_{i-1}, x_i, x_{i+1} where $0 < i < m$, and assume that along any strip there is no hole, then $D_h(x_{i-1}, x_i) \geq r'/2$. To see that this claim is true, assume for the sake of contradiction that $D_h(x_{i-1}, x_i) < r'/2$. The distance $D_h(x_{i-1}, x_{i+1}) > r'$, otherwise x_{i+1} would have been selected as the successor of x_{i-1}. Thus, the distance $D_h(x_i, x_{i+1}) > r'/2$. Since there cannot be any hole in the strip, there exists a node y such that $D_h(x_i, y) < r'/2$. This implies that $D_h(x_{i-1}, y) < r'$. Note that $D_h(y, v) < D_h(x_i, v)$, therefore y should have been chosen as the successor of x_{i-1} by the construction criteria, which is a contradiction. The initial assumption of $D_h(x_{i-1}, x_i) < r'/2$ is thus proven false which proves the truth of the claim.

Since $D_h(x_{i-1}, x_i) \geq r'/2$ for $0 < i < m-1$, the number of hops in the path P is

$$m \leq \left\lceil \frac{D(u,v)}{r'/2} \right\rceil = \left\lceil \sqrt{\alpha^2+4}\frac{D(u,v)}{r} \right\rceil .$$

If $D(u,v) \leq r$ the path is simply the edge connecting u and v and the hop-stretch is trivially 1. Otherwise, $D(u,v) > r$ and so, the hop-stretch is $1 + \sqrt{\alpha^2+4}$.

It remains to show that there is no hole *w.h.p.*

To bound the probability that there is a hole in any strip, consider the sequence of small rectangles (call them *slices*) defined by the spacing parameter, of size $s \times \alpha r'/2$. The slices are numbered in ascending order from u to v.

For any node x_i that is contained in some slice j, let E_i be the event that the node x_{i+1} is contained in the slice $j - 1 + \lceil r'/2s \rceil$ at a horizontal distance greater than r' from x_i. Then,

$$Pr[E_i] \leq \binom{n-1}{1} \frac{\alpha r' s}{2\ell^2} \left(1 - \frac{\alpha r'^2}{4\ell^2}\right)^{n-2} .$$

If x_{i+1} is contained in a slice closer to x_i then there is no hole. If x_{i+1} is contained in a slice farther than $j - 1 + \lceil r'/2s \rceil$ then there is at least one empty bin in the strip. The probability that some bin is empty is bounded by

$$Pr[\text{EmptyBin}] \leq \frac{\max_{(u,v)} D(u,v)}{s} \left(1 - \frac{\alpha r'^2}{4\ell^2}\right)^{n} .$$

Therefore, the probability that there is a hole within any strip is

$$
\begin{aligned}
Pr[\text{Hole}] &\leq \binom{n}{2}\left(n(n-1)\frac{\alpha r' s}{2\ell^2}\left(1-\frac{\alpha r'^2}{4\ell^2}\right)^{n-2} + \frac{\max_{(u,v)} D(u,v)}{s}\left(1-\frac{\alpha r'^2}{4\ell^2}\right)^{n}\right) \\
&\leq n^2 \frac{1}{e^{n\alpha r'^2/4\ell^2}}\left(\frac{n^2\alpha r' s}{2\ell^2} e^{\alpha r'^2/2\ell^2} + \frac{\sqrt{2}\ell}{s}\right) .
\end{aligned}
$$

This expression is minimized when

$$
s = \left(\frac{2\sqrt{2}\ell^3}{n^2\alpha r' e^{\alpha r'^2/2\ell^2}}\right)^{1/2} .
$$

Then,

$$
\begin{aligned}
Pr[\text{Hole}] &\leq \frac{2k^3\ell^6 \ln^3 \ell}{r^6 \ell^{1+(k\alpha/(4+\alpha^2))}}\left(\frac{\alpha r' e^{\alpha r'^2/2\ell^2}}{2\sqrt{2}\ell}\right)^{1/2} \\
&\in O(\ell^{-\gamma_1}) \text{ for } k > 5\frac{4+\alpha^2}{\alpha} .
\end{aligned}
$$

□

A simpler proof of theorem 8 is also possible and follows, though the constant obtained is worse.

Proof. Consider a strip S_j, the probability that a node x_i is contained in S_j is at most $\alpha r'/\sqrt{2}\ell$. The probability that there is a hole within S_j adjoining x_i is at most $(1-\alpha r'^2/4\ell^2)^{n-1}$. Then, the probability that there is a hole in any strip is

$$
\begin{aligned}
Pr[\text{Hole}] &\leq \binom{n}{2} n \frac{\alpha r'}{\sqrt{2}\ell}\left(1-\frac{\alpha r'^2}{4\ell^2}\right)^{n-1} \\
&\in O(\ell^{-\gamma_1}) \text{ for } k > 6\frac{(4+\alpha^2)}{\alpha}
\end{aligned}
$$

□

Lemma 9. *In a $G(n, r, \ell)$ satisfying the parameter conditions of Lemma 10, for every circle of radius $\Theta(r)$ centered on a node, the number of nodes contained in that circle is $\Theta(\log \ell)$ w.h.p.*

Proof. To prove this lemma it is enough to show that the probability that the number of nodes, within any circle of radius βr for some constant β, deviates from $\log \ell$ by more than a constant factor, is polynomially small. Consider the random process of dropping nodes in a square of side length ℓ. Define the random variable X as the number of nodes contained in that circle. For a given node, the probability of falling in the circle is $\pi\beta^2 r^2/\ell^2$. Using Chernoff bounds

$$Pr(X \geq (1+\epsilon)\frac{\pi\beta^2 r^2}{\ell^2}n) \leq e^{-\frac{\epsilon^2}{3}n\frac{\pi\beta^2 r^2}{\ell^2}}$$
$$Pr(X \leq (1-\epsilon)\frac{\pi\beta^2 r^2}{\ell^2}n) \leq e^{-\frac{\epsilon^2}{2}n\frac{\pi\beta^2 r^2}{\ell^2}}$$

Using the parameter conditions

$$Pr(X \geq (1+\epsilon)\pi\beta^2 k \ln \ell) \leq \ell^{-\frac{\epsilon^2 \pi\beta^2 k}{3}}$$
$$Pr(X \leq (1-\epsilon)\pi\beta^2 k \ln \ell) \leq \ell^{-\frac{\epsilon^2 \pi\beta^2 k}{2}}$$

We can use appropriately high values of k and use the union bound so that the required bound holds for all circles *w.h.p.* □

Hop Optimality of the CHSG

Lemma 10. *Consider the RGG $G(n, r, l)$, where n satisfies the parameter conditions of Theorem 8 for a reduced connectivity radius of $r' = (b-a)r/c$. For any pair of nodes (u, v) in the RGG at Euclidean distance $D(u, v)$, there exists a path between them in the CHSG of at most $\lceil c\sqrt{\alpha^2+4}D(u, v)/(b-a)r\rceil - 1 + O(\log \ell)$ edges w.h.p.*

Proof. Theorem 8 states that: In the RGG that satisfies the parameter conditions of Theorem 8, there exists a path of $\lceil \sqrt{\alpha^2+4}D(u, v)/r\rceil$ edges w.h.p. We can thus infer that: If the RGG satisfies the same parameter conditions for a reduced connectivity radius of $r' = (b-a)r/c$, there exists a path between u and v using $\lceil c\sqrt{\alpha^2+4}D(u, v)/(b-a)r\rceil$ edges of length at most $(b-a)r/c$. Let p be such a path and $e_1, e_2, \ldots, e_m$ be its sequence of edges.

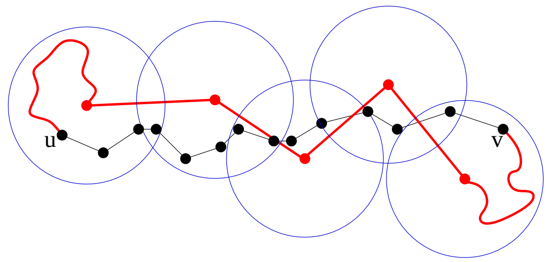


Figure 3.5: Illustration for Lemma 10

In the description of the Disk Covering Scheme, two kinds of disks were defined for clarity: big disks and small disks. In order to prove hop-optimality of the CHSG, we only refer to big disks and simply call them disks. The rest of the proof is illustrated in figure 3.5.

Lemma 6 states that every edge in the path p is completely covered by one disk. Therefore, there exists a sequence $d_1, d_2, \ldots, d_{m'}$ of overlapping disks, where any edge e_i in p is covered by some disk d_j in this sequence. A disk may completely cover more than one edge, hence $m' \leq m$. Let D_i be the bridge (center) of disk d_i.

Define a path p' using only edges of the CHSG as follows. Connect u and the bridge D_1 with a path p_1 of disk-spanner edges defined by the disk d_1. For each edge i, $1 \leq i \leq m$, replace the edge e_i in p with the node D_i. Connect all consecutive bridges D_i and D_{i+1} within the path of overlapping disks with edge $\overline{D_i D_{i+1}}$. Consecutive bridges are adjacent to each other in the RGG, because their disks overlap and the radius of each disk is $br/2$ with $b \leq 1$. Finally, connect the bridge D_m and v with a path p_m of disk-spanner edges defined by the disk d_m. The length of p' is given by: $length(p') \leq length(p_1) + (m-1) + length(p_m)$. Using the stretch bound, $length(p') \leq \lceil c\sqrt{\alpha^2+4}D(u,v)/(b-a)r \rceil - 1 + length(p_1) + length(p_m)$ w.h.p. Only disk-spanner edges are used in p_1 and p_m. It is shown in Lemma 9 that the number of nodes within a disk is $O(\log \ell)$ w.h.p. Therefore, $length(p_1) + length(p_m) = O(\log \ell)$ w.h.p. completing the proof. $\square$

The following theorem shows the main result.

Theorem 11. *For every pair of nodes in an RGG, satisfying the conditions of*

Lemma 10, there is a path in the CHSG, whose length is asymptotically optimal w.h.p.

Proof. The optimal path between any pair of nodes (u, v) separated by a distance $D(u, v)$ has at least $\lceil D(u, v)/r \rceil$ edges. If $\log \ell$ is also an asymptotic lower bound on the length of such a path w.h.p., then $(D(u, v)/r + \log \ell)/2$ is also an asymptotic lower bound, and the result proved in Lemma 10 is a constant factor approximation. It remains to show that $\log \ell$ is an asymptotic lower bound on the length of an optimal path in a constant-degree random geometric graph w.h.p.

In a δ-regular graph, the expected distance between any pair of nodes randomly chosen is at least $\log_{\delta-1} n$. A $\Theta(1)$ degree random geometric graph is a subgraph of some regular graph. Hence, in a $\Theta(1)$ degree random geometric graph, the expected distance between any pair of nodes randomly chosen is in $\Omega(\log n)$. The previous result is true w.h.p. because for some constant β

$$
\begin{aligned}
Pr(D(u, v) < \beta \log n) &\leq \frac{1}{n-1} \sum_{i=0}^{\beta \log n - 2} \delta(\delta - 1)^i \\
&\in O(n^{-\gamma_2}) \text{ for some } \gamma_2 > 2.
\end{aligned}
$$

Using the union bound, under the parameter conditions of Lemma 10, $D(u, v) \in \Omega(\log \ell)$ for all pairs of nodes (u, v) w.h.p. □

3.3 Distributed Algorithm

In this section we describe how to distributedly implement the steps of the Disk Covering Scheme for network formation. Step 2 of the Disk Covering Scheme can be achieved distributedly by means of a Maximal Independent Set (MIS) computation with nodes transmitting in a range of $ar/2$. An algorithm to compute an MIS in a weak model is presented in [MW05]. This algorithm can be tailored to our setting and can be shown to have a running time of $O(\log^2 \ell)$. The details are presented in Section 3.3.1

Steps 3 and 4 of the Disk Covering Scheme require uncolliding transmissions of each bridge in a radius of r and $br/2$ respectively. All nodes assigned to the same bridge will participate in a common spanner construction. Additionally bridge nodes must set up links with all bridge nodes at a distance of at most r. The details are presented in Sections 3.3.2 and 3.3.3. Finally, the constant-degree spanner construction is described in Section 3.3.3.

