

DISTRIBUTED CONNECTIVITY OF WIRELESS NETWORKS

MAGNÚS M. HALLDÓRSSON AND PRADIPTA MITRA

ABSTRACT. We consider the problem of constructing a communication infrastructure from scratch, for a collection of identical wireless nodes. Combinatorially, this means a) finding a set of links that form a strongly connected spanning graph on a set of n points in the plane, and b) scheduling it efficiently in the SINR model of interference. The nodes must converge on a solution in a distributed manner, having no means of communication beyond the sole wireless channel.

We give distributed connectivity algorithms that run in time $O(\text{poly}(\log \Delta, \log n))$, where Δ is the ratio between the longest and shortest distances among nodes. Given that algorithm without prior knowledge of the instance are essentially limited to using uniform power, this is close to best possible. Our primary aim, however, is to find efficient structures, measured in the number of slots used in the final schedule of the links. Our main result is algorithms that match the efficiency of centralized solutions. Specifically, the networks can be scheduled in $O(\log n)$ slots using (arbitrary) power control, and in $O(\log n(\log \log \Delta + \log n))$ slots using a simple oblivious power scheme. Additionally, the networks have the desirable properties that the latency of a converge-cast and of any node-to-node communication is optimal $O(\log n)$ time.

1. INTRODUCTION

We consider the problem of constructing a communication infrastructure from scratch, for a collection of identical wireless nodes. Combinatorially, this means finding a set of links that form a strongly connected spanning graph on a set of points in the plane, and scheduling it efficiently in the SINR model of interference. The nodes must converge on a solution in a distributed manner, having no means of communication beyond the sole wireless channel. The issue is how quickly and how well: the time it takes to form the structure and the efficiency of the final schedule produced.

The importance of creating a connected structure spanning a set of wireless nodes can hardly be overstated. This may underlie a “multi-hop” wireless network, where any two nodes can communicate through path(s) specified by such a structure. In an ad-hoc network, such a structure may provide the underlying backbone for synchronized operation of the network. In a wireless sensor network, the structure can double as an information aggregation mechanism.

The efficiency of a structure is closely intertwined with the issue of interference, the distinguishing feature of wireless communication. Interference implies that only a limited number of transmissions can be successful simultaneously; this number depending on spatial distribution of the links, power settings, etc. We adopt the SINR (or physical) model of interference, that has been shown both theoretically and experimentally to be a more faithful representation of reality than many of the traditional graph-based models [18, 22].

Achieving an efficient schedule involves deciding power levels for the links – which may either be fully instance-dependent (“arbitrary”), or be chosen in an “oblivious” manner, depending only on the length of each link. Recent centralized results show that it is possible to connect any link set using $O(\log n)$ slots [11], whereas the use of oblivious power is bound to involve a factor of $\log \log \Delta$ [8, 4, 11], where Δ is the ratio between shortest and longest distance in the network.

ICE-TCS, SCHOOL OF COMPUTER SCIENCE, REYKJAVIK UNIVERSITY, 101 REYKJAVIK, ICELAND,
E-mail address: mmh@ru.is, ppmitra@gmail.com.

Achieving connectivity is a distributed problem *par excellence*. Distributed algorithms often assume “free” local communication. In contrast, since the purpose in this paper is to build a communication infrastructure from scratch, we assume that the only mode of communication allowed is transmission in the single wireless channel, which succeeds if the required signal-to-interference-and-noise ratio is achieved. We also do not assume a *carrier sensing* primitive (see, e.g., [26]) that allows nodes to estimate the amount of activity on the channel.

Given that the nodes have no information about distances to nearby nodes, they are in effect limited to using a pre-defined fixed power initially. It is known that usage of such a simple power scheme can necessarily require a linear number of slots to connect the nodes [21]. A more refined bound is $\log \Delta$, where Δ is the ratio between maximum to the minimum distance among the nodes. We provide a distributed algorithm that forms a (initial) connected network in time $O(\log \Delta \cdot \log n)$, which is probably close to the best possible.

The quality or efficiency of the *final* structure is another story. Once the initial (and possibly inefficient) network is formed, we are interested in retooling the network, still in a distributed fashion, but using the existing network, to find improved connectivity structures. We provide two approaches to this. First, we show that the initial network has nice geometric properties that allows us to use (distributed) power control to make it much more efficient. Second, we propose a more sophisticated approach — instead of simply changing the power settings of the links of initial network, we leverage the initial tree to construct *new* set of links (and their power settings) that can be scheduled even more efficiently, while still achieving connectivity. This suggest a novel interplay between different layers — a network layer (i.e., the initial tree) that goes back and retools both itself (choosing new links) and the MAC layer (changing power settings and schedules).

The challenge raised in this paper can then be stated as follows:

Is there a distributed algorithm, running in time $O(\text{poly}(\log \Delta, \log n))$, that results in a nearly optimal strongly connected structure in the SINR model?

We answer this question affirmatively, giving algorithms that match the best upper bounds known for centralized algorithms. This holds both for oblivious power assignments as well as when allowing arbitrary power assignments. In particular, using arbitrary power, we find and schedule a bidirectional tree in $O(\log n)$ slots that has the property that both aggregation computation and any pairwise communication can be achieved in optimal logarithmic time.

The rest of the paper is organized as follows. We introduce the model and key definitions in Sec. 3, and discuss related results in Sec. 2. Our results are described in Sec. 4. Section 5 contains technical definitions and clarifications that are essential for the analysis but not needed to understand the results. The algorithm for the initial network construction is given and analyzed in Sec. 6. Our two approaches to finding extremely efficient schedules are presented in Sec. 7 and Sec. 8, respectively. Several proofs and construction details have been deferred to appendices.

