Radio Network Lower Bounds Made Easy*

Calvin Newport

Georgetown University
Washington, DC
cnewport@cs.georgetown.edu

Abstract. Theoreticians have studied distributed algorithms in the synchronous radio network model for close to three decades. A significant fraction of this work focuses on lower bounds for basic communication problems such as *wake-up* (symmetry breaking among an unknown set of nodes) and *broadcast* (message dissemination through an unknown network topology). In this paper, we introduce a new technique for proving this type of bound, based on reduction from a probabilistic hitting game, that simplifies and strengthens much of this existing work. In more detail, in this single paper we prove new expected time and high probability lower bounds for wake-up and global broadcast in single and multi-channel versions of the radio network model both with and without collision detection. In doing so, we are able to reproduce results that previously spanned a half-dozen papers published over a period of twenty-five years. In addition to simplifying these existing results, our technique, in many places, also improves the state of the art: of the eight bounds we prove, four strictly strengthen the best known previous result (in terms of time complexity and/or generality of the algorithm class for which it holds), and three provide the first known non-trivial bound for the case in question. The fact that the same technique can easily generate this diverse collection of lower bounds indicates a surprising unity underlying communication tasks in the radio network model—revealing that deep down, below the specifics of the problem definition and model assumptions, communication in this setting reduces to finding efficient strategies for a simple game.

1 Introduction

In this paper, we introduce a new technique for proving lower bounds for basic communication tasks in the radio network model. We use this technique to unify, simplify, and in many cases strengthen the best known lower bounds for two particularly important problems: wake-up and broadcast.

The Radio Network Model. The radio network model represents a wireless network as a graph $G = (V, E)$, where the nodes in V correspond to the wireless devices and the edges in E specify links. Executions proceed in synchronous rounds.

* This work is supported in part by NSF grant number CCF 1320279 and the Ford Motor Company University Research Program.

F. Kuhn (Ed.): DISC 2014, LNCS 8784, pp. 258–272, 2014.

In each round, each node can choose whether or not to broadcast messages to its neighbors in G. If multiple neighbors of a given node broadcast during the same round, however, the messages are lost due to collision. This model was first introduced by Chlamtac and Kutten [4], who used it to study centralized algorithms. Soon after, Bar-Yehuda et al. [2,3] introduced the model to the distributed algorithms community where variations have since been studied in a large number of subsequent papers; e.g., [1,20,18,21,13,19,6,10,11,17,9,12,8,7,15,14].

Two of the most investigated problems in the radio network model are *wake-up* (basic symmetry breaking among an unknown set of participants in a single hop network) and *broadcast* (propagating a message from a source to all nodes in an unknown multihop network). Lower bounds for these problems are important because wake-up and/or broadcast reduce to most useful communication tasks in this setting, and therefore capture something fundamental about the cost of distributed computation over radio links.

Our Results. In this paper, we introduce a new technique (described below) for proving lower bounds for wake-up and broadcast in the radio network model. We use this technique to prove new expected time and high probability lower bounds for these two problems in the single and multiple channel versions of the radio network model both with and without collision detection. In doing so, we reproduce in this single paper a set of existing results that spanned a half-dozen papers [23,20,18,13,9,7] published over a period of twenty-five years. Our technique simplifies these existing arguments and establishes a (perhaps) surprising unity among these diverse problems and model assumptions. Our technique also strengthens the state of the art. All but one of the results proved in this paper improve the best known existing result by increasing the time complexity and/or generalizing the class of algorithms for which the bound holds (many existing bounds for these problems hold only for *uniform* algorithms that require nodes to use a pre-determined sequence of independent broadcast probabilities; all of our lower bounds, by contrast, hold for all randomized algorithms). In several cases, we prove the first known bound for the considered assumptions.

The full set of our results with comparisons to existing work are described in Figure 1. Here we briefly mention three highlights (in the following, n is the network size and D the network diameter). In Section 6, we significantly simplify Willard's seminal $\Omega(\log\log n)$ bound for expected time wake-up with collision detection [23]. In addition, whereas Willard's result only holds for uniform algorithms, our new version holds for all algorithms. In Section 7, we prove the first tight bound for high probability wake-up with multiple channels and the first known expected time bound in this setting. And in Section 9, we prove that Kushilevitz and Mansour's oft-cited $\Omega(D\log(n/D))$ lower bound for expected time broadcast [20] *still holds* even if we assume multiple channels and/or collision detection—opening an unexpected gap with the wake-up problem for which these assumptions improve the achievable time complexity.

Our Technique. Consider the following simple game which we call *k-hitting*. A *referee* secretly selects a target set $T \subseteq \{1, 2, ..., k\}$. The game proceeds in rounds.

In each round, a *player* (represented by a randomized algorithm) generates a proposal P. If $|P \cap T| = 1$, the player wins. Otherwise, it moves on to the next round. In Section 3, we leverage a useful combinatorial result due to Alon et al. [1] to prove that this game requires $\Omega(\log^2 k)$ rounds to solve with high probability (w.r.t. k), and $\Omega(\log k)$ rounds in expectation. (Notice, you could propose the sets of a (k,k)-*selective family* [5] to solve this problem deterministically, but this would require $\Omega(k)$ proposals in the worst-case.)

These lower bounds are important because in this paper we show that this basic hitting game reduces to solving wake-up and broadcast under all of the different combinations of model assumptions that we consider. In other words, whether or not you are solving wake-up or broadcast, assuming multiple channels or a single channel, and/or assuming collision detection or no collision detection, if you can solve the problem quickly you can solve this hitting game quickly. Our lower bounds on the hitting game, therefore, provide a fundamental speed-limit for basic communication tasks in the radio network model.