3.3.1 MIS Computation (Step 2)

Algorithm

Step 2 of the Disk Covering Scheme can be achieved distributedly by means of an MIS computation with nodes transmitting in a range of $ar/2$. An algorithm to compute an MIS in a weak model for arbitrary graphs was presented in [MW05]. This algorithm can be tailored to our setting and can be shown to have a running time of $O(\log^2 \ell)$. Algorithm 2 gives the details of such MIS computation and we give our analysis in the following section.

Algorithm 2: MIS Computation, δ_1, δ_2, δ_3 and δ_4 are constants.

Transmit the local counter with probability $1/\delta_1 \log \ell$.
if *not transmitting in the current time slot* **then**
    if *a neighbor's counter is received and the difference between the local and neighbor's counter is* $\leq \lfloor \delta_2 \log \ell \rfloor$ **then**
        Set local counter to $-\lfloor \delta_2 \log \ell \rfloor$.
    end
    else if *a neighbor's ID is received* **then**
        Set the local state to *covered* and stop.
    end
end
Increase counter if transmitted at least once.
if *the counter is* $\lceil \delta_3 \log^2 \ell \rceil$ **then**
    Set the local state to *MIS member*.
    Transmit ID forever with probability $q = 1/\delta_4$.
end
Goto step 1 at end of time slot.

Analysis

The analysis of the MIS algorithm turns out to be difficult because nodes running different phases interfere with each other. Hence, necessary assumptions regarding bounds on the total probability of transmission of nodes in other phases cannot be made leading to a circular argument. In order to break the circularity we prove the following lemmas by induction on the time slots in which a given node joins the MIS.

Before the analysis, we recall the following basic fact [MR95]:

Fact 1. *For all $n \geq 1$ and $|x| \leq n$*

$$e^x \left(1 - \frac{x^2}{n}\right) \leq \left(1 + \frac{x}{n}\right)^n \leq e^x.$$

Lemma 12. *Given any node that joins the MIS in a given time slot, the counter of all neighboring nodes is at most $\lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor$ in the same time slot w.h.p.*

Lemma 13. *Every MIS node transmits its MIS status message successfully in the $\lfloor \delta_2 \log \ell \rfloor$ time slots after it joins the MIS w.h.p.*

Proof. We prove both preceding lemmas simultaneously by employing induction on the order in which the nodes join the MIS, with ties broken arbitrarily.

Base case: Consider the first node within the whole network, call it μ_1, that joins the MIS at time t_1.

For the sake of contradiction, assume that there is a node x contained in μ_1's neighborhood whose counter is greater than $L = \lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor$ at t_1. By the definition of the algorithm, μ_1 has first transmitted at time $t_1 - \lceil \delta_3 \log^2 \ell \rceil$ and x has first transmitted within the next $\lfloor \delta_2 \log \ell \rfloor$ time slots. Afterwards, neither μ_1 nor x have sent without collision otherwise one of their counters would have been reset. Let $E(k)$ denote the event that neither μ_1 nor x have sent without collision within k time slots. Using the fact that there are at most $\delta_6 \log \ell$ nodes within the 2-hop neighborhood of μ_1 w.h.p., for some constant $\delta_6 > 0$, as shown in Lemma 9:

$$\begin{aligned} Pr[E(L)] &\leq \left[1 - 2\frac{1}{\delta_1 \log \ell}\left(1 - \frac{1}{\delta_1 \log \ell}\right)^{\delta_6 \log \ell}\right]^{\lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor} \\ &\in O(\ell^{-\gamma_3}) \text{ (Using fact 1, for some } \delta_3, \delta_1 > \sqrt{\delta_6 / \log \ell}). \end{aligned}$$

Now we must additionally prove that within $\lfloor \delta_2 \log \ell \rfloor$ time slots of $\mu_1$joining the MIS, all nodes within range of it receive a message declaring its MIS status. For at least $\lfloor \delta_2 \log \ell \rfloor$ time slots after the node μ_1 joins the MIS, no other nodes in its neighborhood join the MIS w.h.p. as shown. If in this time its MIS status message is received by all its neighbors, then they will all stop counting and transition into the *covered* state. We will now show that this message is received by all its neighbors w.h.p. Let $E(k)$ denote the event that μ_1 does not transmit without collision in k consecutive time slots. The probability of failure in $\lfloor \delta_2 \log \ell \rfloor$ consecutive time slots is:

$$Pr[E(\lfloor \delta_2 \log \ell \rfloor)] = \left[1 - \frac{1}{\delta_4}\left(1 - \frac{1}{\delta_1 \log \ell}\right)^{\delta_6 \log \ell}\right]^{\lfloor \delta_2 \log \ell \rfloor}$$
$$\in O(\ell^{-\gamma_4}) \text{ (Using fact 1 and for some } \delta_2).$$

This shows that μ_1 sends its MIS status message without collision successfully in $\lfloor \delta_2 \log \ell \rfloor$ time slots w.h.p.

*Inductive Step:*Consider the ith node μ_i, $i > 1$, that joins the MIS at time t_i.

Inductive hypothesis: For all nodes μ_j such that $j < i$, joining the MIS at time t_j, the counters of all nodes in the neighborhood of μ_j are at most $\lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor$ at time t_j w.h.p. Additionally all nodes μ_j transmit their MIS status message successfully within the interval $t_j \dots t_j + \lfloor \delta_2 \log \ell \rfloor$ w.h.p.

Therefore by time $t_j + \lfloor \delta_2 \log \ell \rfloor$ all nodes in the range of all MIS nodes $\mu_1 \dots \mu_{i-1}$ will be in the *covered* state. From the previous statements of the inductive hypothesis we can conclude that none of the MIS nodes μ_j (where $j < i$) are neighbors of each other w.h.p.

We want to show that the counters of all nodes in the neighborhood of μ_i are at most $\lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor$ at time t_i w.h.p. and that all neighbors of μ_i are in the *covered* state by time $t_i + \lfloor \delta_2 \log \ell \rfloor$ w.h.p.

If μ_i is out of the two-hop neighborhood of all the previous MIS members, the claim can be easily proved using the same argument as in the base case. Otherwise, μ_i is within a two-hop neighborhood of some MIS members. Since all nodes that previously joined the MIS are not in range of each other, μ_i is within the two-hop neighborhood of at most 12 other MIS members. This is true because a regular polygon with side of length at least r and distance from the center to the vertices at most $2r$ has at most 12 sides.

For the sake of contradiction, assume that there is a node y contained in μ_i's neighborhood whose counter is greater than $L = \lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor$ at t_i. By the definition of the algorithm, μ_i has first transmitted at time $t_i - \lceil \delta_3 \log^2 \ell \rceil$ and y has first transmitted within the next $\lfloor \delta_2 \log \ell \rfloor$ time slots. Afterwards, neither μ_i nor y have sent without collision otherwise one of their counters would have been reset. Let $E(k)$ be the event that neither μ_i or y send without collision for k consecutive time slots.

$$Pr[E(L)] \leq \left[1 - 2\frac{1}{\delta_1 \log \ell}\left(1 - \frac{1}{\delta_1 \log \ell}\right)^{\delta_6 \log \ell}\left(1 - \frac{1}{\delta_4}\right)^{12}\right]^{\lceil \delta_3 \log^2 \ell \rceil - \lfloor \delta_2 \log \ell \rfloor}$$
$$\in O(\ell^{-\gamma_5}) \quad \text{(Using fact 1,for some } \delta_3,\delta_1 > \sqrt{\delta_6/\log \ell}\text{)}.$$

Now we will show that all neighbors of MIS node μ_i will be in the covered state by time slot $t_i + \lfloor \delta_2 \log \ell \rfloor$. Any neighbor of an MIS node has a counter that lags the MIS node's counter by at least $\lfloor \delta_2 \log \ell \rfloor$. Additionally no MIS node can be within range of any other. Hence every MIS node can be subjected to interference by at most 18 other MIS nodes (by a simple geometric packing argument). Let $E(k)$ denote the event that a neighbor of an MIS node does not receive its MIS status message for k consecutive time slots. Thus the probability that a MIS node does not transmit its MIS status message without collision is given by:

$$Pr[E(\lfloor \delta_2 \log \ell \rfloor)] \leq \left[1 - \left(\frac{1}{\delta_4}\right)\left(1 - \frac{1}{\delta_4}\right)^{18}\left(1 - \frac{1}{\delta_1 \log \ell}\right)^{\delta_6 \log \ell}\right]^{\lfloor \delta_2 \log \ell \rfloor}$$
$$\in O(\ell^{-\gamma_6}) \quad \text{(Using fact 1 and for some } \delta_2\text{)}.$$

$\square$

Lemma 14. *No two nodes belonging to the MIS are within transmission range of each other w.h.p.*

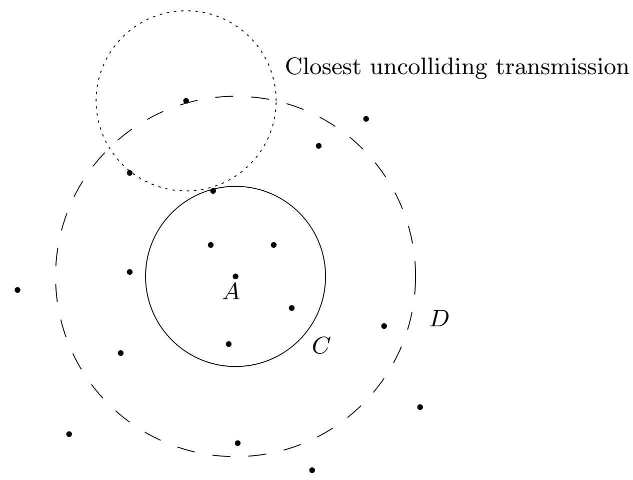


Figure 3.6: Illustration for Lemma 15

Proof. This is a direct conclusion of Lemmas 12 and 13. □

Lemma 15. *For any node running the MIS algorithm with radius r, there is at least one node, in its immediate $r/2$ neighborhood, that transmits without collision within $\lceil \delta_5 \log^2 \ell \rceil$ steps w.h.p., for some constant $\delta_5 > 0$.*

Proof. Consider a node A running the MIS algorithm (refer to figure 3.6). Since A is awake, there is at least one node awake in C at time t. From Lemma 14 it can be seen that no MIS nodes can be within range of each other, therefore there can be at most 9 MIS nodes within D (If there were more then one of them would be in range of A). Let $E(k)$ denote the event that no node in A's $r/2$ neighborhood (including A) transmits without collision in k consecutive time slots. Lemma 9 shows that there are at most $\delta_6 \log \ell$ nodes in D w.h.p., for some constant $\delta_6 > 0$.

$$Pr[E(\lceil \delta_5 \log^2 \ell \rceil)] \leq \left[1 - \left(\frac{1}{\delta_1 \log \ell}\right)\left(1 - \frac{1}{\delta_1 \log \ell}\right)^{\delta_6 \log \ell} \left(1 - \frac{1}{\delta_4}\right)^9\right]^{\lceil \delta_5 \log^2 \ell \rceil}$$

$$\in O(\ell^{-\gamma_7}) \quad \text{(Using fact 1, } \delta_1 > \sqrt{\delta_6/\log \ell}\text{, for some } \delta_5\text{).}$$

□

Theorem 16. *For a given node running the MIS algorithm, at least one node within*

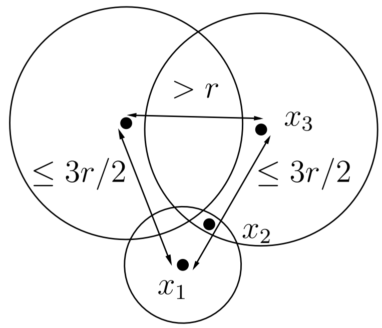


Figure 3.7: Illustration for Theorem 16

its transmission range joins the MIS in $O(\log^2 \ell)$ *time slots and no two MIS nodes are within range of each other w.h.p.*

Proof. The proof is illustrated in figure 3.7. In Lemma 15, it was shown that within a circle of radius $r/2$ centered on any node x_1, there will be a node x_2, transmitting without collision, in less than $\lceil \delta_5 \log^2 \ell \rceil$ steps w.h.p. After this single transmission, there is at least one node, namely x_2, within the neigborhood of x_1 increasing its counter. If x_2 joins the MIS after its counter reaches the value $\lceil \delta_3 \log^2 \ell \rceil$, then the statement of the theorem is proved. Otherwise, some other node, call it x_3, within range of x_2, reaches this value and joins the MIS before. If x_3 is within range of x_1, then the statement of the theorem is proved. Otherwise, x_3 covers at least one node within the $r/2$ neighborhood of x_1, namely x_2, within the next $\lceil \delta_2 \log \ell \rceil$ time slots w.h.p. (as shown in Lemma 13).

