Efficient Symmetry Breaking in Multi-Channel Radio Networks

Sebastian Daum
Faculty of Informatics
U. of Lugano, Switzerland
sebastian.daum@usi.ch

Fabian Kuhn
Dept. of Comp. Science
U. of Freiburg, Germany
kuhn@cs.uni-freiburg.de

Calvin Newport
Dept. of Comp. Science
Georgetown University, USA
cnewport@cs.georgetown.edu

Abstract

We investigate the complexity of basic symmetry breaking problems in multihop radio networks with multiple communication channels. We assume a network of synchronous nodes, where each node can be awakened individually in an arbitrary time slot by an adversary. In each time slot, each awake node can transmit or listen (without collision detection) on one of multiple available shared channels. The network topology is assumed to satisfy a natural generalization of the well-known unit disk graph model.

We study the classic *wake-up* problem and a new variant we call *active wake-up*. For the former we prove a lower bound that shows the advantage of multiple channels disappears for any network of more than one hop. For the active version however, we describe an algorithm that outperforms any single channel solution. We then extend this algorithm to compute a constant approximation for the *minimum dominating set (MDS)* problem in the same time bound. Combined, these results for the increasingly relevant multi-channel model show that it is *often* possible to leverage channel diversity to beat classic lower bounds, but not always.

1 Introduction

An increasing number of wireless devices operate in *multi-channel* networks. In these networks, a device is not constrained to use a single fixed communication channel. Instead, it can choose its channel from among the many allocated to its operating band of the radio spectrum. It can also switch this channel as needed. For example, devices using the 802.11 standard have access to around a dozen channels [1], while devices using the Bluetooth standard have access to around 75 channels [5].

	Single Channel	Multi-Channel
Single Hop Wake-Up	$\Theta(\log^2 n)$	$\mathcal{O}(\log^2 n/\mathcal{F} + \log n)$
Multihop Wake-Up	$\Omega(\log^2 n + D\log{(n/D)})$	$\Omega(\log^2 n + D)$
Single Hop Active Wake-Up	$\Theta(\log^2 n)$	$\mathcal{O}(\log^2 n/\mathcal{F} + \log n)$
Multihop Active Wake-Up	$\Omega(\log^2 n + D\log{(n/D)})$	$\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$
MDS	$\Theta(\log^2 n)$	$\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$

Figure 1: **Summary of the results we study in this paper.** The *single channel* column contains the existing results from the wireless algorithm literature, though we strengthen all these results model-wise in this paper. The *multi-channel* column contains our new results described for the first time (the exception are the single hop wake-up results, which derive from our recent work [11]).

In this paper, we prove new upper and lower bounds for symmetry breaking problems in multi-channel networks. Our goal is to use these problems to compare the computational power of this model with the well-studied *single channel* wireless model first studied by Chlamtac and Kutten [7] in the centralized setting and by Bar-Yehuda et al. [3] in the distributed setting. In more detail, we look at the *wake-up* problem [8, 9, 13, 14], a new variant of this problem we call *active wake-up*, and the *minimum dominating set* (MDS) problem (see [15, 19] for a discussion of MDS in single channel radio networks). Our results are summarized in Figure 1.

Result Details & Related Work. We model a synchronous multi-channel radio network using an undirected graph $G = (V, E)$ to describe the communication topology, where G satisfies a natural geographic constraint (cf. Section 2). We assume $\mathcal{F} \geq 1$ communication channels. In each round, each node u chooses a single channel on which to participate. Concurrent broadcasts on the same channel lead to collision and there is no collision detection. For $\mathcal{F} = 1$, the model is the classical multihop radio network model [3, 7].

The wake-up problem assumes that all nodes in a network begin dormant. Each dormant node can be awakened at the start of any round by an adversary. It will also awaken if a single neighbor broadcasts on the same channel. To achieve strong multi-channel lower bounds, we assume that dormant nodes can switch channels from round to round, using an arbitrary randomized strategy. To achieve strong multi-channel upper bounds, our algorithms assign the dormant nodes to a single fixed channel. In the single channel model, the best known lower bound is $\Omega(\log^2 n{+}D\log{(n/D)})$ (a combination of the $\Omega(\log^2 n)$ wake-up bound of [12, 14] and the $\Omega(D\log{(n/D)})$ broadcast bound of [18], which holds by reduction). The best known upper bound is the near-matching $\mathcal{O}(D\log^2 n)$ randomized algorithm of [9], which generalizes the earlier single hop $\mathcal{O}(\log^2 n)$ bound of [14]. In these bounds, as with all bounds presented here, n is the network size and D the network diameter.

In Section 4.1, we prove our main lower bound result: in a multi-channel network with diameter $D > 1$, $\Omega(\log^2 n + D)$ rounds are required to solve wake-up, regardless of the size of $\mathcal{F}$. This bound holds even if we restrict our attention to networks that satisfy the strong *unit disk graph* (UDG) property.[1]

[1]Many radio network papers assume a geographic constraint on the network topology. The UDG property is arguably

In other words, for multihop wake-up, the difficulty of the single channel and multi-channel settings are (essentially) the same. This bound might be surprising in light of our recent algorithm that solves wake-up in $\mathcal{O}(\log^2 n/\mathcal{F} + \log n)$ rounds in a multi-channel network with diameter $D = 1$ [11]. Combined, our new lower bound and the algorithm of [11] establish a gap in power between the single hop and multihop multi-channel models.

The intuition behind our result is as follows: multiple channels help nearby awake nodes efficiently reduce contention, but they do *not* help these nodes, in a multihop setting, determine which node(s) must broadcast to awaken the dormant nodes at the next hop. The core technical idea driving this bound is a reduction from an abstract hitting game that we bound using a powerful combinatorial result proved by Alon et al. in the early 1990s [2].

In Section 4.2, we are able to leverage this same hitting game to prove a stronger version of the $\Omega(\log^2 n)$ bound of wake-up in single hop, single channel networks [12, 14]. The existing bound holds only for a restricted set of algorithms called *uniform*. Our new bound holds for general algorithms. An immediate corollary is that the $\mathcal{O}(\log^2 n)$ time, non-uniform *maximal independent set* (MIS) algorithm of Moscibroda and Wattenhoffer is optimal [19].

On the positive side, we consider the *active* wake-up problem, which is defined the same as the standard problem except now nodes are only activated by the adversary. The goal is to minimize the time between a node being activated and a node receiving or successfully *delivering* a message. This problem is arguably better motivated than the standard definition, as few real wireless devices are configured to allow nodes to monitor a channel and then awaken on receiving a message. The active wake-up problem, by contrast, uses activation to model a device being turned on or entering the region, and bounds the time for every device to break symmetry, not just the first device.

In a single channel network, the $\Omega(\log^2 n)$ lower bound of standard wake-up still applies to active wake-up. In Section 4.3, we describe a new algorithm that solves active wake-up in a multi-channel network in $\mathcal{O}(\log^2 n/\mathcal{F}+\log n \log\log n)$ rounds—beating the single channel lower bound for non-constant $\mathcal{F}$.

We finally turn our attention to the *minimum dominating set* problem. In the single channel setting the $\Omega(\log^2 n)$ lower bound of [12, 14] (and our own stronger version from Section 4.2) applies via reduction. This is matched in UDGs by the $\mathcal{O}(\log^2 n)$-time MIS algorithm of [19].[2] In Section 5, we describe our main upper bound result, a $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$-time multi-channel algorithm that also provides a constant approximation of a MDS (in expectation)—beating the single channel bounds for non-constant $\mathcal{F}$.

The key idea behind our algorithms is to leverage multi-channel diversity to filter the number of awake nodes from a potential of up to n down to $\mathcal{O}(\log^k n)$, for some constant $k \geq 1$—allowing for more efficient subsequent contention management.

---

the strongest of these constraints. More recently, the trend has been toward looser constraints that generalize UDG (e.g., bounded independence or the clique graph constraint assumed by the algorithms in this paper).

[2]In UDGs, an MIS provides a constant-approximation of an MDS.

