Information Gathering in Ad-Hoc Radio Networks with Tree Topology

Marek Chrobak
Department of Computer Science, University of California at Riverside, USA.

Kevin P. Costello
Department of Mathematics, University of California at Riverside, USA.

Leszek Gasieniec
Department of Computer Science, University of Liverpool, UK.

Dariusz R. Kowalski
Department of Computer Science, University of Liverpool, UK.

Abstract

We study the problem of information gathering in ad-hoc radio networks without collision detection, focusing on the case when the network forms a tree, with edges directed towards the root. Initially, each node has a piece of information that we refer to as a rumor. Our goal is to design protocols that deliver all rumors to the root of the tree as quickly as possible. The protocol must complete this task even though the actual tree topology is unknown when the computation starts. In the deterministic case, assuming that the nodes are labeled with small integers, we give an $O(n)$-time protocol for the model with unbounded message size, and an $O(n \log n)$-time protocol for the model with bounded message size, where any message can include only one rumor. We also consider fire-and-forward protocols, in which a node can only transmit its own rumor or the rumor received in the previous step. We give a deterministic fire-and-forward protocol with running time $O(n^{1.5})$, and we show that it is asymptotically optimal. Our last two results are concerned with randomized algorithms. We present a randomized $O(n \log n)$-time protocol in the model without node labels and without aggregation, and we prove that this bound is asymptotically optimal.

1 Introduction

We consider the problem of *information gathering* in ad-hoc radio networks, where initially each node has a piece of information called a *rumor*, and all these rumors need to be delivered to a designated target node as quickly as possible. A radio network is defined as a directed graph G with n vertices. At each time step any node v of G may attempt to transmit a message. This message is sent immediately to all out-neighbors of v. However, an out-neighbor u of v will receive this message only if u is in the receiving state and no in-neighbor of u other than v attempted to transmit at the same step. The event when two or more in-neighbors of u transmit at the same time is called a *collision*. We do not assume any collision detection mechanism; in other words, not only this u will not receive any message in this time step, but it will not even know that a collision occurred.

One other crucial feature of our model is that the topology of G is not known at the beginning of computation. We are interested in distributed protocols, where the execution of a protocol at a node v depends only on the identifier (label) of v and the information gathered from the received messages. Randomized protocols typically do not use node labels, and thus they work even if the nodes are indistinguishable from each other. The protocol needs to complete its task independent of the topology of G.

Several primitives for information dissemination in ad-hoc radio networks have been considered in the literature. Among these, the two most extensively studied are *broadcasting* and *gossiping*.

The *broadcasting problem* is the one-to-all dissemination problem, where initially one source node has a rumor that needs to be delivered to all nodes in the network (assuming that all other nodes are reachable from the source node). In the model where the nodes of G are labelled with consecutive integers $0, 1, ..., n-1$, the fastest known deterministic algorithm for broadcasting runs in time $O(n \log D \log\log(D\Delta/n))$ [16], where D is the diameter of G and Δ is the maximum in-degree. The best lower bound on the running time in this model is $\Omega(n \log D)$ [15]. (See also [12, 28, 6, 7, 18, 19, 17] for earlier work.) Allowing randomization, broadcasting can be accomplished in time $O(D \log(n/D) + \log^2 n)$ with high probability [17], even if the nodes are not labelled. This matches the lower bounds in [2, 29].

The *gossiping problem* is the all-to-all dissemination problem. Here, each node starts with its own rumor and the goal is to deliver all rumors to each node. Naturally, for gossiping to be well defined, the underlying graph needs to be strongly connected. There is no restriction on the size of messages; in particular, different rumors can be transmitted together in a single message. (Models with restrictions on message size have also been studied, see [24, 10].) With randomization, gossiping can be solved in expected time $O(n \log^2 n)$ [17] (see [30, 13] for earlier work), even if the nodes are not labelled. In contrast, for deterministic algorithms, with nodes labelled $0, 1, ..., n-1$, the fastest known gossiping algorithm runs in time $O(n^{4/3} \log^4 n)$ [25], following earlier progress in [12, 43]. (See also the survey in [23] for more information.) It is quite obvious that no algorithm, deterministic or randomized, can achieve gossiping faster than in time $\Omega(n)$ for arbitrary graphs. To the best of our knowledge, and contrary to folklore, no better lower bounds have been yet established in the literature. (To clarify, we point out that the $\Omega(n \log n)$ lower bound proofs for broadcasting in [15, 7] do not apply to gossiping, because they either use acyclic graphs or assume that nodes are inactive until they receive the broadcast message. The lower bound in [8] for many-to-many communication does not apply for the same reason. The lower bound in [24] is for the model with restricted message size.) Reducing the gap between lower and upper bounds for deterministic gossiping to a poly-logarithmic factor remains a central open problem in the study of radio networks with unknown topology.

Our work has been inspired by this open problem. It is easy to see that for strongly connected directed graphs gossiping is equivalent to information gathering, in the following sense. On one hand, trivially, any protocol for gossiping also solves the problem of gathering. On the other hand, we can apply a gathering protocol and follow it with a protocol that broadcasts all information from the target node r; these two protocols combined solve the problem of gossiping. So if we can solve information gathering in time $O(n \operatorname{polylog}(n))$, then we can also solve gossiping in time $O(n \operatorname{polylog}(n))$. With this observation in mind, we can consider information gathering to be an *extension* of gossiping, because information gathering is well defined for a broader class of graphs, namely all graphs where the target node is reachable from all other nodes.

Our results. To gain better insight into the problem of information gathering in radio networks, we focus on networks with tree topology. Thus we assume that our graph is a tree $\mathcal{T}$ with root r and with all edges directed towards r. In this model, a gathering protocol knows that the network is a tree, but it does not know its topology.

We consider several variants of this problem, for deterministic or randomized algorithms, and with or without restrictions on the message size or processor memory. We provide the following results:

- In the first part of the paper we study deterministic algorithms, under the assumption that the nodes of $\mathcal{T}$ are labelled $0, 1, ..., n-1$. First, in Section 4, we examine the model without any bound on the message size. In particular, protocols for this model are allowed to aggregate any number of rumors into a single message. (This is a standard model in existing gossiping protocols in unknown radio networks; see, for example, the survey in [23]). We give an optimal, $O(n)$-time protocol using unbounded messages.

- Next, in Section 5, we consider the model with bounded messages, where a message may contain only one rumor. For this model we provide an algorithm with running time $O(n \log n)$.

- In Section 6 we consider an even more restrictive model of protocols with bounded messages, that we call fire-and-forward protocols. In those protocols, at each step, a node can only transmit either its own rumor or the rumor received in the previous step (if any). Fire-and-forward protocols are very simple to implement and require very little memory, since nodes do not need to store any received messages. (We remark that our model of fire-and-forward protocols resembles hot-potato packet routing in networks that has been well studied; see, for example [5, 22]). For the fire-and-forward model we provide a deterministic protocol with running time $O(n^{1.5})$ and we show a matching lower bound of $\Omega(n^{1.5})$.

- We then turn our attention to randomized algorithms (Sections 7 and 8). For randomized algorithms we assume that the nodes are not labelled. In this model, we give an $O(n \log n)$-time gathering protocol and we prove a matching lower bound of $\Omega(n \log n)$. The upper bound is achieved by a simple fire-and-forward protocol that, in essence, reduces the problem to the Coupon Collector's Problem. Our main contribution here is the lower bound proof.

We add that our last result (the lower bound result for randomized algorithms) is in fact stronger, as we show that the optimum expected and high-probability running time to achieve information gathering in star graphs (trees of depth 1) is $(c \pm o(1))n \ln n$, where $c = 1/\ln^2 2 \approx 2.08$. This can be also interpreted as a tight bound for randomized contention resolution in multiple-access channels without feedback (see [26, 31, 44] for some related work.)

Other related work. Prior to this paper, there does not seem to be any work in the literature that directly addresses the information gathering problem in our model of ad-hoc radio networks. Some communication protocols for radio networks use forms of information gathering on trees as a sub-routine; see for example [4, 8, 27]. However, these solutions typically focus on undirected graphs, which allow direct feedback. They also solve relaxed variants of information gathering where the goal is to gather only a fraction of rumors in the root, which was sufficient for the applications studied in these papers (since, with feedback, such a procedure can be repeated until all rumors are collected). In contrast, in our work, we study directed trees without any feedback mechanism, and we require all rumors to be collected at the root.

Variants of information gathering problems have been studied, however, in other network models. For example, Onus *et al.* [33] consider information gathering in an ad-hoc network of wireless devices that are located in the plane. Issues related to information gathering in sensor networks are addressed in [9, 35]. Online competitive algorithms for information gathering (in a very different network model) were studied in [3].

In our ad-hoc radio network model we are assuming that each node has a unique label from $0, 1, ..., n-1$. There is some work on information dissemination in ad-hoc radio networks where this assumption is relaxed, allowing the labels to be chosen from a larger domain. For example, the $O(n^{4/3} \log^4 n)$-time gossiping algorithm of Gasieniec *et al.* [25] works even if the labels are drawn from a polynomial size domain $\{0, 1, ..., n^b\}$, for some constant $b \geq 1$.

2 Preliminaries

Radio networks. In this section we give formal definitions of radio networks and gathering protocols. We define a radio network as a directed graph $G = (V, E)$ with n nodes, with each node assigned a different label from the set $[n] = \{0, 1, ..., n-1\}$. (All the results in the paper remain valid if the label range is $[L]$, where $L \geq n$ and $L = O(n)$.) Denote by $\mathsf{label}(v)$ the label assigned to a node $v \in V$. One node r is distinguished as the *target* node, and we assume that r is reachable from all other nodes. Initially, at time 0, each node v has some piece of information that we will refer to as *rumor* and we will denote it by ρ_v. The objective is to deliver all rumors ρ_v to r as quickly as possible, according to the rules described below.

The time is discrete, namely it consists of time steps numbered with non-negative integers $0, 1, 2, \ldots$. At any step, a node v may be either in the *transmit state* or the *receive state*. A gathering protocol $\mathcal{A}$ determines, for each node v and each time step t, whether v is in the transmit or receive state at time t. If v is in the transmitting state, then $\mathcal{A}$ also determines what message is transmitted by v, if any. This specification of $\mathcal{A}$ may depend only on the label of v, time t, and on the content of all messages received by v until time t. We stress that, with these restrictions, $\mathcal{A}$ does not depend on the topology of G nor on the node labeling.

All nodes start executing the protocol simultaneously at time 0. If a node v transmits at a time t, the transmitted message is sent immediately to all out-neighbors of v, that is to all u such that (v, u) is an edge. If (v, u) and (v', u) are edges and both v, v' transmit at time t then a *collision* at u occurs and u does not receive any message. More specifically, an out-neighbor u of v will receive v's message if and only if (i) u is in the receive state at time t, and (ii) no collision at u occurs at time t. We do not assume any feedback from the transmission channel or any collision detection features, so, in case of a collision, neither the sender nor any node within its range knows that a collision occurred.

Throughout the paper, we will focus on the case in which the graph is a rooted tree, with all edges directed towards the root. We will typically use notation $\mathcal{T}$ for this tree and r for its root.

The running time of a deterministic information gathering protocol $\mathcal{A}$ is defined as the minimum time $T(n)$ such that, for any tree $\mathcal{T}$ with root r and n nodes, any assignment of labels from $[n]$ to the nodes of $\mathcal{T}$, and any node v, the rumor ρ_v of v is delivered to r no later than at step $T(n)$. In case of randomized protocols, we evaluate them either using the expectation of their running time $T(n)$, which is now a random variable, or by showing that $T(n)$ does not exceed a desired time bound with high probability.

We consider three types of information gathering protocols:

Unbounded messages: In this model a node can transmit or receive arbitrary information in a single step. In particular, multiple rumors can be aggregated into a single message.

Bounded messages: In this model no aggregation of rumors is allowed. Each message consists of at most one rumor and $O(\log n)$ bits of additional information.

Fire-and-forward: In a fire-and-forward protocol, each message consists of just one rumor, without any other information. If a node v received a message in the previous step, then in the current step v must either transmit (forward) this message, or it can transmit its own rumor (that is, "fire"), or else it can enter the receiving state. Thus each rumor ρ_x, after being fired from its origin node x, travels towards the root one hop at a time, until either it vanishes or it successfully reaches the root. A rumor can vanish when it collides with another rumor, when it is transmitted to a node that is not in the receiving state, or when it is dropped by a node that decided to fire its own rumor.

For illustration, consider a protocol called RoundRobin, where all nodes transmit in a cyclic order, one at a time. Specifically, at any step t, RoundRobin transmits from the node v with $\mathsf{label}(v) = t \bmod n$, with the transmitted message containing all rumors received by v until time t. The running time is $O(n^2)$, because initially each rumor ρ_u is at distance at most $n-1$ from the root and in any consecutive n steps it will decrease its distance to the root by at least 1. (RoundRobin has been used as a subroutine in many protocols in the literature, and it achieves running time $O(n^2)$ for information gathering or gossiping in arbitrary graphs where these problems are well-defined.)

RoundRobin can be adapted to use only bounded messages for information gathering in trees. At any round t and any node v, if v has the rumor ρ_u of node u such that $\mathsf{label}(u) = t \bmod n$, and v has not transmitted ρ_u before, then v transmits ρ_u at time t. Since $\mathcal{T}$ is a tree, each rumor follows the unique path towards the root. Therefore any two sibling nodes will never transmit at the same time, that is, collisions will never occur. This implies that after at most n^2 steps r will receive all rumors. (Later in the paper, we refer to this method as *rumor-based* RoundRobin.)

3 Some Structure Properties of Trees

As we will show later in Sections 4 and 5, the efficiency of our algorithms depends on some subtle topological property of trees — roughly, on how "bushy" the tree is. To capture this property we use the concept of γ-depth. We define this concept in this section and establish its properties needed for the analysis of our algorithms.

γ-Depth of trees. Let $\mathcal{T}$ be the given tree network with root r and n nodes. We use standard terminology for rooted trees: children, descendants, ancestors, and leaves. By the *degree* of a node $v \in \mathcal{T}$ we mean the number of its children. $\mathcal{T}_v$ will denote the subtree of $\mathcal{T}$ rooted at v that contains all descendants of v, including v itself.

Fix an integer γ in the range $1 \le \gamma \le n-1$. We define the *γ-height* of a node v of $\mathcal{T}$, denoted $\mathit{height}_\gamma(v)$, as follows. If v is a leaf then $\mathit{height}_\gamma(v) = 0$. If v is an internal node then let g be the maximum γ-height of a child of v. If at least γ children of v have γ-height equal g then $\mathit{height}_\gamma(v) = g+1$; otherwise $\mathit{height}_\gamma(v) = g$. We then define the *$\gamma$-depth of $\mathcal{T}$* as $D_\gamma(\mathcal{T}) = \mathit{height}_\gamma(r)$.

In the analysis of our algorithms the nodes which contribute to the increase of γ-height will play a special role — they are called γ-frontier nodes. Formally, v is called a *γ-frontier node* if its γ-height is greater than

γ-heights of all its children. The γ-frontier nodes with γ-height 0 are (vacuously, from the definition) the leaves of $\mathcal{T}$.

For any nodes v and $u \in \mathcal{T}_v$, we call u a *(γ, h)-tributary descendant of v* if all the nodes on the path from u to v, including u but excluding v, have γ-height at most h. If $height_\gamma(v) \leq h$ or if v is a frontier node with $height_\gamma(v) = h+1$ then all descendants of v are, by definition, (γ, h)-tributary. (See Figure 1.) This concept is used in Section 5.

In our proofs, we may also consider trees other than the input tree $\mathcal{T}$. If $\mathcal{H}$ is any tree and v is a node of $\mathcal{H}$ then, to avoid ambiguity, we will write $height_\gamma(v, \mathcal{H})$ for the γ-height of v with respect to $\mathcal{H}$. Note that if $\mathcal{H}$ is a subtree of $\mathcal{T}$ and $v \in \mathcal{H}$ then, trivially, $height_\gamma(v, \mathcal{H}) \leq height_\gamma(v)$.

By definition, the 1-height of a node is the same as its height, namely the longest distance from this node to a leaf in its subtree. For a tree, its 1-depth is equal to its depth. Figure 1 shows an example of a tree whose depth equals 4, 2-depth equals 3, and 3-depth equals 1.

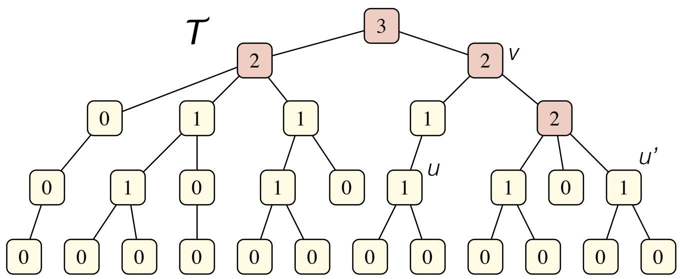

Figure 1: An example illustrating the concept of γ-depth of trees, for $\gamma = 1, 2, 3$. The depth of this tree $\mathcal{T}$ is 4. The number in each node is its 2-height; thus the 2-depth of this tree is 3. All light-shaded nodes have 3-height equal 0 and the four dark-shaded nodes have 3-height equal 1, so the 3-depth of this tree is 1. For $\gamma = 2$, u is a $(2, 1)$-tributary descendant of v but u' is not.

The lemmas below spell out some simple properties of γ-heights of nodes that will be useful for the analysis of our algorithms.

Lemma 1. *$D_\gamma(\mathcal{T}) \leq \log_\gamma n$.*

Proof. It is sufficient to show that $|\mathcal{T}_v| \geq \gamma^{height_\gamma(v)}$ holds for each node v. The proof is by simple induction with respect to the height of v.