Note that the distance between x_1 and x_3 satisfies the following relation :

$$r < D(x_1, x_3) \leq 3r/2 \ . \tag{3.2}$$

All uncovered active nodes within the $r/2$ neighborhood of x_1 are still counting. Hence, the same argument can be repeatedly applied with the restriction that the next MIS node is at least at a distance of r from x_3 (by Lemma 14). There can be at most 9 MIS nodes around x_1 before x_1 or one of its neighbors joins the MIS, as explained in Lemma 15. Thus, this process terminates in at most $10(\lceil \delta_3 \log^2 \ell \rceil + \lceil \delta_5 \log^2 \ell \rceil +$

$\lfloor \delta_2 \log \ell \rfloor$) time slots. □

3.3.2 Broadcast (Steps 3 and 4)

After a node is covered by some neighboring MIS node, it needs to be assigned to that MIS node. All nodes assigned to the same MIS node will participate in a common spanner construction. Additionally MIS nodes must set up links with all MIS nodes at a distance of at most r. Any of these steps only require each MIS node to achieve an uncolliding transmission. In this section an algorithm for achieving this is detailed and a time bound is proved.

Algorithm

The algorithm is simple to describe:

> With probability $1/\beta_1$, each MIS node transmits its ID, within range $\beta_2 r$.

Where β_1 and β_2 are constants whose values depend on which of the aforementioned steps is implemented. For informing the non-MIS nodes about assignment, the transmission is made with $\beta_2 = b/2$. For setting up connections with neighboring MIS nodes, the transmission is made with $\beta_2 = 1$.

Analysis

Lemma 17. *Any MIS node running the broadcast algorithm achieves a transmission without collision within $O(\log \ell)$ steps w.h.p.*

Proof. Let Δ denote the maximum number of interfering MIS neighbors (which depends on β_2). Let $Pr[\text{fail}]$ denote the probability that any node fails to transmit without collision after $\beta_3 \log \ell$ steps for some constant β_3. For appropriate values of β_2 and β_3, using the parameter conditions of theorem 8 and the union bound,

$$Pr[\text{fail}] = n \left(1 - \frac{1}{\beta_1}\left(1 - \frac{1}{\beta_1}\right)^{\Delta}\right)^{\beta_3 \log \ell}$$
$$\in O(\ell^{-\gamma_8}) \text{ for some } \gamma_8 > 0$$

□

3.3.3 Spanner Construction (Step 5)

After nodes are covered by one or more bridges (MIS members), they have to connect locally to neighboring nodes covered by the same bridge, i.e. within the same disk. Nodes can be covered by more than one bridge. Hence, interference of transmissions not only from the local disk but also from neighboring disks must be taken into account to analyze the performance of any spanner construction algorithm. However, any node is covered by at most a constant number of disks as explained in Lemma 7, then the number of interfering transmissions with respect to the local disk is increased only by a constant factor that we fold into the constants involved in this analysis.

Algorithm

Our goal here is to construct a constant-degree spanner graph on the set of nodes assigned to a given bridge node. Since the diameter is not constrained, we adopt the simplest topology, i.e., a linked list. In order to minimize the running time, we avoid handshaking among nodes and all the construction is done by broadcasting. We start with every node choosing an integer index uniformly at random from the interval $[1, \ell]$. Since there are $O(\log \ell)$ nodes within the same range w.h.p. as shown before, no two nodes choose the same index w.h.p.

Analysis

Lemma 18. *Any node running the spanner algorithm joins the spanner within* $O(\log^2 \ell)$ *steps w.h.p.*

Algorithm 3: Spanner construction. β_4 is a constant.

```
for each non-bridge node in parallel do
    predecessor.ID ← bridge.ID;
    successor.ID ← bridge.ID;
    choose an integer index uniformly at random from the interval [1, ℓ];
    while true do
        transmit ← { true   with probability p = 1/β4 log ℓ
                   { false  with probability 1 − p
        if transmit then broadcast ⟨index,ID⟩;
        else if an index is received then
            update predecessor.ID or successor.ID accordingly;
        end
    end
end
```

Proof. In order to prove this lemma it is enough to show that every node covered by the same bridge that is running the spanner algorithm achieves at least one single (i.e. uncolliding) transmission within $O(\log^2 \ell)$ steps w.h.p. It was shown in lemma 9 that there are $\Theta(\log \ell)$ nodes within any disk of radius $O(r)$. Hence, it is enough to show that within any disk with at most $\beta_4 \log \ell$ nodes there are $\beta_4 \log \ell$ *different* single transmissions within $\beta_5 \log^2 \ell$ steps w.h.p., where β_4 and β_5 are constants.

To show that, we use the following balls and bins analysis. Let each node be represented by a bin and each transmission step be represented by a ball. A node achieving a single transmission at a given step is modeled with the ball representing that step falling in the bin representing that node. If at a given transmission step there is no single transmission, we say that the ball falls outside the bins. Now, to prove this lemma it is enough to show that after dropping $\beta_5 \log^2 \ell$ balls in $\beta_4 \log \ell$ bins, no bin is empty w.h.p.

For a given ball, the probability of falling in a given bin is the probability of achieving a single transmission, i.e.

$$Pr = \frac{1}{\beta_4 \log \ell}\left(1 - \frac{1}{\beta_4 \log \ell}\right)^{\beta_4 \log \ell - 1}$$

Hence, the probability of some empty bin is

$$Pr(fail) \leq \sum_{i=1}^{\beta_4 \log \ell} \binom{\beta_4 \log \ell}{i} \left(1 - i \frac{1}{\beta_4 \log \ell} \left(1 - \frac{1}{\beta_4 \log \ell}\right)^{\beta_4 \log \ell - 1}\right)^{\beta_5 \log^2 \ell}$$
$$\leq \left(1 - \frac{1}{\beta_4 \log \ell} \left(1 - \frac{1}{\beta_4 \log \ell}\right)^{\beta_4 \log \ell - 1}\right)^{\beta_5 \log^2 \ell} \sum_{i=1}^{\beta_4 \log \ell} \binom{\beta_4 \log \ell}{i}.$$

Using the binomial theorem,

$$Pr(fail) \leq \left(1 - \frac{1}{\beta_4 \log \ell} \left(1 - \frac{1}{\beta_4 \log \ell}\right)^{\beta_4 \log \ell - 1}\right)^{\beta_5 \log^2 \ell} 2^{\beta_4 \log \ell}$$
$$\in O(\ell^{-\gamma_9}), \gamma_9 > 0 \text{ (using fact 1, for a large enough } \beta_5 > e\beta_4).$$

□

A small-diameter spanner

In the previous construction, the distance between any two nodes is at most the number of nodes within the disk, i.e. $O(\log \ell)$. Although a diameter of $\Theta(\log \ell)$ for the disk spanner is optimal (theorem 11) for a constant-degree random geometric graph, a constant-degree spanner with diameter $o(\log \log \ell)$ is also possible as shown in this section.

The structure we utilize, is popularly known as a *butterfly network*. Butterfly networks are used in many parallel computers to provide paths of length $\log m$ connecting m inputs to m outputs. A labeled instance of a butterfly network with $m = 8$ is shown in figure 3.8. The inputs of the network are on the left and the outputs are on the right. In our case, all nodes have the same role and a message between any pair of nodes can be sent in $O(\log m)$ hops. Then, given that there are $\Theta(\log \ell)$ nodes in any disk, the diameter obtained is $o(\log \log \ell)$. Notice that, once unique consecutive labels are assigned to all nodes, each node can easily compute to which nodes is connected. Then, our goal is to assign unique consecutive indexes to all nodes within the disk.

The distributed algorithm for non-bridge nodes to construct such a network within

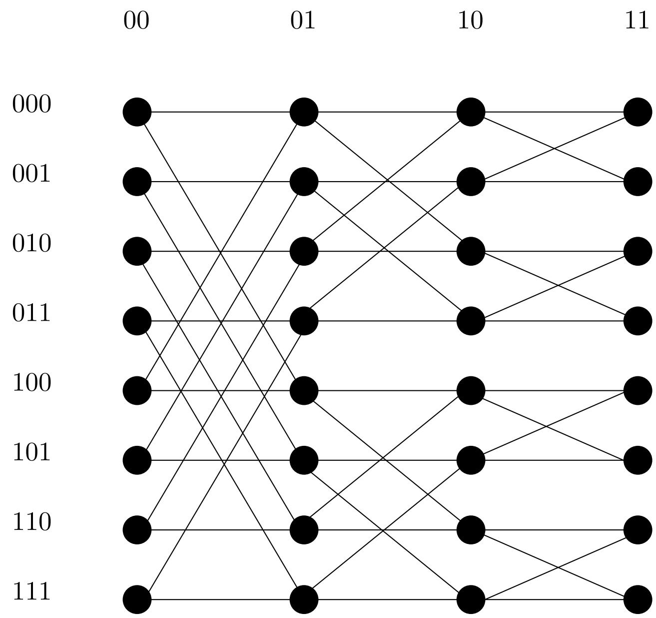


Figure 3.8: A butterfly network with 32 nodes

one disk consists of three phases, as follows. First, every node chooses an index uniformly at random from the interval $[1, \ell]$. As explained before, no two nodes will choose the same index w.h.p. Then, every node broadcasts its index and ID as in algorithm 3, but in this case they keep track of the ID of its predecessor only and the process runs for just $O(\log^2 \ell)$ steps. As shown in lemma 18, at this point all nodes have achieved at least one transmission without collision so, all nodes know who is their predecessor.

To obtain consecutive indexes, the nodes now have to pack the indexes one by one as follows. Upon receiving the new index i of its predecessor, a node redefines its index as $i + 1$ and broadcasts its new index and ID with constant probability for $O(\log \ell)$ steps. As shown in lemma 3.3.2, there will be at least one transmission without collision w.h.p. Obviously, the first node in this ordering will not have any predecessor and will start this phase of the algorithm redefining its index as 1. At this point, all nodes have consecutive indexes and have to connect as a butterfly accordingly but, they do not know yet the ID's of their butterfly neighbors with smaller index so, a final round broadcasting the new index and ID is necessary. The details can be seen in Algorithm 4

The first and third phase take $O(\log^2 \ell)$ time by definition of the algorithm. In the second phase, each of $\Theta(\log \ell)$ nodes in turn transmit for $O(\log \ell)$ steps. Hence, the overall running time of this algorithm is $O(\log^2 \ell)$.

Algorithm 4: A small-diameter spanner construction. β_4 is a constant.

for *each non-bridge node in parallel* **do**
    predecessor.ID $\leftarrow$ *NULL*;
    choose an integer index uniformly at random from the interval $[1, \ell]$;
    for $\beta_6 \log^2 \ell$ *steps* **do**
        $transmit \leftarrow \begin{cases} true & \text{with probability } p = 1/\beta_7 \log \ell \\ false & \text{with probability } 1-p \end{cases}$
        if *transmit* **then** broadcast $\langle index, ID \rangle$;
        else if *an index is received* **then**
            update *predecessor.ID* accordingly;
        end
    end
    $index \leftarrow 0$;
    if predecessor.ID$\neq$NULL **then**
        wait until an index from *predecessor.ID* is received;
    end
    $index \leftarrow index + 1$;
    for $\beta_8 \log \ell$ *steps* **do**
        broadcast $\langle index, ID \rangle$ with probability $1/\beta_9$;
    end
    for $\beta_6 \log^2 \ell$ *steps* **do**
        $transmit \leftarrow \begin{cases} true & \text{with probability } p = 1/\beta_7 \log \ell \\ false & \text{with probability } 1-p \end{cases}$
        if *transmit* **then** broadcast ⟨index,ID⟩;
        else if *an index is received* **then**
            store ID's of butterfly neighbors according with the index;
        end
    end
end

3.4 Conclusions

The bootstrapping protocol described in this chapter, builds a hop-optimal constant-degree Sensor Network under the constraints of the Weak Sensor Model in $O(\log^2 \ell)$ time w.h.p. The time bounds are for the MIS algorithm $O(\log^2 \ell)$, for the broadcast algorithm $O(\log \ell)$, and for the spanner algorithm $O(\log^2 \ell)$. Hence, the total running time is upper bounded by $O(\log^2 \ell)$.