2 Model & Preliminaries

We model a synchronous multihop radio network with (potentially) multiple communication channels. We use an undirected graph $G = (V, E)$ to represent the communication topology for $n = |V|$ wireless *nodes*, one for each $u \in V$, and use $[\mathcal{F}] := \{1, \ldots, \mathcal{F}\}$, $\mathcal{F} \geq 1$, to describe the available communication channels. For each node $u \in V$ we use $N(u)$ to describe the neighbors of u in G, and let $N^k(u)$ be the set $\{v \in V \colon dist(u, v) \leq k\}$. Nodes in our model are awakened *asynchronously*, in any round, chosen by an adversary. At the beginning of each round, each awake node u selects a channel $f \in [\mathcal{F}]$ on which to participate. It then decides to either *broadcast* a message or *receive*. A node's behavior can be probabilistic and based on its execution history up to this point. If u receives and *exactly one* node from $N(u)$ broadcasts on channel f during this round, then u receives the message, otherwise, it detects silence. If u broadcasts, it can not receive anything. That is, we assume concurrent broadcasts by neighbors on the same channel lead to collision, and there is no collision detection. Notice that u gains no direct knowledge of the behavior on other channels during this round (we assume that u only has time to tune into and receive/broadcast on a single channel per round).

When analyzing algorithms, we will assume a global round counter that starts with the first node waking up. This counter is only used for our analysis and is not known to the nodes. Furthermore, we assume nodes know n (or, a polynomial upper bound on n, which would not change our bounds), but do *not* have advanced knowledge of the network topology. In Sections 4.3 and 5, we describe algorithms in which nodes can be in many states, indicated: $\mathbb{W}$, $\mathbb{A}$, $\mathbb{C}$, $\mathbb{D}$, $\mathbb{L}$ and $\mathbb{E}$. We also use this same notation to indicate the *set* of nodes currently in that state. Finally, for ease of calculation we assume that $\log n$, $\log\log n$ and $\log n / \log\log n$ are all integers.

Graph Restrictions. When studying multihop radio networks it is common to assume some type of geographic constraint on the communication topology. In this paper, we assume a constraint that generalizes many of the constraints typically assumed in the wireless algorithms literature, including unit ball graphs with constant doubling dimension [16], which was shown in [20] to generalize (quasi) UDGs [4, 17].

In more detail, let $\mathcal{R} = \{R_1, R_2, \ldots, R_k\}$ be a partition of the nodes in G into regions such that the sub-graph of G induced by each region R_i is a clique. The corresponding *clique graph* (or *region graph*) is a graph $G_{\mathcal{R}}$ with one node r_i for each $R_i \in \mathcal{R}$, and an edge between r_i and r_j iff $\exists u \in R_i, v \in R_j$ such that u and v are connected in G; we write $R(u)$ for the region containing u. In this paper, we assume that G can be partitioned into cliques $\mathcal{R}$ such that the maximum degree of $G_{\mathcal{R}}$ is upper bounded by some constant parameter Δ.

Probability Preliminaries. In the following, if the probability that event A does not occur is exponentially small in some parameter k—i.e., if $\mathbf{P}(A) = 1 - e^{-ck}$ for some constant $c > 0$—we say that A happens with very high probability w.r.t. k, abbreviated as w.v.h.p.(k). We say that an event happens with high probability w.r.t. a parameter k, abbreviated as w.h.p.(k), if it happens with probability $1 - k^{-c}$, where the constant $c > 0$ can be chosen arbitrarily (possibly at the cost of adapting some other involved constants). If an event happens w.h.p.(n), we just say it happens with high probability (w.h.p.). Finally we define the abbreviation w.c.p. for *with constant probability*.

Our algorithm analysis makes use of the following lemma regarding very high probability, proved in our study of wake-up in single hop multi-channel networks [11]:

Lemma 2.1. *Let there be k bins and n balls with non-negative weights $w_1, \ldots, w_n \leq \frac{1}{4}$, as well as a parameter $q \in (0, 1]$. Assume that $\sum_{i=1}^{n} w_i = c \cdot k/q$ for some constant $c \geq 1$. Each ball is independently selected with probability q and each selected ball is thrown into a uniformly random bin. With probability w.v.h.p.(k), there are at least $k/4$ bins in which the total weight of all balls is between $c/3$ and $2c$.*

3 Problem

In this paper, we study two variants of the *wake-up* problem as well as the *minimum dominating set* problem. In all cases, when we say that an algorithm solves one of these problems in a certain number of rounds, then we assume this holds w.h.p.

Wake-Up: The standard definition of the wake-up problem assumes that in addition to being awakened by the adversary, a dormant node u can be awakened whenever a single neighbor broadcasts. In the multi-channel setting we assume that dormant nodes can monitor an arbitrary channel each round and they awaken if a single neighbor broadcasts on the same channel in the corresponding round. The goal of the standard wake-up problem is to minimize the time between the first and last awakening in the whole network.

Active Wake-Up: The active variant of the wake-up problem, which we are introducing in this paper, eliminates the ability for nodes to be awakened by other nodes. We instead focus on the time needed for an awaken node to *successfully* communicate with one of its neighbors. In standard wake-up dormant nodes are limited to listening only and we show that standard wake-up can be global in nature (it can take time for wake-up calls to propagate over a multihop network). The motivation for active wake-up is to have a similar problem, which allows to get past the limits imposed by the global nature of standard wake-up and still capture the most basic need within solving graph problems: communication. It turns out that active wake-up is inherently local, making it a good candidate for capturing the symmetry breaking required of local graph problems.

More formally, we say an awake node u is *completable* if at least one of its neighbors is also awake. We say a node u *completes* if it delivers a message to a neighbor or receives a message from a neighbor. The goal of active wake-up is to minimize the worst case time between a node becoming completable and subsequently completing.

Minimum Dominating Set: Given a graph $G = (V, E)$, a set $\mathbb{D} \subseteq V$ is a *dominating set* (DS) if every node in $\mathbb{E} := V \setminus \mathbb{D}$ neighbors a node in $\mathbb{D}$. A *minimum dominating set* (MDS) is a dominating set of minimum cardinality over all dominating sets for the graph. We say that a distributed algorithm solves the DS problem in time T if upon waking up, within T rounds, w.h.p., every node (irrevocably) decides to be either in $\mathbb{D}$ or in $\mathbb{E}$ such that at all times, all nodes in $\mathbb{E}$ have a neighbor in $\mathbb{D}$. We say that the algorithm computes a *ρ-approximation MDS* if at all times, the size of $\mathbb{D}$ is within a factor ρ of the size of an MDS of the graph induced by all awake nodes.

4 Wake-Up

In this section we prove bounds on both the standard and active versions of wake-up in multi-channel networks.

4.1 Lower Bound for Standard Wake-Up

In the single channel model, there is a near tight bound of $\Omega(\log^2 n + D \log(n/D))$ on the wake-up problem. We prove here that for $D > 1$ the (almost) same bound holds for multi-channel networks.

Theorem 4.1. *In a multi-channel network of diameter $D = 1$, the wake-up problem can be solved in $\mathcal{O}(\log^2 n/\mathcal{F} + \log n)$, but requires $\Omega(\log^2 n + D)$ rounds for $D > 1$, regardless of the size of $\mathcal{F}$. The lower bound holds even if we restrict our attention to network topologies satisfying the unit disk graph property.*

To better capture what makes a multihop network so difficult (and for proving Theorem 4.1), we reduce the following abstract game to the wake-up problem.

The Set Isolation Game. The set isolation game has a *player* face off against an adversarial *referee*. It is defined with respect to some $n > 1$ and a fixed running time $f(n)$, where f maps to the natural numbers. At the beginning of the game, the referee secretly selects a *target set* $T \subset [n]$. In each round, the player generates a *proposal* $P \subseteq [n]$ and passes it to the referee. If $|P \cap T| = 1$, the player wins and the game terminates, otherwise the referee informs the player it did not hit the set, and the game moves on to the next round without the player learning any additional information about T. If the player gets through $f(n)$ rounds without winning, it loses the game. A *strategy* $\mathcal{S}$ for the game is a randomized algorithm that uses the history of previous plays to probabilistically select the new play. We call a strategy $\mathcal{S}$ an *$f(n)$ round solution to the set isolation game*, iff for every T, w.h.p., it guarantees a win within $f(n)$ rounds.

Lemma 4.2. *Let $\mathcal{A}$ be an algorithm that solves wake-up in $f(n, \mathcal{F})$ rounds, for any $n > 0$ and $\mathcal{F} > 0$, when executed in a multi-channel network with diameter at least 2 and a topology that satisfies the unit disk graph property. It follows that there exists a $g_{\mathcal{F}}(n) = f(n+1, \mathcal{F})$ round solution to the set isolation game.*

Proof. Fix some $\mathcal{F}$. Our set isolation solution simulates $\mathcal{A}$ on a 2-hop network topology of size $n + 1$ along with $\mathcal{F}$ channels, as follows. Let $u_1, \ldots, u_{n+1}$ be the simulated nodes. We arrange u_1 to u_n in a clique C, and connect some subset $C' \subseteq C$ to u_{n+1}. Notice, the resulting network topology satisfies the UDG property. In our simulation, the nodes in C are activated in the first round, and the player proposes, in each round of the game, the values from $[n]$ corresponding to the subset of simulated nodes $\{u_1, \ldots, u_n\}$ that broadcast during the round on the same channel chosen by u_{n+1}. (Notice, the simulator is responsible for simulating all communication and all channels.)