If v is a leaf then the inequality is trivial. So suppose now that v is an internal node with $height_\gamma(v) = g$. If v has a child u with $height_\gamma(u) = g$ then, by induction, $|\mathcal{T}_v| \geq |\mathcal{T}_u| \geq \gamma^g$. If all children of v have γ-height smaller than g then $g \geq 1$ and v must have at least γ children with γ-height equal $g - 1$. So, by induction, we get $|\mathcal{T}_v| \geq \gamma \cdot \gamma^{g-1} = \gamma^g$. $\square$

The concept of γ-heights was used earlier in [14], independently and under a different name, to develop some radio network protocols, based on a variant of Lemma 1. (We include a simple proof above for the sake of completeness.) For $\gamma = 2$, the definition of 2-height is equivalent to that of Strahler numbers [36], used in hydrology to measure the complexity of watersheds (see also [38]). In particular, Lemma 1 generalizes the bound from [38] for $\gamma = 2$.

We will be particularly interested in subtrees of $\mathcal{T}$ consisting of the nodes whose γ-height is above a given threshold. Specifically, for $h = 0, 1, ..., D_\gamma(\mathcal{T})$, let $\mathcal{T}^{\gamma,h}$ be the subtree of $\mathcal{T}$ induced by the nodes whose γ-height is at least h (see Figure 2). Note that, since γ-heights are monotonically non-decreasing on the paths from leaves to r, $\mathcal{T}^{\gamma,h}$ is indeed a subtree of $\mathcal{T}$ rooted at r. In particular, for $h = 0$ we have $\mathcal{T}^{\gamma,0} = \mathcal{T}$. The leaves of $\mathcal{T}^{\gamma,h}$ are exactly the γ-frontier nodes with γ-height equal h.

For any h, $\mathcal{T} - \mathcal{T}^{\gamma,h}$ is a collection of subtrees of type $\mathcal{T}_v$, where v is a node of γ-height less than h whose parent is in $\mathcal{T}^{\gamma,h}$. When $h = 1$, all such subtrees contain only nodes of γ-height equal 0, which implies that they all have degree less than γ. In particular, for $\gamma = 2$, each such subtree $\mathcal{T}_v$ is a path from a leaf of $\mathcal{T}$ to v.

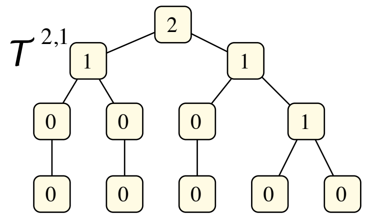

Figure 2: The subtree $\mathcal{T}^{2,1}$ obtained from tree $\mathcal{T}$ in Figure 1. The numbers in the nodes are their 2-heights with respect to $\mathcal{T}^{2,1}$.

Lemma 2. *For any node $v \in \mathcal{T}^{\gamma,h}$ we have $height_\gamma(v, \mathcal{T}^{\gamma,h}) = height_\gamma(v) - h$. Thus, in particular, we also have $D_\gamma(\mathcal{T}^{\gamma,h}) = D_\gamma(\mathcal{T}) - h$.*

Proof. Let $\mathcal{T}' = \mathcal{T}^{\gamma,h}$. The proof is by induction on the height of v in $\mathcal{T}'$. If v is a leaf of $\mathcal{T}'$ then $height_\gamma(v) \geq h$ and $height_\gamma(v, \mathcal{T}') = 0$, by definition. All children of v in $\mathcal{T}$ are outside $\mathcal{T}'$ so their γ-heights are at most $h - 1$. Therefore $height_\gamma(v) = h$ and thus the lemma holds for v.

Suppose now that v is not a leaf of $\mathcal{T}'$ and that the lemma holds for all children of v. This means that each child u of v in $\mathcal{T}$ is of one of two types: either $u \notin \mathcal{T}'$ (that is, $height_\gamma(u) \leq h - 1$), or $u \in \mathcal{T}'$ (that is, $height_\gamma(u) \geq h$) and $height_\gamma(u, \mathcal{T}') = height_\gamma(u) - h$.

Let $height_\gamma(v) = f \geq h$. If v has a child with γ-height equal f then there are fewer than γ such children. By induction, these children will have γ-height in $\mathcal{T}'$ equal $f - h$, and each other child that remains in $\mathcal{T}'$ has γ-height in $\mathcal{T}'$ smaller than $f - h$. So $height_\gamma(v, \mathcal{T}') = f - h$.

If all children of v have γ-height smaller than f then $f \geq h + 1$ (for otherwise v would have to be a leaf of $\mathcal{T}'$) and v must have at least γ children with γ-height $f - 1$. These children will be in $\mathcal{T}'$ and will have γ-height in $\mathcal{T}'$ equal $f - 1 - h$, by induction. So $height_\gamma(v, \mathcal{T}') = f - h$ in this case as well, completing the proof. $\square$

From Lemma 2, we obtain that the operation of taking subtrees $\mathcal{T}^{\gamma,h}$ is, in a sense, transitive.

Corollary 1. *For any $g, h \geq 0$ such that $h + g \leq D_\gamma(\mathcal{T})$, we have $(\mathcal{T}^{\gamma,h})^{\gamma,g} = \mathcal{T}^{\gamma,h+g}$.*

4 Deterministic Algorithms with Aggregation

As explained in Section 2, it is easy to achieve information gathering in time $O(n^2)$. In this section we prove that using unbounded-size messages this running time can be improved to $O(n)$. This is optimal for arbitrary n-node trees, since no better time can be achieved even for paths or star graphs. We start with a simpler protocol with running time $O(n \log n)$, that we use to introduce our terminology and techniques.

We can make some assumptions about the protocols in this section that will simplify their presentation. Since we use unbounded messages, we can assume that each transmitted message contains all information received by the transmitting node, including all received rumors. We also assume that all rumors are different, so that each node can keep track of the number of rumors collected from its subtree. To ensure this we can, for example, have each node v append its label to its rumor ρ_v.

We will also assume that each node knows the labels of its children. To acquire this knowledge, we can precede any protocol by a preprocessing phase where nodes with labels $0, 1, ..., n-1$ transmit, one at a time, in this order. Thus after n steps each node will receive the messages from its children. This does not affect the asymptotic running times of our protocols.

4.1 Warmup: a Simple $O(n \log n)$-Time Algorithm

We now present an algorithm for information gathering on trees that runs in time $O(n \log n)$. In essence, any node waits until it receives the messages from its children, then for $2n$ steps it alternates RoundRobin steps with steps when it always attempts to transmit. A more detailed specification of the algorithm follows.

Algorithm UnbDTree1**.** We divide the time steps into *rounds*, where round $s = 0, 1, 2, ...$ consists of two consecutive steps $2s$ and $2s + 1$, which we call, respectively, the RR-step and the All-step of round s.

For each node v we define its *activation round*, denoted α_v, as follows. If v is a leaf then $\alpha_v = 0$. For any other node v, α_v is the first round such that v has received messages from all its children when this round is about to start.

For each round $s = \alpha_v, \alpha_v + 1, ..., \alpha_v + n - 1$, v transmits in the All-step of round s, and if $\mathsf{label}(v) = s \bmod n$ then it also transmits in the RR-step of round s. In all other steps, v stays in the receiving state.

Analysis. For any node v we say that v is *dormant* in rounds $0, 1, ..., \alpha_v - 1$, v is *active* in rounds $\alpha_v, \alpha_v + 1, ..., \alpha_v + n - 1$, and that v is *retired* in every round thereafter. Since v will make one RR-transmission when it is active, v will successfully transmit its message to its parent before retiring and before this parent is activated. By a simple inductive argument, the message transmitted by v contains all rumors from its subtree $\mathcal{T}_v$. This implies that Algorithm UnbDTree1 is correct, namely that eventually r will receive all rumors from $\mathcal{T}$. This will happen, in fact, in at most $O(n^2)$ steps. Further, we have that at any round Algorithm UnbDTree1 satisfies the following invariants:

- (J1) Any path in $\mathcal{T}$ from a leaf to r consists of a segment of retired nodes, followed by a segment of active nodes, which is then followed by a segment of dormant nodes (each of these segments possibly empty).
- (J2) Each rumor has reached a node that is dormant or active.
- (J3) Any dormant node has at least one active descendant.

We now give a more accurate analysis of the running time of Algorithm UnbDTree1, starting with the lemma below which provides a bound on the activation time α_v of each node v.

Lemma 3. *Let $d = D_2(\mathcal{T})$. For any $h = 0, 1, ..., d$ and any node v with height$_2(v) = h$, we have that*

- (i) *if v is a 2-frontier node then $\alpha_v \leq 2hn$; otherwise,*
- (ii) $\alpha_v \leq (2h + 1)n$.

Proof. The proof is by induction on h — although with a twist. In the base case we observe that, directly from the description of Algorithm UnbDTree1, condition (i) holds for 2-frontier nodes with 2-height equal 0, that is for the leaves of $\mathcal{T}$.

In the inductive step, fix some $h \geq 0$, and assume that Lemma 3 holds for all nodes with 2-height smaller than h, and that, additionally, condition (i) holds for the 2-frontier nodes with 2-height equal h. We then show that Lemma 3 holds for all nodes with 2-height equal h, and that condition (i) holds for the 2-frontier nodes with 2-height equal $h + 1$ (as long as $h < d$). This will be sufficient to prove the lemma.

Given the inductive assumption stated above, we proceed as follows. Consider first a node v with *height*$_2(v) = h$ that is not a 2-frontier node. In order to reduce clutter, denote $\mathcal{Z} = \mathcal{T}^{2,h}$. From Lemma 2, we have that *height*$_2(v, \mathcal{Z}) = 0$, which implies that $\mathcal{Z}_v$ is a path from a leaf of $\mathcal{Z}$ to v. Let $\mathcal{Z}_v = v_1, v_2, ..., v_q$ be this path, where v_1 is a leaf of $\mathcal{Z}$ (that is, a 2-frontier node with 2-height h) and $v_q = v$.

Assume that v is still dormant at step $2nh$, for otherwise (ii) would follow trivially from $\alpha_v \leq 2hn \leq (2h + 1)n$. The nodes in $\mathcal{T} - \mathcal{Z}$ have 2-height smaller than h, so, by the inductive assumption, they are activated no later than in round $(2h - 1)n$, and therefore in round $2hn$ they are already retired. Thus (from invariant (J3) applied to v) some node on $\mathcal{Z}_v$ other than v is active in round $2nh$. Choose the largest p for which v_p is active. In round $2nh$ and later, all children of the nodes $v_p, v_{p+1}, ..., v_q$ that are not on $\mathcal{Z}_v$ do not transmit, because they are already retired. This implies that for each $\ell = 0, ..., q - p - 1$, node $v_{p+\ell+1}$ will get activated in round $2nh + \ell + 1$ as a result of the All-transmission from node $v_{p+\ell}$. In particular, we obtain that $\alpha_v \leq 2nh + q - p \leq (2h + 1)n$, as needed.

Next, suppose that $h < d$ and that v is a 2-frontier node with 2-height equal $h+1$. If u is a child of v then $height_2(u) \le h$, so $\alpha_u \le (2h+1)n$, by the inductive assumption (if $height_2(u) < h$) and by the proof of (ii) above (if $height_2(u) = h$). In one of the rounds $\alpha_u, \alpha_u+1, ..., \alpha_u+n-1$, node u will transmit all by itself, which guarantees that v will receive the message from u before round $\alpha_u + n \le 2(h+1)n$. Since this holds for each child of v, we conclude that $\alpha_v \le 2(h+1)n$, completing the proof. □

We have $height_2(r) = d$ and $d = O(\log n)$, by Lemma 1. Applying Lemma 3, this implies that $\alpha_r \le (2d+1)n = O(n \log n)$, which gives us that the overall running time is $O(n \log n)$. Summarizing, we obtain the following theorem.

Theorem 1. *For any tree with n nodes and any assignment of labels, Algorithm* UnbDTree1 *completes information gathering in time $O(n \log n)$.*

4.2 An $O(n)$-Time Deterministic Algorithm

In this section we show that, for the model with unbounded size messages, the running time of information gathering in trees can be improved to $O(n)$, which is of course optimal. The basic idea is to use (n,k)-strongly-selective families to speed up the computation.

Recall that an (n,k)-*strongly-selective family*, where $1 \le k \le n$, is a collection $F_0, F_1, ..., F_{m-1} \subseteq [n]$ of sets such that for any set $X \subseteq [n]$ with $|X| \le k$ and any $x \in X$, there is an index j for which $F_j \cap X = \{x\}$. It is well known that for any $k = 1, 2, ..., n$, there is a (n,k)-strongly-selective family with $m = O(k^2 \log n)$ sets [20, 15]. Note that in the special case $k = 1$ the family consisting of just one set $F_0 = [n]$ is $(n,1)$-strongly-selective. This corresponds to All-transmissions in the previous section. For $k = \omega(\sqrt{n/\log n})$ we can also improve the $O(k^2 \log n)$ bound to $O(n)$ by using the set family corresponding to the RoundRobin protocol, namely the n singleton sets $\{0\}, \{1\}, ..., \{n-1\}$.

In essence, an (n,k)-strongly-selective family can be used to speed up information dissemination through low-degree nodes. Consider the protocol k-Select that works as follows: for any step t and any node v, if $\mathsf{label}(v) \in F_{t \bmod m}$ then transmit from v, otherwise v stays in the receive state. Suppose that w is a node with at most k children, and that these children collected the messages from their subtrees by time t. Steps $t, t+1, ..., t+m-1$ of this protocol use all sets $F_0, F_1, ..., F_{m-1}$ (although possibly in a different order), so each child of w will make a successful transmission by time $t+m$, which, for $k = o(\sqrt{n/\log n})$, is faster than time $O(n)$ required by RoundRobin.

To achieve linear time for arbitrary trees, we will interleave the steps of protocol k-Select with RoundRobin (to deal with high-degree nodes) and steps where all active nodes transmit (to deal with long paths).

Below, we fix parameters $\kappa = \lceil n^{1/3} \rceil$ and $m = O(\kappa^2 \log n)$, the size of an (n,κ)-strongly-selective family $F_0, F_1, ..., F_{m-1}$. (The choice of κ is somewhat arbitrary; in fact, any $\kappa = \Theta(n^c)$, for $0 < c < \frac{1}{2}$, would work.) In the rest of this section we assume that $m \le n$; we can make this assumption because this inequality will hold for sufficiently large n.

Algorithm UnbDTree2**.** We divide the steps into rounds, where each round $s = 0, 1, 2, ...$ consists of three consecutive steps $3s$, $3s+1$, and $3s+2$, that we will call the RR-step, All-step, and Sel-step of round s, respectively.

For each node v we define its *activation round*, denoted α_v, as follows. If v is a leaf then $\alpha_v = 0$. For any other node v, α_v is the first round such that before this round starts v has received the messages from all its children.

In each round $s = \alpha_v, \alpha_v+1, ..., \alpha_v+m-1$, v transmits in the All-step of round s, and if $\mathsf{label}(v) \in F_{s \bmod m}$ then v also transmits in the Sel-step of round s. In each round $s = \alpha_v, \alpha_v+1, ..., \alpha_v+n-1$, if $\mathsf{label}(v) = s \bmod n$ then v transmits in the RR-step of round s. If v does not transmit according to the above rules then v stays in the receiving state.

Analysis. Similar to Algorithm UnbDTree1, in Algorithm UnbDTree2 each node v goes through three stages. We call v *dormant* in rounds $0, 1, ..., \alpha_v - 1$, *active* in rounds $\alpha_v, \alpha_v+1, ..., \alpha_v+n-1$, and retired thereafter. We will also refer to v as being *semi-retired* in rounds $\alpha_v+m, \alpha_v+m+1, ..., \alpha_v+n-1$ (when it is still active, but only uses RR-transmissions). Assuming that v gets activated in some round, since v makes at least one RR-transmission when it is active, it will successfully transmit its message to its parent before retiring, and before its parent gets activated. By straightforward induction on the depth of $\mathcal{T}$, this implies that each node will eventually get activated, proving that Algorithm UnbDTree2 is correct.

By a similar argument, Algorithm UnbDTree2 satisfies the same invariants (J1), (J2) and (J3) as Algorithm UnbDTree1. In addition, in invariant (J1) we also have that among the active nodes, the semi-retired nodes precede those that are not semi-retired.

It remains to show that the running time of Algorithm UnbDTree2 is $O(n)$. The idea of the analysis is to show that Sel- and All-steps disseminate information very fast, in linear time, through subtrees where all node degrees are at most κ. (In fact, this applies also to nodes with higher degrees, as long as they have at most κ active children left.) The process can stall, however, if all active nodes have parents of degree larger than κ. In this case, a complete cycle of RoundRobin will transmit the messages from these nodes to their parents. We show, using Lemma 1, that, since $\kappa = \lceil n^{1/3} \rceil$, such stalling can occur at most 3 times in total. So the overall running time will be still $O(n)$. We formalize this argument in the remainder of this subsection.

Let $\bar{d} = D_{\kappa+1}(\mathcal{T})$. From Lemma 1, we have $\bar{d} \leq 3$. We fix some integer $g \in \{0, 1, 2, 3\}$, a node w with $height_{\kappa+1}(w) = g$, and we let $\mathcal{Y} = \mathcal{T}_w^{\kappa+1,g}$; that is, $\mathcal{Y}$ is the subtree of $\mathcal{T}^{\kappa+1,g}$ rooted at w. This subtree $\mathcal{Y}$ consists of the descendants of w (including w itself) whose $(\kappa+1)$-height in $\mathcal{T}$ is exactly g. By Lemma 2, the $(\kappa+1)$-height of these nodes within $\mathcal{T}^{\kappa+1,g}$ is equal 0. (See Figure 3 for illustration.) This means that all nodes in $\mathcal{Y}$ have degrees, with respect to $\mathcal{Y}$, at most κ.

We also fix $\bar{s}$ to be the first round when all nodes in $\mathcal{T} - \mathcal{T}^{\kappa+1,g}$ are either active or already retired. In particular, for $g = 0$ we have $\bar{s} = 0$. Our goal now is to show that w will get activated in $O(n)$ rounds after round $\bar{s}$.