There is a trade-off among the maximum degree, the length of the optimal path and the density given by

There is a path of $\leq \left\lceil \frac{D(u,v)}{r} \frac{c\sqrt{4+\alpha^2}}{b-a} \right\rceil - 1 + O(\log \ell)$ hops w.h.p.

The degree of any bridge is $\leq 3\lceil \frac{4}{a\sqrt{3}} \rceil \left(\lceil \frac{4}{a\sqrt{3}} \rceil + 1 \right) + 1$ w.h.p.

The density of nodes is $\frac{n}{\ell^2} > 5\frac{4+\alpha^2}{\alpha} \left(\frac{c}{b-a} \right)^2 \frac{\ln \ell}{r^2}$.

Where $0 < a < 1$, $a < b \leq 1$, $c > 1$ and $0 < \alpha \leq 1$.

The longer the edges covered, the lower density and smaller number of hops in the optimal path but, the degree is bigger.

Notice that in our construction, only three ranges of transmission are used, namely $ar/2$, $br/2$ and r. Hence, the specific values of a and b are hardware dependent.

Notice also that for any of the various parts of the bootstrapping algorithm no synchronicity assumption is needed. Furthermore, neighboring disks do not need to be running the same phase of the algorithm. Regarding failures, the MIS algorithm and its final broadcast algorithm as well as the linked list spanner construction algorithm are also maintenance algorithms since both bridge and non-bridge nodes keep broadcasting forever. If a bridge node fails, after some time non-bridge nodes will detect the absence of their bridge broadcast and will restart the MIS algorithm to obtain a new bridge. On the other hand, if a non-bridge node fails, its successor and predecessor will interconnect within the next round of the spanner construction. If the butterfly network spanner is used instead and a link is lost, the butterfly network can be simply rebuilt locally from scratch.

Chapter 4

Survey: The Wake-up and Broadcast Problems in Radio Networks

I am monarch of all I survey;

William Cowper (1731-1800)

In this chapter, we will survey the literature of the Wake-Up problem and the closely related Broadcast problem in Radio Networks. Our discussion will emphasize the setting of Ad-hoc Radio Networks with limited topology information.

We will begin by reiterating some basic facts about Radio Networks. The theoretical model of Radio Networks used in the algorithmic literature dates to the seminal work of Bar-Yehuda, Goldreich and Itai [BYGI92]. We will restate their model here and results presented will by default be based in this model. A Radio Network consists of nodes capable of communicating via limited-range radio transmissions. We will assume that time is divided into time-slots and all nodes have time-slots of the same length. A node is either transmitting or waiting to receive in any given time-slot.

We say that the network is a single hop network if all nodes are in transmission range of each other. We say that the network is multi-hop if some nodes are not in range of each other. In almost all cases the network topology can be modeled by an undirected strongly connected graph. Most results for Broadcast and Wake-up in multi-hop networks ignore the geometric nature of the underlying connectivity graph and are argued for strongly connected *directed* networks with arbitrary topologies. In both the single-hop and multi-hop networks a node receives a transmission of a neighboring node if and only if exactly one neighboring node is transmitting and all the other nodes are silent. If more than one neighboring node transmits in a given time-slot then we say that a collision has occurred and none of the nodes are capable of identifying that a collision has occurred. A radio network in this setting is often called a **no-CD** radio network.

In our presentation of the results, we will use D to denote the diameter of a multi-hop network, n to represent the number of nodes and ϵ to represent the failure probability of a randomized protocol.

4.1 The Broadcast Problem

In the broadcast problem, we have a single node designated the source that has a message, which needs to be communicated to all other nodes in the network. We assume that the broadcast algorithm proceeds through *non-spontaneous* wake-up, i.e., nodes do not take any action until the source message has reached them. Running time is measured by the number of time-slots before all nodes in the network have the message. It is important to note that for broadcasting we can assume that all the nodes have their clocks globally synchronized, i.e., they all know the time-slot number since the start of the protocol. In the case of multi-hop networks, this is possible because the synchronization information can be propagated with the broadcast message itself. This important property allows Broadcast to be more efficient than Wake-up (defined in Section 4.2 in most cases.

4.1.1 Algorithms for Computing Centralized Broadcasting Schedules

Chlamtac and Kutten [CK85] considered the problem of designing a centralized broadcast schedule assuming that a single node has complete information about the topology of the network and the source of the broadcast. They proved that the problem of determining an optimal deterministic schedule for broadcasting with complete topology information is $\mathcal{NP}$-hard. Chlamtac and Weinstein [CW91] presented the *Wave Expansion Broadcast* (WEB) algorithm, which uses repeated calls to the *Spokesman Election Algorithm* (SEA) to produce a $\mathcal{O}(D \log^2(n/D))$ time-slot deterministic protocol for Broadcast. The WEB algorithm runs in polynomial time. This approach can be used to compute an optimal deterministic broadcasting schedule in a distributed manner using special control channels. However, The number of control messages required may be quadratic in the number of nodes in the Network [Wei87].

Gaber and Mansour [GM03] presented both randomized and deterministic polynomial-time algorithms to compute the optimal broadcast schedule. For any network they demonstrated a randomized polynomial-time algorithm that computes a schedule of length $\mathcal{O}(D+\log^5 n)$ and a deterministic algorithm that computed a schedule of length $\mathcal{O}(D + \log^6 n)$. For $D \in \Omega(\log^5 n)$ the schedule computed by the randomized algorithm is of length $\mathcal{O}(D)$ and hence is asymptotically optimal. It has been proved that there are constant diameter networks for which no schedule can be of length less than $\Omega(\log^2 n)$ [ABNLP91]. This result is described in detail in section 4.1.3. It is not known if there are (optimal) $\mathcal{O}(D)$ schedules for broadcast when D is in the range $\Omega(\log^2 n)$ and $\mathcal{O}(\log^5 n)$.

4.1.2 Randomized Broadcast Protocols

An exponential gap between determinism and randomization for Broadcast protocols in multi-hop networks was proved in [BYGI92]. The authors demonstrated the first randomized distributed protocol for Broadcast. This $\mathcal{O}((D+\log n/\epsilon)\log n)$ time protocol assumed no knowledge of the network topology, save n and an upper bound on the maximum degree in the network, that we can assume to be n. For a constant diameter network, this randomized protocol requires only poly-logarithmic high-probability-time for Broadcast. This was contrasted with an $\Omega(n)$ lower bound for deterministic broadcasting on a two-hop network, even though all nodes have knowledge of n, the number of nodes in the network. The lower bound holds under the situation that a collision may appear as a clear transmission or may appear as channel noise, with the worst case chosen for the algorithm by an adversary. In the case where a collision is always distinguishable from a clear transmission, the lower bound does not hold. An exponential gap between determinism and randomization still exists if we make the weaker assumption as shown in [KP02]. There exist diameter-4 networks such that all deterministic broadcasting protocols take $\Omega(\sqrt[4]{n})$ to complete.

The algorithm presented in [BYGI92] is optimal [ABNLP91, KM98] for broadcast in a network having unknown topology, provided $D \leq n^{1-\epsilon}$. But it is off by a logarithmic factor when D is close to n. Czumaj and Rytter [CR06] provide an improvement on this algorithm for deep networks, i.e., networks with large diameter. Their protocol is critically dependent on global synchronization that can be achieved by a broadcast protocol. Their chief relevant results are:

- A $\mathcal{O}(D \log(n/D) + \log^2 n)$ time, optimal [ABNLP91, KM98], high-probability broadcast algorithm for Radio Networks with unknown topology.
- A randomized algorithm that completes broadcast in $\mathcal{O}(n)$ time on all n-node networks. Using repeated doubling, this algorithm works even if the nodes have no knowledge of n.
- The algorithms presented above are presented in terms of both deterministic and randomized *selecting sequences*. While randomized selecting sequences are easier to understand and analyze, deterministic selecting sequences allow shorter messages to be exchanged between nodes.

The improvement in the running time is obtained because of better selecting sequences. In the analysis the network is decomposed into layers, with the source belonging to layer 0 and all nodes i hops away from the source in layer i. The selecting sequences used are optimized so that the algorithm will clear quickly on the smaller layers and there can only be a small bounded number of large layers. Independently, Kowalski and Pelc [KP05] also demonstrated a Broaadcast Protocol on directed networks of unknown topology that runs in expected time $\mathcal{O}(D \log(n/D) + \log^2 n)$. Their algorithm uses special sequences of probabilities called *universal sequences*.

4.1.3 Lower Bounds on Randomized Broadcast Protocols

The first non-trivial lower bound for Broadcast in a network with unknown topology was presented by Alon at al [ABNLP91]. Using the probabilistic method they demonstrated the existence of a radius-2 network for which no broadcast protocol could complete in less than $\Omega(\log^2 n)$ time-slots. Note that this is a lower bound on *all* protocols, randomized and deterministic. This result is easily extended to show that a lower bound of $\Omega(\log n \log(n/\epsilon))$ on the ϵ-failure probability time to broadcast to a radius-2 network.

Kushilevitz and Mansour [KM98] presented a tighter lower bound of $\Omega(D\log(n/D))$ on the expected time taken by any broadcast protocol even when nodes have knowledge of n and D. They first proved an expected-time lower bound of $\Omega(\log n)$ for the clear transmission problem in one-hop networks. Building on this lower bound in a D-layered network, they proved the Broadcast lower bound. Note that they proved that for any randomized protocol, there exists a network such that the protocol takes $\Omega(D\log(n/D))$ expected time to complete. This lower bound can therefore be combined with the previous lower bound of Alon et al. The combined lower bound is tight for broadcast in an unknown topology network as mentioned in Section 4.1.2. Liu and Prabhakaran [LP02] reproved the Kushilevitz-Mansour lower bound using Yao's Minimax Principle. Notably, their technique does not rely on a reduction to prove the lower bound for non-uniform protocols.

4.2 The Wake-up Problem

The Wake-up problem can be considered as a generalization of the broadcast problem. Given an arbitrary directed radio network, we assume initially that all nodes are asleep. Each node can be woken up spontaneously by an adversary, or it can be activated by receiving a wake-up signal from an in-neighbor. However, if more than one of its in-neighbors transmits in a given time-slot then the transmissions collide, the node does not detect any signal and it does not get activated. Whenever a node is activated, it begins to execute its wake-up protocol. The protocol determines the time-slots during which the node transmits its wake-up signal. We will assume that the adversary that wakes up the nodes can make its decision to wake up a node based on full knowledge of the protocol used and all previous actions taken by nodes. The wake-up time is measured as the number of time-slots till all the nodes in the network are woken up starting from the first spontaneous wake-up. We will restrict ourself here to randomized wake-up protocols in both the one-hop and multi-hop setting. Most protocols considered in this section will have the failure probability ϵ as an input parameter to the protocol.

4.2.1 Randomized One-hop Wake-up Protocols

Gąsieniec, Pelc and Peleg [GPP01] first studied the wake-up problem in **no-CD** Radio Networks. They particularly contrasted protocols for the locally synchronous setting and the globally synchronous setting. In the former, all nodes have access to a global clock, i.e., have knowledge of the time-slot-number since the first node was woken up. In the locally synchronous setting, nodes are assumed to have synchronized time-slots of the same length but do not have access to a common time-slot number. They also consider the cases of protocols with and without knowledge of n, and protocols for labelled and unlabelled nodes. We summarize their results in table 4.1, taken from [JS05].

Known n	Labels	Global Clock	Local Clock
Yes	Yes	$\mathcal{O}(\log n \log(1/\epsilon))$	$\mathcal{O}(n \log(1/\epsilon)$
Yes	No	$\mathcal{O}(\log n \log(1/\epsilon))$	$\mathcal{O}(n \log(1/\epsilon)$
No	Yes	–	$\mathcal{O}(n^2 \log(1/\epsilon))$
No	No	Not considered	$\mathcal{O}(n^2 \log(1/\epsilon))$

Table 4.1: Summary of One-hop wake-up protocols presented in [GPP01].

Jurdziński and Stachowiak [JS05] presented new protocols and improved on almost all of the results present above. In some cases involving local synchronization, their results are an exponential improvement (see Table 4.2). In the case of unknown labels and no knowledge of n, they demonstrated an exponential gap between local and global synchronization.