In this simulation, we want C' to correspond to T in the isolation game. Of course, the player simulating $\mathcal{A}$ does not have explicit knowledge of T. To avoid this problem, our simulation always simulates u_{n+1} as not receiving a message. This is valid behavior in every instance *except* for the case where exactly one node in C' broadcasts. This case, however, defines exactly when the player wins the game. If $\mathcal{A}$ isolates a single player in C' in $f(n+1,\mathcal{F})$ rounds (as is required to solve wake-up in this simulated setting), then our set isolation solution solves the set isolation game in the same time. $\square$

To bound wake-up in multihop multi-channel networks, it is now sufficient to bound the set isolation game. Notice that bounds for a *deterministic* variant of the game could be derived from existing literature on selective families [6, 10], but we are interested here in a *randomized* solution. To obtain this bound, we leverage the following useful combinatorial result proved by Alon et al. in the early 1990s [2]:[3]

Lemma 4.3 (Adapted from [2])**.** *Fix some $n \in \mathbb{N}$. Let $\mathcal{H}$ and $\mathcal{J}$ be families of nonempty subsets of $[n]$. We say that $\mathcal{H}$* hits *$\mathcal{J}$ iff for every $J \in \mathcal{J}$, there is an $H \in \mathcal{H}$ such that $|J \cap H| = 1$. There exists a constant $c > 0$ and family $\mathcal{J}$, with $|\mathcal{J}|$ polynomial in n, such that for every family $\mathcal{H}$ that hits $\mathcal{J}$, $|\mathcal{H}| \geq c\log^2 n$.*

The above lemma applies to the case where there are *multiple* sets to hit, but the sets are *known* in advance. Here we translate the results to the case where there is a *single* set to hit, but the set is *unknown* in advance, and a result must hold with high probability (i.e., the exact setup of the set isolation game).

Lemma 4.4. *Any set isolation game strategy $\mathcal{S}$ needs $f(N) = \Omega(\log^2 N)$ rounds.*

Proof. Fix some $n > 0$. Let $N = n^k$, where $k > 1$ is a constant we fix later. Consider an execution of $\mathcal{S}$ with respect to N. Let $\mathcal{H}_S = (\mathcal{H}_S(r))_{1\leq r\leq f(N)}$ be a sequence of subsets of $[n]$ such that $\mathcal{H}_S(r)$ describes the values from $[n] \subset [N]$ included in the proposal $P \subset [N]$ of $\mathcal{S}$ in round r of the execution under consideration. Let $\mathcal{J}$ be the difficult family identified by Lemma 4.3, defined with respect to n.

Assume for contradiction that $f(N) = o(\log^2 N)$, i.e., $f(N) < c\log^2 n$. But then, as a direct corollary of Lemma 4.3, there is at least one subset $J \in \mathcal{J}$ that is not hit by $\mathcal{H}_S$. With this in mind, we define the following referee strategy for the set isolation game. Choose the target set $T \subset [n] \subset [N]$ from $\mathcal{J}$ uniformly at random. The execution of $\mathcal{S}$ fails to hit T with probability at least $1/|\mathcal{J}|$. By Lemma 4.3, $|\mathcal{J}|$ is polynomial in n.

[3] Our first idea was to try to adapt the strategy used in the existing $\Omega(\log^2 n)$ bound on wake-up in single channel radio networks [12, 14]. This strategy, however, assumes a strong *uniformity* condition among the nodes, which makes sense in a single channel world—where no nodes can communicate until the problem is solved—but is too restrictive in our multi-channel world, where nodes can coordinate on the non-wake-up channels, and therefore break uniformity in their behavior.

Therefore, we can choose our constant k such that $1/|\mathcal{J}| > 1/n^k$.[4] It follows that the probability of failure to win the game in $f(N)$ rounds is at least $1/|\mathcal{J}| > 1/n^k = 1/N$, a contradiction to the definition of a set isolation game strategy. □

The $D > 1$ term of Theorem 4.1 now follows from Lemmas 4.2 and 4.4, plus a straightforward argument that $\Omega(D)$ rounds are needed to propagate information D hops, while the $D = 1$ term comes from [11].

4.2 A Stronger Single Channel Wake-Up Bound

Before continuing with our multi-channel results, we make a brief detour. By leveraging our set isolation game and Lemma 4.4, we can prove a stronger version of the classic $\Omega(\log^2 n)$ lower bound on wake-up in a single hop single channel network [12, 14]. This existing bound holds only for *uniform* algorithms (i.e., nodes use a uniform fixed broadcast probability in each round). The version proved here holds for *general* randomized algorithms (i.e., each node's probabilistic choices can depend on its IDs and its execution history).

The argument is a variation on the simulation strategy used in Lemma 4.2.

Theorem 4.5. *Let $\mathcal{A}$ be a general randomized algorithm that solves wake-up in $f(n)$ rounds in a single hop single channel network. It follows that $f(n) = \Omega(\log^2 n)$.*

Proof. Here we follow the same general strategy exhibited by Lemma 4.2: showing how to use $\mathcal{A}$ to solve set isolation. Though the idea of this reduction is the same, we must alter the argument to deal with the fact that we are now in a single hop network.

In more detail, simulate all n wake-up nodes as awake and not receiving messages. In each round, propose the set of simulated wake-up nodes that broadcast in that round. Notice, if we knew T, the obvious thing to do would be to simulate *only* the nodes corresponding to T, because by the definition of the wake-up problem, there would be a round in which exactly one of those nodes broadcasts (as required to solve wake-up). We are instead simulating all nodes. However, this does not cause a problem because each node's simulation looks the same regardless of the other nodes being simulated—in the single channel wake-up problem, nodes do not communicate with each other before the problem is solved. Consequently, for the nodes corresponding to T, this simulation is indistinguishable from one in which only these nodes were being simulated. Therefore, in some round $r \leq f(|T|) \leq f(n)$, exactly one of these nodes from T has to broadcast. The resulting proposal set will contain only one element from T (potentially in addition to some other elements from $[n] \setminus T$): solving set isolation. □

The wake-up problem reduces to the MIS problem, so a bound on wake-up applies to MIS. The best known MIS algorithm for single channel radio networks is the $\mathcal{O}(\log^2 n)$-time algorithm of Moscibroda and Wattenhoffer [19]. Because their algorithm is *non-uniform*, we cannot reduce from the uniform wake-up bounds of [12, 14]. Using Theorem 4.5, however, the reduction now holds, proving the conjecture that the result of [19] is optimal.

[4] In the proof construction used in [2], the size $\mathcal{J}$ is bounded around n^8.

4.3 Upper Bound for Active Wake-Up

In this section we present a $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$ time solution to the active wake-up problem in a multi-channel network. For non-constant $\mathcal{F}$ this beats the $\Omega(\log^2 n)$ lower bound for this problem in the single channel setting.

Algorithm Description. Our algorithm, Algorithm 1, requires that $\mathcal{F} \geq 9$ and that $\mathcal{F} = \mathcal{O}(\log n)$ (if $\mathcal{F}$ is larger we can simply restrict ourselves to use a subset of the channels). It uses the first channel as a *competition* channel, and the remaining $\mathrm{F} = \mathcal{F} - 1$ channels for nodes in an *active* state (denoted $\mathbb{A}$). Nodes begin the algorithm in state $\mathbb{A}$, during which they choose active state channels with uniform probability and broadcast with a probability that increases exponentially from $1/n$ to $1/4$, spending only $\mathcal{O}(\log n/\mathrm{F})$ rounds at each probability. During this state, if a node receives a message it is *eliminated* ($\mathbb{E}$), at which point it receives on the competition channel for the remainder of the execution. A node that survives the active state moves on to the competition state ($\mathbb{C}$) during which it broadcasts on the competition channel with probabilities that exponentially increase from $1/\log^2 n$ to $1/2$, spending $\Theta(\log n)$ rounds at each probability. As before, receiving a message eliminates a node ($\mathbb{E}$). Finally, a node that survives the competition state advances to the leader state ($\mathbb{L}$) where it broadcasts on the competition channel with probability $1/2$ in each round.