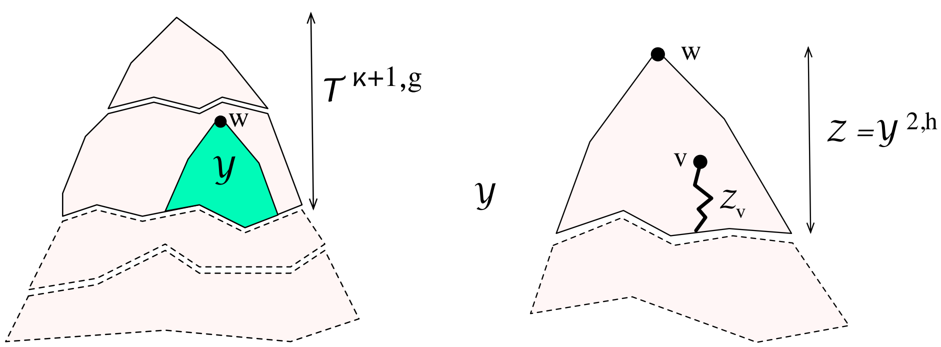

Figure 3: On the left, tree $\mathcal{T}$, partitioned into layers consisting of nodes with the same $(\kappa+1)$-height. (This figure should not be interpreted literally; for example, nodes of $(\kappa+1)$-height 3 may have children of $(\kappa+1)$-height 0 or 1.) The figure on the right shows subtree $\mathcal{Y}$ of $\mathcal{T}$, partitioned into $\mathcal{Z}$ and $\mathcal{Y} - \mathcal{Z}$. Within $\mathcal{Z}$, subtree $\mathcal{Z}_v$ is a path from v to a leaf.

Lemma 4. *$\alpha_w \leq \bar{s} + O(n)$.*

Proof. Let $d = D_2(\mathcal{Y})$. By Lemma 1, $d = O(\log |\mathcal{Y}|) = O(\log n)$. For $h = 0, ..., d$, let l_h be the number of nodes $u \in \mathcal{Y}$ with $height_2(u, \mathcal{Y}) = h$. The overall idea of the proof is similar to the analysis of Algorithm UnbDTree1. The difference is that now, since all degrees in $\mathcal{Y}$ are at most κ, the number of rounds required to advance through the h-th layer of $\mathcal{Y}$, consisting of the nodes in $\mathcal{Y}$ of 2-height (with respect to $\mathcal{Y}$) equal h, can be bounded by $O(m + l_h)$, while before this bound was $O(n)$. Adding up the bounds for all layers, all terms $O(l_h)$ will now amortize to $O(n)$, and the terms $O(m)$ will add up to $O(md) = O(n^{2/3} \log^2 n) = O(n)$ as well. We now fill in the details.

Claim 1. *Let v be a node in $\mathcal{Y}$ with $height_2(v, \mathcal{Y}) = h$. Then the activation round of v satisfies $\alpha_v \leq s_h$, where $s_h = \bar{s} + 2n + \sum_{i \leq h} l_i + 2hm$.*

First, we observe that Claim 1 implies the lemma. This is because for $v = w$ we get the bound $\alpha_w \leq \bar{s} + 2n + \sum_{i \leq d} l_i + 2dm \leq \bar{s} + 2n + n + 2 \cdot O(\log n) \cdot O(n^{2/3} \log n) = \bar{s} + O(n)$, as needed. Thus, to complete the proof, it remains to justify Claim 1. We proceed by induction on h.

Consider first the base case, when $h = 0$. We focus on the computation in the subtree $\mathcal{Y}_v$, which (for $h = 0$) is simply a path $v_1, v_2, ..., v_q = v$, from a leaf v_1 of $\mathcal{Y}$ to v. In round $\bar{s} + n$ all the nodes in $\mathcal{T} - \mathcal{Y}$ must be already retired. If v is active or retired in round $\bar{s} + n$, we are done, because $\bar{s} + n \leq s_0$. If v is dormant, at least one node in $\mathcal{Y}_v$ must be active, because of the invariant (J3) and the fact that all nodes in $\mathcal{T} - \mathcal{Y}$ are already retired. So choose p to be the maximum index for which v_p is active in round $\bar{s} + n$. Each node v_i, for $i = p + 1, p + 2, ..., q$ has already received messages from all its children except v_{i-1}. The activation round of v_p satisfies $\bar{s} < \alpha_{v_p} \leq \bar{s} + n$. Thus v_{p+1} will get a message from v_p before round $\alpha_{v_p} + n$ (using some RR-transmission, if not other), so its activation round satisfies $\bar{s} + n < \alpha_{v_{p+1}} \leq \bar{s} + 2n$. Since we have no interference from outside $\mathcal{Y}_v$, v will then be activated in round $\alpha_{v_{p+1}} + q - p - 1$ using a sequence of All-transmissions. As $q \leq l_0$, we conclude that $\alpha_v = \alpha_{v_{p+1}} + q - p - 1 \leq \bar{s} + 2n + l_0 = s_0$, which is exactly the bound from Claim 1 for $h = 0$.

In the inductive step, fix some $h > 0$, and assume that Claim 1 holds for nodes with 2-height smaller than h. Denoting $\mathcal{Z} = \mathcal{Y}^{2,h}$, we consider the computation in $\mathcal{Z}_v$, the subtree of $\mathcal{Z}$ rooted at v. (See Figure 3.) $\mathcal{Z}_v$ is a path $v_1, v_2, ..., v_q = v$ from a leaf v_1 of $\mathcal{Z}$ to v.

The argument is similar to the base case. There are two twists, however. One, we need to show that v_1 will get activated no later than in round $s_{h-1} + m$; that is, after delay of only m, not $O(n)$. Two, in the time interval between $s_{h-1} + m$ and s_h (when we need the message from v_1 to reach v) the children of the nodes on $\mathcal{Z}_v$ that are not on $\mathcal{Z}_v$ are not guaranteed to be retired anymore. However, they are semi-retired, which is good enough for our purpose.

Consider v_1. We want to show first that v_1 will get activated no later than in round $s_{h-1} + m$. All children of v_1 in $\mathcal{T}$ can be grouped into three types:

Type 1: The children of v_1 in $\mathcal{T} - \mathcal{Y}$. These are activated no later than in round $\bar{s}$, so they retire no later than in round $\bar{s} + n$.

Type 2: The children of v_1 in $\mathcal{Y} - \mathcal{Z}$ that were activated before round $\bar{s} + n$.

Type 3: The children of v_1 in $\mathcal{Y} - \mathcal{Z}$ that were activated at or after round $\bar{s} + n$.

This classification is well defined because all children of v_1 are either in $\mathcal{T} - \mathcal{Y}$ or in $\mathcal{Y} - \mathcal{Z}$.

Clearly, v_1 will receive the messages from its children of Type 1 and Type 2, using RR-transmissions (if not other), before round $\bar{s} + 2n \leq s_{h-1} + m$. (The reason we need to consider Type 2 children separately from those of Type 3 is that children of Type 2 may activate when v_1 still has more than κ active children, so their Sel-transmissions may not be successful.)

The children of v_1 of Type 3 are in $\mathcal{Y} - \mathcal{Z}$ and they activate no earlier than in round $\bar{s} + n$. Since they are in $\mathcal{Y} - \mathcal{Z}$, their 2-height in $\mathcal{Y}$ is strictly less than h, so they activate no later than in round s_{h-1}, by induction. (Note that $s_{h-1} \geq \bar{s} + 2n$.) Thus each child u of v_1 in $\mathcal{Y}$ of Type 3 will complete all its Sel-transmissions, that include the complete (n, κ)-strongly-selective family between rounds $\bar{s} + n$ and $s_{h-1} + m - 1$ (inclusive). In these rounds all children of v_1 in $\mathcal{T} - \mathcal{Y}$ are retired, so at most κ children of v_1 are active in these rounds. This implies that the message of u will be received by v_1 using a Sel-transmission, if no other. Putting it all together, v_1 will receive messages from all its children before round $s_{h-1} + m$, and thus it will be activated no later than in round $s_{h-1} + m$.

Since $\alpha_{v_1} \leq s_{h-1} + m$, we obtain that in round $s_{h-1} + m$ either there is an active node in $\mathcal{Z}_v$ or all nodes in $\mathcal{Z}_v$ are already retired. The remainder of the argument is similar to the base case. If v itself is active or retired in round $s_{h-1} + m$ then we are done, because $s_{h-1} + m \leq s_h$. So suppose that v is still dormant in round $s_{h-1} + m$. Choose p to be the largest index for which v_p is active in this round. Thus $\alpha_{v_p} \leq s_{h-1} + m$ and $\alpha_{v_p} > s_{h-1} + m - n \geq \bar{s} + n$. The last inequality gives us that during the whole time when v_p is active, v_{p+1} has at most κ active children (because its degree in $\mathcal{Y}$ is at most κ and all nodes in $\mathcal{T} - \mathcal{Y}$ are already retired). So v_{p+1} will receive the message from v_p in at most m steps, and therefore $\alpha_{v_{p+1}} \leq \alpha_{v_p} + m \leq s_{h-1} + 2m$. By the choice of p, we also have that, at round $s_{h-1} + m$ and later, for each $i = p + 1, p + 2, ..., q$, all children of v_i except v_{i-1} are either retired or semi-retired, so they do not participate in All-transmissions anymore. Therefore, since there is no interference, v will get activated in $q - p - 1$ additional rounds using consecutive All-transmissions. So $\alpha_v \leq \alpha_{v_{p+1}} + q - p - 1 \leq s_{h-1} + 2m + l_h = s_h$, completing the inductive step, the proof of Claim 1, and the lemma. □

From Lemma 4, all nodes in $\mathcal{T}$ with $(\kappa+1)$-height equal 0 will get activated in at most $O(n)$ rounds. For $g = 1, 2, 3$, all nodes with $(\kappa+1)$-height equal g will activate no later than $O(n)$ rounds after the last node with $(\kappa+1)$-height less than g is activated. This implies that all nodes in $\mathcal{T}$ will be activated within $O(n)$ rounds. Summarizing, we obtain the main result of this section.

Theorem 2. *For any tree with n nodes and any assignment of labels, Algorithm* UnbDTree2 *completes information gathering in time $O(n)$.*

Algorithm UnbDTree2 can be modified to work if we allow labels to be from the range $[L]$, for some $L \geq n$, assuming that all nodes know the value of L. There are two ways to do that. If L is not too large, say $L = O(n)$, then in the algorithm we simply replace n by L. The running time will be still $O(n)$. If L is large, then we can replace the RR-steps by an (L, n)-strongly-selective family of size $O(n^2 \log L)$. This will give us an algorithm with running time $O(n^2 \log L)$. (This explanation assumes that n is known. If n is not known, an algorithm can "guess" n using a standard doubling trick that successfully tries larger and larger values of n. Although the nodes will compute forever, they will still complete gathering within the claimed time bound.) Combining these two approaches gives running time $O(\min(L, n^2 \log L))$.

5 Deterministic Algorithms without Aggregation

In this section we consider deterministic information gathering without aggregation, where each message can contain at most one rumor, plus additional $O(\log n)$ bits of information. For this model, we give an algorithm with running time $O(n \log n)$.

In our algorithm we will assume that each node in $\mathcal{T}$ knows its 2-height. These values can be computed in linear time using a modification of Algorithm UnbDTree2. In this modified algorithm, the message from each node contains its 2-height. When a node v receives such messages from all its children, it can compute its own 2-height, which it can then transmit to its parent. Using 2-heights allows us to synchronize the computation of different nodes; in particular all nodes with the same 2-height will be working in lockstep, executing the same code. This leads to a relatively simple analysis.

To streamline the description of the algorithm, we will also temporarily assume that we are allowed to receive and transmit at the same time. Later, we will show how to remove this assumption.

Algorithm BndDTree**.** Let $\ell = \lceil \log n \rceil$. We divide the computation into $\ell + 1$ phases. Phase h, for $h = 0, 1, ..., \ell$, consists of $3n$ steps $3nh, 3nh + 1, ..., 3n(h + 1) - 1$. In phase h, only the nodes of 2-height equal h participate in the computation. Specifically, consider a node v with $height_2(v) = h$. We have two stages:

Stage AllTransmit: In each step $t = 3nh, 3nh + 1, ..., 3nh + 2n - 1$, if v contains any rumor ρ_u that it still has not transmitted, v transmits ρ_u. (Recall that ρ_u is the rumor that originated in node u.)

Stage RumorRR: In each step $t = 3nh + 2n + u$, for $u = 0, 1, ..., n - 1$, if v has rumor ρ_u, then v transmits ρ_u.

In any other step, v does not transmit.

To clarify, in Stage AllTransmit the algorithm keeps track of the rumors that it already transmitted, say by marking them. Newly arriving, non-duplicate rumors are not marked. At each step, if a node has any non-marked rumors, it transmits an arbitrary non-marked rumor and marks it.

Stage RumorRR can be thought of as "rumor-based" version of RoundRobin, where the decision whether a node v transmits is not based on its own label, but on the label associated with the rumors it has already collected (or, more precisely, with the labels of nodes where these rumors originated). In each step of RumorRR multiple nodes may transmit, although they all must be on the same leaf-to-root path.

Analysis. By Lemma 1, each node has 2-height at most ℓ, so all nodes will participate in the computation and the whole algorithm will complete computation in $O(n \log n)$ steps. It remains to show that r (the root of $\mathcal{T}$) will receive all rumors. The proof of this relies on the following lemma.

Lemma 5. *For each $h = 0, 1, ..., \ell$, at the beginning of phase h, every node $v \in \mathcal{T}$ has rumors from all its $(2, h-1)$-tributary descendants. (In particular, if $height_2(v) < h$ or if v is a frontier node with $height_2(v) = h$ then v has all rumors from T_v.)*

We should stress that this lemma applies to *all* nodes v, without any restriction on their 2-height. Recall that a $(2, h')$-tributary descendant of a node v is a node $u \in \mathcal{T}_v$ such that all the nodes on the path from u to v (including u but excluding v) have 2-height at most h'.

Proof. The proof is by induction on h, the index of the phase. The lemma is vacuously true at the beginning, when $h = 0$.

In the inductive step, assume that the lemma holds for some $h < \ell$, and consider phase h. We want to show that when phase h ends then each node v has rumors from all $(2, h)$-tributary descendants. By the inductive assumption, when phase h ends (in fact, even when it starts) node v already has all rumors from its $(2, h-1)$-tributary descendants. So if v does not have any proper descendants of 2-height h (that is, if $height_2(v) \leq h - 1$ or if v is a frontier node with $height_2(v) = h$) then we are done.

Let v be a node with $height_2(v) \geq h$. It remains to prove that if v has a child u with $height_2(u) = h$ then right after phase h all rumors from $\mathcal{T}_u$ will also be in v.

Since $height_2(u) = h$, the subtree $\mathcal{T}_u^{2,h}$, namely the subtree consisting of the descendants of u with 2-height equal h, is a path $u_1, u_2, ..., u_q = u$, where u_1 is a 2-frontier node and a leaf of $\mathcal{T}^{2,h}$. From the inductive assumption, when phase h starts all rumors from $\mathcal{T}_u$ are in $\mathcal{T}_u^{2,h}$. Also, the children of the nodes in $\mathcal{T}_u^{2,h}$, except possibly for one child that is in $\mathcal{T}_u^{2,h}$, do not transmit in phase h, so all transmissions from $u_1, u_2, ..., u_{q-1}$ in Stage AllTransmit will be successful. We show that, thanks to pipelining, all rumors that are in $\mathcal{T}_u^{2,h}$ when phase h starts will reach u during Stage AllTransmit.

For any step $3nh + s$, $s = 0, 1, ..., 2n - 1$, and for $i = 1, 2, ..., q - 1$, we let $\phi_{s,i}$ to be the number of rumors in u_i which are still not in u_{i+1}. Our argument uses a potential function defined by $\Phi_s = \sum_{i=a_s}^{q-1} \max(\phi_{s,i}, 1)$, where a_s is the smallest index for which $\phi_{s,a_s} \neq 0$.

Claim 2. *For $s = 0, 1, ..., 2n - 2$, if $\Phi_s > 0$ then the value of the potential function Φ_s will decrease in step $3nh + s$, that is $\Phi_{s+1} < \Phi_s$.*

Indeed, for $i < q$, each node u_i with $\phi_{s,i} > 0$ will transmit a new rumor to u_{i+1} in step $3nh + s$. Since $\phi_{s,i} = 0$ for $i < a_s$, node u_{a_s} will not receive any new rumors. We have $\phi_{s,a_s} > 0$, by the choice of a_s. If $\phi_{s,a_s} > 1$ then $\max(\phi_{s,a_s}, 1)$ will decrease by 1. If $\phi_{s,a_s} = 1$ then the index a_s itself will increase. In either case, u_{a_s}'s contribution to Φ_s will decrease by 1. For $i > a_s$, even if u_i receives a new rumor from u_{i-1}, the term $\max(\phi_{s,i}, 1)$ cannot increase, because if $\phi_{s,i} > 0$ then u_i transmits a new rumor to u_{i+1}, and if $\phi_{s,i} = 0$ then this term is 1 anyway. Therefore, overall, Φ_s will decrease by at least 1, completing the proof for Claim 2.

We have $\Phi_0 \leq q + n$, because each rumor contributes to at most one term in Φ_0. Since Φ_s strictly decreases in each step, Φ_s will become 0 in at most $2n$ steps. In other words, in $2n$ steps u will receive all rumors that were in $\mathcal{T}_u^{2,h}$ when phase h started, and thus all rumors from $\mathcal{T}_u$.

In Stage RumorRR, u will transmit all collected rumors to v, without collisions. As a result, at the beginning of the next phase v will contain all rumors from $\mathcal{T}_u$, completing the proof of the inductive step and the lemma. ☐

We still need to explain how to modify Algorithm BndDTree to eliminate the assumption that we can transmit and receive at the same time. This can be accomplished by adding $3n$ more steps to each phase, as follows:

- We first add a new stage at the beginning of each phase, consisting of n steps. Note that each node v with $height_2(v) = h$ knows h and also it knows whether it has a child of 2-height h. If v does not have such a child, then v is a frontier node and the initial node of a path consisting of nodes of 2-height equal h. At the very beginning of phase h, v sends a message along this path, so that any node on this path can determine whether it is an even- or odd-numbered node along this path.

- Then we double the number of steps in Stage AllTransmit, increasing its length from $2n$ to $4n$. In this stage, among the nodes with 2-height h, "even" nodes will transmit in even steps, and "odd" nodes will transmit in odd steps. In this way, each node will never receive and transmit at the same time.