2. RELATED WORK

Connectivity was the first problem studied from a worst-case perspective in the SINR model. In a seminal paper, Moscibroda and Wattenhofer [21] formalized the problem and proposed an algorithm that connects arbitrary set of n points in $O(\log^4 n)$ slots. This was improved to $O(\log^3 n)$ [23], $O(\log^2 n)$ [20], and recently to $O(\log n)$ [11]. All these works deploy centralized algorithms. No non-trivial lower bound is known. Somewhat orthogonally, a large body of work exists on randomly deployed wireless networks, starting with the influential work by Gupta and Kumar [7]. Work in this setting for connectivity includes [1], which studied the probability of there existing a path between two nodes in a randomly deployed network. In [25], minimum energy connectivity structures is studied for randomly deployed networks, but interference is essentially ignored.

Distributed connectivity of wireless networks has also been the subject of research. In [28], connectivity in mobile networks was studied from a graph-theoretic perspective with no explicit interference model. Indeed, connectivity maintenance problem has been well studied in control theory and robotics [28, 19, 3], but with different underlying assumptions, typically without the use of the SINR interference model. Sensor connectivity has also been studied [13] without reference to any particular interference model. In [24], a heuristic was proposed for connectivity maintenance in multi-hop wireless networks. A more rigorous study was done in [27] but with the assumption of an underlying MAC layer that resolves interference problems.

Two fundamental problems that deal with a given set of links relate to this work. *Capacity*: find the largest feasible subset of links, and *Scheduling*: partition the link set into the fewest number of feasible sets. For the former, constant-factor algorithms were given for uniform power [5, 12], mean and linear power (and most other oblivious power assignments) [10], and power control [14]. These imply a logarithmic factor for the corresponding scheduling problems. Distributed algorithm was given for *Scheduling* with oblivious power [15] and shown to achieve $O(\log n)$ -approximation [9].

Distributed algorithms have also been given for local broadcasting [6] and dominating set [26] in the SINR model. Both of these problem are, however, local in nature.

The Minimum-Latency Aggregation Scheduling problem is closely related to connectivity, where the latency for transmitting messages to a sink is to be minimized. A large literature is known, but the first worst-case analysis in the SINR model was given in [16], with a $O(\log^3 n)$ bound on the schedule length by a centralized algorithm and $O(\log \Delta)$ by a distributed algorithm. The centralized bound was improved to optimal $O(\log n)$ in [11].

3. MODEL AND PRELIMINARIES

Given is a set P of n wireless nodes located at points on the plane. Without loss of generality, assume that the minimum distance between any two points is 1. The nodes have synchronized clocks, and start running the distributed algorithm simultaneously using slotted time. Each node knows its location and has a globally unique ID. A single message is large enough to contain the ID and the location of a node. A receiver of a message thus always knows its distance from the sender and can identify the sender uniquely.

A *link* is a directed edge between two nodes, indicating a transmission from the first node (the sender) to the second (the receiver). A link between u and v is denoted by (u, v) ; ℓ will also be used to indicate a generic link. A link set L naturally induces a set of senders $S(L)$ and a set of receivers $R(L)$. The link (y, x) is known as the *dual* of link (x, y) , following [15]. A link set X is a dual of set Y if X consists of the duals of the links in Y . The degree of a node u in a linkset L is the number of links incident on u in L . The *distance* between two nodes u and v is denoted by $d(u, v)$ (this is also the *length* of the link (u, v)). Let Δ denote the maximum length of a possible link. A *length class* refers to a set of links whose lengths differ by a factor of at most 2.

In the SINR model of interference, a non-transmitting node v successfully receives a message transmitted by node u if,

$$(1) \quad \frac{P_u/d(u, v)^\alpha}{N + \sum_{w \in S \setminus \{u\}} P_w/d(w, v)^\alpha} \geq \beta ,$$

where N is the ambient *noise*, β is the required SINR level, $\alpha > 2$ is the so-called path loss constant, P_w is the *power* used by node w , and S is the set of senders transmitting simultaneously. A set L of links is *feasible* if the above constraint holds for all $v \in R(L)$ where $S = S(L)$. We do not impose any limit on the power a node can use.

The goal is to identify a set \mathcal{T} of links that both *strongly connects* the wireless nodes and can be scheduled efficiently (i.e., can be partitioned into few feasible sets). Additionally, we seek low latency constructions.

A *converge-cast* tree is a directed rooted spanning tree where all links are oriented towards the root (i.e., for each link, the receiver is a parent of the sender). An *aggregation tree* is a converge-cast tree along with a schedule of the links that has the property that each link (x, y) in the tree is scheduled after all links involving descendants of x . A *dissemination tree* is the opposite: a broadcast tree (spanning arborescence) with links oriented away from the root, with the opposite property for the schedule. In both cases, the scheduling order follows link directions and paths in the trees.

Definition 1. A *bi-tree* is an aggregation tree with a complementary dissemination tree, using the same links in the opposite direction and same schedule in opposite order.

Note that with a bi-tree, any node-node communication can be achieved within time equal to the length of the schedule. The same holds for computing an aggregation or a broadcast.