We analyze the algorithm below.

Theorem 4.6. *Algorithm 1 solves the active wake-up problem in multi-channel networks within* $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$ *rounds.*

As detailed in Section 2, we assume the graph can be partitioned into cliques with certain useful properties. In this proof we refer to those cliques as *regions*, which we label $R_1, R_2, \ldots, R_k$, where $k \leq n$. We also make use of the "very high probability" notation, and corresponding Lemma 2.1, also presented in Section 2.

For a given round and node u, let $p(u)$ be the probability that u broadcasts in that round. Similarly, for a given round and region R we define the probability masses $P_{\mathbb{A}}(R) := \sum_{u \in \mathbb{A} \cap R} p(u)$ and $P_{\mathbb{C}}(R) := \sum_{u \in (\mathbb{C} \cup \mathbb{L}) \cap R} p(u)$. When it is clear which region is meant, we sometimes omit the (R) in this notation. We begin by bounding $P_{\mathbb{A}}$ for every region R. The following lemma is a generalization of Lemma 4.8 from [11], modified to now handle a multihop network.

Lemma 4.7. *W.h.p., for every round and region:* $P_{\mathbb{A}} = \mathcal{O}(\mathrm{F}) = \mathcal{O}(\mathcal{F})$.

Proof. Assume our lemma does not hold. It would follow that at least one region would have its probability mass exceed our $\mathcal{O}(\mathrm{F})$ threshold with probability larger than $1/n^d$, for appropriate constant d. Fix $4\mathrm{F} + 1 = \mathcal{O}(\mathrm{F})$ as our specific threshold. Let R be the first region to exceed this threshold and let $\alpha_{\mathbb{A}} = c_d \log n/\mathrm{F}$ be the phase length used in Algorithm 1 during the active state, where c_d is a constant we can choose arbitrarily. Notice, in any interval of $\alpha_{\mathbb{A}}$ rounds, $P_{\mathbb{A}}$ can only increase by a multiplicative factor of 2 and an additive factor of 1 (the latter contributed by newly activated nodes, which start with a broadcast probability of $1/n$). Hence, if $P_{\mathbb{A}}$ exceeds $4\mathrm{F} + 1$, it must have been between $2\mathrm{F}$ and $4\mathrm{F} + 1$ for $\alpha_{\mathbb{A}}$ consecutive rounds before this point.

Algorithm 1: Active Wake-Up Algorithm

State description: $\mathbb{A}$ – active, $\mathbb{C}$ – competitor, $\mathbb{L}$ – leader, $\mathbb{E}$ – eliminated

```
begin
    α_A = Θ(log n/F);    α_C = Θ(log n)
    set count := 0; phase := 0; state := A
    while state ≠ E do
        count := count + 1
        uniformly at random pick: k ∈ {2, ..., F};    q ∈ [0, 1)
        switch state do
            case A
                if q > 2^phase / n then listen on k
                else send on k

                if count > α_A then phase := phase + 1; count := 0

                if phase > log (n/4) then phase := 0; state := C

            case C
                if q > 2^phase / log^2 n then listen on 1
                else send on 1

                if count > α_C then phase := phase + 1; count := 0

                if phase > log ((log^2 n)/2) then state := L

            case L
                if q ≥ 1/2 then listen on 1
                else send on 1

    Listen on 1 perpetually
Upon receiving a message:
 if state ≠ L then state := E
```

Consider such a round r, during which $P_{\mathbb{A}} \in [2\mathrm{F}, 4\mathrm{F}+1]$. We can apply Lemma 2.1 to show that, w.v.h.p.(F), in round r there are at least $\mathrm{F}/4$ channels on which the total broadcast probability of all nodes in R is between $2/3$ and 10 (we call such channels *balanced*). The parameters for the lemma are $k = \mathrm{F}$, $q = 1$, $w_v = p(v)$ and consequently, $c \in [2, 5]$.

Now consider one such channel with a total broadcast probability between $2/3$ and 10. Because the total broadcast probability is at least $2/3$, and because an individual node cannot broadcast with probability more than $1/4$, we know we are dealing with at least 2 nodes on this channel in this region.

At this point, we must consider interference from neighboring regions. Because we assumed that R is the first region to exceed our broadcast probability threshold, every region R_j that neighbors R must have $P_{\mathbb{A}}(R_j) \leq 4\mathrm{F} + 1 \leq 5\mathrm{F}$ during this round. Denote with $\mathcal{R}$ the union of all those regions R_j neighboring R excluding R itself. Then within $\mathcal{R}$ denote with $\tilde{P}_{\mathbb{A}}$ the total probability mass, thus $\tilde{P}_{\mathbb{A}} \leq 5\mathrm{F}\Delta$. We denote with $\tilde{P}_{\mathbb{A}}^f$ the broadcasting probability of all nodes within $\mathcal{R}$ that decide to operate on channel f this round. Call a channel *crowded* if $\tilde{P}_{\mathbb{A}}^f \geq 40\Delta$, and *non-crowded* otherwise. Obviously, at most $\mathrm{F}/8$ of all channels can be crowded.

Since at least $1/4$ of all channels are balanced and at most $1/8$ are crowded, we have that at least $1/8$ of all channels are balanced, but *not* crowded (it is here that we see the justification for the technical requirement that $\mathcal{F} = \mathrm{F} + 1 \geq 9$). For a given balanced and non-crowded channel f, the probability $\tilde{p}$ that *no* node from $\mathcal{R}$ broadcasts on f is bounded:

$$\tilde{p} \geq \prod_{v \in \mathbb{A} \cap \mathcal{R}} (1 - p(v)) > (1/4)^{40\Delta} = \Omega(1).$$

Pulling together the pieces, we have shown that w.v.h.p.(F), a constant fraction of our channels are balanced and non-crowded, i.e., that the probability of a single node from R broadcasting without interference from neighboring regions, is constant. Whenever such a broadcast succeeds, all other nodes in R receive the message and are eliminated: decreasing $P_{\mathbb{A}}$. A balanced channel houses a probability mass of at least $2/3$, of which a successful sender can occupy at most $1/4$, i.e., at least a probability mass of $1/3$ is eliminated in the case that a successful transmission happens on that channel. Since w.v.h.p.(F) at least $\mathrm{F}/8$ channels are balanced and non crowded, a constant fraction of the probability mass is eliminated if on a constant fraction of these a successful transmission happens. As shown above we have a constant chance of this event on each of these. A standard Chernoff argument finally implies that a constant fraction of the probability mass is eliminated w.v.h.p.(F) in *each* round.

Assume that for constant $\hat{\gamma} > 0$ and $\hat{s} > 1$, with probability $p := 1 - e^{-\hat{\gamma}\mathrm{F}}$, a $1/\hat{s}$-fraction of the total probability mass is eliminated and let us call a round successful if an $1/\hat{s}$-fraction of the total probability mass is eliminated. Note that we can not influence the constant $\hat{\gamma}$—it is given by the system. In order to get from some $P_{\mathbb{A}} < 4\mathrm{F} + 1$ to $P_{\mathbb{A}} < 2\mathrm{F}$ w.h.p. (to get a contradiction), we need $\log_{\hat{s}} 3 = \Theta(1)$ successful rounds in a time span of $\alpha_{\mathbb{A}} = \Theta(\log n/\mathrm{F})$ rounds with sufficiently high probability, say $1 - n^{-d-2}$.

If $1 - p = e^{-\hat{\gamma}\mathrm{F}}$ is more than e^{-3}, then F is less than $3/\hat{\gamma} = \mathcal{O}(1)$, i.e., constant, having a phase lasting $\Omega(\log n)$ rounds, each being successful with a constant probability. A standard Chernoff argument gives us that enough of them are successful w.p. $1 - n^{-d-2}$ if we choose c_d large enough.