As mentioned earlier, it is possible that several nodes transmit at the same step of Stage RumorRR. If that happens, these nodes must be on the same leaf-to-root path and must transmit the same rumor. In that case, only the node highest in the tree is relevant for the algorithm, so this situation does not cause a problem.

We thus obtain the main result of this section:

Theorem 3. *For any tree with n nodes and any assignment of labels, Algorithm* BndDTree *completes information gathering in time $O(n \log n)$.*

Similar to Algorithm UnbDTree2, we can adapt Algorithm BndDTree to the model where labels are from $[L]$, for some $L \geq n$. If we use Algorithm UnbDTree2 as is, but using labels $0, 1, ..., L-1$, the running time will be $O(L \log L)$. Alternatively, we can replace Stage RumorRR by an (L, n)-strongly-selective family of size $O(n^2 \log L)$. Overall, choosing the better of these two options, the running time will be $O(\min(L \log L, n^2 \log n \log L))$. In particular, for $L = O(n)$, the running time will remain $O(n \log n)$.

6 Deterministic Fire-and-Forward Protocols

We now consider a simple type of protocols that we call *fire-and-forward* protocols. As defined in Section 2, in such protocols each message consists of just one rumor and no other information. They also satisfy the following condition: for any node v, at any time t, v can only choose one of three actions: (i) enter the receiving state, (ii) if v received a rumor ρ_z in step $t-1$ then it can transmit (forward) ρ_z, or (iii) it can transmit (fire) its own rumor ρ_v. The idea behind fire-and-forward protocols is that all messages travel towards the root without any delay. A message may vanish along the way, if it collides with another message, or if it gets refused by a node that is in the transmitting state, or if it gets dropped by a node that decides to fire.

In Section 7 we will show that there exists a randomized fire-and-forward protocol that accomplishes information gathering in time $O(n \log n)$. This raises the question whether this running time can be achieved by a deterministic fire-and-forward protocol. (As before, in the deterministic case we assume that the nodes are labelled $0, 1, ..., n-1$.) There is a trivial deterministic fire-and-forward protocol with running time $O(n^2)$: release all rumors one at a time, spaced at intervals of length n. In this section we show that this can be improved to $O(n^{1.5})$ and that this bound is optimal.

It is convenient to formulate our arguments in terms of a yet simpler and mathematically more elegant model of *basic fire-and-forward* protocols, that we define shortly. We will refer to the above definition as the *standard fire-and-forward* model. After proving our upper and lower bounds for the basic model, we show later how to adapt our proofs to the standard model.

6.1 Basic Fire-and-Forward (BFF) Protocols

In the *basic fire-and-forward (BFF)* model, we allow nodes to receive and transmit at the same time. Each message consists of one rumor and no other information. If a node v receives a rumor ρ_z at time $t-1$, v must *forward* (that is, transmit) ρ_z at time t. At any time v can also decide to *fire*, that is transmit its own rumor ρ_v. The situation when both events occur (receiving a rumor at step $t-1$ and firing at step t) is treated as a collision at v, and in this case both messages disappear.

If rumors fired from two nodes collide at all, they will collide at their lowest common ancestor. This can happen only when the difference in times between these two firings is equal to the difference of their depths in the tree. More precisely, let $\mathcal{T}$ be the tree on input, denote by *depth*(v) the depth of a node v in $\mathcal{T}$ (its distance to the root), and suppose that some node v fires its rumor ρ_v at time t. Then ρ_v is guaranteed to reach the root providing that no other node u fires at time $t + \mathit{depth}(v) - \mathit{depth}(u)$. (It may happen, however that ρ_v will reach the root even if u fires at time $t + \mathit{depth}(v) - \mathit{depth}(u)$, since ρ_u could collide with some other rumor before meeting ρ_v.)

The BFF protocol we develop below is *oblivious*, in the sense that the decision whether to fire or not depends only on the label of the node and the current time. Such protocols are defined by specifying, for each label $l \in [n]$, the corresponding set F_l of firing times. Then a node v with $\mathsf{label}(v) = l$ fires at the times in F_l.

In fact, it is easy to show that any BFF protocol can be turned into an oblivious one without affecting its asymptotic running time. The idea is that leaves of the tree receive no information at all during the computation. For any BFF protocol $\mathcal{A}$ that runs in time $f(n)$, and for any tree $\mathcal{T}$, imagine that we run this protocol on the tree $\mathcal{T}'$ obtained by adding a leaf to every node v and giving it the label of v. Label the original nodes with the remaining labels. This modification exactly doubles the number of nodes, so $\mathcal{A}$ will complete in time $O(f(n))$ on $\mathcal{T}'$. (We tacitly assume here that $f(cn) = \Theta(f(n))$ for any constant $c > 0$, which is true for the time bounds we consider.) In the execution of $\mathcal{A}$ on $\mathcal{T}'$ the leaves receive no information and all rumors from the leaves will reach the root. For $l \in [n]$, let F_l be the set of firing times in $\mathcal{T}'$ of the leaf v with $\mathsf{label}(v) = l$. Now apply $\mathcal{A}$ to $\mathcal{T}$, with all nodes firing obliviously according to the sets F_l, and completely ignoring all information received during the computation. Since the relative depths of the nodes with labels from $[n]$ in $\mathcal{T}'$ are exactly the same as in $\mathcal{T}$, all rumors will also reach the root in $\mathcal{T}$. In other words, after this modification, we obtain an oblivious protocol $\mathcal{A}'$ with running time $O(f(n))$.

6.2 An $O(n^{1.5})$ Upper Bound in the BFF Model

We now present our $O(n^{1.5})$-time fire-and-forward protocol for the basic model. As explained above in Section 6.1, this protocol should specify a set of firing times F_l for each label l. For the protocol to be correct, it is sufficient that it has the following property: for any mapping $[n] \to [n]$, that maps each label to the depth of the node with this label, and for any label l, there is a firing time $t \in F_l$ that does not "collide" with firing times of other nodes, in the sense that each other set $F_{l'}$ does not have a firing time t' such that $t - t'$ is equal to the difference of depths of the nodes with labels l and l'. We also want each of these firing times to be at most $O(n^{1.5})$. To this end, we will partition all labels into batches, each of size roughly $\sqrt{n}$, and show that for any batch we can define such collision-avoiding firing times from an interval of length $O(n)$. Since we have about $\sqrt{n}$ batches, this will give us running time $O(n^{1.5})$.

Our construction of firing times is based on a concept of dispersers, defined below, which are reminiscent of various *rulers* studied in number theory, including Sidon sequences [40], Golomb rulers [41], or sparse rulers [42]. The particular construction we give in the paper is, in a sense, a multiple set extension of a Sidon-set construction by Erdös and Turán [21].

We now give the details. For $z \in \mathbb{Z}$ and $X \subseteq \mathbb{Z}$, let $X + z = \{x + z : x \in X\}$. Also, let s be a positive integer. A set family $D_1, ..., D_m \subseteq [s]$ is called an (n, m, s)-*disperser* if for each function $\delta : \{1, ..., m\} \to [n]$ and each $j \in \{1, ..., m\}$ we have

$$D_j + \delta(j) \not\subseteq \bigcup_{i \in \{1,...,m\} - \{j\}} (D_i + \delta(i)).$$

The intuition is that D_j represents the set of firing times of node identified by j (in the sense to be explained later) and $\delta(j)$ represents this node's depth in the tree. Then $D_j + \delta(j)$ are the times when this node's messages would arrive at the root, if they did not collide along the way. Therefore the disperser condition says that, for each depth function δ, some firing in D_j will not collide with firings of other nodes.

Lemma 6. *There exists an (n, m, s)-disperser with $m = \Omega(\sqrt{n})$ and $s = O(n)$.*

Proof. Let p be the smallest prime such that $p^2 \geq n$. For each $a = 1, 2, ..., p - 1$ and $x \in [p]$ define

$$d_a(x) = (ax \bmod p) + 2p \cdot (ax^2 \bmod p).$$

We claim that for any $a \neq b$ and any $t \in \mathbb{Z}$ the equation $d_a(x) - d_b(y) = t$ has at most two solutions $(x, y) \in [p]^2$. For the proof, fix a, b, t and one solution $(x, y) \in [p]^2$. Suppose that $(u, v) \in [p]^2$ is a different solution. Thus we have $d_a(x) - d_b(y) = d_a(u) - d_b(v)$. After substituting and rearranging, this can be written as

$$\begin{aligned}(ax \bmod p) - (by \bmod p) - (au \bmod p) + (bv \bmod p)& \\ = 2p[\, -(ax^2 \bmod p) + (by^2 \bmod p) + (au^2 \bmod p) - (bv^2 \bmod p)\,]&.\end{aligned}$$

The expression on the left-hand side is strictly between $-2p$ and $2p$, so both sides must be equal 0. This implies that

$$ax - au \equiv by - bv \pmod{p} \quad \text{and} \tag{1}$$

$$ax^2 - au^2 \equiv by^2 - bv^2 \pmod{p}. \tag{2}$$

From equation (1), the assumption that $(x, y) \neq (u, v)$ implies that $x \neq u$ and $y \neq v$. We can then divide the two equations, getting

$$x + u \equiv y + v \pmod{p}. \tag{3}$$

With addition and multiplication modulo p, $\mathbb{Z}_p$ is a field. Therefore for any x and y, and any $a \neq b$, equations (1) and (3) uniquely determine u and v, completing the proof of the claim.

Now, let $m = (p-1)/2$ and $s = 2p^2 + p$. By Bertrand's postulate we have $\sqrt{n} \leq p < 2\sqrt{n}$, which implies that $m = \Omega(\sqrt{n})$ and $s = O(n)$. For each $i = 1, 2, ..., m$, define $D_i = \{d_i(x) : x \in [p]\}$. It is sufficient to show that the sets $D_1, D_2, ..., D_m$ satisfy the condition of the (n, m, s)-disperser.

The definition of the sets D_i implies that $D_i \subseteq [s]$ for each i. Fix some δ and j from the definition of dispersers. It remains to verify that $D_j + \delta(j) \nsubseteq \bigcup_{i \neq j}(D_i + \delta(i))$. For $x \in [p]$ and $i \in \{1, 2, ..., m\}$, we say that i *kills* x if $d_j(x) + \delta(j) \in D_i + \delta(i)$. Our earlier claim (with $t = \delta(i) - \delta(j)$) implies that any $i \neq j$ kills at most two values in $[p]$. Thus all indices $i \neq j$ kill at most $2(m-1) = p - 3$ integers in $[p]$, which implies that there is some $x \in [p]$ that is not killed by any i. For this x, we will have $d_j(x) + \delta(j) \notin \bigcup_{i \neq j}(D_i + \delta(i))$, completing the proof that $D_1, ..., D_m$ is indeed an (n, m, s)-disperser. ☐

We now describe our algorithm.

Algorithm BFFDTree**.** Let $D_1, D_2, ..., D_m$ be the (n, m, s)-disperser from Lemma 6. We partition all labels (and thus also the corresponding nodes) arbitrarily into batches $B_1, B_2, ..., B_k$, for $k = \lceil n/m \rceil$, with each batch B_i having m nodes (possibly except the last batch, that could be smaller). Order the nodes in each batch arbitrarily, for example according to increasing labels.

The algorithm has k phases. Each phase $q = 1, 2, ..., k$ consists of $s' = s + n$ steps in the time interval $[s'(q-1), s'q - 1]$. In phase q, the algorithm transmits rumors from batch B_q, by having the j-th node in B_q fire at each time $s'(q-1) + \tau$, for $\tau \in D_j$. Note that in the last n steps of each phase none of the nodes fires.

Analysis. We now show that Algorithm BFFDTree correctly performs gathering in any n-node tree in time $O(n^{1.5})$. Since $m = \Omega(\sqrt{n})$, we have $k = O(\sqrt{n})$ phases. Also, $s' = O(n)$, so the total running time of the protocol is $O(n^{1.5})$.

It remains to show that during each phase q each node in B_q will have at least one firing that will send its rumor to the root r without collisions. Fix some tree $\mathcal{T}$ and let $\delta(j) \in [n]$ be the depth of the jth node in batch B_q. For any batch B_q and any $v \in B_q$, if v is the jth node in B_q then v will fire at times $s'(q-1) + \tau$, for $\tau \in D_j$. From the definition of dispersers, there is $\tau \in D_j$ such that $\tau + \delta(j) - \delta(i) \notin D_i$ for each $i \neq j$. This means that the firing of v at time $s'(q-1) + \tau$ will not collide with any firing of other nodes in batch B_q. Since the batches are separated by empty intervals of length n, this firing will not collide with any firing in other batches. So v's rumor will reach r.

Summarizing, we obtain our main result of this section.

Theorem 4. *There is a BFF protocol for information gathering in trees with running time $O(n^{1.5})$.*

6.3 An $\Omega(n^{1.5})$ Lower Bound in the BFF Model

In this section we show that any basic fire-and-forward protocol needs time $\Omega(n^{1.5})$ to deliver all rumors to the root for an arbitrary tree with n nodes. Fix some fire-and-forward protocol $\mathcal{A}$. Without loss of generality, as explained earlier in this section, we can assume that $\mathcal{A}$ is oblivious, namely that $\mathcal{A}$ is specified by the sets F_l of firing times, for each label l. A node v with $\mathsf{label}(v) = l$ fires at the time steps in F_l and not in other time steps.

Let T be the running time of $\mathcal{A}$, and denote by $\#(\mathcal{A}) = \sum_{l \in [n]} |F_l|$ the total number of firings in $\mathcal{A}$, among all possible nodes. We will first prove the lower bound under the assumption that $\mathcal{A}$ fires at most T times in total, that is $\#(\mathcal{A}) \leq T$. Later we will show how to extend our argument to protocols with an arbitrary number of firings.

We will show that there is a constant $c > 0$ such that if $T < cn^{1.5}$ and n is sufficiently large then $\mathcal{A}$ will fail even on a "caterpillar" tree, consisting of a path $\mathcal{P}$ of length n with n leaves attached to the nodes of this path. (For convenience we use $2n$ nodes instead of n, but this does not affect the asymptotic lower bound.) This path $\mathcal{P}$ is fixed, and the label assignment to the nodes on $\mathcal{P}$ is not important for the proof, but, for the ease of reference, we will label them $n, n+1, ..., 2n-1$, in order in which they appear on the path, with node labeled $2n-1$ being the root. The leaves have labels from the set $[n] = \{0, 1, ..., n-1\}$. To simplify the argument we will identify the labels with nodes, and we will refer to the node with label l simply as "node l". (See Figure 4.) The objective is to show that if $T < cn^{1.5}$, then, for any collection $\{F_0, F_1, \ldots, F_{n-1}\}$ of firing time sets, there is a way to attach the nodes from $[n]$ to $\mathcal{P}$ to make $\mathcal{A}$ fail. Here "failure" means that there is at least one node w such that all firings of w will collide with firings from other nodes.

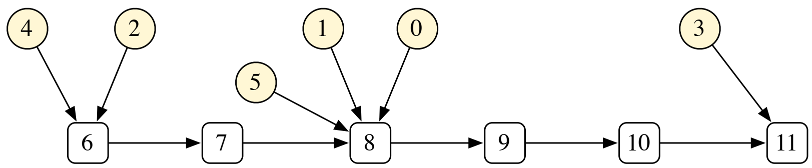

Figure 4: A caterpillar graph from the proof, for $n = 6$. The nodes on the path $\mathcal{P}$ are represented by rectangles, and the leaves are represented by circles. In this example, the root is 11.

Without loss of generality, assume that T is a multiple of n, and let $k = T/n$. We divide the time range $0, 1, ..., T-1$ into k bins of size n, where the ith bin is the interval $[in, (i+1)n-1]$, for $i = 0, 1, ..., k-1$. If a node $v \in [n]$ fires at time t, we say that a node $u \in [n] - \{v\}$ *covers* this firing if u has a firing at time t' such that $t' \geq t$ and $\lfloor t'/n \rfloor = \lfloor t/n \rfloor$, that is t, t' are in the same bin. For a subset $L \subseteq F_v$ of firings of v, denote by $C(L)$ the *cover of* L, that is the set of nodes that cover the firings in L.

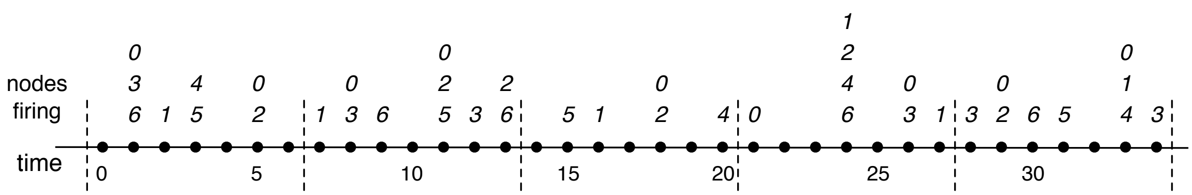

Figure 5: An example illustrating firing sets, bins, and the cover relation. Here, $n = 7$, $k = 5$ and $T = 35$. The bins are separated by vertical lines. The numbers above each time represent the vertices that fire at that time. For example, $F_4 = \{3, 20, 24, 33\}$. For a set $L = \{3, 20, 33\} \subseteq F_4$, its cover is $C(L) = \{0, 1, 2, 3, 5\}$. For a set $L' = \{20, 24\} \subseteq F_4$, its cover is $C(L') = \{0, 1, 2, 3, 6\}$.

Lemma 7. *Suppose that $\{F_0, F_1, \ldots, F_{n-1}\}$ is the collection of firing time sets of a protocol $\mathcal{A}$ successful on all caterpillar trees. Then, for each node $v \in [n]$, there is a set of firings $L \subseteq F_v$ such that $|C(L)| < |L|$.*

Proof. The proof of the lemma is by contradiction. Suppose that there exists a node $w \in [n]$ with the property that for each $L \subseteq F_w$ we have $|C(L)| \geq |L|$. Then Hall's Theorem implies that there is a perfect matching between the firing times in F_w and the nodes in $[n] - \{w\}$ that cover these firings. Let the firing times of w be $F_w = \{t_1, t_2, \ldots, t_j\}$, and for each $i = 1, 2, ..., j$, let u_i be the node matched to t_i in this matching. By the definitions of bins and covering, each u_i fires at some time $t_i + s_i$, where $0 \leq s_i \leq n-1$. We can then construct a caterpillar tree by attaching w to node n and attaching each u_i to node $n + s_i$ on $\mathcal{P}$. In this caterpillar tree, the firing of w at each time t_i will collide with the firing of u_i at time $t_i + s_i$ (if not earlier). So the rumor from w will not reach the root, contradicting the correctness of $\mathcal{A}$. This completes the proof of the lemma. $\square$

Now, consider the sets $F_0, F_1, \ldots, F_{n-1}$ of firing times for all nodes, as specified by $\mathcal{A}$. Using Lemma 7, for each $v \in [n]$, we can now choose a set $L_v \subseteq F_v$ with $|C(L_v)| < |L_v|$. Obviously, each set L_v is non-empty.