The following power assignments are of interest. An oblivious power assignment is one where power assigned to a sender u is a (simple) function of $d(u, v)$, where v is the intended receiver. The oblivious assignment we are most interested in is the “mean power” assignment \mathcal{M} where $P_u^{\mathcal{M}} = d(u, v)^{\alpha/2}$. We also use uniform power \mathcal{U} that assigns the same power to all transmitting nodes, and the “linear power” assignment \mathcal{L} where $P_u^{\mathcal{L}} = d(u, v)^\alpha$. Note that a sender can transmit to different receivers at different times, and may use different powers. Finally, we also consider solutions achievable with arbitrary power assignments, where the algorithm is free to use any assignment. We let $\Upsilon = O(\log \log \Delta + \log n)$ denote the best ratio known for the cost of using oblivious power; namely, it is known that for any set of links, the ratio between the maximum size of feasible subset using arbitrary power vs. using mean power is at most Υ [8, 10].

4. OUR RESULTS

We give the first distributed algorithms with performance guarantees for connectivity problems in the SINR model. We first provide a basic algorithm:

Theorem 2. *There exists a distributed algorithm that computes a bi-tree \mathcal{T} in $O(\log \Delta \cdot \log n)$ slots.*

We can improve this solution by using scheduling with non-uniform (but oblivious) power assignments. Recall $\Upsilon = O(\log \log \Delta + \log n)$.

Theorem 3. *The bi-tree \mathcal{T} can be re-scheduled in $O(\Upsilon \cdot \log^3 n)$ slots using mean power.*

We then intersperse the connectivity-building and the scheduling to get solutions matching the best centralized solutions known.

Theorem 4. *There exists a distributed algorithm (building on the first one) that finds and schedules a bi-tree in $O(\log n)$ slots (with arbitrary power), using time $O(\Upsilon \cdot \log \Delta \cdot \log n)$. A variation finds and schedules a bi-tree in $O(\Upsilon \cdot \log n)$ slots with mean power, using time $O(\Upsilon \log \Delta \cdot \log^2 n)$.*

In particular, the bi-tree property ensures that aggregation, broadcast, and pairwise communication can all be achieved in optimal $O(\log n)$ steps.

Technically, this work combines ingredients from numerous recent works on the SINR model [8, 10, 14, 15, 9, 11]. In addition, we derive a number of properties, most of which deal with the concept of *affectance* in relation to connectivity structures; intuitively, affectance measures the interference of one transmission on the reception of another transmission, relative to the signal strength of the latter. We explicitly define a previously considered geometric property of *sparsity*, and show it to imply small average affectance. We give novel algorithms for finding large feasible subsets in such sparse link sets. And, we introduce randomized transmission strategies to estimate affectance in terms of transmission successes.

5. TECHNICAL NOTES

Our algorithms require the following knowledge about the instance: The number of nodes, n , up to a polynomial factor; the minimum distance (assumed to be 1); and the maximum distance Δ . We do not treat Δ as a constant, although it is small in many systems. Knowledge of Δ is mainly needed for stopping criteria; it can be avoided by computing the size of the tree, if precise knowledge of n is available.

In describing our algorithm, we refer to some messages as *broadcasts* and some as *acknowledgments*. In terms of if and how these messages succeed, they are identical and work as dictated by Eqn. 1. The difference lies in when these messages are transmitted and what they contain. A broadcast refers to an exploratory message sent to no node in particular, only containing the sender's ID and location. An acknowledgment is transmitted as a response to a previous message (typically a broadcast) and contains IDs of both the sender (the acknowledger) and the initial broadcaster. Thus, receivers receiving an acknowledgment can determine if it was addressed to them or not.

All our results are proved to be true “with high probability” (w.h.p., for short), where the term means that the relevant event occurs with probability $1 - \frac{1}{n^c}$, for some suitably large c^1 . We frequently prove a lemma to hold, w.h.p., for a node u , or a link (u, v) . It will always be clear that such a result can be safely union bounded over all nodes, or all possible links, to derive a high probability result for the whole algorithm. The only case that needs care is when we union bound over slots in the algorithm. The number of slots in our first algorithm is a function of $\log \Delta$, which can be arbitrarily larger than n . Union bounding is still safe for the following reason. The algorithm proceeds by considering links belonging to the same length class, and there can be at most $\log \Delta$ of such classes (thus the dependence on $\log \Delta$). However, since there are at most n^2 links in the network, only n^2 classes can actually be non-empty (in the full version, we provide a more refined $O(n)$ upper bound). During empty length classes, nothing happens with probability 1 and thus the union bounding incurs no “cost”.

Affectance. We use the notion of *affectance*, introduced in [5, 12] and refined in [15] to the threshold-ed form used here. The affectance $a_w^{\mathcal{P}}(\ell)$ on link $\ell = (u, v)$ from a sender w , with a given power assignment \mathcal{P} , is the interference of w on u relative to the power received, or

$$a_w^{\mathcal{P}}(\ell) = \min \left\{ 1 + \epsilon, c(u, v) \frac{P_w}{P_u} \cdot \left(\frac{d(u, v)}{d(w, v)} \right)^\alpha \right\},$$

where ϵ is some arbitrary fixed constant (say 0.1), $c(u, v) = \beta / (1 - \beta N d(u, v)^\alpha / P_u)$ depends only on the parameters of the link ℓ . We drop \mathcal{P} when clear from context. For a set S of senders and a link ℓ , $a_S(\ell) = \sum_{w \in S} a_w(\ell)$.