4.2.2 Lower Bounds on Randomized One-hop Wake-up

Kushilevitz and Mansour proved the first known lower bound for one-hop wake-up in no-CD Radio Networks. In the setting with globally synchronized labelled nodes, they

Known n	Labels	Global Clock	Local Clock
Yes	Yes	$\mathcal{O}(\log n \log(1/\epsilon))$	$\mathcal{O}(\log n \log(1/\epsilon)$
Yes	No	$\mathcal{O}(\log n \log(1/\epsilon))$	$\mathcal{O}(\log n \log(1/\epsilon)$
No	Yes	$\mathcal{O}(\log n \log(1/\epsilon)$	$\mathcal{O}(\log n \log(1/\epsilon))$
No	No	$\mathcal{O}(\log^2 n(\log\log n)^3 \log(1/\epsilon(n))$	$\mathcal{O}\left(\frac{n\log(1/\epsilon)}{\log n}\right)$

Table 4.2: Summary of One-hop wake-up protocols presented in [JS05].

proved a lower bound of $\Omega(\log n)$ on the time till wake-up. This shows that the upper bounds in the first 3 rows of table 4.2 are tight in expectation.

Jurdziński and Stachowiak [JS05] proved a lower bound of $\Omega(\log n \log(1/\epsilon)/(\log\log n + \log\log(1/\epsilon))$ on the wake-up problem for globally synchronized, unlabelled nodes running fair protocols (defined in Section 5.1). This lower bound is weaker than the previous one in expectation, but is parameterized by ϵ. The paper also considers protocols in the interesting cases where n is not known but wake-up is required with ϵ_n-failure-probability (ϵ_n is a decreasing function of n). The other significant lower bound proved is for the $w$eakest model, where there are no labels, n is not known to the nodes, and the nodes are only locally synchronized. In this setting a lower bound of $\Omega(n/\log n)$ is proved on expected time till wake-up.

4.2.3 Protocols for Multi-hop Wake-up

Deterministic and Randomized wake-up protocols for **no-CD** Radio Networks were first studied by Chrobak, Gąsieniec and Kowalski [CGK04]. They presented a simple doubling-probability randomized protocol for multi-hop wakeup in Radio Networks with no topology information. Their protocol runs in ϵ-failure-probability time of $\mathcal{O}(D \log n \log(n/\epsilon))$.

Previous to this work, the best lower bound for multi-hop wake-up in this setting is the broadcast lower bound [ABNLP91, KM98], presented in Section 4.1.3. In Chapter 6, we present an improved lower bound of $\Omega(D \log n \log(1/\epsilon))$ on fair protocols, which applies to networks with diameter of $D \in O(n/\log^2 n)$.

Chapter 5

Lower Bounds on Clear Transmissions in Radio Networks

> If only God would give me some clear sign! Like making a large deposit in my name in a Swiss bank.
>
> Woody Allen, 1935–

Any network where transmissions may collide needs a protocol for *collision-free transmissions*. Different networks provide different information about collisions. For example, on some hardware, transmitters can distinguish amongst three states at each time step: no transmission, single transmission, and collision, whereas on other hardware, transmitters can not distinguish between no transmission and collisions. In some networks, transmitters know an upper bound on their number. Sometimes, transmitters may not *snoop*, i.e., listen to the channel when not transmitting; whereas at the other extreme, transmitters may only snoop, i.e., they get no information on the channel when they are transmitting. In some networks collisions are transitive. The properties of a shared channel have a profound impact on the protocols usable on such a channel.

Sensor networks are a heavily studied example of a shared-channel network. A sensor network consists of small devices with processing, sensing and communication capabilities. These *sensor nodes* are randomly deployed over an area in order to achieve sensing tasks after self-organizing as a wireless radio network. Sensor nodes have strong limitations and operate under harsh conditions. Some of the important limitations of sensor nodes include: lack of collision detection hardware, non-simultaneous transmission and reception, and one channel of communication. We call any such network a Radio Network. Additionally, nodes in sensor networks wake up at arbitrary times. Sensor networks are even more restricted in various ways that will not concern us here. The Radio Network restrictions, along with these further restrictions, are part of the

Weak Sensor Model presented in chapter 2.

Algorithms for achieving a *clear*, that is, uncolliding, transmission have been studied in several shared-channel contention settings. In a one-hop Radio Network, the clear transmission problem is equivalent to the so-called *wake-up* and *leader election* problems. These problems differ in multi-hop networks, although, a clear transmission is necessary to achieve wake-up and leader election since, indeed, a clear transmission is necessary to solve any problem on a Radio Network.

5.1 Related Work

In this section, we will briefly survey upper and lower bounds on The Clear Transmission Problem. A more comprehensive survey can be found in [Mos07, Chapter 7]. We will cite bounds for the clear transmission problem in the literature, even when the bounds were originally state for the other problems, like Wake-Up, Leader Election and Broadcast. Our survey will focus only on the settings for which we prove lower bounds in this chapter. Note first that a *fair* randomized protocol is one in which each node transmits with the same probability of transmission in the same time-slot after starting the protocol. Such protocols are interesting because they are simple and most protocols and many lower bounds for the clear transmission problem are presented for *fair* protocols.

Hayashi, Nakano and Olariu [HNO99] presented the first $\mathcal{O}(\log^2 n)$ high-probability algorithm for the Leader-Election problem in one-hop Radio Networks. In their paper, they make the assumption that nodes can sense the channel while transmitting. While this is not realistic, in the model, we consider, all the nodes other than the elected leader will know who the leader is. Thus their algorithm works fine for the clear transmission or wake-up problems.

An ϵ-failure-probability time *fair* protocol for a clear transmission a one-hop Radio Network running in $\mathcal{O}(\log n \log(1/\epsilon))$ was introduced in [GPP00]. Strikingly, when $\epsilon = 1/n$ the same time bound can be obtained for the much more complicated problem of computing a Maximal Independent Set (MIS) in the multi-hop Weak Sensor

Model [MW05](For the model, see chapter 2).

Kushilevitz and Mansour [KM98] proved the first lower bound of $\Omega(\log n)$ on the expectation of the running time of any randomized protocol for clear transmissions in radio networks. A lower bound of $\Omega(\log n \log(1/\epsilon)/(\log\log n + \log\log(1/\epsilon))$ for achieving an ϵ-failure-probability clear transmission in a one-hop, globally-synchronized Radio Network was proved in [JS05] (see section 4.2.2). The latter lower bound is tighter than the previous one if ϵ is $o(1/\log n)$ and applies only to fair protocols. A more detailed survey of randomized protocols for wake-up in one-hop Radio Networks can be found in sections 4.2.1 and 4.2.2.

5.2 Our Results

In this chapter, we close the gap between the best upper and lower bounds for fair protocols for the clear transmission problem by proving a stronger lower bound: it takes time $\Omega(\log n \log(1/\epsilon))$ to solve the problem of achieving an ϵ-failure-probability clear transmission using a fair protocol in a one-hop setting, which implies, for example, the $\Omega(\log n)$ lower bound on the expectation of any fair randomized algorithm for clear transmission. Our lower bounds apply to any network with the following characteristics:

- *Shared channel of communication:* All nodes communicate with their neighbors using broadcasts that are transmitted on a shared channel.

- *Lack of a collision detection mechanism:* Nodes do not have the ability to distinguish between a collision on the channel or lack of a transmission.

- *Non-simultaneous transmission and reception:* Nodes cannot snoop on the channel while transmitting.

- *Local synchronization:* Time is assumed to be divided into slots and all nodes have the same clock frequency.

- *Adversarial wake-up schedule:* Nodes are woken up by an adversary.

Indeed, we will prove our lower bound with the following weak adversary: the adversary may chose an $i \in [1, \log n]$, and 2^i nodes wake up at time 0. Our techniques also give us a lower bound of $\Omega(\log \log n \log(1/\epsilon))$ on clear transmissions using fair protocols in the important multi-hop setting of nodes in a Connected RGG.

5.3 Fair Protocols in One-Hop Radio Networks

We first define what the clear transmission problem is in the one-hop setting. The nodes are all connected to a common broadcast channel and each transmission is available for snooping to all non-transmitting nodes. The connectivity of the nodes can be modeled as a clique. In this case we assume that all nodes know an upper bound on the number of their neighbors. In this setting, a clear transmission is achieved if exactly one node transmits in a time slot.

We prove our lower bounds under the assumption of the existence of a weak adversary that, at a given time, wakes up (i.e. turns on) some subset of nodes. We call them *active* nodes. Upon waking up, the active nodes start the execution of a protocol to achieve a clear transmission. All non-active nodes do not participate in the protocol.

We define a *randomized fair protocol* for clear transmission to be a sequence $p_1, p_2, \ldots$ where each node transmits with probability p_ℓ in the ℓ^{th} time step after waking up. Given our adversary, this means that all active nodes transmits with same probability as each other in each time slot.

We seek a lower bound on the number of time-slots required to achieve an ϵ-failure-probability clear transmission. We simplify the analysis in two ways. First, we further weaken the adversary by requiring that the number of nodes participating can only be one of $\{2^i | 0 \leq i \leq \log_2 n\}$. Secondly, we assume that all $p_\ell \in \{2^{-j} | 1 \leq j \leq \log_2 n\}$. If this assumption is not true of a particular algorithm A, we can always produce an algorithm A' from A by replacing one attempt in A by a constant number of attempts in A' where the probabilities of transmission in A' have been rounded off to the closest power of $1/2$.

One of the principal benefits of our weak adversary is that, the probability P_ℓ of a clear transmission by time ℓ is the same for any permutation of $p_1, p_2, \ldots, p_\ell$. Therefore, we need not bother with what order the steps are taken in, but only how many times the protocol fires with each probability.

Let t_j be the number of time-slots that nodes are transmitting with probability 2^{-j}. Let p_{ij} denote the probability that 2^i nodes fail to clear when they all transmit with probability 2^{-j}. Thus we know that:

$$\begin{aligned} p_{ij} &= 1 - 2^i \frac{1}{2^j}\left(1 - \frac{1}{2^j}\right)^{2^i-1} \\ &= 1 - 2^{i-j}(1-2^{-j})^{2^i-1} \end{aligned}$$

The total probability of failure for any number of active nodes, 2^i, needs to be bounded by:

$$\begin{aligned} &\prod_j p_{ij}^{t_j} \leq \epsilon \\ \iff &\sum_j t_j \ln(p_{ij}) \leq \ln(\epsilon). \end{aligned}$$

A lower bound is achieved by minimizing the total number of time-slots needed to satisfy the previous constraints. This can be formulated as the following *primal* linear program:

$$\text{Minimize } \mathbf{1}^T\mathbf{t},$$

$$\text{subject to:}$$

$$\mathbf{P}\mathbf{t} \geq \boldsymbol{\epsilon}$$

$$\mathbf{t} \geq \mathbf{0}$$

$$\text{where:}$$

$$\mathbf{t}^T \triangleq [t_1, t_2, \ldots]_{\log_2 n},$$

$$\boldsymbol{\epsilon}^T \triangleq [-\ln(\epsilon), -\ln(\epsilon), \ldots]_{\log_2 n+1},$$

$$\mathbf{P} \triangleq \begin{bmatrix} -\ln(p_{01}) & -\ln(p_{02}) & \ldots \\ -\ln(p_{11}) & -\ln(p_{12}) & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix}_{(\log_2 n+1)\times\log_2 n},$$

which yields the following *dual*:

$$\text{Maximize } \boldsymbol{\epsilon}^T\mathbf{u},$$

$$\text{subject to:}$$

$$\mathbf{P}^T\mathbf{u} \leq \mathbf{1}$$

$$\mathbf{u} \geq \mathbf{0}.$$

$$\text{where:}$$

$$\mathbf{u}^T \triangleq [u_1, u_2, \ldots]_{\log_2 n+1}$$

The primal linear program has a finite minimum solution, and hence its dual has a finite maximum solution. The value of the objective function for every feasible solution of the dual is a lower bound on the minimum value of the objective function for the primal. Thus any feasible solution for the dual will give a lower bound on the number of time-slots required to achieve a clear transmission with failure probability ϵ.

Suppose that the j^{th} row, $(\mathbf{P}^T)_j$, of $\mathbf{P}^T$ has the maximum row sum, and let $r(\mathbf{P}^T) = (\mathbf{P}^T)_j\mathbf{1}$. Now we set $\mathbf{u} = [1/r(\mathbf{P}^T), 1/r(\mathbf{P}^T), \ldots]$. This value of $\mathbf{u}$ satisfies all constraints of the dual. The value of the objective function of the dual is simply $\boldsymbol{\epsilon}^T\mathbf{u}$. To obtain the value of the objective function of the dual we need to find the row of $\mathbf{P}^T$ with the largest row sum which is the same as the column of $\mathbf{P}$ with the largest column sum.