Let Q be the set of ordered pairs (u, v) of different nodes $u, v \in [n]$ for which there is a bin which contains a firing from L_u and a firing from L_v, in this order in time. So $(u, v) \in Q$ implies that $v \in C(L_u)$.

We will bound $|Q|$ in two different ways. On the one hand, each $u \in [n]$ appears as the first element in at most $|C(L_u)|$ pairs in Q. Adding up over all u, using Lemma 7 and the assumption that the total number of firings is at most T, we get

$$\begin{aligned} |Q| &\le \textstyle\sum_{u\in[n]} |C(L_u)| \\ &< \textstyle\sum_{u\in[n]} |L_u| \\ &\le \textstyle\sum_{u\in[n]} |F_u| = \#(\mathcal{A}) \le T. \end{aligned} \tag{4}$$

On the other hand, we can also establish a lower bound on $|Q|$, as follows. Choose a specific representative firing t_v from each L_v. For each bin $i = 0, 1, ..., k-1$, let n_i be the number of representatives in the ith bin. Any two representatives in bin i contribute one pair to Q. So $|Q| \ge \frac{1}{2}\sum_{i=0}^{k-1} n_i(n_i - 1)$. Since $\sum_{i=0}^{k-1} n_i = n$, if we let all n_i take real values then the value of $\sum_{i=0}^{k-1} n_i(n_i - 1)$ will be minimized when all n_i are equal to n/k. This implies that

$$\begin{aligned} |Q| &\ge \tfrac{1}{2} \cdot k \left(\left(\tfrac{n}{k}\right)^2 - \tfrac{n}{k} \right) \\ &= \tfrac{1}{2} \cdot (\tfrac{n^2}{k} - n) = \tfrac{1}{2} \cdot (\tfrac{n^3}{T} - n). \end{aligned} \tag{5}$$

Combining bounds (4) and (5), we obtain that $T \ge |Q| \ge \frac{1}{2}(n^3/T - n)$, which implies that $T \ge \frac{1}{3} n^{3/2}$ for any sufficiently large n. (The choice of constant $\frac{1}{3}$ is arbitrary; any coefficient strictly smaller than $\frac{1}{\sqrt{2}}$ would work.) This completes the proof of the lower bound, with the assumption that $\#(\mathcal{A}) \le T$.

We now consider the general case, without any restriction on $\#(\mathcal{A})$, the total number of firings in $\mathcal{A}$. Let $\mathcal{A}$ be any protocol whose running time T satisfies $T \le \frac{1}{50} n^{1.5}$ for sufficiently large n. Using a probabilistic argument and a reduction to the above special case, we will show that we can construct a caterpillar tree and a node w for which $\mathcal{A}$ will fail.

We start by choosing a random set $\tilde{V}$ of nodes, where each node is included in $\tilde{V}$ independently with probability $\frac{1}{3}$. Let $\tilde{n} = |\tilde{V}|$, and let $\tilde{Z}$ be the set of times at which only nodes from $\tilde{V}$ fire in $\mathcal{A}$. Define $\tilde{\mathcal{A}}$ to be a sub-protocol of $\mathcal{A}$ obtained by restricting $\mathcal{A}$ to the steps in $\tilde{Z}$. In other words, for each time $t = 0, 1, ..., T-1$, if $t \in \tilde{Z}$ then $\tilde{\mathcal{A}}$ has exactly the same firings as $\mathcal{A}$; otherwise, for $t \notin \tilde{Z}$, $\tilde{\mathcal{A}}$ does not fire at all. The idea of the argument is to construct a bad caterpillar tree for $\tilde{\mathcal{A}}$ and then extend it to a bad tree for $\mathcal{A}$.

Lemma 8. $\mathbb{P}[\, \#(\tilde{\mathcal{A}}) \le T \,] \ge \frac{2}{3}$.

Proof. Consider the random variable $\#(\tilde{\mathcal{A}})$. If t is a time where $\mathcal{A}$ has $j \ge 1$ firings, then the probability that $t \in \tilde{Z}$ is $(\frac{1}{3})^j$, so the contribution of t to $\mathbb{E}[\#(\tilde{\mathcal{A}})]$, the expected number of firings in $\tilde{\mathcal{A}}$, is $j(\frac{1}{3})^j \le \frac{1}{3}$. It follows that $\mathbb{E}[\#(\tilde{\mathcal{A}})] \le \frac{1}{3}T$. Therefore, using Markov's inequality, we get

$$\mathbb{P}[\, \#(\tilde{\mathcal{A}}) \ge T \,] \le \frac{\mathbb{E}[\#(\tilde{\mathcal{A}})]}{T} \le \tfrac{1}{3},$$

and the lemma follows. □

Lemma 9. *For sufficiently large n we have $\mathbb{P}[\, T \le \frac{1}{3}\tilde{n}^{1.5} \,] \ge \frac{2}{3}$.*

Proof. Since the events "$v \in \tilde{V}$" are independent for different nodes v, the distribution of $\tilde{n}$ is binomial with parameters n and p. In particular, the expected value of $\tilde{n}$ is $\mathbb{E}[\tilde{n}] = \frac{1}{3}n$, and its variance is $\mathsf{Var}[\tilde{n}] = \frac{2}{9}n$. Using Chebyshev's inequality, we get

$$\begin{aligned} \mathbb{P}[\tilde{n} < \tfrac{1}{6}n] &\le \mathbb{P}[\, |\tilde{n} - \tfrac{1}{3}n| \ge \tfrac{1}{6}n \,] \\ &= \mathbb{P}[\, |\tilde{n} - \mathbb{E}[\tilde{n}]| \ge \tfrac{1}{6}n \,] \\ &\le \frac{\mathsf{Var}[\tilde{n}]}{(\frac{1}{6}n)^2} = \frac{8}{n} \le \tfrac{1}{3}, \end{aligned}$$

for $n \geq 24$. So, for such n, $\mathbb{P}[\tilde{n} \geq \frac{1}{6}n] \geq \frac{2}{3}$. But whenever $\tilde{n} \geq \frac{1}{6}n$, as long as n is large enough, we have

$$T \leq \tfrac{1}{50}n^{1.5} \leq \tfrac{1}{3}(\tfrac{1}{6}n)^{1.5} \leq \tfrac{1}{3}\tilde{n}^{1.5},$$

which implies that $\mathbb{P}[\, T \leq \frac{1}{3}\tilde{n}^{1.5}\,] \geq \frac{2}{3}$ as well. □

From Lemmas 8 and 9, we have

$$\begin{aligned}\mathbb{P}[\,\#(\tilde{\mathcal{A}}) \leq T \leq \tfrac{1}{3}\tilde{n}^{1.5}\,] &\geq \mathbb{P}[\,\#(\tilde{\mathcal{A}}) \leq T\,] + \mathbb{P}[\,T \leq \tfrac{1}{3}\tilde{n}^{1.5}\,] - 1\\ &\geq \tfrac{2}{3} + \tfrac{2}{3} - 1 = \tfrac{1}{3}.\end{aligned}$$

So there must be some set $\bar{V}$ of nodes, such that letting $\bar{n} = |\bar{V}|$, denoting by $\bar{Z}$ the set of times when only nodes from $\bar{V}$ fire, and by $\bar{\mathcal{A}}$ the restriction of $\mathcal{A}$ to $\bar{Z}$, we have that

$$\#(\bar{\mathcal{A}}) \leq T \leq \tfrac{1}{3}\bar{n}^{1.5}.$$

Therefore we can apply our earlier construction to obtain a caterpillar tree $\mathcal{T}$ with nodes from $\bar{V}$ and a node $w \in \bar{V}$ for which $\bar{\mathcal{A}}$ fails. (Note that $\bar{V}$ may not be of the form $[\bar{n}]$, but our construction of the "bad" caterpillar tree, where all firings of some node are unsuccessful, did not depend on the labels of the leaves being of this form.)

We still need to show that we can modify $\mathcal{T}$ so that not only $\bar{\mathcal{A}}$, but also $\mathcal{A}$ itself will fail. Let $\mathcal{T}'$ be a modified tree obtained from $\mathcal{T}$ by adding all nodes that are not in $\bar{V}$ as siblings of w. Consider now a firing of w in $\mathcal{T}'$ at some time t. If $t \in \bar{Z}$, then t is used by $\bar{\mathcal{A}}$, so this firing collides with another firing from $\bar{V}$. If $t \notin \bar{Z}$, then, by the definition of $\bar{Z}$, there is a node u outside of $\bar{V}$ that fires at the same time (otherwise t would be included in $\bar{Z}$) and is a sibling of w in $\mathcal{T}'$. So again, this firing from w will collide with the firing of u. We can thus conclude that the rumor from w will not reach the root, completing the proof of the following lower bound.

Theorem 5. *If $\mathcal{A}$ is a deterministic BFF protocol for information gathering in trees, then the running time of $\mathcal{A}$ is $\Omega(n^{1.5})$.*

6.4 Extension to the Standard Model

It remains to explain how we can extend this result to the standard model, where nodes are not allowed to receive and transmit at the same time. The lower bound from the previous section applies to the standard model without any changes. (In particular, it is still valid to restrict the argument to oblivious algorithms.) This is because, for the caterpillar tree $\mathcal{T}$ constructed in this proof, and for any collection of firing time sets $\{F_v\}_{v\in[n]}$ for the leaves, if a message from w (the leaf that is attached to node n and whose all messages are forced to collide in the lower-bound construction) collides in the basic model then it also either collides or gets dropped in the standard model (possibly earlier), no matter how the nodes along the main path behave.

For the upper bound, we argue that our Algorithm BFFDTree from Section 6.2 can be adapted to the standard model. We only give a sketch of the argument. The idea is that if, at some time step, some node z transmits a rumor ρ_v and receives a rumor ρ_u, and if v fired at time t, then u fired at time $t + \mathit{depth}(v) - \mathit{depth}(u) + 1$. We can then extend the definition of collisions to include this situation. Incorporating this into the construction from Lemma 6, any index i may now kill more than two x's, but not more than four. So taking $m = \lfloor (p-1)/4 \rfloor$, we still will always have an x that is not killed by any i. By using such modified concept of dispersers, we obtain an $O(n^{1.5})$-time protocol in the standard fire-and-forward model.

Theorem 6. (i) *There is a deterministic fire-and-forward protocol for information gathering in trees with running time $O(n^{1.5})$.* (ii) *Any deterministic fire-and-forward protocol for information gathering in trees has running time $\Omega(n^{1.5})$.*

7 An $O(n \log n)$-Time Randomized Algorithm

We now show a randomized algorithm for information gathering in trees, in the model without rumor aggregation, with expected running time $O(n \log n)$. The algorithm does not use node labels. It is specified as a fire-and-forward algorithm, as defined in Sections 2 and 6; that is, each message consists of just one rumor and no other information, and at each step each node v chooses one of three options: it can be in the receiving mode, it can fire (transmit its own rumor), or it can forward the rumor received in the previous step, if any.

In the description of the algorithm we assume that the number n of nodes is known. Using a standard doubling trick that tries values of n that are successive powers of 2, the algorithm can be extended to one that does not depend on n. (This new algorithm will complete the task in expected time $O(n \log n)$, but it will keep running forever.)

In our presentation, we first specify our algorithm in the basic fire-and-forward (BFF) model (see Section 6); that is, we allow nodes to receive and transmit at the same time. Thus we only need to specify the (random) set of firing times for each node. For the sake of uniformity we assume that the root r also needs to fire successfully in order to receive its own rumor (at the same time step). We explain later how to modify this algorithm to make it compatible with the standard fire-and-forward model. The algorithm is extremely simple:

Algorithm RTree**.** Any node v, at each step t, fires with probability $1/n$, independently of other nodes.

Analysis. At the fundamental level, the problem resembles the Coupon Collector's Problem (see, for example, [32]), with rumors playing the role of coupons. There are two differences though. One, in our case the rumors "choose themselves" (when their origin nodes fire) independently of each other, so it's possible that no rumor will be selected at some steps, either because of all nodes deciding not to fire or because of collisions. Two, we need to account for the delay from the time when a node fires until its rumor reaches the root. The analysis can be carried out through a reduction to the "distributed" variant of Coupon Collector's Problem in [13]. We present an independent argument, for the sake of self-containment. We start with the following lemma.

Lemma 10. *For each node z, and for each time step $t \geq n-1$, the probability that r receives rumor ρ_z at step t is at least $\frac{1}{n}(1-\frac{1}{n})^{n-1}$. Further, for different t, the events of r receiving ρ_z are independent.*

Proof. To prove the lemma, it helps to view the computation in a slightly different, but equivalent way. Imagine that we ignore collisions, and we allow each message to consist of a set of rumors. If some children of v transmit at a time t, then v receives all rumors in the transmitted messages, and at time $t+1$ it transmits a message containing all these rumors, possibly adding its own rumor, if v also fires at step $t+1$. We will refer to these messages as *rumor bundles*. In this interpretation, if r receives a rumor bundle that is a singleton set $\{\rho_z\}$, at some time t, then in Algorithm RTree this rumor ρ_z will be received by r at time t. (Note that the converse is not true, because it may happen that r will receive a rumor bundle that has other rumors besides ρ_z, but in Algorithm RTree these other rumors may collide between themselves and disappear before meeting ρ_z, allowing ρ_z to reach r.)

For any time t and node v, define two random events, one of v firing its rumor and the other of r receiving only ρ_v:

$$\begin{aligned} \mathsf{fire}(v,t) &= \text{“node } v \text{ fires at time } t\text{”}, \\ \mathsf{singl}(v,t) &= \text{“rumor bundle received by } r \text{ at time } t \text{ is the singleton } \{\rho_v\}\text{”}. \end{aligned}$$

For any fixed time $t \geq n-1$ and node $z \in \mathcal{T}$ we can then express r receiving only ρ_z at time t with a boolean expression involving firing events, as follows:

$$\mathsf{singl}(z,t) \;\equiv\; \mathsf{fire}(z, t - \mathit{depth}(z)) \,\wedge\, \forall_{u \in \mathcal{T}-\{z\}} \neg\, \mathsf{fire}(u, t - \mathit{depth}(u)). \tag{6}$$

By the algorithm, all the firing events in this expression are independent, so the probability of the combined event (6) is equal exactly $\frac{1}{n}(1-\frac{1}{n})^{n-1}$. As explained in the first paragraph of the proof, this event's occurrence implies that in the BFF model ρ_z will be received by r at time t; thus the first claim of the lemma follows.

To show the second claim, consider some fixed vertex z. The event that r receives ρ_z (in the BFF model) can also be described by a boolean expression composed of firing events $\mathsf{fire}(u, t - \mathit{depth}(u))$. This expression is significantly more complicated than the one in (6) and it is dependent on the tree topology: in addition to $\mathsf{fire}(z, t - \mathit{depth}(z))$ it needs to say that all rumors $\rho_u \neq \rho_z$ for which $\mathsf{fire}(u, t - \mathit{depth}(u))$ holds (that is, they might potentially collide with ρ_z) collide between themselves before meeting ρ_z. The actual formula for this expression is not important – what matters is that, like in (6), the expressions corresponding to different times t involve disjoint sets of firing events. All these firing events are independent. Therefore the events of r receiving ρ_z, for different times t, are independent as well. ☐

Instead of analyzing rumor arrival times directly, we will reduce the problem to the *slowed down* variant of the Coupon Collector's Problem (SD-CCP) where, at each time step:

- With probability $p = \left(1 - \frac{1}{n}\right)^{n-1}$ we make a decision to collect (otherwise we skip the step).
- If we make a decision to collect, then each of the n coupons is chosen uniformly at random.

Lemma 10 implies that, beginning at time $n-1$, the rate at which Algorithm RTree gathers rumors at r is at least as large as the rate at which coupons are collected in SD-CCP. More formally, for each time $t \geq n-1$ and integer j, the probability that r has received at least j different rumors by time t is at least as large as the probability of collecting at least j coupons in the first t steps. Therefore, to estimate the expected running time and its concentration bounds for Algorithm RTree, it is sufficient to establish the corresponding results for SD-CCP.

In the SD-CCP process, let X_1 be the number of steps until the first decision to collect, and for $i \geq 2$ let X_i be the number of steps between the $(i-1)$th and ith decision to collect. All variables X_i are independent, identically distributed, and their expectation is $\mathbb{E}[X_i] = 1/p \leq \mathrm{e}$. Let also T_{cc} denote the random variable equal to the number of steps in the standard Coupon Collector's Problem (or, equivalently, the number of decisions to collect in SD-CCP) until all coupons are collected. It is well known that $\mathbb{E}[T_{\mathsf{cc}}] = nH_n$, where H_n is the n-th Harmonic number (see, for example, Section 3.6 of [32]). Treating T_{cc} as the stopping time, we can apply Wald's Equation [39], which gives us that the expected time until all coupons are collected in SD-CCP is

$$\mathbb{E}[\textstyle\sum_{i=1}^{T_{\mathsf{cc}}} X_i\,] = (1/p) \cdot \mathbb{E}[\,T_{\mathsf{cc}}\,] \leq \mathrm{e}nH_n = (\mathrm{e} + o(1))n \ln n.$$

As explained earlier, Lemma 10 implies that $(\mathrm{e} + o(1))n \ln n$ is also an upper bound for the expected time for all rumors to reach the root in Algorithm RTree.