Using such notation, Eqn. 1 can be rewritten as $a_S(\ell) \leq 1$, which we adopt. When dealing with links $\ell = (u, v)$ and $\ell' = (u', v')$ we mean $a_\ell(\ell')$ to mean $a_u(\ell')$. Extending this to a link set L , we use the notation $a_L(\ell)$ to mean $a_S(\ell)$ where $S = S(L)$ are the senders in L . For two sets X and Y , $a_X(Y)$ thus means $\sum_{\ell \in Y} a_{S(X)}(\ell)$. From its definition, it is clear that $c(u, v) \geq \beta$. We require that $c(u, v) \leq 2\beta$, and point out how to achieve this during the description of the algorithms. This simply means that nodes always transmit with power high enough for the intended (or potentially intended, in case of a broadcast) links to comfortably succeed in the presence of noise (but no other interference).

6. INITIAL TREE CONSTRUCTION

The general template for the algorithm is as follows. At any given time, a subset of the nodes is *active*, with initially all nodes active and in the end only one node. Links are formed between pairs of active nodes, by a node u broadcasting, and another node v acknowledging that message

¹This can be amplified to hold for *any* c , by scaling up the constant factors.

in the next round. When such a communication succeeds, links (u, v) and (v, u) become part of the network and node u becomes inactive (and forms no further links). The still active node v is u 's parent in the eventual aggregation tree. The link (u, v) is then part of the aggregation tree and the link (v, u) is part of the dissemination tree.

In what follows, $\lambda_1, \lambda_2 \dots, \gamma_1, \gamma_2 \dots$ are constants.

The algorithm proceeds in $\lceil \log \Delta \rceil$ rounds, each containing $\lambda_1 \log n$ slot-pairs (a slot-pair is simply two consecutive slots). Each node u maintains a link set L_u storing incoming and outgoing links along with a time stamp. The final set \mathcal{T} is then simply $\cup_u L_u$. In this initial tree construction, slots in the schedule of the links correspond simply to the time stamps.

At the beginning of each slot-pair in round r , each active node decides to be a *broadcaster* with iid probability p ($p \leq \frac{1}{2}$ to be determined), and *listener* otherwise. Then,

- During the first slot, a broadcaster u transmits a message and a listener v listens for messages.
- During the second slot, a listener v that received a message from u such that $2^{r-1} \leq d(u, v) < 2^r$ in the previous slot does the following with iid probability p : add the links (u, v) and (v, u) to L_v with appropriate slot numbers and return an acknowledgment. A broadcaster u listens for acknowledgments during this slot, and on receiving one (say, from v) adds (u, v) and (v, u) to L_u , and becomes *inactive*.

Note that a node only forms links with nodes at distance in the range $[2^{r-1}, 2^r)$ during round r . Since each node knows this range it can easily choose a power that ensures $c(u, v) \leq 2\beta$ for all $d(u, v) \in [2^{r-1}, 2^r)$. Setting the power to $2\beta N 2^{r\alpha}$ suffices. We say that a link (u, v) is *successfully formed* between nodes u and v during a slot-pair if all of the following happen: a) the transmission (u, v) is successful in the first slot, b) it is successfully acknowledged in the second slot (i.e., the link (v, u) successfully transmits), and c) both nodes store (u, v) and (v, u) in their set of links with the appropriate time stamps. Note that when this happens, u becomes inactive, by the description of the algorithm. The sole link that is outgoing from a given node is also the last one to be scheduled, thus ordering satisfies the leaf-to-root order of aggregation trees.

Remarks. Two technical clarifications. First, note that a listener v can store a failed link, since it does not necessarily know whether an acknowledgment (v, u) succeeded. However, this is not a problem, since: a) Node u remains active if the acknowledgment fails and connects itself later to some node (or eventually becomes the root), b) Transmission of the link (v, u) is not problematic for other links, since links transmitting in that slot *did* succeed in the presence of that transmission. In any case, it is easy to efficiently “clean up” such stray links after the whole network is formed. Second, as constructed, the dissemination tree has the opposite order of links in the schedule (links closer to the root are scheduled later, instead of earlier, as the definition calls for). This is also easily fixable after the network is formed by a reversal process initiated by the root. We omit these details in this version.

6.1. Analysis. We first show that short links have a high probability of succeeding.

Lemma 5. *Assume that at the beginning of round r , the minimum distance between active nodes is at least 2^{r-1} . Consider any slot-pair in the round and active nodes u and v with $d(u, v) < 2^r$. Then, with probability at least $\frac{1}{4}p^2(1-p)$, the link (u, v) is successfully formed in that slot-pair. Similarly, with probability at least $\frac{1}{4}p^2(1-p)$, the link (v, u) is successfully formed.*

Proof. Let $\rho = 2^{r-1}$. Let M_r be the set of currently active nodes and let $\ell = (u, v)$. Let B_r be the set of broadcasters during the slot-pair. First, note that by the description of the algorithm

$$\mathbb{P}(u \in B_r \text{ and } v \notin B_r) = p(1-p) .$$

For $t = 0, 1, \dots$ define C_t to be the ball around v of radius $\rho(t+1)$ and define the annulus A_t as $A_0 = C_0$, $A_t = C_t \setminus C_{t-1}$ for $t \geq 1$. From this it is easily computed that the area of A_t is

$$(2) \quad \text{Area}(A_t) = \pi\rho^2(2t+1)$$

Now, by the definition of ρ , balls of radius $\frac{\rho}{4}$ around any pair of points in M_r do not intersect (since the minimum distance between active nodes is ρ). Combining this with Eqn. 2, we see that A_t contains at most $16(2t+1) \leq 48t$ nodes in M_r .