If $1-p = e^{-\hat{\gamma}\mathrm{F}}$ is less than e^{-3}, then we apply Chernoff again to show that less than a constant fraction of $\alpha_{\mathbb{A}}$ rounds are *not* successful. By choosing $\delta + 1 := e^{\hat{\gamma}\mathrm{F}-1} \geq e^2$ and letting X count the number of unsuccessful rounds we get ($\mu = e^{-\hat{\gamma}\mathrm{F}}\alpha_{\mathbb{A}}$):

$$\begin{aligned}\mathbf{P}(X \geq \alpha_{\mathbb{A}}/e) &= \mathbf{P}(X \geq (1+\delta)\mu) \leq e^{-\mu \overbrace{((\delta+1)\ln(\delta+1)-\delta)}^{>0}} \\ &\leq e^{-\alpha_{\mathbb{A}}(e^{-1}(\hat{\gamma}\mathrm{F}-1)-e^{-1}+e^{-\hat{\gamma}\mathrm{F}})} \overset{\alpha_{\mathbb{A}} \geq c_d \frac{\log n}{\mathrm{F}}}{\leq} n^{-\frac{c_d}{\mathrm{F}}\frac{\hat{\gamma}\mathrm{F}-2}{e}} \leq n^{-c_d\frac{\hat{\gamma}}{3e}}\end{aligned}$$

Thus, for sufficiently large c_d we have that more than $\alpha_{\mathbb{A}}/e \geq \log_{\hat{s}} 3$ rounds are successful with probability $1 - n^{-d-2}$. We then apply a union bound over all regions ($< n$) and relevant rounds ($< n$) and get that the probability of any region ever to exceed $4\mathrm{F}+1$ for its $P_{\mathbb{A}}$ is less than $1/n^d$, contradicting our assumption. □

Lemma 4.8. *W.h.p., for every round and region R: $P_{\mathbb{C}} = \mathcal{O}(1)$.*

Proof. Notice, when a node transitions from $\mathbb{A}$ to $\mathbb{C}$, it must have been contributing a constant probability (i.e., $1/4$) to $P_{\mathbb{A}}$ in its region before the transition. Combining this observation with Lemma 4.7, it follows: (*) w.h.p., for every round and region, no more than $\mathcal{O}(\mathrm{F})$ nodes enter the competition phase in this region during this round.

Armed with observation (*), we turn our attention to our lemma statement. Assume our lemma did not hold. It would follow that a least one region would have its probability mass exceed our constant threshold with sufficiently high probability. As before let R be this first region and we will fix our threshold at some constant c (which we fix later). Let $\alpha_{\mathbb{C}}$ be the phase length in the competitor phase (see Figure 1). Assume $P_{\mathbb{C}} > c$ for the first time at some round r. In the $\alpha_{\mathbb{C}}$ rounds preceding r, $P_{\mathbb{C}}$ could have only increased by a multiplicative factor of 2 and a constant additive factor. The additive factor comes from nodes newly entering $\mathbb{C}$. To see why this is bounded as a constant, notice (*) tells us that only around $\mathcal{O}(\log n)$ nodes can join per round during these $\alpha_{\mathbb{C}}$ rounds, leading to $\mathcal{O}(\log^2 n)$ total new nodes. Since these nodes start with broadcast probability $1/\log^2 n$, they add only a constant mass.

At this point, we can use a simpler version of the argument from Lemma 4.7. During each of these $\Theta(\log n)$ rounds there is a constant probability of a single broadcaster on the competitor channel in R. Because R is the first region to exceed our threshold, we can also argue that the probability of no interference from neighboring regions is also constant. Combined, there is a constant chance that a single node in R will broadcast on the competitor channel and eliminate all other nodes in $R \cap \mathbb{C}$: reducing $P_{\mathbb{C}}$ to no more than $1/2$, with sufficiently high probability. If we choose our threshold c to be sufficiently high, it will be impossible for region i to add enough mass to $P_{\mathbb{C}}$ in time—creating a contradiction. A union bound can be used to apply the result to all regions and relevant rounds. □

Proof of Theorem 4.6. In the following, let $T = \mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$ be the time required to get from waking up to $\mathbb{L}$. Consider a node u that wakes up in region R in round r. We consider two cases. In the first case, u is eliminated before it reaches $\mathbb{L}$. Therefore, u received a message in T rounds—satisfying the theorem statement.

In the second case, u reaches $\mathbb{L}$ without receiving a message. At this point T rounds have elapsed. If u is not already completable, wait until it next becomes so. Let v be the first node to make u completable. Within T rounds from waking up, v is either eliminated or in $\mathbb{C}$. In either case, it will remain on the competition channel for the remainder of the execution, where it has a chance of receiving a message from $u \in \mathbb{L}$, which would complete u. In each such round, u broadcasts with constant probability. We apply Lemma 4.8 to establish that the broadcast probability sum of interfering nodes (both in R and neighboring regions) is constant. Combined, u has a constant probability of delivering a message to v. For sufficiently large constant c, $c \log n$ additional rounds are sufficient for u to complete with high probability. $\square$

5 Minimum Dominating Set

In this section, we present an algorithm that computes a constant-factor (in expectation) approximation for the MDS problem in time $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$. For $\mathcal{F} = \omega(1)$ this outperforms the fastest known algorithm to solve MDS in the single channel model.
For $\mathcal{F} = \mathcal{O}(\log n/\log\log n)$ the speed-up is in the order of $\Theta(\mathcal{F})$.

Algorithm Description. Algorithm 2 builds on the ideas of the active wake-up algorithm of the previous section as follows. For simplicity, we assume that $\mathcal{F} = \mathcal{O}(\log n)$, as more frequencies are not exploited. For an easier handling of the analysis we partition and rename the $\mathcal{F}$ available channels $[\mathcal{F}]$ into $\{\mathcal{A}_1, \ldots, \mathcal{A}_{\mathrm{F}}\}\dot\cup\{\mathcal{D}_1, \ldots, \mathcal{D}_{n_\mathcal{D}}\}\dot\cup\{\mathcal{C}\}$, such that $\mathrm{F} = \Theta(\mathcal{F})$ and $n_\mathcal{D} = \mathcal{O}(\min\{\log\log n, \mathcal{F}\})$.

After being woken up, a node u starts in the *waiting state* $\mathbb{W}$, in which it listens uniformly at random on channels $\mathcal{D}_1, \ldots, \mathcal{D}_{n_\mathcal{D}}$. Its goal is to hear from a potentially already existing neighboring dominator before it moves on to the *active state* $\mathbb{A}$. Once in $\mathbb{A}$ node u starts broadcasting on the channels $\{\mathcal{A}_1, \ldots, \mathcal{A}_{\mathrm{F}}\}$ with probability $1/n$ in each round. It acts in phases and at the beginning of each phase it doubles its broadcasting probability until it reaches probability $1/4$. As in the wake-up protocol, u chooses its channel uniformly at random, allowing us to reduce the length of each phase from the usual $\Theta(\log n)$ in a single channel setting to $\Theta(\log n/\mathrm{F})$, while still keeping the broadcasting probability mass in each region bounded w.h.p.

Unlike the wake-up algorithm, a node is not done when it receives a message in state $\mathbb{A}$. Instead it restarts with state $\mathbb{W}$. If a node manages to broadcast in state $\mathbb{A}$, it immediately moves on to the *candidate state* $\mathbb{C}$. Because the probability mass $P_\mathbb{A}$ in $\mathbb{A}$ is bounded in every region, the number of nodes moving to the candidate state can also be bounded by $\mathcal{O}(\mathrm{polylog}\, n)$.

State $\mathbb{C}$ starts with a long *sleeping phase* (phase 0) in which nodes act as in state $\mathbb{W}$, i.e., they listen on channels $\mathcal{D}_1, \ldots, \mathcal{D}_{n_\mathcal{D}}$: to find out about potential dominators created while they were in state $\mathbb{A}$. If a node u does not receive the message of a neighboring dominator in that time, it moves on to yet another long phase called *readying phase* (phase 1), in which nodes listen exclusively on $\mathcal{C}$. If again no message is received during this phase, u becomes a *competing candidate* by advancing to much shorter phases $2, 3, \ldots$, during which u tries to become a dominator by broadcasting on channel $\mathcal{C}$. Unlike in state $\mathbb{A}$, u can start with broadcasting probability $1/\log^2 n$, without risk of too much congestion. This allows us to reduce the total number of phases to $\mathcal{O}(\log\log n)$. A candidate that manages to broadcast, immediately moves on to the *dominating state* $\mathbb{D}$, while candidates receiving a message from another candidate move to the *eliminated state* $\mathbb{E}$, because they know that the sender of that message is now a dominator. Assuming that $\mathcal{F} = \Omega(\log\log n)$, dominators run the following protocol. In each round, they choose a channel $\mathcal{D}_i$ uniformly at random and broadcast on it with probability 2^{-i}. We can show that the number of dominators in each node v's neighborhood is at most poly-logarithmic in n. Then, as soon as v has at least one dominator in its neighborhood, there is always a channel $\mathcal{D}_\lambda$ on which v can receive a message from a dominator with constant probability. On average v will choose the right channel within $\mathcal{O}(\log\log n)$ rounds, so $\mathcal{O}(\log n \log\log n)$ rounds are enough to ensure high probability. In the case $\mathcal{F} = o(\log\log n)$ a constant number of channels with appropriate broadcasting probabilities suffice to make a dominator heard within $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$ rounds.