An extension of this argument establishes concentration. We claim that with probability $1 - o(1)$ Algorithm RTree will collect all rumors in the first $(\mathrm{e} + o(1))n \ln n$ steps. By Lemma 10, it is sufficient to show the corresponding bound for SD-CCP.

Let $T = \mathrm{e}n(\ln n + 2 \ln \ln n)$. Let also D_t be the number of decisions to collect in the first t steps of the SD-CCP process. The expectation of D_t is $\mathbb{E}[D_t] = tp \geq t/\mathrm{e}$. According to the Chernoff bound (see [32], for example), for any $\delta \in (0, 1)$ we have

$$\mathbb{P}[\,D_T < (1-\delta)T/\mathrm{e}\,] \leq \mathbb{P}[\,D_T < (1-\delta)Tp\,] < \mathrm{e}^{-Tp\delta^2/2} \leq \mathrm{e}^{-T\delta^2/2\mathrm{e}}.$$

Taking $\delta = 1/\ln n$, the above bound implies that, for sufficiently large n,

$$\begin{aligned} \mathbb{P}[\,D_T < n(\ln n + \ln \ln n)\,] &\leq \mathbb{P}[\,D_T < (1-\delta)T/\mathrm{e}\,] \\ &\leq \mathrm{e}^{-T\delta^2/2\mathrm{e}} \;=\; o(1). \end{aligned} \tag{7}$$

In other words, with probability $1 - o(1)$, in T steps of SD-CCP we will make at least $n(\ln n + \ln \ln n)$ decisions to collect. Conditioned on this occurring, the expected number U of uncollected coupons can be bounded as follows:

$$\begin{aligned} \mathbb{E}[U] &\leq n\left(1 - \tfrac{1}{n}\right)^{n(\ln n + \ln \ln n)} \\ &\leq n\mathrm{e}^{-\ln n - \ln \ln n} \\ &= \mathrm{e}^{-\ln \ln n} \;=\; \tfrac{1}{\ln n} = o(1). \end{aligned}$$

It follows from Markov's inequality that $\mathbb{P}[U \geq 1] \leq \mathbb{E}[U] = o(1)$. Thus, conditioned on making at least $n(\ln n + \log\log n)$ decisions to collect, the probability of collecting all coupons in SD-CCP is also $1 - o(1)$. Combining it with (7), we conclude that the probability that in T steps of SD-CCP all coupons wil be collected is $1 - o(1)$ as well.

In terms of our gathering problem, applying Lemma 10, the bound in the above paragraph gives us that with probability $1 - o(1)$ Algorithm RTree will gather all rumors in the root in at most $n - 1 + T = (\mathrm{e} + o(1))n \ln n$ steps.

It remains to argue that we can convert Algorithm RTree into the standard model, where receiving and transmitting at the same time is not allowed. We modify the algorithm in two ways: (i) the firing probability is reduced to $\frac{1}{2n}$ (this is done both to simplify analysis and to improve the constant in Theorem 7 resulting from the below argument), and (ii) each node enters the transmitting state whenever it transmits and the receiving state otherwise. If a node v received a rumor in the previous step and now the outcome of its random choice is to fire, the action of v is arbitrary (it can transmit its own message, forward the message received in the previous step, or go into the receiving state). Note that now some rumors may get rejected when they are transmitted (without collision) to a node that happens to be in the transmitting state, or when they get transmitted to a node whose random choice is to fire in the next step.

We claim that our analysis above can be extended to this new algorithm. The key is to consider only rumors that arrive at r at even-numbered times. The outline of the argument is this:

- We first prove an analog of Lemma 10, claiming that for each even-numbered time $t \geq n$ the probability that r receives rumor ρ_z at time t is at least $\frac{1}{2n}(1 - \frac{1}{2n})^{2n-1}$. The proof also uses the reduction with rumor bundles. We observe that if the rumor bundle received by r at time t is a singleton $\{\rho_z\}$ and the rumor bundle received at time $t - 1$ is empty then in our algorithm ρ_z will reach r at time t. By the same argument as in Lemma 10, the probability of this event is at least $\frac{1}{2n}(1 - \frac{1}{2n})^{2n-1}$, as needed. The proof that the events of r receiving ρ_z at different even times are independent is also analogous to the argument in the proof of Lemma 10.

- Then, as before, we proceed by reduction to the slowed-down variant of Coupon Collector's Problem, but now we use an instance with $2n$ coupons, even though we only have n rumors. (Say, we can associate coupons numbered $0, 1, ..., n - 1$ with rumors, while the remaining coupons $n, n + 1, ..., 2n - 1$ are "dummy" coupons.) In the slowed-down variant, the decision to collect is now made with probability $(1 - \frac{1}{2n})^{2n-1}$, and then each of $2n$ coupons is collected with equal probability. The slow-down factor is still $\mathrm{e} + o(1)$, but since we only use even steps, and since we have $2n$ coupons, the expected (and high-probability) running time will now be $(4 + o(1))\mathrm{e}n \ln n$.

Theorem 7. *The modified Algorithm* RTree *has expected running time at most* $(4 + o(1))\,\mathrm{e}n \ln n$ *in the standard model. In fact, it will complete gathering in time* $(4 + o(1))\,\mathrm{e}n \ln n$ *with probability at least* $1 - o(1)$.

8 An $\Omega(n \log n)$ Lower Bound for Randomized Algorithms

In this section we show that Algorithm RTree is within a constant factor of optimal for the model without node labels, even if the topology of the tree is known in advance. Actually we will show something a bit stronger, namely that there is a constant c such that any algorithm for unlabelled trees with running time less than $cn \ln n$ will almost surely have some rumors fail to reach the root on certain trees.

The specific tree we will use here is that of the *n-star*, illustrated in Figure 6, consisting of the root with n children that are also the leaves in the tree. (We use $n + 1$ nodes instead of n, for convenience, but this does not affect our asymptotic lower bound.) In the n-star, these leaves are entirely isolated: no leaf receives any information from the root or from any other leaf. Thus at each time step t, each leaf v transmits with a probability that can depend only on t and on the set of previous times at which v transmitted. As the actions of the root r are irrelevant, we will use the terms "node" and "leaf" synonymously.

For the n-star, Algorithm RTree will have each leaf node transmit with probability $\frac{1}{n+1}$ at each time step, regardless of when or whether it had transmitted previously. What makes our argument quite complicated is that, in general, the actions of a node v at different steps may not be independent. In fact, as it turns out, allowing dependence between time steps does lead to a constant factor improvement on star graphs (see Theorem 9), but improvement beyond this constant factor is not possible (see Theorem 8).

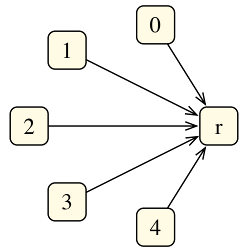

Figure 6: The n-star graph for $n = 5$.

Any randomized protocol $\mathcal{R}$ for the n-star that does not use node labels and completes in T time steps can be represented as a probability distribution over all subsets of $\{0, 1, \ldots, T-1\}$. In this algorithm, each node v independently picks a subset F_v according to this distribution (the same for all nodes), and then transmits only at the times in F_v. We say that node v *succeeds in transmitting* if there is a time $t \in \{0, 1, ..., T-1\}$ such that $t \in F_v$, but $t \notin F_w$ for any $w \neq v$.

The main result of this section is the following lower bound.

Theorem 8. *If $\mathcal{R}$ is a randomized protocol for information gathering on trees then, for certain trees, the expected running time of $\mathcal{R}$ is $\Omega(n \ln n)$. More specifically, if we run $\mathcal{R}$ on the n-star graph for $T \leq cn \ln n$ steps, where $c < \frac{1}{\ln^2 2}$, then with probability $1 - o(1)$ at least one rumor will fail to reach the root.*

As we show, our lower bound is very tight, in the sense that the value of the constant $\frac{1}{\ln^2 2} \approx 2.08$ in the above theorem is best possible for star graphs.

Theorem 9. *If $T = cn \ln n$, where $c > \frac{1}{\ln^2 2}$, then there is a protocol which succeeds on the n-star in time at most T with probability $1 - o(1)$.*

Recall from Section 7 that on the same star graph, Algorithm RTree takes time approximately $\mathrm{e}n \log n$. So allowing dependence between time steps leads to a roughly 25% improvement in running time.

The proof of Theorem 8 is given in Sections 8.1-8.3. The proof of Theorem 9 is in Section 8.4.

8.1 Proof of Theorem 8 (Under Two Assumptions)

Overloading notation, we will simply write $\mathcal{R}$ for the probability distribution on transmission sets $F_v \subseteq \{0, 1, ..., T-1\}$ induced by Algorithm $\mathcal{R}$. To prove Theorem 8, we will first assume that $\mathcal{R}$ satisfies two additional simplifying assumptions for every node v and $n \geq 3$:

Assumption 1: $|F_v| \leq \ln^{10} n$ with probability 1.

Assumption 2: $\frac{1}{n^3} \leq \mathbb{P}[t \in F_v] \leq \frac{\ln^4 n}{n}$, for each time t.

Later, in Section 8.3, we will show how to remove these assumptions.

For each time t, let $q_t = \mathbb{P}[t \in F_v]$ be the (unconditional) probability that a given node transmits at time t (which, by Assumption 2, is between $\frac{1}{n^3}$ and $\frac{\ln^4 n}{n}$). Then the conditional probability that a node v fails to transmits its rumor at time t, given that it attempted to transmit, is

$$\mathbb{P}\left[\, v \text{ fails at time } t \mid t \in F_v \,\right] = 1 - (1 - q_t)^{n-1} \geq 1 - \mathrm{e}^{-(n-1)q_t}.$$

What we would like to do is to use this inequality to lower-bound the probability of v's rumor not reaching r after T steps, given v's transmission set F_v. If the transmissions were independent, this would be easy: just take the product of the above failure probabilities at each time in F_v. We cannot do this for arbitrary distributions because of possible dependencies between transmissions, but, as the lemma below shows, this idea still nearly works.

Lemma 11. *For any transmission distribution $\mathcal{R}$ that satisfies Assumptions 1 and 2, and any fixed v and F_v, we have*

$$\mathbb{P}[\, v \text{ fails} \mid F_v\,] \geq \prod_{t \in F_v} \left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)}\right),$$

where "v fails" is the event that the rumor from v does not reach the root by time T.

In general, the lower bound in Lemma 11 is not tight. As an example, consider the case where $T = 2n$ and each node transmits at times $2i$ and $2i+1$, for a uniformly chosen $i \in \{0, 1, ..., n-1\}$. Then if a node's first transmission collides, it increases the likelihood of failure in the next step; in fact its second transmission is *guaranteed* to fail. What the lemma states, in essence, is that dependencies between transmissions at different times cannot make transmission failures significantly *less* likely.

For now we will assume that Lemma 11 holds and we will continue with the proof of Theorem 8. We will return to the proof of Lemma 11 later, in Section 8.2.

From now on, fix some constant $c < \frac{1}{\ln^2 2}$. To allow asymptotic notation in the calculations, we consider n to be some sufficiently large integer. Let X be the random variable equal to the number of nodes whose rumors fail to reach the root in the first T steps of Algorithm $\mathcal{R}$, where $T \leq cn \ln n$. We next show that the expectation of X is large, which is equivalent to showing that the probability that any fixed node v fails is significantly larger than $1/n$. By Lemma 11, we have

$$\begin{aligned}
\mathbb{P}[\, v \text{ fails}\,] &= \mathbb{E}_{F_v}\left[\mathbb{P}\left[v \text{ fails} \,|\, F_v\right]\right] \\
&\geq \mathbb{E}_{F_v}\left[\prod_{t\in F_v}\left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)}\right)\right] \\
&= \mathbb{E}_{F_v}\left[\prod_{t=0}^{T-1}\left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)} \cdot \chi(t \in F_v)\right)\right].
\end{aligned}$$

Here $\chi(t \in F_v)$ is an indicator function equal to 1 if $t \in F_v$ and 0 otherwise. We now apply a variant of Jensen's inequality (see, e.g., Section 6.5.2 of [34]) which states that $f(\mathbb{E}[Y]) \geq \mathbb{E}[f(Y)]$, for a random variable Y and a concave function $f(Y)$. Using this inequality, we have

$$\begin{aligned}
\ln \mathbb{P}[\, v \text{ fails}\,] &\geq \ln \mathbb{E}_{F_v}\left[\prod_{t=0}^{T-1}\left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)} \cdot \chi(t \in F_v)\right)\right] \\
&\geq \mathbb{E}_{F_v}\left[\ln\left(\prod_{t=0}^{T-1}\left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)} \cdot \chi(t \in F_v)\right)\right)\right] \\
&= \sum_{t=0}^{T-1} \mathbb{E}_{F_v}\left[\ln\left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)} \cdot \chi(t \in F_v)\right)\right] \\
&= \sum_{t=0}^{T-1} q_t \ln\left(1 - \mathrm{e}^{-nq_t\left(1-n^{-5/6+o(1)}\right)}\right) \\
&\geq \frac{T}{n\left(1-n^{-5/6+o(1)}\right)} \inf_{x>0}\left(x \ln\left(1 - \mathrm{e}^{-x}\right)\right) \\
&= -(c + o(1)) \ln^2 2 \ln n.
\end{aligned}$$

The last equality holds because $x \ln(1 - \mathrm{e}^{-x})$, for $x > 0$, is minimized when $x = \ln 2$.

Let $\epsilon = 1 - c \ln^2 2 > 0$. The above bound on $\ln \mathbb{P}[v \text{ fails}]$ implies that

$$\begin{aligned}
\mathbb{P}[\, v \text{ fails}\,] &\geq \mathrm{e}^{-(c+o(1))\ln^2 2 \ln n} \\
&= n^{-c\ln^2 2 - o(1)} = n^{-1+\epsilon-o(1)}.
\end{aligned}$$

This bound holds for any individual node v, so the expected number of nodes that fail is $\mathbb{E}[X] \geq n \cdot n^{-1+\epsilon-o(1)} = n^{\epsilon-o(1)}$.

To complete the proof of Theorem 8 (i.e. to show that X is almost surely positive), we need to establish concentration around this expectation of X, for which we utilize Talagrand's inequality. Think of X as a function of transmission sets, $X = X(F_0, F_1, \ldots, F_{n-1})$. This function X is Lipschitz, in the sense that changing a single F_w can only change X by at most $\ln^{10} n$. (By Assumption 1, node w transmits at most $\ln^{10} n$ times, and each transmission can only interfere with at most one otherwise successful transmission.) Furthermore, X is locally certifiable in the following sense: If $X \geq k$ for some k, then there is a subset I of at most $k(\ln^{10} n + 1)$ nodes such that X remains larger than k no matter how we change the transmission patterns of the nodes outside I. For example, we can construct I as follows. Start with I_0 being a set of k nodes that failed to transmit successfully. Then, for each $v \in I_0$ let J_v be a set of at most $\ln^{10} n$ nodes such that whenever v transmits then at least one of the nodes in J_v also transmits. Then the set $I = I_0 \cup \bigcup_{v \in I_0} J_v$ has the desired properties.

Let m be the median value of X (not the mean). It follows from the above two properties of X, together with Talagrand's Inequality [37] (see Section 7.7 of [1] for the specific version used here), that for any $\gamma > 0$ we have

$$\mathbb{P}\left[\, |X - m| \geq \gamma \ln^{15} n \sqrt{2m} \,\right] \leq 4\mathrm{e}^{-\gamma^2/4}. \tag{8}$$

In particular, applying this bound with $\gamma = \sqrt{m/2} \ln^{-15} n$ gives

$$\mathbb{P}[\, X = 0 \,] \leq \mathbb{P}[\, |X - m| \geq m \,] \leq 4\mathrm{e}^{-m \ln^{-30} n/8} \tag{9}$$

To finish, it is therefore enough to show that m is large.

Taking $\gamma = \sqrt{2} \ln n$ in (8), we have

$$\mathbb{P}\left[\, X \geq m + 2\ln^{16} n \sqrt{m} \,\right] \leq 4\mathrm{e}^{-\ln^2 n/2}. \tag{10}$$

Since X is always at most n, inequality (10) implies that the contribution to $\mathbb{E}[X]$ from the values of X that are at least $m + 2\ln^{16} n\sqrt{m}$ is at most $4\mathrm{e}^{-\ln^2 n/2} \cdot n = o(1)$. On the other hand, the contribution to $\mathbb{E}[X]$ from the remaining values of X can be at most $m + 2\ln^{16} n\sqrt{m}$. It follows that

$$n^{\epsilon - o(1)} \leq \mathbb{E}[X] \leq m + 2\ln^{16} n\sqrt{m} + o(1),$$

from which we have $m \geq n^{\epsilon - o(1)}$. Substituting this bound for m into the right-hand side of (9), we obtain that

$$\mathbb{P}[\, X = 0 \,] \leq 4\mathrm{e}^{-n^{\epsilon - o(1)} \ln^{-30} n/8} = o(1),$$

which is exactly the claim in the second part of Theorem 8.

It remains to prove Lemma 11 and to remove Assumptions 1 and 2, that we originally placed on the distribution of transmission times.

8.2 Proof of Lemma 11

We present first a rough idea behind our argument: Suppose we fixed F_v and all of the q_t, and tried to choose our distribution so as to minimize the probability that v fails. (Here and throughout this section we will say that v "fails" if no transmission from v reaches the root in T steps. Otherwise we say that v "succeeds"). As the example discussed following the statement of Lemma 11 shows, we can increase the probability of v's failure by choosing a distribution in which a single other node interferes with multiple transmissions from v. Conversely, one might hope that to minimize v's failure probability the distribution should be such each node interferes with at most one transmission time in F_v. In this section we give a rigorous argument that formalizes this intuition.

For the remainder of this section, we will treat F_v as fixed (for $u \neq v$, F_u remains distributed according to $\mathcal{R}$). For each $U \subseteq F_v$ and a node $u \neq v$, define y_U by

$$y_U = \mathbb{P}[\, F_u \cap F_v = U \,].$$

This definition does not depend on our choice of u, as the distributions of sets F_u are identical for all $u \neq v$. Note that we allow U to be $\emptyset$, and that $q_t = \sum_{t \in U \subseteq F_v} y_U$, for $t \in F_v$. We can think of y as the restriction of our distribution $\mathcal{R}$ to time steps in F_v. Since the success or failure of v depends only on what happens at times in F_v, the probability that v fails remains identical if we replace distribution $\mathcal{R}$ by that of y, for each $u \neq v$. The intuition in the previous paragraph is formalized in the following claim.