For $x \in M_r \cap A_0$, $a_x(\ell) \leq 1 + \epsilon$, simply by the definition of affectance. For $x \in M_r \cap A_t$ for $t \geq 1$, $d(x, v) \geq \rho \cdot t$ and thus $a_x(\ell) \leq c_\ell \frac{(2\rho)^\alpha}{(\rho \cdot t)^\alpha} \leq 2\beta \left(\frac{2}{t}\right)^\alpha$, where $c_\ell \equiv c(u, v) \leq 2\beta$. Note that for any x , $\mathbb{P}(x \in B_r) = p$.

Thus,

$$\begin{aligned} \mathbb{E}(a_{B_r}(\ell)) &= \mathbb{E}(a_{B_r \cap A_0}(\ell)) + \sum_{t \geq 1} \mathbb{E}(a_{B_r \cap A_t}(\ell)) \\ &\leq 16(1 + \epsilon)p + p2\beta \sum_{t \geq 1} \left(\frac{2}{t}\right)^\alpha 48t \\ &\leq 16(1 + \epsilon)p + 96p\beta 2^\alpha \frac{1}{\alpha - 2}, \end{aligned}$$

using the bound $\zeta(x) = \sum_{n \geq 1} \frac{1}{n^x} \leq \frac{1}{x-1}$ on the Riemann zeta function. Thus, for any $p \leq (64(1 + 6\beta 2^\alpha \frac{1}{\alpha-2}))^{-1}$, we get that $\mathbb{E}(a_{B_r}(\ell)) \leq 1/2$. By Markov's inequality, $a_{B_r}(\ell) \leq 1$ with probability at least $\frac{1}{2}$ (recall that this means that the link ℓ succeeds). Thus,

$$\mathbb{P}(a_{B_r}(\ell) \leq 1 \text{ and } u \in B_r \text{ and } v \notin B_r) \geq \frac{1}{2}p(1-p),$$

A similar argument proves that the probability of $\ell_r = (v, u)$ succeeding is at least $\frac{1}{2}p$ and thus the link (u, v) is formed with probability at least $\frac{1}{4}p^2(1-p)$. The argument for the potential formation of link (v, u) is identical. \square

Now we can claim that,

Lemma 6. *At the beginning of each round r , the distance between active nodes is at least 2^{r-1} , w.h.p.*

Proof. (Sketch.) The claim is clearly true for round 1 (since the minimum distance in the system is 1). Now inductively assume that it is true for round r . Consider any two nodes u, v that are active at the beginning of round $r+1$ with $d(u, v) < 2^{r+1}$. Consider any slot-pair in which they are both active. By Lemma 5, the probability of both of them remaining active after this slot pair is at most $1 - \frac{1}{4}p^2(1-p) \leq 1 - \frac{1}{8}p^2$. Thus, the probability of both of them remaining active over $\lambda_1 \log n$ slot-pairs is $\leq (1 - \frac{1}{8}p^2)^{\lambda_1 \log n}$. Setting $\lambda_1 = \frac{80}{p^2}$, this probability can be upper bounded by $\frac{1}{n^{10}}$. This proves the Lemma (after union bounding). \square

We can now prove the first main result.

Proof. [of Thm. 2] By Lemma 6 it is clear that within $O(\log \Delta)$ rounds, and thus $O(\log \Delta \cdot \log n)$ slots, at most one active node remains (since the maximum distance among nodes is Δ). Since nodes only cease to be active by forming links with an active node, it is also clear that exactly one node remains active. When nodes cease to be active, they do so only by connecting in both directions to still-active nodes (by the description of the algorithm). By induction, the whole network is then strongly connected to the single node active at the end. This last active node is the root of both the aggregation and dissemination trees. \square

We can also show that the network formed has low degree, where the degree of a node is its number $|L_u|$ of incident links.

Theorem 7. *The probability of a link having degree d is at most $e^{-\frac{p^2 d}{8}}$. As a result, the maximum degree is $O(\log n)$, w.h.p.*

Proof. Let u be a node and consider any round r and any slot-pair in the round where u is active. Suppose there is another active node v with $d(u, v) < 2^r$. Then by Lemma 5, u ceases to be active after this slot-pair, with probability at least $\frac{1}{4}p^2(1-p) \geq \frac{1}{8}p^2$. Note that in slot-pairs where no such v exists, u does not form a link. Thus, the degree of a node is upper bounded by the number of slot pairs where such a v exists, and u remains active after wards. The probability of there being d such slot pairs is at most $(1 - \frac{1}{8}p^2)^d \leq e^{-\frac{p^2 d}{8}}$.

Setting $d = \frac{1}{p^2} 80 \log n$ gives us the second part of the lemma. \square

7. SPARSITY AND POWER CONTROL

In this section, we show that the link set \mathcal{T} produced by the algorithm of Sec. 6 can actually be scheduled in considerably fewer slots (in terms of dependence on Δ) using mean power, thus proving Thm. 3. This leads to an algorithm to reschedule the same links with this improved power assignment. The main idea is to show that the produced link set has certain geometric properties that allows such improved scheduling.

Definition 8. *A set L of links is ψ -sparse if, for every closed ball B in the plane,*

$$B \cap L(8 \cdot \text{rad}(B)) \leq \psi,$$

where $\text{rad}(B)$ is the radius of B , $L(d)$ is the set of links in L of length at least d , and $B \cap Q$ denotes the links in a set Q with at least one endpoint in ball B .