We analyze the algorithm below.

Theorem 5.1. *Algorithm 2 computes a ρ-approximation for the MDS problem in time $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$, where ρ is constant in expectation, i.e., $\mathbf{E}[\rho] = \mathcal{O}(1)$.*

Let us start out with some definitions and notations. We define $P_\mathbb{A}$ and $P_\mathbb{C}$ analogously to Section 4.3. Further, we call a node *decided* if it belongs to $\mathbb{D}$ or $\mathbb{E}$.
Finally, we define $T' := \alpha_\mathbb{W} + 2\alpha_{sleep} + (\alpha_\mathbb{A} + \alpha_\mathbb{C} + 2)\log n$ and $T := 3(\Delta^2+1)T'$.

Lemma 5.2. *W.h.p., at all times and for every region R, the probability mass $P_\mathbb{A}$ in region R is bounded by $\mathcal{O}(\mathrm{F})$.*

Proof. The proof is identical to the proof of Lemma 4.7 for the wake-up algorithm. □

Lemma 5.3. *W.h.p., at most $\mathcal{O}(\mathrm{F} + \log n) = \mathcal{O}(\log n)$ nodes switch to the candidate phase in any region R in any round r.*

Proof. By Lemma 5.2, w.h.p., the probability mass $P_\mathbb{A}$ is always bounded by $c\mathrm{F}$ for some known constant c. For each node v in a fixed region R, define X_v as the Bernoulli random variable that indicates whether v moves to the candidate phase in round r and let $X := \sum_{v\in R} X_v$ and $\mu := \mathbf{E}[X] \le P_\mathbb{A} \le c\mathrm{F}$. For an arbitrary $d > 0$ let $\delta := \mu^{-1}(e^2 c\mathrm{F} + d\log n) - 1$. Then, applying a standard Chernoff bound, we get

$$\mathbf{P}(X \ge (1+\delta)\mu) = \mathbf{P}(X \ge (e^2 c\mathrm{F} + d\log n)) \le e^{-\mu(\delta+1)} \le e^{-d\log n} = n^{-d}. \qquad \square$$

Lemma 5.4. *W.h.p., at all times and for every region R, the probability mass $P_\mathbb{C}$ in region R is bounded by $\mathcal{O}(1)$.*

Proof. By Lemma 5.3, in no round more than $\mathcal{O}(\log n)$ nodes move from state $\mathbb{A}$ to state $\mathbb{C}$. Thus at most $\mathcal{O}(\log^2 n)$ nodes do so within the length $\alpha_\mathbb{C}$ of one phase – other than phase 0 and 1 – of state $\mathbb{C}$. The claim then follows analogously to the proof of Lemma 4.8. □

The purpose of the sleeping phases in state $\mathbb{W}$ and at the beginning of state $\mathbb{C}$ is for nodes to detect if they have a dominator in their neighborhood and thus getting eliminated before going to $\mathbb{A}$ or to start competing in $\mathbb{C}$. The following lemma shows that both sleep phases do their job and that a full sleep phase is enough for a dominator to eliminate a neighbor in state $\mathbb{W}$ or phase 0 of state $\mathbb{C}$.

Lemma 5.5. *Assume that a node u starts with state $\mathbb{W}$ or phase 0 of state $\mathbb{C}$ in round r and there is already a dominator in $N(u)$. Further, assume at all times $t \in [r, r+\alpha_{\mathbb{W}}] = [r, r+\alpha_{sleep}]$ that $k_t := |\mathbb{D} \cap (N^1(u))| = \mathcal{O}(\log^3 n/\mathcal{F} + \log^2 n \log\log n)$. Then, w.h.p., u switches to state $\mathbb{E}$ by round $r+\alpha_{\mathbb{W}} = r+\alpha_{sleep}$.*

Proof. We have to consider two cases. For the first assume that $n_{\mathcal{D}} = c\log\log n$ for a sufficiently large constant c (if we have more channels available, we do not make use of them). For a particular channel $\mathcal{D}_i$ and node u define the broadcast probability sum $P^i_{\mathbb{D}}(u) := \sum_{w \in \mathbb{D}\cap N^1(u)} 2^{-i} = k_t 2^{-i}$. Since k_t is bounded, there exists a 'favored' channel $\mathcal{D}_{\lambda_t}$ for which $P^{\lambda_t}_{\mathbb{D}}(u) = k_t 2^{-\lambda_t} \in [1/2, 1)$. If u listens on channel $\mathcal{D}_{\lambda_t}$ at time t, then w.c.p. exactly one dominator in $N^1(u)$ broadcasts on that channel (denote this event with A):

$$\mathbf{P}(A) = k_t 2^{-\lambda_t} \cdot (1-2^{-\lambda_t})^{k_t - 1} \geq \frac{1}{2} \cdot \left((1-2^{-\lambda_t})^{2^{\lambda_t}}\right)^{2^{-\lambda_t} k_t} \geq \frac{1}{2}\cdot\frac{1}{4} = \Omega(1)$$

As long as the number of neighboring dominators is bounded, in each round u listens on that round's favored channel $\mathcal{D}_{\lambda_t}$ with probability $1/c\log\log n$ for some constant c, dictated by the number $n_{\mathcal{D}}$ of channels available. Thus, for a suitably chosen α, the probability that u manages to receive a message within $\alpha \log\log n$ rounds is more than $1/2$. For a given constant $d > 0$ let $\alpha_{\mathbb{W}} = \alpha_{sleep} \geq d\alpha \log n \log\log n$. Then node u hears one of its neighboring dominators with probability $1 - n^{-d}$ before it ends state $\mathbb{W}$ or phase 0 of state $\mathbb{C}$ at time $r + \alpha_{\mathbb{W}} = r + \alpha_{sleep}$.

For the second case assume that $n_{\mathcal{D}} < c\log\log n$. In that case the algorithm makes use of only 4 channels as it sets $n_{\mathcal{D}}$ to 4. Also we have that $\alpha_{\mathbb{W}} = \alpha_{sleep} = \Omega(\log^2 n/\mathcal{F})$. In any round the number of neighboring dominators is either in $\mathcal{O}(\log n/\mathcal{F})$, in $\mathcal{O}(\log^2 n/\mathcal{F}^2) \cap \omega(\log n/\mathcal{F})$, in $\mathcal{O}(\log^3 n/\mathcal{F}^3) \cap \omega(\log^2 n/\mathcal{F}^2)$ or in $\mathcal{O}(\log^4 n/\mathcal{F}^4) \cap \omega(\log^3 n/\mathcal{F}^3) = \mathcal{O}(k_t)$. In either case u has a $1/4$ probability to hit a channel on which the probability mass is in $\Omega(\mathcal{F}/\log n) \cap \mathcal{O}(1)$, therefore having a probability of $\Omega(\mathcal{F}/\log n)$ to receive the message of a neighboring dominator. I.e., for a well chosen α, within $\alpha \log n/\mathcal{F}$ rounds this accumulates to at least probability $1/2$. Thus, within time $\alpha_{\mathbb{W}} = \alpha_{sleep} \geq d\alpha \log^2 n/\mathcal{F}$ it hears a neighboring dominator with probability at least $1 - n^{-d}$. $\square$

The following lemma shows that the number of dominators in each region is bounded and that as soon as there is a dominator in a region, the region also becomes decided within bounded time.

Lemma 5.6. *The lemma statement is in three parts:*

(a) W.h.p., in every region R and round r: only $\mathcal{O}(\log n)$ nodes move to state $\mathbb{D}$.

(b) W.h.p., if there is a node u in state $\mathbb{D}$, then all nodes that are already awake in $N^1(u) \supset R(u)$ are decided within time T'.

(c) W.h.p., in every region R: $|\mathbb{D}| = \mathcal{O}(\log^3 n/\mathcal{F} + \log^2 n \log\log n)$.

Proof. For *(a)* we refer to the proof of Lemma 5.3. The bound is obtained analogously by applying Lemma 5.4.