Lemma 19. *The trace of every column vector of the constraint matrix* $\mathbf{P}$ *of the primal is* $\mathcal{O}(1)$.

Proof. We begin by stating the following useful inequality [Mit64, Section 2.68]:

$$e^{-x/(1-x)} \leq 1 - x \leq e^{-x}, 0 < x < 1. \tag{5.1}$$

The sum of the elements of a column j of $\mathbf{P}$ is:

$$\begin{aligned} S_j &\leq \sum_i -\ln(1 - 2^{i-j}(1-2^{-j})^{2^i-1}) \\ &\leq \sum_i -\ln\left(e^{-2^{i-j}(1-2^{-j})^{2^i-1}/(1-2^{i-j}(1-2^{-j})^{2^i-1})}\right) \text{ (By Inequality 5.1)} \\ &= \sum_i \frac{2^{i-j}(1-2^{-j})^{2^i-1}}{1-2^{i-j}(1-2^{-j})^{2^i-1}}. \end{aligned}$$

Let $y_{ij} \triangleq 2^{i-j}(1-2^{-j})^{2^i-1}$.

$$\begin{aligned} S_j &= \sum_i \frac{y_{ij}}{1-y_{ij}} \\ &\leq \sum_i \frac{y_{ij}}{1-y_{max}} \text{ (where } y_{max} = \max_{ij}\{y_{ij}\}). \end{aligned}$$

Now we derive an upper bound on y_{max}:

$$\begin{aligned}
y_{max} &= \max_{ij} y_{ij} \\
&= \max_{ij} 2^{i-j}(1-2^{-j})^{2^i-1} \\
&\leq \max_{ij} 2^{i-j}e^{-2^{i-j}+2^{-j}} \text{ (By Inequality 5.1)} \\
&\leq \max_{ij} \sqrt{e}\frac{2^{i-j}}{e^{2^{i-j}}} \ (\because j \geq 1) \\
&\leq \frac{1}{\sqrt{e}} \text{ (The function is maximized, when } i = j\text{).}
\end{aligned}$$

Therefore:

$$S_j \leq \frac{\sqrt{e}}{\sqrt{e}-1}\sum_i y_{ij}$$

We derive an upper bound on the right hand side sum.

$$\begin{aligned}
\sum_i y_{ij} &= \sum_i 2^{i-j}(1-2^{-j})^{2^i-1} \\
&\leq \sum_i 2^{i-j}(e^{-2^{-j}})^{2^i-1} \text{ (By Inequality 5.1)} \\
&= \sum_i 2^{i-j}e^{-2^{i-j}+2^{-j}} \\
&\leq \sqrt{e}\left(\sum_{i\geq j} 2^{i-j}e^{-2^{i-j}} + \sum_{i<j} 2^{i-j}e^{-2^{i-j}}\right) (\because j \geq 1) \\
&\leq \sqrt{e}\left(\sum_{k\geq 0} 2^{k}e^{-2^{k}} + \sum_{k\geq 1} 2^{-k}e^{-2^{-k}}\right) \\
&\leq \sqrt{e}\left(\sum_{k\geq 0} 2^{k}e^{-2^{k}} + \sum_{k\geq 1} 2^{-k}\right) \\
&\in O(1) \text{ (Because both the sums are bounded by a constant)}
\end{aligned}$$

$$\implies S_j \in O(1).$$

□

Theorem 20. *Every fair randomized algorithm to achieve an ϵ-failure-probability clear* *transmission in a one-hop Radio Network requires* $\Omega(\log n \log(1/\epsilon))$ *time-slots.*

Proof. From lemma 19, we know that $r(P^T) \in O(1)$, then $\boldsymbol{\epsilon}^T\mathbf{u} \in \Omega(\log n \log(1/\epsilon))$. From this we can conclude that the dual linear program has a feasible solution with objective function evaluating to $\Omega(\log n \log(1/\epsilon))$. Since we showed earlier that the solution to the primal linear program gives a lower bound on the number of time-slots required to achieve a clear transmission with probability $1 - \epsilon$, the statement of the theorem holds. $\square$

5.4 Fair Protocols for Geometrically Distributed Nodes

Here we consider the problem of achieving a clear transmission under the following conditions:

The nodes are connected by a broadcast channel to some subset of nodes and each transmission made by a node is available to its neighbors only, but it can interfere with all transmissions originating in a two-hop neighborhood. The specific case we will derive a lower bound for is the case of nodes consistent with the Weak Sensor Model distributed randomly in the plane with limited transmission range but adequate density to ensure connectivity. The connectivity of the nodes can be modeled as a *Random Geometric Graph (RGG)*(Refer to Section 2.1.2). In this case, we assume that nodes know an upper bound on the number of their neighbors with a probability given by the parameter conditions for connectivity.

In this setting, we say that a clear transmission occurred if exactly one node is transmitting and no other nodes within two hops of it are transmitting. Then, the clear transmission problem in a multi-hop setting is solved after every node either produces or receives a clear transmission.

In a $G(n, r, \ell)$ satisfying the connectivity conditions explained previously, the number of nodes contained in any circle of radius $\Theta(r)$ is $\Theta(\log n)$ with high probability. This can be proved by a simple application of the Chernoff-Hoeffding bounds. Then, we complete our lower bounds with the following corollary, which can be obtained as a simple application of Theorem 20.

Corollary 21. *Every fair protocol to solve the clear transmission problem with* ϵ*-failure-probability in a Radio Network with geometrically distributed nodes requires* $\Omega(\log\log n \log(1/\epsilon))$ *time slots, where* $\epsilon \geq 1/n^{\gamma}$ *for some constant* $\gamma > 0$*.*

Proof. Replacing the appropriate density for any one-hop neighborhood in this setting, i.e. $\Theta(\log n)$ instead of n, in theorem 20 the corollary follows. $\square$

Chapter 6

A Lower Bound on Fair Wake-up Protocols in Multi-hop Networks

> Wake! For the Sun, who scatter'd into flight
> The Stars before him from the Field of Night,
> Drives Night along with them from Heav'n, and strikes
> The Sultan's Turret with a Shaft of Light.
>
> Omar Khayyám / Edward Fitzgerald
> The Rubáiyát of Omar Khayyám

An ad-hoc multi-hop Radio Network can be modeled as a directed, strongly connected graph with n vertices. The nodes of the graph represent processors with radio transmission capability and the outgoing edges represent their transmission links. We assume that the nodes have no knowledge of the network topology, save the total number of nodes n and the diameter D. For any node its in-neighbors (out-neighbors) are those that are connected to it with incoming edges (outgoing edges). We assume that all transmitters have time divided into identical length time-slots but are not synchronized. Each transmitter has a unique label (identifier or ID) that is $\Theta(\log n)$ bits in length. In our discussion, we do not consider protocols that use the ID in any way.

We will now re-define the Wake-up problem. We assume initially that all nodes are asleep. Each node can be woken up spontaneously, or it can be activated by receiving a wake-up signal from an in-neighbor. However, if more than one of its in-neighbors transmits in a given time-slot then the transmissions collide, the node does not detect any signal and it does not get activated. Additionally, since we assume in radio networks that there is no channel feedback [BYGI92], the transmitters have no knowledge of the collision either. Whenever a node is activated, it begins to execute its wake-up protocol. The protocol determines the time-slots during which the node transmits it wake-up signal. In our model we assume that the nodes are not globally synchronized and hence determining if a wake-up signal is to be sent during a given time-slot is done on the basis of time-slot number since activation and any previous activation signals received. The running time of the protocol is the number of time-slots from the first spontaneous activation till the time that all nodes are activated. Additionally a *fair* protocol is one in which all the nodes transmit with the same probability in the same time-slot after waking up. This pre-condition is stronger than that for a *uniform* protocol in that nodes running a uniform protocol may have different probabilities of transmission in the *same* time-slot after activation, given that they have different histories.

The wake-up problem is closely related to the problem of broadcasting in a network of unknown topology. In one-hop networks, the wake-up problem is closely related to the Leader Election problem and is identical to the Clear Transmission problem. The related literature is comprehensively dealt with in Chapter 4.

6.1 Preliminaries

We present some easily provable bounding results.

Lemma 22. *Given a set of probabilities $p_1, p_2, \ldots, p_\ell \leq 1/2$, such that $\sum_i p_i = x$,*

$$\sum_{i=1}^{\ell} p_i \prod_{j \neq i} (1 - p_j) \leq \begin{cases} x e^{-x+1/2} & x \geq 1/2 \\ x & 0 < x < 1/2. \end{cases}$$

Proof.

$$\begin{aligned} \sum_{i=1}^{\ell} p_i \prod_{j \neq i} (1 - p_j) &\leq \sum_{i=1}^{\ell} p_i \prod_{j \neq i} e^{-p_j} \\ &\leq \sum_{i=1}^{\ell} p_i e^{1/2 - \sum_{j=1}^{\ell} p_j} \\ &= \sum_{i=1}^{\ell} p_i e^{-x+1/2} \\ &= x e^{-x+1/2} \end{aligned}$$

Proving the upper bound in the interval $[0, 1/2)$ only requires the observation that the function to be upper-bounded is convex in that interval. $\square$

We re-iterate an inequality from Chapter 5 and present another from [Mit64]:

$$e^{-x/(1-x)} \leq 1 - x \leq e^{-x},\ 0 < x < 1. \tag{5.1}$$

$$e^{-x-x^2} \leq 1 - x \leq e^{-x},\ 0 < x < \frac{1}{2}. \tag{6.1}$$

6.2 Lower Bound for Fair Protocols

In this section, we will prove a lower bound on fair randomized protocols for the multi-hop wake-up problem. Any fair randomized protocol for wake-up can be modeled by a sequence of probabilities of transmission.

Furthermore, we will assume in our fair protocol that the transmission probability in all time-slots lies in the interval $[1/n, 1/2]$. Through our wake-up schedule, we will ensure that at least two nodes transmit simultaneously in all time-slots, therefore having probability of transmission exceeding $1/2$ only reduces the probability of success. Additionally rounding up transmission probabilities in the interval $(0, 1/n)$ changes our lower bound by a constant factor only. We will also assume that $1/n \leq \epsilon \leq 1/2$. This restriction on ϵ is technical and is explained at the end of Lemma 24, but it still allows for many meaningful values. Note that, in a multi-hop network, we require at least $D - 1$ clear transmissions in order to wake up the entire network. We will prove our lower bound for a network with the following topology (as illustrated in figure 6.1):

- The network has $n = N + 2(D-2) + 1$ nodes in $D + 1$ layers indexed from 0 to D.
- Layer D has only one node.
- Of the remaining D layers, $D - 1$ of them have 2 nodes each and one of them has N nodes. The *large* layer is selected arbitrarily. The other layers are called *small* layers.

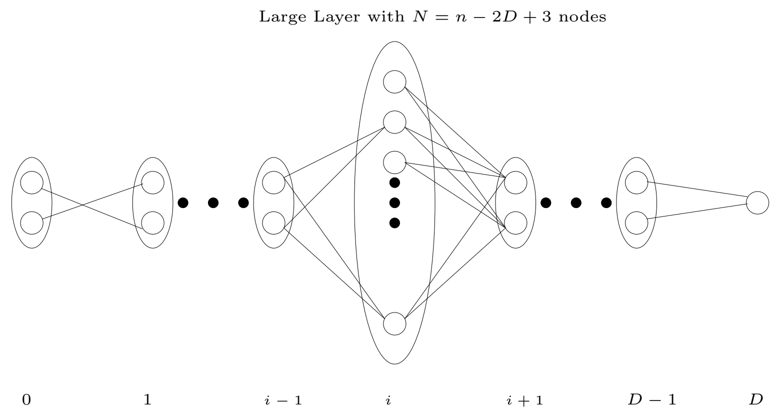


Figure 6.1: Network Topology for Wake-up Lower Bound

- All nodes in each layer are connected to each other.
- If layers i and $i+1$ are small layers then both nodes in layer i are connected to both nodes in layer $i+1$.
- If layer i is the large layer, then all nodes in layer i are connected to both nodes in layer $i+1$.
- If layer i is a small layer and layer $i+1$ is large, then both nodes in layer i are connected to two nodes in layer $i+1$.