It was shown in [11] that the sparsity property (not explicitly defined there) is connected to a property named *amenability* in [11], which via an algorithm in [14] and results in [10] imply the following:

Theorem 9 ([11]). *Let L be a ψ -sparse link set, for some $\psi \geq 1$. Then any $L' \subseteq L$ contains a feasible subset of size $\Omega\left(\frac{|L'|}{\psi}\right)$. The set L can be scheduled in $O(\psi \log n)$ slots. Furthermore, any $L' \subseteq L$ contains a subset of size $\Omega\left(\frac{|L'|}{\psi \cdot \Upsilon}\right)$ that is feasible under mean power assignment. The set L can be scheduled in $O(\psi \cdot \Upsilon \cdot \log n)$ slots using mean power.*

We provide a short overview of these ideas for reference in Appendix B. We now claim a sparsity result for the network \mathcal{T} formed by the algorithm.

Lemma 10. *If D is a disc of radius ρ in the plane, then the number of links in \mathcal{T} longer than 8ρ that have at least one endpoint in D is $O(\log n)$, w.h.p.*

Proof. Let $L = L(8 \cdot \rho) \cap D$. We first claim that at most one node inside D is incident to a link in L . For contradiction, assume that there are two such nodes u and v . Now, by the description of the algorithm, links of length 8ρ or higher can only be formed in rounds $\log \rho + 4$ or higher. Thus, both u and v were active during round $\log \rho + 4$. However, $d(u, v) \leq 2\rho$ and thus by Lemma 6, at the end of round $\log \rho + 2$, at most one of them could remain active. This is a contradiction. The proof of the Lemma is now complete by Thm. 7. \square

By union bounding over all ρ and all balls (by careful selection, there are only polynomially many of them that are relevant), this implies:

Theorem 11. *The set \mathcal{T} of links produced by the algorithm is $O(\log n)$ -sparse.*

We now propose the following extension of the algorithm to schedule the links using significantly fewer slots.

The sender of each link ℓ in \mathcal{T} sets its power to mean power, $\ell^{\alpha/2}$. The links then use the distributed algorithm from [15] to compute a schedule of the links using this power assignment.

We can now prove Thm. 3.

Proof. Thm. 9 and Thm. 11 imply that \mathcal{T} can be scheduled in $O(\Upsilon \cdot \log^2 n)$ slots using mean power. The distributed scheduling algorithm of [15] produces a $O(\log n)$ -approximation [9], giving the Theorem. (See Appendix C for a technical note on the approximation factor in [9]). \square

The resulting schedule, however, does not necessarily satisfy the ordering property of bi-trees.

8. MATCHING CENTRALIZED BOUNDS

In this section, we prove Thm. 4. The difference with Sec. 7 are threefold. First, we achieve more efficient final schedules. Second, unlike Sec. 7, we produce bi-trees. The third is a difference in approach. While the algorithm in Sec. 7 merely rescheduled the links in the original tree, in this section, we shall actually build a *new* tree with superior properties, but will do so by using the original tree.

We use **Init** to refer to the algorithm from Sec. 6 that constructs the initial bi-tree. For any link set L which is a subset of a directed rooted tree, we call a node u a “top level node” with respect to L if no link of form (v, w) is in L (i.e., the link between v and its parent in the rooted tree, if such a link exists, is not present in L).

In what follows, we focus on forming the aggregation tree part for simplicity (constructing the dissemination tree portion of the bi-tree is essentially identical). The algorithmic framework is as follows.

Algorithm 1 TreeViaCapacity

- 1: Set $i = 0$ and $P_i = P$ (the original input set).
 - 2: **for** $i = 0, 1, 2 \dots$ **until** $|P_i| = 1$ **do**
 - 3: Construct (aggregation) tree \mathcal{T} on P_i using **Init**.
 - 4: Find a feasible subset $\mathcal{T}' \subset \mathcal{T}$
 - 5: Let P_{i+1} be the set of top level nodes w.r.t. \mathcal{T}' .
 - 6: **end for**
-

If \mathcal{T}' is large, then this process ends quickly.

Theorem 12. *Assume that in each iteration, $\mathbb{E}(|\mathcal{T}'|) = \delta|\mathcal{T}|$ for some $\delta > 0$. Then, the process ends after $O(\frac{1}{\delta} \log n)$ iterations and the links produced form an aggregation tree connecting the nodes in $O(\frac{1}{\delta} \log n)$ slots, w.h.p.*

Proof. First we show that:

Claim 8.1. $\mathbb{E}(|P_{i+1}|) \leq (1 - \frac{1}{2}\delta)|P_i|$, for any P_i such that $|P_i| \geq 2$.

Proof. Suffices to prove Recall that $|\mathcal{T}'| \geq \delta|\mathcal{T}| = \delta(|P_i| - 1)$. Consider any link $(u, v) \in \mathcal{T}'$. Clearly, this link rules out u as a top level node. Also, since \mathcal{T} is an aggregation tree, there can be at most one outgoing link from each node u . Thus, $\mathbb{E}(|P_{i+1}|) \leq |P_i| - \mathbb{E}(|\mathcal{T}'|) \leq |P_i| - \delta(|P_i| - 1) \leq (1 - \frac{1}{2}\delta)|P_i|$ (for $|P_i| \geq 2$). \square

This can be used to show that the process ends in $O(\frac{1}{\delta} \log n)$ steps, w.h.p.

Claim 8.2. $\mathbb{P}(|P_t| > 1) \leq \frac{1}{n^4}$ for $t = 10\frac{1}{\delta} \log n$.