The trick in applying this method is identifying the proper reduction argument for the assumptions in question. Consider, for example, our reduction for wake-up with a single channel and no collision detection. Assume some algorithm $\mathcal{A}$ solves wake-up with these assumptions in $f(n)$ rounds, in expectation. As detailed in Section 5, we can use $\mathcal{A}$ to define a player that solves the k-hitting game in $f(k)$ rounds with the same probability—allowing the relevant hitting game lower bound to apply. Our strategy for this case is to have the player simulate $\mathcal{A}$ running on all k nodes in a network of size k. For each round of the simulation, it proposes the ids of the nodes that broadcast, then simulates all nodes receiving nothing. This is not necessarily a valid simulation of $\mathcal{A}$ running on k nodes: *but it does not need to be.* What we care about are the simulated nodes with ids in T: the (unknown to the player) target set for this instance of the hitting game. The key observation is that in the *target execution* where only the nodes in T are active, they will receive nothing until the first round where one node broadcasts alone—solving wake-up. In the player's simulation, these same nodes are also receiving nothing (by the the player's fixed receive rule) so they will behave the same way. This will lead to a round of the simulation where only one node from T (and perhaps other nodes outside of T) broadcast. The player will propose these ids, winning the hitting game.

These reductions get more tricky as we add additional assumptions. Consider, for example, what happens when we now assume collision detection. Maintaining consistency between the nodes in T in the player simulation and the target execution becomes more complicated, as the player must now correctly simulate a collision event whenever two or more nodes from T broadcast—even though the player *does not know* T. Adding multiple channels only further complicates this need for consistency. Each bound in this paper, therefore, is built around its own clever method for a hitting game player to correctly simulate a wake-up or broadcast algorithm in such a way that it wins the hitting game with the desired efficiency. These arguments are simple to understand and sometimes surprisingly elegant once identified, but can also be elusive before they are first pinned down.

	Existing (exp. | high)	This Paper (exp. | high)
wake-up	$\Omega(\log n) \mid \Omega(\log^2 n)$ [18,13]	$\Omega(\log n) \mid \Omega(\log^2 n)$ (*)
wake-up/cd	$\Omega(\log \log n) \mid \Omega(\log n)$ [23]	$\Omega(\log \log n) \mid \Omega(\log n)$ (*)
wake-up/mc	? $\mid \Omega(\frac{\log^2 n}{\mathcal{C} \log \log n} + \log n)$ [9,7]	$\Omega(\frac{\log n}{\mathcal{C}} + 1) \mid \Omega(\frac{\log^2 n}{\mathcal{C}} + \log n)$ (*)
wake-up/cd/mc	$\Omega(1) \mid$?	$\Omega(1) \mid \Omega(\frac{\log n}{\log \mathcal{C}} + \log \log n)$
broadcast	$\Omega(D \log (n/D))$ [20]	$\Omega(D \log (n/D))$
broadcast/cd/mc	?	$\Omega(D \log (n/D)$

Fig. 1. This table summarizes the expected time (exp.) and high probability (high) results for wake-up and broadcast in the existing literature as well as the new bounds proved in this paper. In these bounds, n is the network size, $\mathcal{C}$ the number of channels, and D the network diameter. In the problem descriptions, "cd" indicates the collision detection assumption and "mc" indicates the multiple channels assumption. In the existing results we provide citation for the paper(s) from which the results derive and use "?" to indicate a previously open problem. In all cases, the new results in this paper simplify the existing results. We marked some of our results with "(*)" to indicate that the existing results assumed the restricted *uniform* class of algorithms. All our algorithms hold for all randomized algorithms, so any result marked by "(*)" is strictly stronger than the existing result. We do not separate expected time and high probability for the broadcast problems as the tight bounds are the same for both cases.

Roadmap. A full description of our results and how they compare to existing results is provided in Figure 1. In addition, before we prove each bound in the sections that follow, we first discuss in more detail the relevant related work. In Section 2, we formalize our model and the two problems we study. In Section 3, we formalize the hitting games at the core of our technique then bound from below their complexity. In Section 4, we detail a general simulation strategy that we adopt in most of our wake-up bounds (by isolating this general strategy in its own section we reduce redundancy). Sections 5 to 8 contain our wake-up lower bounds, and Section 9 contains our broadcast lower bound. (We only need one section for broadcast as we prove that the same result holds for all assumptions considered in this paper.)

2 Model and Problems

In this paper we consider variants of the standard *radio network model*. This model represents a radio network with a connected undirected graph $G = (V, E)$ of diameter D. The $n = |V|$ nodes in the graph represent the wireless devices and the edges in E capture communication proximity. In more detail, executions in this model proceed in synchronous rounds. In each round, each node can choose to either *transmit* a message or *receive*. In a given round, a node $u \in V$ can receive a message from a node $v \in V$, if and only if the following conditions hold: (1) u is receiving and v is transmitting; (2) v is u's neighbor in G; and (3) v is the *only* neighbor of u transmitting in this round. The first condition captures the half-duplex nature of the radio channel and the second condition captures message collisions. To achieve the strongest possible lower bounds, we assume nodes are provided unique ids from $[n]$. In the following, we say an algorithm is *uniform* if (active) nodes use a predetermined sequence of independent broadcast probabilities to determine whether or not to broadcast in each round, up until they first receive a message. A uniform algorithm, for example, cannot select its broadcast probability in a given round based on the outcome of a coin flip during a previous round. This prohibits, among other strategies, allowing nodes to change their behavior based on whether or not they previously chose to broadcast (e.g., as in [21]).