We present first a summary of our lower bounding technique. If a fair protocol produces a clear transmission quickly when only two nodes are concurrently contending, then the probability of transmission is high in the early time-slots (lemma 23). A high probability of transmission in the early time-slots enables us to design a wake-up schedule in the *large* layer that maintains high contention[1] for many time-slots (lemmas 24 and 25). The high-contention time-slots give us a low probability of clearance and high-delay. This delay is long enough to prove the lower bound. On the other hand if the early time-slots of the protocol have low probabilities of transmission, then the protocol will clear each layer of our network slowly enough to prove the lower bound.

[1]We refer to the total probability of transmission of all contending nodes in a single time-slot during execution as the *contention* of that time-slot

Theorem 26 states the main theorem of this chapter and Corollary 27 shows that our lower bound is tight for high-probability wake-up protocols when $D \in O(n/\log^2 n)$.

Lemma 23. *If two nodes in a one-hop network that wake up at the same time and run fair randomized protocols (with probabilities of transmission: $p_1, p_2, \dots$) for wake-up, achieve a clear transmission in ϵ-failure-probability time t, then $\sum_{i=1}^{t} p_i \geq \delta_1 \log(1/\epsilon)$, where $\delta_1 = 27\ln(1/\epsilon)/(26 + 5\sqrt{10})$.*

Proof. The probability of failure in t time-steps is upper bounded by:

$$\prod_{i=1}^{t} [1 - 2p_i(1 - p_i)] \leq \epsilon.$$

Since the success probability never exceeds 1/2, we can use equation 6.1:

$$\Longrightarrow \prod_{i=1}^{t} e^{-2p_i(1-p_i)-(2p_i(1-p_i))^2} \leq \epsilon$$

$$\Longrightarrow \sum_{i=1}^{t} 2p_i(1 - p_i) + (2p_i(1 - p_i))^2 \geq \ln(1/\epsilon)$$

$$\Longrightarrow \sum_{i=1}^{t} 2p_i(1 + p_i - 4p_i^2 + 2p_i^3) \geq \ln(1/\epsilon)$$

The term in parentheses is maximized in $[0, 1]$ by $p_i = \dfrac{2}{3} - \dfrac{\sqrt{10}}{6}$

$$\Longrightarrow \sum_{i=1}^{t} p_i \geq \frac{27\ln(1/\epsilon)}{26 + 5\sqrt{10}}$$

□

Lemma 24. *Suppose a randomized, fair wake-up protocol running on two nodes in a one-hop network succeeds in making a clear transmission with ϵ-failure-probability time $t = \delta_2 \log n \log(1/\epsilon)$ time-slots (where $1/n \leq \epsilon \leq 1/2$)then there exists a wake-up schedule for $N/2$ nodes so that for the time-interval $(t+1)\dots(t+T)$, where $T \in \Omega(N\log(1/\epsilon)/\log N)$, the failure probability exceeds $\sqrt{\epsilon}$.*

Proof. Suppose in a given time-slot we have total contention of $h \geq 1/2$, then we know from lemma 22 that the probability of success is less than $he^{-h+1/2}$. Suppose we denote

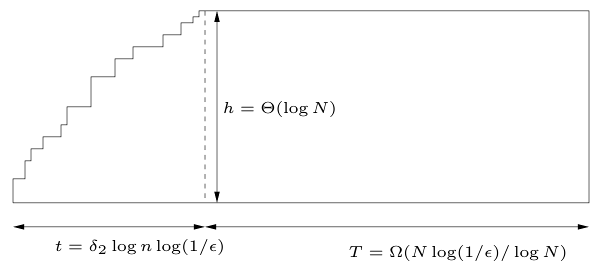


Figure 6.2: Contention over time using wake-up schedule in lemma 24

the maximum length of time T, that we can maintain such high contention and not allow failure probability to fall below ϵ then:

$$\left(1 - he^{-h+1/2}\right)^T \geq \sqrt{\epsilon}.$$

Using inequality 5.1 we get:

$$\Longleftrightarrow T \ln\left(e^{-he^{-h+1/2}/(1-he^{-h+1/2})}\right) \geq \ln\left(\sqrt{\epsilon}\right)$$

$$\Longleftrightarrow T\frac{he^{-h+1/2}}{1 - he^{-h+1/2}} \leq \ln\left(\frac{1}{\sqrt{\epsilon}}\right)$$

$$\Longleftarrow 2he^{-h+1/2}T \leq \ln\left(\frac{1}{\sqrt{\epsilon}}\right).$$

We thus obtain:

$$Th \leq \frac{e^{h-1/2}}{2} \ln\left(\frac{1}{\sqrt{\epsilon}}\right). \tag{6.2}$$

We maintain contention h from time-slot $t+1$ till time-slot $t+T$, by waking up equal numbers of nodes in each time-slot from 1 to $t+T$. We observe that the protocol has total probability in the first t time-slots of greater than $\delta_1 \log(1/\epsilon)$ as given by lemma 23. Given that we can wake-up at most $N/2$ nodes,

$$(T+t)h \geq \frac{\delta_1 N \log(1/\epsilon)}{2}. \tag{6.3}$$

Combining equations 6.2 and 6.3, we get the following condition from which we can derive satisfactory values of h and T:

$$\frac{\delta_1 N \log(1/\epsilon)}{2} - th \leq \frac{e^{h-1/2}\ln(1/\sqrt{\epsilon})}{2} \tag{6.4}$$

$$h = 2\log N$$

$$T \geq \frac{\delta_1}{4}\frac{N}{\log N}\log\left(\frac{1}{\epsilon}\right) - \delta_2 \log n \log\left(\frac{1}{\epsilon}\right) \tag{6.5}$$

This means that the number of nodes woken up in each time-slot is $\frac{2\log N}{\delta_1 \log(1/\epsilon)}$. In order for this to be integral, we have the restriction on ϵ, $1/n \leq \epsilon \leq 1/2$. □

We have thus shown how to construct a wake-up schedule such that the probability of success in the large layer is bounded in the interval $[(t+1)\ldots(t+T)]$. We must now construct a wake-up schedule that bounds the success probability in the first t time-slots. This wake-up schedule must in combination with the previous one make sure that the probability of success is not high in the early time-slots. The existence of such a wake-up schedule is the topic of the next lemma.

Lemma 25. *Given $N/2$ nodes running a uniform protocol such that two nodes running a uniform protocol produce a clear transmission with ϵ-failure-probability in less than $\Delta_2 = \delta_2 \log n \log(1/\epsilon)$ time-slots, there exists a wake-up schedule so that the failure probability of having a clear transmission within the first Δ_2 time-slots is greater than $\sqrt{\epsilon}$.*

Proof. In order to prove the lemma, we will construct a wake-up schedule consisting of *meta-schedules*. Each meta-schedule will consist of a node woken up in each of Δ_2 consecutive time-slots. Since we have $N/2$ nodes overall, we can have between 1 and $N/2\Delta_2$ meta-schedules. However, referring to lemma 24, we restrict ourselves to choosing from $\frac{2}{\delta_1}\frac{\log N}{\log(1/\epsilon)}$ to $N/2\Delta_2$ meta-schedules.

In the first Δ_2 time-slots, each meta-schedule will have time-slots with contention ranging from $1/n$ to Δ_1. We will show that if each number of meta-schedules started has to result in a clear transmission with probability of failure less than $\sqrt{\epsilon}$, then Δ_2 has to be *long*.

Consider that each time-slot of a meta-schedule is thrown into bins depending on what the contention is in a given time-slot. Thus, a time-slot where the contention is in the range $[2^k, 2^{k+1}]$ is consigned to bin k and the number of time-slots in bin k, we denote by t_k. Let us suppose that we can start 2^j meta-schedules, where j ranges from $\lceil \log \frac{2}{\delta_1} \frac{\log N}{\log(1/\epsilon)} \rceil$ to $\lfloor \log(N/2\Delta_2) \rfloor$. Also let p_{jk} denote the probability of failure in a given time-slot given that 2^j meta-schedules are concurrently in a time-slot belonging to bin k. Thus the minimum time Δ_2 such that each of these number of meta-schedules produces a clear transmission with failure probability bounded by $\sqrt{\epsilon}$ is given by the following linear program (The approach here is similar to that given in Chapter 5). A detailed proof is repeated here for clarity. 1

$$\text{Minimize} \sum_{k=-\log n}^{\log \Delta_1} t_k$$

$$\text{subject to:}$$

$$\sum_{k=-\log n}^{\log \Delta_1} t_k \ln p_{jk} \leq \ln(\sqrt{\epsilon}).$$

This can be written in matrix-form as:

$$\begin{aligned}
&\text{Minimize } \mathbf{1}^T \mathbf{t},\\
&\text{subject to:}\\
&\qquad \mathbf{P}\mathbf{t} \geq \boldsymbol{\epsilon}\\
&\qquad \mathbf{t} \geq \mathbf{0}\\
&\text{where:}\\
&\qquad \mathbf{t}^T \triangleq [t_{-\log n}, t_{-\log n+1}, \ldots]_{\Theta(\log n)},\\
&\qquad \boldsymbol{\epsilon}^T \triangleq [-\ln(\sqrt{\epsilon}), -\ln(\sqrt{\epsilon}), \ldots]_{\Theta(\log n)},\\
&\qquad \mathbf{P} \triangleq \begin{bmatrix} \ddots & \vdots & \vdots \\ \ldots & -\ln(p_{11}) & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix}_{\Theta(\log n)\times\Theta(\log n)}.
\end{aligned}$$

The dual of this is given by:

$$\begin{aligned}
&\text{Maximize } \boldsymbol{\epsilon}^T \mathbf{u},\\
&\text{subject to:}\\
&\qquad \mathbf{P}^T\mathbf{u} \leq \mathbf{1}\\
&\qquad \mathbf{u} \geq \mathbf{0}\\
&\text{where:}\\
&\qquad \mathbf{u}^T \triangleq [u_1, u_2, \ldots]_{\Theta(\log n)}.
\end{aligned}$$

The primal linear program has a finite minimum solution, and hence its dual has a finite maximum solution. The value of the objective function for every feasible solution of the dual is a lower bound on the minimum value of the objective function for the primal. Thus any feasible solution for the dual will give a lower bound on the number of time-slots required to achieve a clear transmission with failure probability ϵ.

Suppose that the k^{th} row, $\mathbf{P}_k^T$, of $\mathbf{P}^T$ has the maximum row sum, and let $r(\mathbf{P}^T) = (\mathbf{P}^T)_j\mathbf{1}$. Now we set $\mathbf{u} = [1/r(\mathbf{P}^T), 1/r(\mathbf{P}^T), \ldots]$. This value of $\mathbf{u}$ satisfies all constraints of the dual. The value of the objective function of the dual is simply $\boldsymbol{\epsilon}^T\mathbf{u}$. To obtain the value of the objective function of the dual we need to find the row of $\mathbf{P}^T$ with the largest row sum which is the same as the column of $\mathbf{P}$ with the largest column sum.

The sum of the elements of a column j of $\mathbf{P}$ is:

$$\begin{aligned}
S_k &\leq \sum_j -\ln(1-p_{jk})\\
&\leq \sum_j -\ln\left(e^{p_{jk}/(1-p_{jk})}\right) \text{ (By Inequality 6.1)}\\
&= \sum_j \frac{p_{jk}}{1-p_{jk}}\\
&\leq \sum_i \frac{p_{jk}}{1-p_{max}} \text{ (where } p_{max} = \max_{jk}\{p_{jk}\}).
\end{aligned}$$

Now we derive an upper bound on p_{max}:

$$\begin{aligned} p_{max} &= \max_{jk} p_{jk} \\ &\leq \frac{1}{\sqrt{e}} \text{ (By lemma 22)} \end{aligned}$$

Therefore:

$$S_k \leq \frac{\sqrt{e}}{\sqrt{e}-1} \sum_j p_{jk}$$

We derive an upper bound on the right hand side sum.

$$\begin{aligned} \sum_j p_{jk} &\leq \left(\sum_{\ell \geq 0} 2^{\ell} e^{-2^{\ell}} + \sum_{\ell \leq 0} 2^{\ell} \right) \\ &\in O(1) \text{ (Because both the sums are bounded by a constant)} \\ \implies S_k &\in O(1). \end{aligned}$$

From this we can conclude that the dual linear program has a feasible solution with objective function evaluating to $\Omega(\log n \log(1/\epsilon))$. This shows that we can find a value of constant δ_2 such that there is a wake-up schedule made up of meta-schedules using less than $N/2$ nodes such that the probability of failure exceeds $\sqrt{\epsilon}$.