Proof. Since P_i is non-increasing in i , for contradiction, condition on all $P_i \geq 2$ for $i \leq t$. Then we can apply the above Lemma to show that

$$\mathbb{E}(|P_t|) \leq \left(1 - \frac{1}{2}\delta\right)^{10\frac{1}{\delta} \log n} \frac{1}{n} \leq \frac{1}{n^4},$$

from which the claim follows by Markov's inequality. \square

By the definition of top level nodes, nodes not in P_{i+1} are connected to some node in P_i by a link. Thus, the final structure is clearly a converge-cast tree. The ordering on schedules is also guaranteed by the way the algorithm proceeds (it is easy to see that nodes can be involved in at most one link in a feasible set, thus the ordering is not violated within \mathcal{T}').

Finally, since each iteration uses a single slot, the bound on iterations immediately implies the bound on the number of slots in the schedule. The theorem follows. \square

To implement the above scheme, we need to show that \mathcal{T}' can always be found for a large enough δ to claim the results in Thm. 4.

We do this in two steps: in the first step a $O(1)$ -sparse subset $\mathcal{T}(M) \subseteq \mathcal{T}$ is chosen, and in the second step a subset of $\mathcal{T}(M)$ is chosen as \mathcal{T}' . The first step is identical for mean power and arbitrary power case. The set $\mathcal{T}(M)$ is defined in the following result, whose proof is in Appendix A.

Theorem 13. *Let M be the set of nodes of degree at most $\rho = \frac{160}{p^2}$ in \mathcal{T} , and let $\mathcal{T}(M)$ be the links in \mathcal{T} induced by M . Then, $\mathcal{T}(M)$ is $O(1)$ -sparse and $\mathbb{E}(|\mathcal{T}(M)|) = \Omega(|\mathcal{T}|)$.*

To actually compute $\mathcal{T}(M)$ in a distributed fashion, note that nodes can easily decide if they are in M (by counting the number of links adjacent to them). One sweep through the existing network \mathcal{T} is enough for each node to detect which of their links (if any) are in $\mathcal{T}(M)$.

Selecting \mathcal{T}' is also reasonably easy for mean power, but more involved for arbitrary powers. The following two subsections deal with these cases separately. Note that we keep the original network around at all times, which is useful for controlling the construction of the new one. Running these networks in parallel can be achieved with simple time-division multiplexing.

8.1. Finding \mathcal{T}' with mean power. Assume that $\mathcal{T}(M)$ is known. It can be shown that the *average affectance* in the linkset $\mathcal{T}(M)$ (under mean power) is small, or $O(\Upsilon)$ (proof in Appendix A).

Lemma 14. *Affectance within $\mathcal{T}(M)$ under mean power satisfies $a_{\mathcal{T}(M)}^{\mathcal{M}}(\mathcal{T}(M)) = \gamma_1 \Upsilon |\mathcal{T}(M)|$, for some constant γ_1 .*

Lemma 14 implies, after some basic manipulation, that there exists Q with $|Q| \geq \frac{1}{2}|\mathcal{T}(M)|$, such that $a_{\mathcal{T}(M)}^{\mathcal{M}}(\ell) \leq 2\gamma_1 \Upsilon$ for all $\ell \in Q$.

The following *sampling* mechanism produces a large feasible set in expectation (see [4]): Each link in $\mathcal{T}(M)$ transmits with iid probability $\frac{1}{4\gamma_1 \Upsilon}$, with the successful links forming the set \mathcal{T}' . Since each transmitting link in Q succeeds with probability $\geq \frac{1}{2}$, the expected size of \mathcal{T}' is at least $\frac{1}{2\gamma_1 \Upsilon} |Q| = \Omega(\frac{1}{\Upsilon} |\mathcal{T}(M)|)$. Combining this with Thm. 13, we get that

Lemma 15. $\mathbb{E}(|\mathcal{T}'|) = \Omega(\frac{1}{\Upsilon} \mathbb{E}(|\mathcal{T}(M)|)) = \Omega(\frac{1}{\Upsilon} |\mathcal{T}|)$.

Thus, Thm. 12 can be invoked with $\delta = \Omega(\frac{1}{\Upsilon})$, to obtain the second half of Thm. 4:

Theorem 16. *There exists a distributed algorithm that forms and schedules a bi-tree in $O(\Upsilon \cdot \log n)$ slots using mean power. This algorithm completes in time $O(\Upsilon \log \Delta \cdot \log^2 n)$.*

Proof. The performance of the final solution follows from Thm. 12, as mentioned above. Let us bound the total running time. The algorithm **Init** needs to be invoked $O(\Upsilon \cdot \log n)$ times, for a total cost of $O(\Upsilon \cdot \log \Delta \cdot \log^2 n)$. After forming \mathcal{T} with each such invocation, identifying $\mathcal{T}(M)$ costs $O(\log \Delta \log n)$ (the cost of \mathcal{T}). Computing \mathcal{T}' is cheap since the sampling is done in parallel. One technical aspect to note is that while the nodes choose \mathcal{T}' , they need to know if their transmission succeeded; this can be done without substantial loss of performance using an extra acknowledgment slot, as we have seen before. The runtime bound of the theorem then follows. \square

This theorem completes the proof of the second half of Thm. 4.

8.2. Finding \mathcal{T}' with arbitrary power. In this case, we want to find a large set \mathcal{T}' , given $\mathcal{T}(M)$, and then choose a power assignment making the set feasible.