In the *collision detection* variant of the radio network model, a receiving node u can distinguish between silence (no neighbor is transmitting) and collision (two or more neighbors are transmitting) in a given round. In this paper, to achieve the strongest possible lower bounds, when studying single hop networks we also assume that a transmitter can distinguish between broadcasting alone and broadcasting simultaneously with one or more other nodes. In the *multichannel* variant of the radio network model, we use a parameter $\mathcal{C} \geq 1$ to indicate the number of orthogonal communication channels available to the nodes. When $\mathcal{C} > 1$, we generalize the model to require each node to choose in each round a single channel on which to participate. The communication rules above apply separately to each channel. In other words, a node u receives a message from v on channel c in a given round, if and only if in this round: (1) u receives on c and v transmits on c; (2) v is a neighbor of u; and (3) no other neighbor of u transmits on c.

We study both *expected time* and *high probability* results, where we define the latter to mean probability at least $1-\frac{1}{n}$. We define the notation $[i, j]$, for integers $i \leq j$, to denote the range $\{i, i+1, ..., j\}$, and define $[i]$, for integer $i > 0$, to denote $[1, i]$.

Problems. The *wake-up* problem assumes a single hop network consisting of *inactive* nodes. At the beginning of the execution, an arbitrary subset of these nodes are *activated* by an adversary. Inactive nodes can only listen, while active nodes execute an arbitrary randomized algorithm. We assume that active nodes have no advance knowledge of the identities of the other active nodes. The problem is solved in the first round in which an active node broadcasts alone (therefore *waking up* the listening inactive nodes). When considering a model with collision detection, we still require that an active node broadcasts alone to solve the problem (e.g., to avoid triviality, we assume that the inactive nodes need to receive a message to wake-up, and that simply detecting a collision is not sufficient[1]). When considering multichannel networks, we assume the inactive nodes are all listening on the same known *default* channel (say, channel 1). To solve the problem, therefore, now requires that an active node broadcasts alone on the default channel.

The *broadcast* problem assumes a connected multihop graph. At the beginning of the execution, a single *source* node u is provided a message m. The problem

[1] Without this restriction, the problem is trivially solved by just having all active nodes broadcast in the first round.

is solved once every node in the network has received m. We assume nodes do not have any advance knowledge of the network topology. As is standard, we assume that nodes are inactive (can only listen) until they first receive m. As in the wake-up problem, detecting a collision alone is not sufficient to activate an inactive node, and in multichannel networks, we assume inactive nodes all listen on the default channel.

3 The k-Hitting Game

The *k-hitting game*, defined for some integer $k > 1$, assumes a *player* that faces off against an *referee*. At the beginning of the game, the referee secretly selects a *target set* $T \subseteq \{1, ..., k\}$. The game then proceeds in rounds. In each round, the player generates a *proposal* $P \subseteq \{1, ..., k\}$. If $|P \cap T| = 1$, then the player wins the game. Otherwise, the player moves on to the next round learning no information other than the fact that its proposal failed. We formalize both entities as probabilistic automata and assume the player does not know the referee's selection and the referee does not know the player's random bits. Finally, we define the *restricted* k-hitting game to be a variant of the game where the target set is always of size two.

A Useful Combinatorial Result. Before proving lower bounds for our hitting game we cite an existing combinatorial result that will aid our arguments. To simplify the presentation of this result, we first define some useful notation. Fix some integer $\ell > 0$. Consider two sets $A \subseteq \{1, 2, ..., \ell\}$ and $B \subseteq \{1, 2, ...\ell\}$. We say that A *hits* B if $|A \cap B| = 1$. Let an *ℓ-family* be a family of non-empty subsets of $\{1, 2, ..., \ell\}$. The *size* of an ℓ-family $\mathscr{A}$, sometimes noted as $|\mathscr{A}|$, is the number of sets in $\mathscr{A}$. Fix two ℓ-families $\mathscr{A}$ and $\mathscr{B}$. We say $\mathscr{A}$ *hits* $\mathscr{B}$, if for every $B \in \mathscr{B}$ there exists an $A \in \mathscr{A}$ such that A hits B. Using this notation, we can now present the result:

Lemma 1 ([1,15]). *There exists a constant $\beta > 0$, such that for any integer $\ell > 1$, these two results hold:*

1. *There exists an ℓ-family $\mathscr{R}$, where $|\mathscr{R}| \in O(\ell^8)$, such that for every ℓ-family $\mathscr{H}$ that hits $\mathscr{R}$, $|\mathscr{H}| \in \Omega(\log^2 \ell)$.*
2. *There exists an ℓ-family $\mathscr{S}$, where $|\mathscr{S}| \in O(\ell^8)$, such that for every $H \subseteq \{1, 2, ..., \ell\}$, H hits at most a $(\frac{1}{\beta \log(\ell)})$-fraction of the sets in $\mathscr{S}$.*

The first result from this lemma was proved in a 1991 paper by Alon et al. [1]. It was established using the probabilistic method and was then used to prove a $\Omega(\log^2 n)$ lower bound on *centralized* broadcast solutions in the radio network model. The second result is a straightforward consequence of the analysis used in [1], recently isolated and proved by Ghaffari et al. [15].

Lower Bounds for the k-Hitting Game. We now prove lower bounds on our general and restricted k-hitting games. These results, which concern probabilities, leverage Lemma 1, which concerns combinatorics, in an interesting way which depends on the size of $\mathscr{R}$ and $\mathscr{S}$ being polynomial in ℓ.