For *(b)* consider some node $v \in N(u)$ and let r be the round in which it neighbors a dominator for the first time. Since the number of regions neighboring v is bounded by Δ, in the time interval $I = [r, r+T']$ at most $\mathcal{O}(\Delta T' \log n) = \mathcal{O}(\log^3 n/\mathcal{F} + \log^2 n \log\log n)$ nodes can move to state $\mathbb{D}$ in $N(v)$, according to *(a)*. If v is in state $\mathbb{A}$, then, w.h.p., after at most $(\alpha_{\mathbb{A}}+1)\log n$ rounds it either restarts in state $\mathbb{W}$ or it starts with phase 0 of state $\mathbb{C}$. We apply Lemma 5.5 to get that v is decided within $T' - \alpha_{\mathbb{W}}$ rounds. Let v be in state $\mathbb{W}$. It takes at most $\alpha_{\mathbb{W}}$ rounds for v to move on to state $\mathbb{A}$ and we can apply our previous result to get that v is decided within T' rounds. If v is in state $\mathbb{C}$, then, because it can not be set back to state $\mathbb{W}$ or $\mathbb{A}$, it is decided within $2\alpha_{\mathit{sleep}} + \alpha_{\mathbb{C}}(\log\log n + 1) \leq T'$ rounds, finishing our claim.

For *(c)*, consider the first round r in which a node becomes a dominator in region R. Then by the previous part, by time $r + T'$, all nodes of region R that are awake in round r become decided. Further, by Lemma 5.5, all nodes that wake up after round r switch to state $\mathbb{E}$ and do not become dominators. Finally, by *(a)*, during the T' rounds in which region R can be populated by dominators, at most $\mathcal{O}(T' \log n) = \mathcal{O}(\log^3/\mathcal{F} + \log^2 n \log\log n)$ dominators are created. $\square$

Lemma 5.7. *W.h.p., each node u is decided within $T = \mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$ rounds after waking up.*

Proof. A region R is called *decided* in round r, if no node in R is in state $\mathbb{A}$ or $\mathbb{C}$ in *any* round $r' \geq r$ (i.e., all nodes are in $\mathbb{D}$, $\mathbb{E}$, $\mathbb{W}$ or still dormant). Hence, in particular after a region R becomes decided, no more dominators will be created in R (since a node must go through states $\mathbb{A}$ and $\mathbb{C}$ before becoming a dominator). Note that a region is decided if all its nodes are decided, but not necessarily the other way round.

Now, according to Lemma 5.6, if in some region R a dominator v gets created, then the remaining awake nodes in R are decided within T' rounds. Further, by Lemma 5.5, if nodes in R are activated after v becomes a dominator, then w.h.p. they are decided while being in state $\mathbb{W}$ and therefore never enter state $\mathbb{A}$. Thus, within T' rounds no newly awoken node in R will ever be in state $\mathbb{A}$ or $\mathbb{C}$. But that means that any region R with a dominator in it becomes decided at most T' rounds after its first dominator is created.

We now focus on some node u. Note that once u leaves states $\mathbb{W}$ and $\mathbb{A}$ behind, it gets decided within T' rounds. We thus only have to analyze how much time u could possibly spend in those two states.

Let u be in state $\mathbb{W}$ or $\mathbb{A}$. Whenever u gets reset in round r to the beginning of $\mathbb{W}$, some neighbor v of u becomes a candidate. We have seen that then v becomes decided within time T' w.h.p.

When v becomes decided in round $r' \leq r + T'$, some node $w \in N^1(v) \subset N^2(u)$ (possibly v itself) has to become a dominator in that round r'. Note that because w becomes a dominator in round r', the region $R(w)$ cannot be decided in round $r' - 1$, as w is in state $\mathbb{C}$ in that round. By Lemma 5.6 and our initial argumentation, however, the region $R(w)$ becomes decided in some round $r'' \leq r' + T' \leq r + 2T'$. When u gets set back to the beginning of state $\mathbb{W}$ again after that round r'', we can again conclude that a non-decided region $R(w'')$ of some dominator w' within distance 2 from u becomes decided within $2T'$ rounds. As there can only be $\Delta^2 + 1$ regions within distance 2 from u, this can only repeat that many times until u has to either hear from a dominator or proceed to state $\mathbb{C}$ itself. The claim then follows. □

Lemma 5.8. *For each region R, the expected number of nodes that become dominators in region R is bounded by $\mathcal{O}(1)$.*

Proof. Consider some fixed region R and let t_0 be the first time when a node becomes a candidate in region R. For $i = 1, 2, \ldots$, let $P_{\mathbb{C},i}$ be the sum of the broadcast probabilities of all the candidates in region R on channel $\mathcal{C}$ in round t_0+i. As nodes have to be candidates before becoming dominators, dominators in region R can only be created after time t_0. Let X_i be the number of dominators created in R in round $t_0 + i$ and let $X = \sum_{i\geq 1} X_i$. To prove the lemma, we have to show that $\mathbf{E}[X] = \mathcal{O}(1)$.

We say that a newly created dominator v in round r *clears* its region R iff v is the only dominator created in region R in round r and all candidates in region R hear v's message on channel $\mathcal{C}$ in round r. Clearly, all nodes that are candidates in region R in round r switch to state $\mathbb{E}$ when this occurs – except those in phase 0. Therefore, the only nodes in R that can still become dominators must either be in another state ($\mathbb{W}$, $\mathbb{A}$), not yet awake or sleeping candidates. By Lemma 5.6, $|\mathbb{D} \cap R|$ is always bounded such that by Lemma 5.5, w.h.p., any such nodes but the sleeping candidates do not become dominators.

Having established the power of clearing, we bound the probability of such events. In more detail, let $\mathcal{E}_i$ be the event that in round $t_0 + i$ some node v in region R becomes a dominator by clearing R. Assume for the moment that if a clearance occurs, no sleeping candidate makes it to the readying phase, phase 0 – either because there are no sleeping candidates or because they learn of a dominator.

We next show that a clearance happens with probability at least $\delta P_{\mathbb{C},i}$ for some constant $\delta > 0$. To see why, recall that by Lemma 5.4, we know that for all $i \geq 1$, $P_{\mathbb{C},i}$ as well as $P_{\mathbb{C},i}(R')$ in round $t_0 + i$ for every neighboring region R' are upper bounded by some constant $\hat{P}_{\mathbb{C}}$. For each candidate v in region R let $p(v)$ be its broadcasting probability. Then the probability that exactly one candidate from region R broadcasts on channel $\mathcal{C}$ in round $t_0 + i$ is lower bounded by

$$\sum_{v\in R\cap\mathbb{C}} p(v) \prod_{u\in R\cap\mathbb{C},\, u\neq v} (1 - p(u)) \geq P_{\mathbb{C},i} 4^{-\hat{P}_{\mathbb{C}}} = \Omega(P_{\mathbb{C},i}).$$

The probability that no candidate from any neighboring region R' (of which there are at most Δ) broadcasts on channel $\mathcal{C}$ in round $t_0 + i$ is at least $4^{-\Delta\hat{P}_{\mathbb{C}}} = \Omega(1)$. Hence, there exists a constant $\delta > 0$ such that $\mathbf{P}(\mathcal{E}_i) \geq \delta P_{\mathbb{C},i}$.

In the following, we define $Q_i := \sum_{j=1}^{i} P_{\mathbb{C},i}$. The probability that no node v clears region R by some time $t_0 + \tau$ can be upper bounded by

$$\mathbf{P}\left(\bigcap\nolimits_{i=1}^{\tau} \overline{\mathcal{E}_i}\right) \leq \prod\nolimits_{i=1}^{\tau} (1 - \delta P_{\mathbb{C},i}) < e^{-\delta \sum_{i=1}^{\tau} P_{\mathbb{C},i}} = e^{-\delta Q_\tau}.$$

As discussed above and using our assumption that no sleeping candidate manages to reach the readying phase, w.h.p., a clearance in R prevents new nodes from subsequently becoming dominators in this region. Let $\mathcal{G}$ be the event that this high probability property holds. When we condition on $\mathcal{G}$, it holds that a dominator can join a region in a given round only if there have been no previous clearances in that region. Hence,

$$\mathbf{E}\left[X_i|\mathcal{G}\right] \leq \mathbf{P}\left(\bigcap\nolimits_{j=1}^{i-1} \overline{\mathcal{E}}_i\right) \cdot P_{\mathbb{C},i}$$

and therefore

$$\mathbf{E}\left[X|\mathcal{G}\right] \leq \sum\nolimits_{i\geq 1} P_{\mathbb{C},i} \cdot e^{-\delta Q_{i-1}} = \mathcal{O}(1).$$

Because $\mathcal{G}$ happens w.h.p., and $|\mathbb{D}| \leq n$, we get $\mathbf{E}[X] = \mathcal{O}(1)$.