Claim 3. *There is a distribution $\tilde{y}$ on subsets of F_v with the following properties:*

(i) *$\tilde{y}$ is supported on $\{\{t\} : t \in F_v\} \cup \{\emptyset\}$, namely on subsets of F_v of cardinality at most 1. (That is, $\tilde{y}_U = 0$ for all $U \subseteq F_v$ with $|U| \geq 2$.)*

(ii) *If a node $u \neq v$ transmits according to $\tilde{y}$ then the probability that u transmits at a time $t \in F_v$ is q_t (same as in y).*

(iii) *The probability that v fails if all nodes other than v transmit according to $\tilde{y}$ is no larger than if they transmit according to y.*

Above, when we say that "u transmits according to y" (resp. $\tilde{y}$), we mean that it chooses each transmission set $U \subseteq F_v$ with probability y_U (resp. $\tilde{y}_U$) and then, once its transmission set U is chosen, it transmits at each time in U.

Let us for now assume the truth of Claim 3. (The proof of Claim 3 is given in Section 8.2.1.) In the rest of this section we assume that all nodes $u \neq v$ transmit according to distribution $\tilde{y}$, and we prove a lower bound on the probability of failure (analogous to Lemma 11) under this assumption. By Claim 3(iii), this is sufficient to prove Lemma 11.

Let us now fix $F_v = \{t_1, t_2, ..., t_k\}$, where $t_1 < t_2 < \cdots < t_k$. By Assumption 1, we have $k \leq \ln^{10} n = n^{o(1)}$. For each $i = 1, 2, ..., k$, let g_i be the random variable equal to the number of nodes other than v that transmit at time t_i, and let E_i be the event that $1 \leq g_i \leq n^{1/6}$. We can bound the conditional probability that v fails given F_v from below by the probability that every event E_i occurs, that is

$$\mathbb{P}[\, v \text{ fails} \mid F_v \,] \geq \mathbb{P}\Big[\bigwedge_{i=1}^{k} E_i \Big] = \prod_{i=1}^{k} \mathbb{P}\left[\, E_i \mid E_1 \wedge \cdots \wedge E_{i-1} \,\right]. \tag{11}$$

To estimate the right-hand side of (11), we will bound each term in this product directly by separately bounding the conditional probabilities that $g_i = 0$ and that $g_i > n^{1/6}$.

We start with some auxiliary estimates. Let p_{i-1} be the (unconditional) probability that a node $u \neq v$ transmits at least once during $t_1, t_2, \ldots, t_{i-1}$. By Claim 3(i), u transmits at most once in F_v, so we have

$$p_{i-1} \;=\; \textstyle\sum_{j=1}^{i-1} q_{t_j} \;\leq\; i \cdot \max_t q_t \;\leq\; \ln^{10} n \cdot \frac{\ln^4 n}{n} \;=\; \frac{\ln^{14} n}{n},$$

where the second inequality follows from Assumptions 1 and 2. Let q'_{t_i} be the probability with which a node $u \neq v$ transmits at a time t_i, provided that it did not transmit at times $t_1, ..., t_{i-1}$. Then we have

$$q'_{t_i} \;=\; \tfrac{q_{t_i}}{1-p_{i-1}} \;\leq\; \tfrac{q_{t_i}}{1-(\ln^{14} n)/n} \;=\; q_{t_i}(1+o(1)).$$

Thus, using Assumption 2 again, probabilities q_{t_i} and q'_{t_i} satisfy:

$$q_{t_i} \;\leq\; q'_{t_i} \;=\; q_{t_i}(1+o(1)) \;\leq\; \tfrac{\ln^4 n}{n}(1+o(1)). \tag{12}$$

Now, to show (11), we fix some i and assume that all events $E_1, \ldots, E_{i-1}$ have occurred. Then the set A of nodes that did not transmit at any time $t_1, t_2, ..., t_{i-1}$ has cardinality at least $n - 1 - n^{1/6}k = n - n^{1/6+o(1)}$ and at step i each of them will transmit with probability q'_{t_i}. Restricting attention to the nodes in A and using (12), we estimate the probability that $g_i = 0$, as follows:

$$\begin{aligned}
\mathbb{P}\left[\, g_i = 0 \mid E_1 \wedge \cdots \wedge E_{i-1} \,\right] &\leq \left(1 - q'_{t_i}\right)^{|A|} && (13)\\
&\leq \left(1 - q'_{t_i}\right)^{n - n^{1/6+o(1)}} \\
&\leq \mathrm{e}^{-q'_{t_i}\left(n - n^{1/6+o(1)}\right)} \;=\; \mathrm{e}^{-nq_{t_i}\left(1 - n^{-5/6+o(1)}\right)}. && (14)
\end{aligned}$$

(The second inequality in the above derivation follows from $1 - x \leq \mathrm{e}^{-x}$, for $x \geq 0$.)

To bound the probability that g_i is large, we use the union bound. If $g_i > n^{1/6}$, there must be some set of nodes B of cardinality $n^{1/6}$, not including v, all of whose nodes transmit at time t_i. (To avoid clutter, here and in the calculations below we write $n^{1/6}$ to mean $\lceil n^{1/6} \rceil$.) Denote by D_B the event that all nodes in B transmit at time t_i. If H is an event that describes the complete history of transmissions from B at steps $t_1, ..., t_{i-1}$, then either $\mathbb{P}[D_B \mid H] = 0$, if some node in B already transmitted at one of these times, or otherwise $\mathbb{P}[D_B \mid H] = {q'_{t_i}}^{|B|}$. This implies that $\mathbb{P}[D_B \mid E_1 \wedge \cdots \wedge E_{i-1}] \le {q'_{t_i}}^{|B|}$. Applying this inequality and $|B| = n^{1/6}$, we get

$$\begin{aligned}
\mathbb{P}\big[\, g_i > n^{1/6} \mid E_1 \wedge \cdots \wedge E_{i-1} \,\big] &\le \textstyle\sum_{|B|=n^{1/6}} \mathbb{P}[\, D_B \mid E_1 \wedge \cdots \wedge E_{i-1} \,] \\
&\le \tbinom{n}{n^{1/6}} {q'_{t_i}}^{n^{1/6}} \\
&\le \tbinom{n}{n^{1/6}} \Big(\tfrac{(1+o(1))\ln^4 n}{n}\Big)^{n^{1/6}} && (15)\\
&\le \Big(\tfrac{2\mathrm{e}\ln^4 n}{n^{1/6}}\Big)^{n^{1/6}} && (16)\\
&= \big(n^{-1/6+o(1)}\big)^{n^{1/6}} = \mathrm{e}^{(-1/6+o(1))n^{1/6}\ln n} = \mathrm{e}^{-\omega(n^{1/6})}, && (17)
\end{aligned}$$

where inequality (15) follows from (12), and inequality (16) follows from $1 + o(1) \le 2$ (for n large enough) and from $\binom{n}{k} \le (\frac{n\mathrm{e}}{k})^k$. Summing the bounds (14) and (17) and taking logarithms, we obtain

$$\begin{aligned}
\ln \mathbb{P}\,[\, \neg E_i \mid E_1 \wedge \cdots \wedge E_{i-1} \,] &\le \ln\Big(\mathrm{e}^{-nq_{t_i}\left(1-n^{-5/6+o(1)}\right)} + \mathrm{e}^{-\omega(n^{1/6})}\Big) \\
&\le -nq_{t_i}\Big(1 - n^{-5/6+o(1)}\Big) + \frac{\mathrm{e}^{-\omega(n^{1/6})}}{\mathrm{e}^{-nq_{t_i}\left(1-n^{-5/6+o(1)}\right)}} && (18)\\
&\le -nq_{t_i}\Big(1 - n^{-5/6+o(1)}\Big) + \mathrm{e}^{-\omega(n^{1/6})}\mathrm{e}^{\ln^4 n} && (19)\\
&= -nq_{t_i}\Big(1 - n^{-5/6+o(1)}\Big). && (20)
\end{aligned}$$

We now justify the steps of this derivation. Inequality (18) follows from $\ln(y+z) \le \ln y + \frac{z}{y}$ for all $y, z > 0$. Inequality (19) holds because, by Assumption 2, $nq_{t_i} \le \ln^4 n$. In expression (19), the value of $nq_{t_i} \cdot n^{-5/6+o(1)}$ is at least $(1-o(1))n^{-17/6}$ in magnitude (by the lower bound in Assumption 2) while the second term is smaller than any polynomial. This leads to the absorption in the final step (20).

We thus have

$$\mathbb{P}\,[\, E_i \mid E_1 \wedge \cdots \wedge E_{i-1} \,] \ge 1 - \mathrm{e}^{-nq_{t_i}\left(1-n^{-5/6+o(1)}\right)}. \tag{21}$$

Substituting the bound (21), for each i, into the right-hand side of (11), we obtain the inequality in Lemma 11.

To complete the proof of Lemma 11, it remains to prove Claim 3.

8.2.1 Proof of Claim 3

The idea of this proof will be to repeatedly apply a series of transformations on the distribution y. Each transformation will in turn decrease the total mass of y on non-singleton subsets, without increasing the failure probability. In the arguments below we assume (tacitly) that n is sufficiently large.

We first observe that the total mass of y on any particular subset U of size at least 2 cannot be too large:

Claim 4. *For any $U \subseteq F_v$ with $|U| \ge 2$, we have $y_U < y_\emptyset$.*

Proof. Let f_u be the expected number of times a node $u \ne v$ transmits during F_v. (Note that f_u does not actually depend on u.) On the one hand, we have

$$f_u = \sum_{t \in F_v} q_t \le |F_v| \cdot \max_{t \in F_v} q_t \le \ln^{10} n \cdot \frac{\ln^4 n}{n} < 1, \tag{22}$$

where in the second inequality we used Assumptions 1 and 2. On the other hand, for a set $U \subseteq F_v$ with

$|U| \geq 2$, we can write

$$\begin{aligned} f_u &= \sum_{S \subseteq F_v} |S| y_S \\ &= |U| y_U + \sum_{\substack{S \subseteq F_v \\ S \neq U, \emptyset}} |S| y_S \\ &\geq 2 y_U + \sum_{\substack{S \subseteq F_v \\ S \neq U, \emptyset}} y_S = 1 + y_U - y_\emptyset. \end{aligned}$$

Comparing the two bounds on f_u, the inequality $y_U < y_\emptyset$ follows. $\square$

Consider some $U \subseteq F_v$ with $|U| \geq 2$. We now define a new distribution y' as follows: Fix some $\tau \in U$, and define

$$y'_U = 0 \qquad y'_{U-\{\tau\}} = y_{U-\{\tau\}} + y_U$$
$$y'_{\{\tau\}} = y_{\{\tau\}} + y_U \qquad y'_\emptyset = y_\emptyset - y_U$$

with $y'_B = y_B$ for all $B \neq U$. Effectively, what we are doing here is moving mass y_U from each of U and $\emptyset$ to each of $U - \{\tau\}$ and $\{\tau\}$ (Claim 4 guarantees that $\emptyset$ has enough mass to perform this operation). Note that this operation does not change the value of q_t for any $t \in F_v$. This can be seen by expressing

$$q_t = \sum_{W : t \in W \subseteq F_v} y_W.$$

If $t \notin U$, all sets containing t have their probabilities unchanged by the transformation. If $t \neq \tau$ and $t \in U$, then the loss in y_U is exactly cancelled out by the gain in $y_{U-\{\tau\}}$. Similarly, if $t = \tau$, then the loss in y_U exactly cancels with the gain in $y_{\{\tau\}}$.

The intuition above is that this separates the transmission times, so this should reduce the failure probability of v. Actually, a stronger monotonicity property is true:

Claim 5. *Fix some node $w \neq v$, and assume that, for each node $u \in V - \{v, w\}$, the transmission set of u within F_v is distributed either according to y or according to y'. Then the probability that v fails if w transmits according to y' is not larger than the probability that v fails if w transmits according to y.*

Proof. It is sufficient to prove the claim for the failure probabilities conditioned on all transmission patterns $F_u \cap F_v$, for $u \in V - \{v, w\}$, having some fixed (but arbitrary) values. This is because, once we prove the claim for such conditional probabilities, the claim for non-conditional probabilities will follow by simple averaging. (In fact, this remains true if the transmission sets within F_v of all $u \notin \{v, w\}$ are arbitrary, not necessarily restricted to y or y'.)

So let us fix all sets $F_u \cap F_v$, for $u \in V - \{v, w\}$. The argument below will be conditioned on these patterns being fixed. From this point on, the proof of the claim essentially comes from direct verification.

Let $S = F_v - \bigcup_{u \notin \{v,w\}} F_u$; that is, S is the set of times at which v transmits but no node $u \in V - \{v, w\}$ transmits. So v can only succeed at the times in S when w does not transmit. Let $\hat{F}_w = F_v \cap F_w$ be the random variable representing the transmission times of w within F_v. In this notation, v will fail if and only if $S \subseteq \hat{F}_w$.

We now analyze how moving the mass between distributions y and y' affects the probability of v failing, that is the probability of the event that $S \subseteq \hat{F}_w$. It is sufficient to consider only the four sets $\hat{F}_w \in \{U, \{\tau\}, U - \{\tau\}, \emptyset\}$, since the probabilities of all other sets $\hat{F}_w$ remain unchanged. We have four cases:

Case 1: $S = \emptyset$. Then v fails for all four choices of $\hat{F}_w$, so moving the mass does not affect the failure probability.

Case 2: $S - U \neq \emptyset$. Then moving the mass again has no impact on the success probability, because v succeeds for all four choices of $\hat{F}_w$.

Case 3: $S \subseteq U$ and $|S| \geq 2$. Then v fails when $\hat{F}_w = U$, but succeeds when $\hat{F}_w = \{\tau\}$. So we are moving mass from a pair of elements with one failure ($\hat{F}_w = U$) and one success ($\hat{F}_w = \emptyset$) to a pair of elements with at least one success (for $\hat{F}_w = \{\tau\}$); therefore we cannot increase the failure probability.

Case 4: $S = \{\tau\}$. Then the operation moves mass from a pair of elements with one failure ($\hat{F}_w = U$) and one success ($\hat{F}_w = \emptyset$) to a pair of elements with one failure ($\hat{F}_w = \{\tau\}$) and one success ($\hat{F}_w = U - \{\tau\}$). So the failure probability is unchanged.

This completes the proof of Claim 5. □

The complete process of converting y into $\tilde{y}$ is this. We start with all nodes different from v transmitting according to y. Then, we proceed in phases. In each phase we choose some $U \subseteq F_v$ in the support of y that satisfies $|U| \geq 2$, and we apply the $y \to y'$ change of distribution, described above, in succession to each node $w \neq v$. Observe that within a phase, some nodes $u \in V - \{v, w\}$ will use distribution y while others will use the new distribution y'. At the end, all nodes other than v will transmit according to y'. Since Claim 5 holds at each step, we obtain that the probability that v fails when all of the nodes other than v transmit according to y' is no larger than when they all transmit according to y.

As explained earlier, y' has the same values of all probabilities q_t as y. Also, in y' each node transmits on average less than once, which is the only specific property (namely inequality (22)) that we needed in order to transform y into y'. Therefore, if after the phase the support of y' still contains a set U with $|U| \geq 2$, we can proceed to the next phase where we apply again the same transformation with y' in place of y.

After each phase the sum of the squares of the cardinalities of the sets in the support of y decreases, so this process must terminate after a finite number of phases. After the last phase, the resulting distribution y' must only have singletons and the empty set in its support. Thus letting $\tilde{y}$ be this y', we obtain Claim 3.

8.3 Removing Assumptions 1 and 2

So far, in Sections 8.1 and 8.2, we have shown that any probability distribution $\mathcal{R}$ on the sets F_v of transmission times that satisfies both Assumptions 1 and 2 will lead to at least one rumor failing to reach the root with high probability. Recall that Assumption 1 states that $|F_v| \leq \ln^{10} n$ with probability 1, while Assumption 2 states that $\frac{1}{n^3} \leq q_t \leq \frac{\ln^4 n}{n}$, where $q_t = \mathbb{P}[t \in F_v]$ is the (unconditional) probability that a given node v transmits at a time t. To complete the proof of Theorem 8, we now show how to eliminate these assumptions.

Eliminating Assumption 1. We first consider distributions satisfying Assumption 2, but not necessarily Assumption 1. For such a distribution, the expected number of times when a given node v transmits is

$$\mathbb{E}[\,|F_v|\,] = \textstyle\sum_{t=0}^{T-1} q_t \leq cn \ln n \cdot \frac{\ln^4 n}{n} = c \ln^5 n. \tag{23}$$

We now think of the transmission sets F_v as being generated in a two step exposure process:

- First, we expose the set V' of all nodes v for which $|F_v| \leq \ln^9 n$, and then,
- in the second step, the actual transmission sets F_v, for all nodes v, are exposed.

According to our assumption, $c < \frac{1}{\ln^2 2}$. Fix a sufficiently small $\epsilon > 0$ such that for $c' = \frac{c}{1-3\epsilon}$ we have $c' < \frac{1}{\ln^2 2}$. Let $n' = |V'|$ and let E' be the event that $n' > (1-2\epsilon)n$. Let also E_{succ} be the event that all rumors are successfully transmitted in the first T steps. Our goal is to show that $\mathbb{P}[E_{\text{succ}}] = o(1)$. We have

$$\mathbb{P}[\,E_{\text{succ}}\,] \leq \mathbb{P}[\,\neg E'\,] + \mathbb{P}[\,E_{\text{succ}} \wedge E'\,]. \tag{24}$$

We bound each term on the right side of (24) separately.