Theorem 1. *Fix some player $\mathcal{P}$ that guarantees, for all $k > 1$, to solve the k-hitting game in $f(k)$ rounds, in expectation. It follows that $f(k) \in \Omega(\log k)$.*

Proof. Fix any $k > 1$. Let β and $\mathscr{S}$ be the constant and ℓ-family provided by the second result of Lemma 1 applied to $\ell = k$. The lemma tells us that for any $P \subseteq [k]$, P hits at most a $(\frac{1}{\beta \log k})$-fraction of the sets in $\mathscr{S}$. It follows that for any k-family $\mathscr{H}$, such that $|\mathscr{H}| < \frac{\beta \log k}{2}$, $\mathscr{H}$ hits less than half the sets in $\mathscr{S}$.

We now use these observations to prove our theorem. Let $\mathcal{P}$ be a k-hitting game player. Consider a referee that selects the target set by choosing a set T from $\mathscr{S}$ with uniform randomness. Let $\mathscr{H}$ be the first $\lfloor \frac{\beta \log k}{2} \rfloor - 1$ proposals generated by $\mathcal{P}$ in a given instance of the game. By our above observation, this sequence of proposals hits less than half the sets in $\mathscr{S}$. Because the target set was chosen from $\mathscr{S}$ with randomness that was uniform and independent of the randomness used by $\mathcal{P}$ to generate its proposals, it follows that the probability that $\mathscr{H}$ hits the target is less than 1/2. To conclude, we note that $f(k)$ must therefore be larger than $\lfloor \frac{\beta \log k}{2} \rfloor - 1 \in \Omega(\log k)$, as required by the theorem.

Theorem 2. *Fix some player $\mathcal{P}$ that guarantees, for all $k > 1$, to solve the k-hitting game in $f(k)$ rounds with probability at least $1 - \frac{1}{k}$. It follows that $f(k) \in \Omega(\log^2 k)$.*

Proof. Fix any $\ell > 1$. Let $\mathscr{R}$ be the ℓ-family provided by the first result of Lemma 1 applied to this value. Let $t = |\mathscr{R}|$. We know from the lemma that $t \in O(\ell^8)$.

To achieve our bound, we will consider the behavior of a player $\mathcal{P}$ in the k-hitting game for $k = t + 1$. As in Theorem 1, we have our referee select its target set by choosing a set from $\mathscr{R}$ with uniform randomness. (Notice, in this case, our referee is actually making things *easier* for the player by restricting its choices to only the values in $[\ell]$ even though the game is defined for the value set $[k]$, which is larger. As we will show, this advantage does not help the player much.)

Let $c \log^2(\ell)$, for some constant $c > 0$, be the exact lower bound from the first result of Lemma 1. Let $\mathscr{H}$ be the first $\lfloor c \log^2(\ell) \rfloor - 1$ proposals generated by $\mathcal{P}$ in a given instance of the game. Lemma 1 tells us that there is at least one set $R \in \mathscr{R}$ that $\mathscr{H}$ does not hit. Because the target set was chosen from $\mathscr{R}$ with randomness that was uniform and independent of the randomness used by $\mathcal{P}$, it follows that the probability that $\mathscr{H}$ misses the target is at least $1/t$ (recall that t is the size of $\mathscr{R}$). Inverting this probability, it follows that the probability that $\mathcal{P}$ wins the game with the proposals represented by $\mathscr{H}$ is less than or equal to $1 - \frac{1}{t} = 1 - \frac{1}{k-1} < 1 - \frac{1}{k}$. It follows that $f(k)$ must be larger than $|\mathscr{H}|$ and therefore must be of size at least $c \log^2(\ell) \in \Omega(\log^2(\ell))$. To conclude the proof, we note that $k \in O(\ell^8)$, from which it follows that $\ell \in \Omega(k^{1/8})$ and therefore that $\log^2(\ell) \in \Omega(\log^2 k)$, as required by the theorem.

The below theorem is proved similar to Theorem 2. The details can be found in the full version of this paper [22].

Theorem 3. *Fix some player $\mathcal{P}$ that guarantees, for all $k > 1$, to solve the* restricted *k-hitting game in $f(k)$ rounds with probability at least $1 - \frac{1}{k}$. It follows that $f(k) \in \Omega(\log k)$.*

4 Simulation Strategy

Most of our bounds for the *wake-up* problem use a similar simulation strategy. To reduce redundancy, we define the basics of the strategy and its accompanying notation in its own section. In more detail, the *wake-up simulation strategy*, defined with respect to a wake-up algorithm $\mathcal{A}$, is a general strategy for a k-hitting game player to generate proposals based on a local simulation of $\mathcal{A}$. The strategy works as follows. The player simulates $\mathcal{A}$ running on all k nodes in a k-node network satisfying the same assumptions on collision detection and channels assumed by $\mathcal{A}$. For each simulated round, the player will generate one or more proposals for the hitting game. In more detail, at the beginning of a new simulated round, the player simulates the k nodes running $\mathcal{A}$ up until the point that they make a broadcast decision. At this point, the player applies a *proposal rule* that transforms these decisions into one or more proposals for the hitting game. The player then makes these proposals, one by one, in the game. If none of these proposals wins the hitting game, then the player most complete the current simulated round by using a *receive rule* to specify what each node receives; i.e., silence, a message, or a collision (if collision detection is assumed by $\mathcal{A}$). In other words, a given application of the wake-up simulation strategy is defined by two things: a definition of the *proposal rule* and *receive rule* used by the player to generate proposals from the simulation, and specify receive behavior in the simulation, respectively.