Now we strip off the assumption that no sleeping nodes arrive at phase 1 after the first clearance. Those nodes are the only ones possibly being able to become dominators. For that, however, they need to reach phase 2. When the first such node reaches phase 2, all other nodes that were sleeping candidates at the time of the first clearance, have either already been eliminated by a dominator, or they advanced out from phase 0. But this means that the next clearance eliminates all readying and competing candidates, while all sleeping candidates, dormant nodes and nodes in state $\mathbb{W}$and $\mathbb{A}$become eliminated by previously created dominators. We can apply the same logic as above to bound the dominators being created until the second clearance happens by $\mathcal{O}(1)$. □

Proof of Theorem 5.1. By Lemma 5.7, w.h.p., after it wakes up every node is decided within $\mathcal{O}(\log^2 n/\mathcal{F} + \log n \log\log n)$ rounds. Since a node only goes to state $\mathbb{E}$ after hearing from a neighboring dominator, the computed dominating set is valid. Finally, by Lemma 5.8, in expectation, the algorithm computes a constant approximation of the optimal MDS solution. □

References

[1] I. 802.11. Wireless LAN MAC and Physical Layer Specifications, June 1999.

[2] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg. A Lower Bound for Radio Broadcast. *Journal of Computer and System Sciences*, 43(2):290–298, 1991.

[3] 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(1):104–126, 1992.

[4] L. Barriére, P. Fraigniaud, and L. Narayanan. Robust position-based routing in wireless ad hoc networks with unstable transmission ranges. In *Proc. 5th Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM)*, pages 19–27, 2001.

[5] Bluetooth Consortium. *Bluetooth Specification Version 2.1*, July 2007.

[6] A. D. Bonis, L. Gasieniec, and U. Vaccaro. Generalized fFamework for Selectors with Applications in Optimal Group Testing. In *Proceedings of the International Colloquium on Automata, Languages and Programming*, 2003.

[7] I. Chlamtac and S. Kutten. On Broadcasting in Radio Networks–Problem Analysis and Protocol Design. *IEEE Transactions on Communications*, 33(12):1240–1246, 1985.

[8] B. Chlebus and D. Kowalski. A Better Wake-Up in Radio Networks. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, pages 266–274. ACM, 2004.

[9] M. Chrobak, L. Gasieniec, and D. Kowalski. The Wake-Up Problem in Multi-Hop Radio Networks. In *Proceedings of the Annual Symposium on Discrete Algorithms*, pages 992–1000. Society for Industrial and Applied Mathematics, 2004.

[10] A. E. F. Clementi, A. Monti, and R. Silvestri. Distributed Broadcast in Radio Networks of Unknown Topology. *Theoretical Computer Science*, 302(1-3), 2003.

[11] S. Daum, S. Gilbert, F. Kuhn, and C. Newport. Leader Election in Shared Spectrum Networks. In *Proceedings of the Principles of Distributed Computing (to appear)*, 2012.

[12] M. Farach-Colton, R. J. Fernandes, and M. A. Mosteiro. Lower Bounds for Clear Transmissions in Radio Networks. In *The Proceedings of the Latin American Symposium on Theoretical Informatics*, 2006.

[13] L. Gasieniec, A. Pelc, and D. Peleg. The Wakeup Problem in Synchronous Broadcast Systems. In *Proceedings of the ACM symposium on Principles of Distributed Computing*, pages 113–121, 2000.

[14] T. Jurdzinski and G. Stachowiak. Probabilistic Algorithms for the Wakeup Problem in Single-Hop Radio Networks. In *Proceedings of the International Symposium on Algorithms and Computation*, pages 535–549, 2002.

[15] F. Kuhn, T. Moscibroda, and R. Wattenhofer. Initializing Newly Deployed Ad Hoc and Sensor Networks. In *Proceedings of International Conference on Mobile Computing and Networking*, pages 260–274. ACM, 2004.

[16] F. Kuhn, T. Moscibroda, and R. Wattenhofer. On the locality of bounded growth. In *Proc. 24th Symp. on Principles of Distributed Computing (PODC)*, pages 60–68, 2005.

[17] F. Kuhn, R. Wattenhofer, and A. Zollinger. Ad hoc networks beyond unit disk graphs. *Wireless Networks*, 14(5):715–729, 2008.

[18] E. Kushilevitz and Y. Mansour. An (D log (N/D)) Lower Bound for Broadcast in Radio Networks. *SIAM Journal on Computing*, 27(3):702–712, 1998.

[19] T. Moscibroda and R. Wattenhofer. Maximal independent sets in radio networks. In *Proc. 24th Symp. on Principles of Distributed Computing (PODC)*, pages 148–157, 2005.

[20] S. Schmid and R. Wattenhofer. Algorithmic models for sensor networks. In *Proc. 14th Int. Workshop on Parallel and Distributed Real-Time Systmes*, pages 1–11, 2006.

Algorithm 2: Dominating Set Algorithm

States: $\mathbb{W}$ – *waiting*, $\mathbb{A}$ – *active*, $\mathbb{C}$ – *candidate*, $\mathbb{D}$ – *dominator*, $\mathbb{E}$ – *eliminated*

Channels: $\mathcal{A}_1, \ldots, \mathcal{A}_{\mathrm{F}}$ – *filtering*, $\mathcal{D}_1, \ldots, \mathcal{D}_{n_{\mathcal{D}}}$ – *notification*, $\mathcal{C}$ – *competition*

begin

$\alpha_{\mathbb{W}} = \alpha_{sleep} = \Theta(\log^2 n/\mathcal{F} + \log n \log\log n)$; $\alpha_{\mathbb{A}} = \Theta(\log n/\mathrm{F})$; $\alpha_{\mathbb{C}} = \Theta(\log n)$

set *count* $:= 0$; *state* $:= \mathbb{W}$

if $\mathcal{F} = \Omega(\log\log n)$ **then** $n_{\mathcal{D}} := \Theta(\log\log n)$

else $n_{\mathcal{D}} := 4$

while *state* $\neq \mathbb{E}$ **do**

- *count* := *count* $+ 1$
- uniformly at random pick: $i \in \{1, \ldots, n_{\mathcal{D}}\}$; $k \in \{1, \ldots, \mathrm{F}\}$; $q \in [0, 1)$
- **switch** *state* **do**
  - **case** $\mathbb{W}$
    - listen on $\mathcal{D}_i$
    - **if** *count* $= \alpha_{\mathbb{W}}$ **then** *count* $:= 0$, *state* $:= \mathbb{A}$, *phase* $:= 0$
  - **case** $\mathbb{A}$
    - **if** *count* $= \alpha_{\mathbb{A}}$ **then** *count* $:= 0$, *phase* $:= \min\{phase + 1, \log(n/4)\}$
    - **if** $q > \frac{2^{phase}}{n}$ **then** listen on $\mathcal{A}_k$
    - **else** send on $\mathcal{A}_k$; *count* $:= 0$, *phase* $:= 0$, *state* $:= \mathbb{C}$
  - **case** $\mathbb{C}$
    - **if** *phase* $= 0$ **then**
      - listen on $\mathcal{D}_i$
      - **if** *count* $= \alpha_{sleep}$ **then** *count* $:= 0$, *phase* $:= 1$
    - **else if** *phase* $= 1$ **then**
      - listen on $\mathcal{C}$
      - **if** *count* $= \alpha_{sleep}$ **then** *count* $:= 0$, *phase* $:= 2$
    - **else**
      - **if** *count* $= \alpha_{\mathbb{C}}$ **then** *count* $:= 0$, *phase* $:= \min\{phase + 1, 2\log\log n\}$
      - **if** $q > \frac{2^{phase-2}}{\log^2 n}$ **then** listen on $\mathcal{C}$
      - **else** send on $\mathcal{C}$; *state* $:= \mathbb{D}$
  - **case** $\mathbb{D}$
    - **if** $n_{\mathcal{D}} = 4$ **then** $p := \left(\frac{\mathcal{F}}{\log n}\right)^i$
    - **else** $p := 2^{-i}$
    - with probability p send on $\mathcal{D}_i$

Upon receiving a message:

if *state* $= \mathbb{A}$ **then** *count* $:= 0$, *state* $:= \mathbb{W}$

else *state* $:= \mathbb{E}$