For the first term, we note that by Markov's inequality and (23), we have for any particular v that

$$\mathbb{P}[\,v \notin V'\,] = \mathbb{P}[\,|F_v| > \ln^9 n\,] \leq \frac{\mathbb{E}[|F_v|]}{\ln^9 n} \leq \frac{c}{\ln^4 n},$$

which implies that

$$\mathbb{E}[n'] \geq \left(1 - \frac{c}{\ln^4 n}\right) n \geq (1-\epsilon)n, \tag{25}$$

for sufficiently large n. Thus, for n large enough, using (25) and the Chernoff bound [32] (which is applicable, because the events $v \in V'$ are independent for different nodes v), we have

$$\begin{aligned}\mathbb{P}[\,\neg E'\,] &= \mathbb{P}[\,n' \le (1-2\epsilon)n\,]\\ &\le \mathbb{P}[\,n' \le (1-\epsilon)\mathbb{E}[n']\,]\\ &\le \mathrm{e}^{-\mathbb{E}[n']\epsilon^2/2} \;\le\; \mathrm{e}^{-\epsilon^2(1-\epsilon)n/2} \;=\; o(1).\end{aligned} \tag{26}$$

We now turn to the second term of (24). First, we note that for $n' > (1-2\epsilon)n$, we have

$$T = cn\ln n < c\tfrac{n'}{1-2\epsilon}\ln(\tfrac{n'}{1-2\epsilon}) < \tfrac{c}{1-3\epsilon}n'\ln n' = c'n'\ln n',$$

where the last inequality holds for sufficiently large n. For a fixed set $V_0 \subseteq V$ with $|V_0| > (1-2\epsilon)n$, we now consider the distribution $\mathcal{R}'$ given by conditioning $\mathcal{R}$ on $V' = V_0$. (Thus for $\mathcal{R}'$ event E' holds.) Note that $\mathcal{R}'$ satisfies Assumption 1 for vertex set V_0 and sufficiently large n, because for $v \in V_0$ we have $|F_v| \le \ln^9 n \le \ln^{10} n'$. As shown in Section 8.1, since $c' < \frac{1}{\ln^2 2}$, this gives that with probability $1-o(1)$ there is a node $w \in V_0$ that fails to transmit successfully in the first $c'n'\ln n'$ steps, even if the nodes in $V - V_0$ do not transmit at all. Of course, including the nodes in $V - V_0$ can only make the failure probability larger. Since $c'n'\ln n' \ge T$, this implies an analogous statement for the original transmission distribution $\mathcal{R}$ of all nodes in V, namely that with probability $1-o(1)$ there will be a node $w \in V$ that will fail to deliver its rumor to r in the first $T = cn\ln n$ steps.

It follows that for any particular choice of V_0 satisfying $|V_0| > (1-2\epsilon)n$ we have $\mathbb{P}[E_{\text{succ}}|V' = V_0] = o(1)$. Averaging over all possible choices of such sets V_0, we have $\mathbb{P}[E_{\text{succ}}|E'] = o(1)$, which in turn implies $\mathbb{P}[E_{\text{succ}} \wedge E'] = o(1)$, because we also have $\mathbb{P}[E'] = 1-o(1)$, from (26).

Since both terms on the right side of (24) are $o(1)$, it follows that $\mathbb{P}[E_{\text{succ}}] = o(1)$ as well, completing the proof that Assumption 1 can be eliminated from Theorem 8.

Eliminating Assumption 2. It remains to consider distributions which do not necessarily satisfy Assumption 2. Let L_{low} and L_{high} be the time steps where Assumption 2 does not hold; specifically

$$L_{\text{low}} = \{t : q_t < \tfrac{1}{n^3}\} \quad \text{and} \quad L_{\text{high}} = \{t : q_t > \tfrac{\ln^4 n}{n}\}.$$

Let E_{low} and E_{high} be the events that at least one node transmits successfully at time steps in L_{low} and L_{high}, respectively. Then

$$\begin{aligned}\mathbb{P}[\,E_{\text{low}}\,] &\le \sum_{t\in L_{\text{low}}}\sum_{v}\mathbb{P}[\,v \text{ succeeds at time } t\,]\\ &\le \sum_{t\in L_{\text{low}}}\sum_{v}\mathbb{P}[\,v \text{ transmits at time } t\,]\\ &= \sum_{t\in L_{\text{low}}}\sum_{v} q_t \;\le\; Tn\cdot\tfrac{1}{n^3} \;\le\; \tfrac{c\ln n}{n} \;=\; o(1)\,,\end{aligned}$$

and

$$\begin{aligned}\mathbb{P}[\,E_{\text{high}}\,] &\le \sum_{t\in L_{\text{high}}}\sum_{v}\mathbb{P}[\,v \text{ succeeds at time } t\,]\\ &\le \sum_{t\in L_{\text{high}}}\sum_{v}\mathbb{P}[\,\text{each node } u \ne v \text{ does not transmit at time } t\,]\\ &\le \sum_{t\in L_{\text{high}}}\sum_{v}(1-q_t)^{n-1}\\ &\le Tn\left(1-\tfrac{\ln^4 n}{n}\right)^{n-1}\\ &\le cn^2\ln n\cdot(1+o(1))\mathrm{e}^{-\ln^4 n} \;=\; cn^{-\ln^3 n+2}\ln n\cdot(1+o(1)) \;=\; o(1).\end{aligned}$$

Let E_{mid} be the event that all nodes transmit successfully in a time not in $L_{\text{low}} \cup L_{\text{high}}$. Consider a modified distribution with all time steps in $L_{\text{low}} \cup L_{\text{high}}$ removed. More precisely, for each time step t let $\ell_t = |(L_{\text{low}} \cup L_{\text{high}}) \cap \{0, 1, ..., t-1\}|$. We replace each set $F_v \subseteq \{0, 1, ..., T-1\}$ of transmissions by

$$F'_v = \{t - \ell_t : t \in F_v - (L_{\text{low}} \cup L_{\text{high}})\}.$$

In this new distribution set F'_v has the same probability as F_v had in the original distribution, and $F'_v \subseteq \{0, 1, ..., T'-1\}$, for some $T' \le cn \ln n$. Since this new distribution satisfies Assumption 2 in the time interval $\{0, 1, ..., T'-1\}$, by the previous analysis we have $\mathbb{P}[E_{\text{mid}}] = o(1)$.

The event that all nodes transmit their rumors in T steps is a subset of $E_{\text{low}} \cup E_{\text{high}} \cup E_{\text{mid}}$. So, using the above bounds on the probabilities of these events, the total success probability is at most $\mathbb{P}[E_{\text{low}}] + \mathbb{P}[E_{\text{high}}] + \mathbb{P}[E_{\text{mid}}] = o(1) + o(1) + o(1) = o(1)$. The proof of Theorem 8 is now complete.

8.4 Proof of Theorem 9

We now show that for $c > \frac{1}{\ln^2 2}$ there is a protocol which succeeds on the n-star in time $T = cn \ln n$ with probability approaching 1. The protocol is straightforward: The set $\{0, 1, \ldots, T-1\}$ of possible transmission times is partitioned into $\frac{T \ln 2}{n} = c \ln 2 \ln n$ disjoint time intervals of length $\frac{n}{\ln 2}$ each, that we will refer to as *phases*. (To avoid clutter, we will treat the values of $c \ln 2 \ln n$ and $\frac{n}{\ln 2}$ as if they were integral. In reality, they need to be appropriately rounded. This rounding will not affect asymptotic values of the estimates below.) Each node independently and uniformly chooses a single time step in each phase at which to transmit.

Fix some arbitrary node $v \neq r$. Clearly, v will succeed in a given phase iff all other nodes choose their transmission times in this phase to be different than that of v. Thus the probability that v succeeds in this phase is at least

$$\left(1 - \tfrac{\ln 2}{n}\right)^{n-1} \ge \tfrac{1}{2}.$$

In other words, the probability that v fails in this phase is at most $\frac{1}{2}$. Let $\epsilon = c \ln^2 2 - 1 > 0$. By the independence of v's transmission times in different phases, the probability that v fails in *all* phases is at most

$$\left(\tfrac{1}{2}\right)^{c \ln 2 \ln n} = n^{-c \ln^2 2} = n^{-1-\epsilon}.$$

This holds for each node $v \neq r$, so, using the union bound, the probability that there exists a node that fails in all phases is at most $n \cdot n^{-1-\epsilon} = n^{-\epsilon} = o(1)$. We can thus conclude that with probability $1 - o(1)$ all rumors reach r, completing the proof of Theorem 9.

9 Final Comments

Our work leaves several open problems, with the most intriguing one being the time complexity of deterministic information gathering in trees in the model without rumor aggregation, when each message may contain only one rumor. In Section 5 we gave an $O(n \log n)$-time algorithm for this problem. In very recent work, Chrobak and Costello [11] improved this upper bound to $O(n \log \log n)$. The question whether this can be further improved to $O(n)$ still remains open.

It would also be interesting to refine our results by expressing the running time in terms of the tree depth D and the maximum degree Δ. Consider, for example, deterministic algorithms in the model with rumor aggregation, for which we gave an $O(n)$-time information gathering algorithm in Section 4.2. In a certain sense this bound is optimal: trivially, it cannot be improved for trees that have depth $\Omega(n)$ or contain a vertex of degree $\Omega(n)$. But it is conceivable that sub-linear time algorithms may be possible for trees with smaller depth and smaller maximum degree.

We have established a tight bound of $\Theta(n \log n)$ for randomized algorithms, in the model where the nodes in the tree have no labels. One research direction we have not yet explored is to consider randomized algorithms for information gathering in trees where the nodes *are* labelled. The combination of labels and randomization may be sufficient to overcome the issue of congestion at high-degree nodes, and it is conceivable that such algorithms could achieve running time $O(n)$ in the model where aggregation is not allowed, even if it's not possible to do it deterministically.

All our algorithms for deterministic protocols can be extended to the model where the node labels are drawn from a set $0, 1, ..., L-1$, for $L \geq n$. In this model we assume that all nodes know L. If $L = O(n)$ then the running times of our algorithms are not affected. If L is arbitrary, the algorithms can be modified to achieve running times $O(n^2 \log L)$ and $O(n^2 \log n \log L)$, respectively, although this requires that all nodes known both L and n. (See Section 4.2, near the end, for more discussion.) These bounds are far from satisfactory, and improving them, or showing some non-trivial lower bounds, would be of interest.

In regard to broader research directions, information gathering in other classes of graphs, beyond trees, is not well understood and deserves an in-depth study. As explained in the introduction, in strongly connected graphs information gathering is equivalent to gossiping, for which the computational complexity remains an open problem. In fact, it does not seem possible to adapt the sub-quadratic gossiping algorithms in [12, 25] to perform information gathering without also achieving gossiping. Roughly, the reason is that these algorithms need to occasionally broadcast some information to all nodes in the network, for example to synchronize the computation when electing a leader. It would be interesting to clarify to what degree this indirect feedback is needed for efficient information gathering. The most natural research direction here would be to consider information gathering in graphs where such feedback is just not possible — namely in directed acyclic graphs.

Acknowledgements. We would like to thank the anonymous referee, who spotted a number of potentially embarrassing mistakes in the earlier version of this paper and provided numerous thoughtful comments that helped us improve the presentation.

References

[1] A. Alon and J. Spencer. *The Probabilistic Method, 3rd edition.* Wiley, Hoboken, 2008.

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

[3] Yossi Azar and Rafi Zachut. Packet routing and information gathering in lines, rings and trees. In *13th Annual European Conference on Algorithms (ESA'05)*, pages 484–495, 2005.

[4] Reuven Bar-Yehuda, Amos Israeli, and Alon Itai. Multiple communication in multihop radio networks. *SIAM J. Comput.*, 22(4):875–887, 1993.

[5] Allan Borodin, Yuval Rabani, and Baruch Schieber. Deterministic many-to-many hot potato routing. *IEEE Trans. Parallel Distrib. Syst.*, 8(6):587–596, 1997.

[6] Danilo Bruschi and Massimiliano Del Pinto. Lower bounds for the broadcast problem in mobile radio networks. *Distrib. Comput.*, 10(3):129–135, April 1997.

[7] Bogdan S. Chlebus, Leszek Gasieniec, Alan Gibbons, Andrzej Pelc, and Wojciech Rytter. Deterministic broadcasting in ad hoc radio networks. *Distributed Computing*, 15(1):27–38, 2002.

[8] Bogdan S. Chlebus, Dariusz R. Kowalski, and Tomasz Radzik. Many-to-many communication in radio networks. *Algorithmica*, 54(1):118–139, 2009.

[9] Hongsik Choi, Ju Wang, and Esther A. Hughes. Scheduling for information gathering on sensor network. *Wirel. Netw.*, 15(1):127–140, 2009.

[10] Malin Christersson, Leszek Gasieniec, and Andrzej Lingas. Gossiping with bounded size messages in ad hoc radio networks. In *29th International Colloquium on Automata, Languages and Programming (ICALP'02)*, pages 377–389, 2002.

[11] Marek Chrobak and Kevin P. Costello. Faster information gathering in ad-hoc radio tree networks. In *12th Latin American Symposium on Theoretical Informatics (LATIN'16)*, pages 275–289, 2016.

[12] Marek Chrobak, Leszek Gasieniec, and Wojciech Rytter. Fast broadcasting and gossiping in radio networks. *J. Algorithms*, 43(2):177–189, 2002.

[13] Marek Chrobak, Leszek Gasieniec, and Wojciech Rytter. A randomized algorithm for gossiping in radio networks. *Networks*, 43(2):119–124, 2004.

[14] Ferdinando Cicalese, Fredrik Manne, and Qin Xin. Faster centralized communication in radio networks. In *17th International Symposium on Algorithms and Computation (ISAAC'06)*, pages 339–348. Springer Berlin Heidelberg, 2006.

[15] Andrea E. F. Clementi, Angelo Monti, and Riccardo Silvestri. Distributed broadcast in radio networks of unknown topology. *Theor. Comput. Sci.*, 302(1-3):337–364, 2003.

[16] Artur Czumaj and Peter Davies. Faster deterministic communication in radio networks. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP'16)*, pages 139:1–139:14, 2016.

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

[18] Gianluca De Marco. Distributed broadcast in unknown radio networks. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'08)*, pages 208–217, 2008.

[19] Gianluca De Marco. Distributed broadcast in unknown radio networks. *SIAM Journal on Computing*, 39:2162–2175, 2010.

[20] P. Erdös, P. Frankl, and Z. Füredi. Families of finite sets in which no set is covered by the union of r others. *Israel Journal of Mathematics*, 51(1–2):79–89, 1985.

[21] Paul Erdös and Pál Turán. On a problem of Sidon in additive number theory and on some related problems. *Journal of London Mathematical Society*, 16:212 – 215, 1941.

[22] Uriel Feige and Prabhakar Raghavan. Exact analysis of hot-potato routing (extended abstract). In *33rd Annual Symposium on Foundations of Computer Science (FOCS'92)*, pages 553–562, 1992.

[23] Leszek Gasieniec. On efficient gossiping in radio networks. In *16th Int. Colloquium on Structural Information and Communication Complexity (SIROCCO'09)*, pages 2–14, 2009.

[24] Leszek Gasieniec and Igor Potapov. Gossiping with unit messages in known radio networks. In *Foundations of Information Technology in the Era of Networking and Mobile Computing, IFIP 17th World Computer Congress - TC1 Stream / 2nd IFIP International Conference on Theoretical Computer Science (TCS'02)*, pages 193–205, 2002.

[25] Leszek Gasieniec, Tomasz Radzik, and Qin Xin. Faster deterministic gossiping in directed ad hoc radio networks. In *Scandinavian Workshop on Algorithm Theory (SWAT'04)*, pages 397–407, 2004.

[26] L. Gyorfi and I. Vajda. Constructions of protocol sequences for multiple access collision channel without feedback. *IEEE Transactions on Information Theory*, 39:1762–1765, 1993.

[27] Majid Khabbazian and Dariusz R. Kowalski. Time-efficient randomized multiple-message broadcast in radio networks. In *30th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '11, pages 373–380, 2011.

[28] Dariusz R. Kowalski and Andrzej Pelc. Faster deterministic broadcasting in ad hoc radio networks. *SIAM J. Discrete Math.*, 18(2):332–346, 2004.

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

[30] Ding Liu and Manoj Prabhakaran. On randomized broadcasting and gossiping in radio networks. In *8th Annual Int. Conference on Computing and Combinatorics (COCOON'02)*, pages 340–349, 2002.

[31] J. Massey and P. Mathys. The collision channel without feedback. *IEEE Trans. Inf. Theor.*, 31:192–204, 2006.

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

[33] Melih Onus, Andréa W. Richa, Kishore Kothapalli, and Christian Scheideler. Efficient broadcasting and gathering in wireless ad-hoc networks. In *8th International Symposium on Parallel Architectures, Algorithms, and Networks (ISPAN'05)*, pages 346–351, 2005.

[34] S. Resnick. *A Probability Path.* Birkhäuser, Basel, 2014.

[35] Yoram Revah, Michael Segal, and Liron Yedidsion. Real-time data gathering in sensor networks. *Discrete Applied Mathematics*, 158(5):543 – 550, 2010.

[36] A.N. Strahler. Hypsometric (area-altitude) analysis of erosional topology. *Bull. Geol. Soc. Amer.*, 63:117–1142, 1952.

[37] M. Talagrand. Concentration of measure and isoperimetric inequalities in product spaces. *Publications Mathematiques de l'I.H.E.S.*, 81:73–205, 1995.

[38] X.G. Viennot. A Strahler bijection between Dyck paths and planar trees. *Discrete Mathematics*, 246:317–329, 2003.

[39] Abraham Wald. On cumulative sums of random variables. *The Annals of Mathematical Statistics*, 15(3):283–296, 1944.

[40] Wikipedia. Sidon sequence. http://en.wikipedia.org/wiki/Sidon_sequence.

[41] Wikipedia. Golomb ruler. http://en.wikipedia.org/wiki/Golomb_ruler, 2011.

[42] Wikipedia. Sparse ruler. http://en.wikipedia.org/wiki/Sparse_ruler, 2011.

[43] Ying Xu. An $O(n^{1.5})$ deterministic gossiping algorithm for radio networks. *Algorithmica*, 36(1):93–96, 2003.

[44] Dongxiao Yu, Qiang-Sheng Hua, Weiguo Dai, Yuexuan Wang, and Francis C. M. Lau. *Dynamic Contention Resolution in Multiple-Access Channels*, pages 232–243. Springer-Verlag, 2012.