To analyze a wake-up simulation strategy for a given instance of the k-hitting game with target set T, we define the *target execution* for this instance to be the execution that would result if $\mathcal{A}$ was run in a network where only the nodes corresponding to T were active and they used the same random bits as the player uses on their behalf in the simulation. We say the simulation strategy is *consistent* with its target execution through a given round, if the nodes corresponding to T in the simulation behave the same (e.g., send and receive the same messages) as the corresponding nodes in the target execution through this round.

5 Lower Bounds for Wake-Up

We begin by proving tight lower bounds for both expected and high probability solutions to the wake-up problem in the most standard set of assumptions used with the radio network model: a single channel and no collision detection. As explained below, our bounds are tight and generalize the best know previous bounds, which hold only for uniform algorithms, to now apply to all randomized algorithms. (We note that a preliminary version of our high probability bound below appeared as an aside in our previous work on structuring multichannel radio networks [8]).

In terms of related work, the *decay* strategy introduced Bar-Yehuda et al. [3] solves the wake-up problem in this setting with high probability in $O(\log^2 n)$ rounds and in expectation in $O(\log n)$ rounds. In 2002, Jurdzinski and Stachowiak [18] proved the necessity of $\Omega\left(\frac{\log n \log(1/\epsilon)}{\log\log n + \log\log(1/\epsilon)}\right)$ rounds to solve wake-up with probability at least $1-\epsilon$, which proves decay optimal within a $\log\log n$ factor. Four years later, Farach-Colton et al. [13] removed the $\log\log n$ factor by applying linear programming techniques. As mentioned, these existing bounds only apply to uniform algorithms in which nodes use a predetermined sequence of broadcast probabilities. (Section 3.1 of [13] claims to extend their result to a slightly more general class of uniform algorithms in which a node can choose a uniform algorithm to run based on its unique id.)

Theorem 4. *Let $\mathcal{A}$ be an algorithm that solves wake-up with high probability in $f(n)$ rounds in the radio network model with a single channel and no collision detection. It follows that $f(n) \in \Omega(\log^2 n)$.*

Proof. Fix some wake-up algorithm $\mathcal{A}$ that solves wake-up in $f(n)$ rounds with high probability in a network with one channel and no collision detection. We start by defining a wake-up simulation strategy that uses $\mathcal{A}$ (see Section 4). In particular, consider the *proposal rule* that has the player propose the id of every node that broadcasts in the current simulated round, and the *receive rule* that always has all nodes receive nothing.

Let $\mathcal{P}_\mathcal{A}$ be the k-hitting game player that uses this simulation strategy. We argue that $\mathcal{P}_\mathcal{A}$ solves the k-hitting game in $f(k)$ rounds with high probability in k. To see why, notice that for a given instance of the hitting game with target T, $\mathcal{P}_\mathcal{A}$ is consistent with the target execution until the receive rule of the first round in which exactly one node in T broadcasts. (In all previous rounds, $\mathcal{P}_\mathcal{A}$ correctly simulates the nodes in T receiving nothing, as its receive rule has all nodes always receive nothing.) Assume $\mathcal{A}$ solves wake-up in round r in the target execution. It follows that r is the first round in which a node in T broadcasts alone in this execution. By our above assumption, $\mathcal{P}_\mathcal{A}$ is consistent with the target execution up to the application of the receive rule in r. In particular, it is consistent when it applies the proposal rule for simulated round r. By assumption, this proposal will include exactly one node from T—winning the hitting game.

We assumed that $\mathcal{A}$ solves wake-up in $f(n)$ rounds with high probability in n. Combined with our above argument, it follows that $\mathcal{P}_\mathcal{A}$ solves the k-hitting game in $f(k)$ rounds with high probability in k. To complete our lower bound, we apply a contradiction argument. In particular, assume for contradiction that there exists a wake-up algorithm $\mathcal{A}$ that solves wake-up in $f(n) \in o(\log^2 n)$ rounds, with high probability. The hitting game player $\mathcal{P}_\mathcal{A}$ defined above will therefore solve k-hitting in $o(\log^2 n)$ rounds with high probability. This contradicts Theorem 2.

Theorem 5. *Let $\mathcal{A}$ be an algorithm that solves wake-up in $f(n)$ rounds, in expectation, in the radio network model with a single channel and no collision detection. It follows that $f(n) \in \Omega(\log n)$.*

Proof (Idea). It is sufficient to apply the same argument as in Theorem 4. The only change is in the final contradiction argument, where we simply replace $\log^2 n$ with $\log n$, and now contradict Theorem 1.

6 Lower Bounds for Wake-Up with Collision Detection

We prove tight lower bounds for expected and high probability bounds on the wake-up problem in the radio network model with collision detection. In terms of related work, a seminal paper by Willard [23] describes a wake-up algorithm (he called the problem "selection resolution," but the definition in this setting is functionally identical) which solves the problem in $O(\log\log n)$ rounds, in expectation. He also proved the result tight with an $\Omega(\log\log n)$ lower bound for uniform algorithms. As Willard himself admits, his lower bound proof is mathematically complex. Below, we significantly simplify this bound and generalize it to hold for all algorithms. From a high-probability perspective, many solutions exist in folklore for solving wake-up (and related problems) in $O(\log n)$ rounds. Indeed, leveraging collision detection, wake-up can be solved *deterministically* in $O(\log n)$ rounds (e.g., use the detector to allow the active nodes to move consistently through a binary search tree to identify the smallest active id). The necessity of $\Omega(\log n)$ rounds seems also to exist in folklore.

We begin with our high probability result. Our simulation strategy is more difficult to deploy here because the player must now somehow correctly simulate the collision detection among the nodes in the (unknown) target set T. To overcome this difficulty, we apply our solution to networks in which only two nodes are activated and then achieve a contradiction with our lower bound on *restricted* hitting. The details of this proof are deferred to the full version [22].