□

Theorem 26. *There exists a topology and a wake-up schedule so that any randomized fair protocol takes ϵ-failure-probability time $\Omega(\min\{D \log n \log(1/\epsilon), D + \frac{n \log(1/\epsilon)}{\log n}\})$ to wake up the entire network.*

Proof. We use the network topology defined earlier (see figure 6.1). Now, we have two possibilities. Either the premise of Lemma 23 is false. In this case all the small layers take more than Δ_2 time-slots to clear. We wake up no other nodes in the large layer other than the ones woken up by receiving the activation transmission from one of the nodes in the in-layer. In this case, each and every layer wakes up with ϵ-failure-probability time $\Omega(\log n \log(1/\epsilon))$ and the network wake-up time is $\Omega(D \log n \log(1/\epsilon))$.

If the premise of Lemma 23 is satisfied, then we can use the spontaneous wake-up schedules describes in Lemmas 24 and 25 to construct a long-delay wake-up protocol. Lemma 25 gives us a way of constructing a spontaneous wake-up schedule that bounds the wake-up failure-probability in the first Δ_2 time-slots by $\sqrt{\epsilon}$. Lemma 24 allows us to construct a spontaneous wake-up schedule that bounds the wake-up failure-probability for the time-slots $[(t+1)\ldots(t+T)]$ by $\sqrt{\epsilon}$. Thus the overall success probability is bounded by $(1-\epsilon)$. Thus the wake-up protocol has ϵ-failure-probability time of $\Omega(D + n\log(1/\epsilon)/\log n)$

We have thus shown a lower bound of $\Omega(\min\{D\log n\log(1/\epsilon), D+\frac{n\log(1/\epsilon)}{\log n}\})$ on the ϵ-failure-probability time to wake up the entire network with a fair protocol.

□

The theorem just stated leads to the following corollary.

Corollary 27. *There is a network topology (with diameter $D \in O(n/\log^2 n)$) and wake-up schedule for every fair wake-up protocol such that the high-probability wake-up time has a lower bound of $\Omega(D\log^2 n)$, which matches the best known upper bound.*

References

[ABNLP91] Noga Alon, Amotz Bar-Noy, Nathan Linial, and David Peleg. A lower bound for radio broadcast. *J. Comput. Syst. Sci.*, 43(2):290–298, 1991.

[ACGL02] W. Aiello, F. Chung-Graham, and L. Lu. *Random evolution of massive graphs*, pages 97–122. Kluwer Academic Publishers, Boston/Dordrecht/London, November 2002.

[AR97a] M. J. Appel and R. P. Russo. The maximum vertex degree of a graph on uniform points in $[0, 1]^d$. *Adv. Applied Probability*, 29:567–581, 1997.

[AR97b] M. J. Appel and R. P. Russo. The minimum vertex degree of a graph on uniform points in $[0, 1]^d$. *Adv. Applied Probability*, 29:582–594, 1997.

[BFNO03] Lali Barrière, Pierre Fraigniaud, Lata Narayanan, and Jaroslav Opatrny. Dynamic construction of bluetooth scatternets of fixed degree and low diameter. In *Proc. of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 781–790. Society for Industrial and Applied Mathematics, 2003.

[BLRS03] D. M. Blough, M. Leoncini, G. Resta, and P. Santi. The k-neigh protocol for symmetric topology control in ad hoc networks. In *MobiHoc*, 2003.

[Blu] Bluetooth Special Interest Group. http://www.bluetooth.com.

[BS01] J. Bray and C. Sturman. *Bluetooth: Connect Without cables.* Prentice Hall, 2001.

[BYGI92] R. Bar-Yehuda, O. Goldreich, and A. Itai. On the time-complexity of broadcast in multi-hop radio networks: An exponential gap between determinism and randomization. *Journal of Computer and System Sciences*, 45:104–126, 1992.

[CGK04] Marek Chrobak, Leszek Gąsieniec, and Dariusz Kowalski. The wake-up problem in multi-hop radio networks. In *Proc. of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 992–1000, 2004.

[CK85] I. Chlamtac and S. Kutten. On broadcasting in radio networks–problem analysis and protocol design. *IEEE Trans. Comm.*, 33(2):1240–1246, August 1985.

[CR06] Artur Czumaj and Wojciech Rytter. Broadcasting algorithms in radio networks with unknown topology. *J. Algorithms*, 60(2):115–143, 2006.

[CW91] I. Chlamtac and O. Weinstein. The wave expansion approach to broadcasting in multihop radio networks. *IEEE Trans. Comm.*, 39(3):426–433, March 1991.

[ER59] P. Erdos and A. Renyi. On random graphs–i. *Publicationes Matematicae*, 6:290–297, 1959.

[FCFM05] M. Farach-Colton, R. J. Fernandes, and M. A. Mosteiro. Bootstrapping a hop-optimal network in the weak sensor model. *Lecture Notes in Computer Science*, 3669:827–838, 2005.

[FCFM06] M. Farach-Colton, R. J. Fernandes, and M. A. Mosteiro. Lower bounds for clear transmissions in radio networks. *Lecture Notes in Computer Science*, 3887:447–454, 2006.

[FMPC04] F. Ferraguto, G. Mambrini, A. Panconesi, and C.Petrioli. A new approach to device discovery and scatternet formation in bluetooth networks. In *Proc. of the 18th International Parallel and Distributed Processing Symposium*, 2004.

[FT40] L. Fejes-Tóth. über einen geometrischen satz. *Mathematische Zeitschrift*, 46(1):83–85, 1940.

[GK98] P. Gupta and P. R. Kumar. Critical power for asymptotic connectivity in wireless networks. In *Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W. H. Fleming.*, pages 547–566. Birkhauser, Boston, 1998.

[GKR04] A. Goel, B. Krishnamachari, and S. Rai. Sharp thresholds for monotone properties in random geometric graphs. In *Proc. of STOC*, 2004.

[GM03] Iris Gaber and Yishay Mansour. Centralized broadcast in multihop radio networks. *J. Algorithms*, 46(1):1–20, January 2003.

[GPP00] L. Gasieniec, A. Pelc, and D. Peleg. The wakeup problem in synchronous broadcast systems. In *Proceedings 19th Annual ACM Symposium on Principles of Distributed Computing*, 2000.

[GPP01] L. Gąsieniec, A. Pelc, and D. Peleg. The wakeup problem in synchronous broadcast systems. *SIAM J. Discret. Math.*, 14(2):207–222, April 2001.

[HNO99] T. Hayashi, K. Nakano, and S. Olariu. Randomized initialization protocols packet radio networks. In *IPPS*, 1999.

[JS02] T. Jurdziński and G. Stachowiak. Probabilistic algorithms for the wakeup problem in single-hop radio networks. In *ISAAC*, 2002.

[JS05] T. Jurdziński and G. Stachowiak. Probabilistic algorithms for the wakeup problem in single-hop radio networks. *Theory of Computing Systems*, 38:347–367, 2005.

[KKR$^+$99] J. Kleinberg, S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. The web as a graph: Measurements, models and methods. In *Annual Conference on Computing and Combinatorics*, 1999.

[KM98] Eyal Kushilevitz and Yishay Mansour. An $\Omega(D \log(N/D))$ lower bound for broadcast in radio networks. *SIAM J. Comput.*, 27(3):702–712, 1998.

[KMPS04] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan. End-to-end packet-scheduling in wireless ad-hoc networks. In *Proc. of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2004.

[KP02] Dariusz R. Kowalski and Andrzej Pelc. Deterministic broadcasting time in radio networks of unknown topology. In *Proc. of the 43rd IEEE Annual Symp. on Foundation of Computer Science*, pages 63–72, 2002.

[KP05] Dariusz R. Kowalski and Andrzej Pelc. Broadcasting in undirected ad hoc radio networks. *Distributed Computing*, 18(1):43–57, 2005.

[LP02] D. Liu and M. Prabhakaran. On randomized broadcasting and gossiping in radio networks. *Lecture Notes in Computer Science*, 2387:340–349, 2002.

[LS01] Ching Law and Kai-Yeung Siu. A Bluetooth scatternet formation algorithm. In *Proceedings of the IEEE Symposium on Ad Hoc Wireless Networks*, November 2001.

[MB00] B. Miller and C. Bisdikian. *Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communication.* Prentice Hall, 2000.

[Mit64] D. S. Mitrinović. *Elementary Inequalities.* P. Noordhoff Ltd. - Groningen, 1964.

[Mos07] Miguel Angel Mosteiro. *Upper and Lower Bounds in Radio Networks.* PhD thesis, Rutgers University, Department of Computer Science, Piscataway NJ, 08854, USA, May 2007.

[MP05] S. Muthukrishnan and Gopal Pandurangan. The bin-covering technique for thresholding random geometric graph properties. In *Proc. of ACM-SODA*, 2005.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[MW05] Thomas Moscibroda and Roger Wattenhofer. Maximal independent sets in radio networks. In *PODC*, 2005.

[NO00] K. Nakano and S. Olariu. Energy-efficient initialization protocols for radio networks with no collision detection. In *Proc. of the International Conference on Parallel Processing and Applications*, 2000.

[Pen99] Mathew D. Penrose. On k-connectivity for a geometric random graph. *Random Struct. Algorithms*, 15(2):145–164, 1999.

[Pen03] Mathew Penrose. *Random Geometric Graphs.* Oxford University Press, May 2003.

[Rob75] L. G. Roberts. Aloha packet system with and without slots and capture. *Computer Communication Review*, 5(2):28–42, 1975.

[SBTL01] T. Salonidis, P. Bhagwat, L. Tassiulas, and R. LaMaire. Distributed topology construction of bluetooth personal area networks. In *INFOCOM*, 2001.

[SWLF04] W. Song, Y. Wang, X. Li, and O. Frieder. Localized algorithms for energy efficient topology in wireless ad hoc networks. In *MobiHoc*, 2004.

[TAGH02] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman. A taxonomy of wireless micro-sensor network problems. *Mobile Computing and Communications Review*, 6(2), 2002.

[Wei87] Orly Weinstein. The wave expansion approach to broadcasting in multihop radio networks. Master's thesis, Computer Science Dept., Technion, Haifa, Israel, 1987.

[WTH02] Z. Wang, R. J. Thomas, and Z. Haas. Bluenet - A new scatternet formation algorithm. In *HICSS*, 2002.

[ZBC01] G. V. Zaruba, S. Basagni, and I. Chlamtac. Bluetrees - Scatternet formation to enable Bluetooth-based ad hoc networks. In *Proc. of IEEE ICC*, 2001.

Vita

Rohan Jude Fernandes

2003-2007	Ph. D. in Computer Science, Rutgers University
2000-2002	M. S. in Computer Science, Stony Brook University
1994-1998	B. E. in Computer Engineering, University of Mumbai

2007	Associate, Quantitative Research Group, JPMorgan Chase Bank, New York, NY, USA
2007	Research Assistant, Department of Computer Science, Rutgers University
2004-2006	Teaching Assistant, Department of Computer Science, Rutgers University
2003-2004	Research Assistant, Department of Computer Science, Rutgers University
2002	Summer Intern, Invision Inc. New York, NY, USA
2001-2002	Research Assistant, Department of Computer Science, Stony Brook University
2000-2001	Teaching Assistant, Department of Computer Science, Stony Brook University
1998-2000	Engineer, Systems Integration, CMC Ltd, Mumbai, India

2007	MultiPrimer: A System for Microarray PCR Primer Design. With Steven Skiena in *PCR Primer Design*, volume 402 of Methods in Molecular Biology, Humana Press Inc., Totowa, New Jersey, USA
2006	Lower Bounds for Clear Transmissions in Radio Networks. With Martín Farach-Colton and Miguel A. Mosteiro in *Proceedings of the 7th Latin American Theoretical Informatics Symposium (LATIN)*, volume 3887 of Lecture Notes in Computer Science, pages 447-454, Springer-Verlag, Berlin, Germany

2005 Bootstrapping a Hop-Optimal Network in the Weak Sensor Model. With Martín Farach-Colton and Miguel A. Mosteiro in *Proceedings of the 13th Annual European Symposium on Algorithms (ESA)*, volume 3669 of Lecture Notes in Computer Science, pages 827-838,Springer-Verlag, Berlin, Germany

2002 Microarray Synthesis through Multiple-use PCR Primer Design. With Steven Skiena in *Bioinformatics*, volume 18, supplement 1, pages S128-S135, Oxford University Press, Oxford, UK