Theorem 6. *Let $\mathcal{A}$ be an algorithm that solves wake-up with high probability in $f(n)$ rounds in the radio network model with a single channel and collision detection. It follows that $f(n) \in \Omega(\log n)$.*

We now simplify and strengthen Willard's bound of $\Omega(\log\log n)$ rounds for expected time wake up. At the core of our result is a pleasingly simple but surprisingly useful observation: if you can solve wake-up in t rounds with collision detection, you can then use this strategy to solve the hitting game in 2^t rounds by simulating (carefully) all possible sequences of outcomes for the collision detector behavior in a t round execution. Solving the problem in $o(\log\log n)$ rounds (in expectation) with collision detection, therefore, yields a hitting game solution that requires only $2^{o(\log\log k)} = o(\log k)$ rounds (in expectation), contradicting Theorem 1—our lower bound on expected time solutions to the hitting game.

Theorem 7. *Let $\mathcal{A}$ be an algorithm that solves wake-up in $f(n)$ rounds, in expectation, in the radio network model with a single channel and collision detection. It follows that $f(n) \in \Omega(\log\log n)$.*

Proof. Fix some algorithm $\mathcal{A}$ that solves wake-up in $f(n)$ rounds, in expectation, in this setting. We start by defining a player $\mathcal{P}_{\mathcal{A}}$ that simulates $\mathcal{A}$ to solve k-hitting in no more than $2^{f(k)+1}$ rounds, in expectation. Our player will use a variant of the simulation strategy defined in Section 4 and used in the preceding proofs, and we will, therefore, adopt much of the terminology of this approach (with some minor modifications). In more detail, in this variant, $\mathcal{P}_{\mathcal{A}}$ will run a different fixed-length simulation of $\mathcal{A}$, starting from round 1, to generate each of its guesses in the hitting game. Most of these simulations will *not* be consistent with the relevant target execution. We will show, however, that in the case that the target execution solves wake-up, at least one such simulation is consistent and will therefore win the game.

In more detail, for a given k, let $B_{f(k)}$ be a full rooted binary tree of depth $f(k)$. We define a tree node labeling ℓ, such that for every non-root node u, $\ell(u) = 0$ if u is a left child of its parent and $\ell(u) = 1$ if u is a right child (by some consistent orientation). Let d be the depth function (i.e., $d(u)$ is the depth of u in the tree with $d(root) = 0$). Finally, let $p(u)$ return the $d(u)$-bit binary string defined by the sequence of labels (by ℓ) on the path that descends from the root to u (including u). For example, if the path from the root to u goes from the root to its right child v, then from v to its left child u, $p(u) = 10$.

Our player $\mathcal{P}_{\mathcal{A}}$, when playing the k-hitting game, generate one guess for each node in $B_{f(k)}$. Fix some such node u. To generate a guess for u, the player first executes a $d(u)$-round simulation of $\mathcal{A}$, running on all k nodes in a k-node network, using $p(u)$ to specify collision detector behavior (in a manner described below). After it simulates these $d(u)$ full rounds, it then simulates just enough of round $d(u) + 1$ to determine the simulated nodes' broadcast decisions in this round. The player proposes the id of the nodes that choose to broadcast in this final partial round. (When generating a guess for the root node, the player simply proposes the nodes that broadcast in the first round.)

In more detail, for each round $r \leq d(u)$ of the simulation for tree node u, if the r^{th} bit of $p(u)$ is 0, the player simulates all nodes detecting silence, and if the bit is 1, it simulates all nodes detecting a collision. As a final technicality, let κ be the random bits provided to the player to resolve its random choices. We assume that for each simulated node i, the players uses the same bits from κ for i in each of its simulations. We do not, therefore, assume independence between different simulations.

Consider the target execution of $\mathcal{A}$ for a given instance of the hitting game with target set T and random bits κ. Assume that the target execution defined for these bits and target set solves wake-up in some round $r \leq f(k)$. Notice that in every round $r' < r$, there are only two possible behaviors: (1) no nodes broadcast (and all nodes therefore receive and detect nothing); and (2) two or more nodes broadcast (and all nodes therefore detect a collision). By definition, there exists a node u in $B_{f(k)}$ such that $p(u)$ is a binary string of length $r - 1$, where for each bit position i in the string, $i = 0$ if no nodes broadcast in that round of the target execution, and $i = 1$ if two or more nodes broadcast in that round of the target execution. It follows that the first $r - 1$ rounds of the simulation associated with tree node u are consistent with the target execution. Because exactly one node from T broadcasts in round r of the target execution, and the u-simulation is consistent through round $r - 1$, then this same single node from T will broadcast in the simulated beginning of round r. The player's proposal associated with u will therefore win the hitting game.

Pulling together the pieces, by assumption, the target execution for a given T and κ solves wake-up in $f(k)$ rounds, in expectation. It follows that our player solves k-hitting with the same probability. The number of guesses required to solve the problem in this case is bounded by the number of nodes in $B_{f(k)}$ (as there is one guess per node), which is $2^{f(k)+1} - 1$. We can now conclude with our standard style of contradiction argument. Assume for contradiction that there exists an algorithm $\mathcal{A}$ that solves wake-up with a single channel and collision detection in $f(n) \in o(\log \log n)$ rounds, in expectation. It follows that $\mathcal{P}_{\mathcal{A}}$ wins the k-hitting game in $2^{f(k)+1} \in o(\log k)$ rounds, in expectation. This contradicts Theorem 1.

7 Lower Bounds for Wake-Up with Multiple Channels

In recent years, theoreticians have paid increasing attention to multichannel versions of the radio network model (e.g., [10,11,17,9,12,8,7]). These investigations are motivated by the reality that most network cards allow the device to choose its channel from among multiple available channels. From a theoretical perspective, the interesting question is how to leverage the parallelism inherent in multiple channels to improve time complexity for basic communication problems. Daum et al. [7], building on results from Dolev et al. [9], prove a lower bound of $\Omega\big(\frac{\log^2 n}{\mathcal{C} \log \log n} + \log n\big)$ rounds for solving wake-up with high probability and uniform algorithms in a network with $\mathcal{C}$ channels. A lower bound for expected-time solutions was left open. The best known upper bound solves the problem in $O\big(\frac{\log^2 n}{\mathcal{C}} + \log n\big)$ rounds with high probability and in $O\big(\frac{\log n}{\mathcal{C}} + 1\big)$ rounds in expectation [7].

In the theorems that follow, we prove new lower bounds that match the best known upper bounds. These bounds close the $\log \log n$ gap that exists with the best known previous results, establish the first non-trivial expected time bound, and strengthen the results to hold for all algorithms. To prove our high probability bound, both terms in the sum are tackled separately. To prove the first term, we show that a player can simulate an algorithm using $\mathcal{C}$ channels by making $\mathcal{C}$ proposals for each simulated round—one for each channel—to test if T has an isolated broadcast on any channel. The second term uses a reduction from the restricted hitting game. The expected time result adopts a similar strategy as the first term. The proofs for these theorems are deferred to the full version [22].

Theorem 8. *Let $\mathcal{A}$ be an algorithm that solves wake-up with high probability in $f(n, \mathcal{C})$ rounds in the radio network model with $\mathcal{C} \geq 1$ channels. It follows that for every $\mathcal{C} \geq 1$, $f(n, \mathcal{C}) \in \Omega(\log^2 n/\mathcal{C} + \log n)$.*

Theorem 9. *Let $\mathcal{A}$ be an algorithm that solves wake-up in $f(n, \mathcal{C})$ rounds, in expectation, in the radio network model with $\mathcal{C} \geq 1$ channels. It follows that for every $\mathcal{C} \geq 1$, $f(n, \mathcal{C}) \in \Omega(\log n/\mathcal{C} + 1)$.*

8 Lower Bound for Wake-Up with Collision Detection and Multiple Channels

The final combination of model parameters to consider for wake-up is collision detection *and* multiple channels. No non-trivial upper or lower bounds are currently known for this case. We rectify this omission by proving below that $\Omega(\log n/\log \mathcal{C} + \log\log n)$ rounds are necessary to solve this problem with high probability in this setting. Notice, this bound represents an interesting split with the preceding multichannel results (which assume no collision detection), as the speed-up is now logarithmic in $\mathcal{C}$ instead of linear. On the other hand, the $\log^2 n$ term in the previous case is replaced here with a faster $\log n$ term. Collision detection, in other words, seems to be powerful enough on its own that adding extra channels does not yield much extra complexity gains. We do not consider an expected time result for this setting. This is because even *without* collision detection, the best known upper bound for multichannel networks [7] approaches $O(1)$ time (which is trivially optimal) quickly as the number of channels increases. The proof for the below theorem, which combines techniques from both Section 6 and Section 7, is deferred to the full version[22].

Theorem 10. *Let $\mathcal{A}$ be an algorithm that solves wake-up with high probability in $f(n, \mathcal{C})$ rounds in the radio network model with $\mathcal{C} \geq 1$ channels and collision detection. It follows that for every $\mathcal{C} \geq 1$, $f(n, \mathcal{C}) \in \Omega(\log n/\log \mathcal{C} + \log\log n)$.*

9 Lower Bound for Global Broadcast

We now turn our attention to proving a lower bound for global broadcast. The tight bound for this problem is $\Theta(D\log(n/D) + \log^2 n)$ rounds for a connected multihop network of size n with diameter D. The lower bound holds for expected time solutions and the matching upper bounds hold with high probability [3,19,6]. The $\log^2 n$ term was established in [1], where it was shown to hold even for centralized algorithms, and the $D\log(n/D)$ term was later proved by Kushilevitz and Mansour [20]. Below, we apply our new technique to reprove (and significantly simplify) the $\Omega(D\log(n/D))$ lower bound for expected time solutions to global broadcast. (We do not also reprove the $\Omega(\log^2 n)$ term because this bound is proved using the same combinatorial result from [1] that provides the mathematical foundation for our technique. To reprove the result of [1] using [1] is needlessly circular.)

Perhaps surprisingly, we show that this bound holds even if we allow multiple channels and collision detection, both of which are assumptions that break the original lower bound from [20]. Notice, this indicates a interesting split with the wake-up problem for which these assumptions *improve* the achievable time complexity.

It is important to remind the reader at this point that the definition of collision detection we consider in this paper *does not* allow a collision to activate a node. Instead, activation still requires that a node receive a message. Once activated, however, nodes can use collision detection to speed up or otherwise simplify contention management. The assumption that collisions *can* activate nodes (essentially) reduces the problem to the less well-studied *synchronous start* variation in which all nodes activate in round 1 (if collisions can activate nodes then the source can instigate a wave of collisions that activates the entire network quickly). Recent work solved the synchronous start broadcast problem in $O(D+\text{polylog}(n))$ rounds using collision detection [16]. The problem's complexity without collision detection remains open.

Returning to our result, the proof details for the theorem below are deferred to the full version of this paper [22]. The intuition, however, is straightforward to describe. Given n nodes, we can construct a network consisting of D ordered *layers* each containing n/D nodes. Imagine that only a subset of the nodes in each layer are connected to the next layer. The only way to advance the message from one layer to the next, therefore, is to isolate a single node from this unknown set of connected nodes. Accordingly, it is not hard to reduce our hitting game to this task, reducing the challenge of broadcast to solving D sequential instances of the (n/D)-hitting game, where each instance requires $\Omega(\log(n/D))$ rounds.

Theorem 11. *Let $\mathcal{A}$ be an algorithm that solves global broadcast in $f(n,\mathcal{C},D)$ rounds, in expectation, in the radio network model with collision detection, $\mathcal{C} \geq 1$ channels, and a network topology with diameter D. It follows that for every $\mathcal{C}, D \geq 1$, $f(n,\mathcal{C},D) \in \Omega(D\log(n/D))$.*

Acknowledgments. The author acknowledges Mohsen Ghaffari for his helpful conversations regarding the combinatorial results from [1,15]. The author also acknowledges Sebastian Daum and Fabian Kuhn for their feedback on early applications of this technique to the wake-up problem.

References

1. Alon, N., Bar-Noy, A., Linial, N., Peleg, D.: A Lower Bound for Radio Broadcast. Journal of Computer and System Sciences 43(2), 290–298 (1991)
2. Bar-Yehuda, R., Goldreigch, O., Itai, A.: On the Time-Complexity of Broadcast in Multi-Hop Radio Networks: An Exponential Gap between Determinism and Randomization. In: Proceedings of the ACM Conference on Distributed Computing (1987)
3. Bar-Yehuda, R., Goldreigch, O., Itai, A.: 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. Chlamtac, I., Kutten, S.: On Broadcasting in Radio Networks–Problem Analysis and Protocol Design. IEEE Transactions on Communications 33(12), 1240–1246 (1985)
5. Clementi, A.E.F., Monti, A., Silvestri, R.: Distributed Broadcast in Radio Networks of Unknown Topology. Theoretical Computer Science 302(1-3) (2003)
6. Czumaj, A., Rytter, W.: Broadcasting algorithms in radio networks with unknown topology. Journal of Algorithms 60, 115–143 (2006)

7. Daum, S., Gilbert, S., Kuhn, F., Newport, C.: Leader Election in Shared Spectrum Radio Networks. In: Proceedings of the ACM Conference on Distributed Computing. ACM (2012)
8. Daum, S., Kuhn, F., Newport, C.: Efficient Symmetry Breaking in Multi-Channel Radio Networks. In: Aguilera, M.K. (ed.) DISC 2012. LNCS, vol. 7611, pp. 238–252. Springer, Heidelberg (2012)
9. Dolev, S., Gilbert, S., Guerraoui, R., Kuhn, F., Newport, C.: The Wireless Synchronization Problem. In: Proceedings of the ACM Conference on Distributed Computing (2009)
10. Dolev, S., Gilbert, S., Guerraoui, R., Newport, C.: Gossiping in a Multi-channel Radio Network. In: Pelc, A. (ed.) DISC 2007. LNCS, vol. 4731, pp. 208–222. Springer, Heidelberg (2007)
11. Dolev, S., Gilbert, S., Guerraoui, R., Newport, C.: Secure Communication Over Radio Channels. In: Proceedings of the ACM Conference on Distributed Computing (2008)
12. Dolev, S., Gilbert, S., Khabbazian, M., Newport, C.: Leveraging Channel Diversity to Gain Efficiency and Robustness for Wireless Broadcast. In: Proceedings of the ACM Conference on Distributed Computing (2011)
13. Farach-Colton, M., Fernandes, R.J., Mosteiro, M.A.: Lower Bounds for Clear Transmissions in Radio Networks. In: Correa, J.R., Hevia, A., Kiwi, M. (eds.) LATIN 2006. LNCS, vol. 3887, pp. 447–454. Springer, Heidelberg (2006)
14. Ghaffari, M., Haeupler, B.: Near Optimal Leader Election in Multi-Hop Radio Networks (2013)
15. Ghaffari, M., Haeupler, B., Khabbazian, M.: A Bound on the Throughput of Radio Networks. CoRR (ArXiv), abs/1302.0264 (February 2013)
16. Ghaffari, M., Haeupler, B., Khabbazian, M.: Randomized Broadcast in Radio Networks with Collision Detection. In: Proceedings of the ACM Conference on Distributed Computing (2013)
17. Gilbert, S., Guerraoui, R., Kowalski, D., Newport, C.: Interference-Resilient Information Exchange. In: Proceedings of the IEEE International Conference on Computer Communications (2009)
18. Jurdziński, T., Stachowiak, G.: Probabilistic Algorithms for the Wakeup Problem in Single-Hop Radio Networks. In: Bose, P., Morin, P. (eds.) ISAAC 2002. LNCS, vol. 2518, pp. 535–549. Springer, Heidelberg (2002)
19. Kowalski, D., Pelc, A.: Broadcasting in Undirected Ad Hoc Radio Networks. Distributed Computing 18(1), 43–57 (2005)
20. Kushilevitz, E., Mansour, Y.: An Ω(D\log(N/D)) Lower Bound for Broadcast in Radio Networks. SIAM Journal on Computing 27(3), 702–712 (1998)
21. Moscibroda, T., Wattenhofer, R.: Maximal Independent Sets in Radio Networks. In: Proceedings of the ACM Conference on Distributed Computing (2005)
22. Newport, C.: Radio Network Lower Bounds Made Easy. CoRR (ArXiv), abs/1405.7300 (May 2014)
23. Willard, D.: Log-Logarithmic Selection Resolution Protocols in a Multiple Access Channel. SIAM Journal on Computing 15(2), 468–477 (1986)