We start with the link selection step. Leveraging the fact that our input instance $\mathcal{T}(M)$ is sparse, we implement a distributed version of a centralized algorithm for choosing such a set proposed in [14].

The following algorithm was shown in [14] to give constant factor approximation for finding the largest feasible subset of any given linkset: Given a linkset R , let the selected set be L , initially empty. Go through all links in ascending order of length (breaking ties arbitrarily). If the condition

$$(3) \quad a_L^{\mathcal{L}}(\ell) + a_L^{\mathcal{U}}(L) \leq \tau,$$

holds, for a constant τ , then the link ℓ is added to L (Eqn. 1 of [14] can be seen to be essentially equivalent to the above equation).

For simplicity, we assume in this abstract that receivers can measure the SINR of a successful link (i.e., can measure if the link succeeded with a desired threshold τ or not). This assumption can be removed.

Assume the formation of \mathcal{T} using **Init** required R rounds. Our selection algorithm **Distr-Cap** has the following outline.

Distr-Cap contains R phases. In phase i , links in $\mathcal{T}(M)$ that were formed in round i of **Init** decide whether or not to add themselves to the selected set \mathcal{T}' .

By the description of **Init**, links formed in the same round belong to the same length class (also, links formed in a particular round are smaller than all links formed in later rounds).

For all i , phase i of **Distr-Cap** consists of one slot-pair. Let Q be the links participating in this phase (i.e., links formed during round i of **Init**). During the first slot of the phase, the following happens:

- (1) All links ℓ in \mathcal{T}' (the set selected so far) transmit using linear power (i.e. $P_\ell = \ell^\alpha$).
- (2) Links in Q transmit with iid probability p (small constant) using linear power.
- (3) Receivers in Q record a success if they received a message across the link with $\text{SINR} \leq \tau/4$.
Let \tilde{Q} be the set of links that recorded success.

During the second slot:

- (1) Links in \mathcal{T}'_d (dual of \mathcal{T}') transmit using linear power (i.e., the receivers of \mathcal{T}' transmit using linear power).
- (2) Links in \tilde{Q}_d (dual of \tilde{Q}) transmit with iid probability $\gamma_2^2 \cdot p$ for some $\gamma_2 < 1$, using linear power.
- (3) Receivers in \tilde{Q}_d record a success if they received a message across the link with $\text{SINR} \leq \frac{\gamma_2 \cdot \tau}{4}$.

Thus, at the end of a second slot, a success is recorded at a sender of a (original) link in Q , if the transmission succeeded in both directions (the original link and the dual) with the required SINR

threshold. Let Q^* be the set of links that succeeded. The updated solution is then $\mathcal{T}' \leftarrow \mathcal{T}' \cup Q^*$, which simply means that links add themselves to \mathcal{T}' if they succeeded in both directions.

We now analyze this algorithm. The following sub-subsections show that the selected solution is feasible and large (a constant factor approximation to the largest feasible subset), respectively.

8.2.1. \mathcal{T}' is feasible. We now show that \mathcal{T}' satisfies Eqn. 3. It suffices to show that for all $\ell \in \mathcal{T}'$, if $L \subseteq \mathcal{T}'$ are the links no larger than ℓ then:

$$a_L^{\mathcal{L}}(\ell) + a_\ell^{\mathcal{U}}(L) \leq \tau .$$

The following two Lemmas imply the above.

Lemma 17. $a_L^{\mathcal{L}}(\ell) \leq \frac{\tau}{4}$.

Proof. To see this, note the selection of \tilde{Q} in the first slot of each slot-pair. We claim that during this slot, all links in L are transmitting with linear power. For links in L that were selected in an earlier phase, this is obviously true. For links in Q that will be selected in L , this is true as well, since eventual admission in L is only possible (though not guaranteed) if the link decided to transmit during the first slot.

The proof of the Lemma is completed by noting the SINR threshold used in the selection of \tilde{Q} . \square

Lemma 18. $a_\ell^{\mathcal{U}}(L) \leq \frac{\tau}{4}$.

Proof. The selection process implemented during the second slot guarantees that $a_{L_d}^{\mathcal{L}}(\ell_d) \leq \frac{\gamma_2 \tau}{4}$, where L_d is the dual set of L and ℓ_d is the dual of ℓ (this follows the proof of the previous Lemma almost verbatim).

To complete the proof, we use a result from [15, Obs. 4]. It was shown that for a constant γ_2 , and links ℓ and ℓ' ,

Claim 8.3. $\gamma_2 a_{\ell'_d}^{\mathcal{L}}(\ell_d) \leq a_\ell^{\mathcal{U}}(\ell') \leq \frac{1}{\gamma_2} a_{\ell'_d}^{\mathcal{L}}(\ell_d)$.

Using this claim, we get that

$$a_\ell^{\mathcal{U}}(L) = \sum_{\ell' \in L} a_\ell^{\mathcal{U}}(\ell') \leq \sum_{\ell' \in L_d} \frac{1}{\gamma_2} a_{\ell'_d}^{\mathcal{L}}(\ell_d) = \frac{1}{\gamma_2} a_{L_d}^{\mathcal{L}}(\ell_d) \leq \frac{\tau}{4} ,$$

as required. \square

8.2.2. \mathcal{T}' is large. Define, following [11, 14],

$$f_\ell(\ell') = \begin{cases} a_{\ell'}^{\mathcal{U}}(\ell) + a_\ell^{\mathcal{L}}(\ell') & \text{if } \ell \leq \ell', \\ 0 & \text{otherwise.} \end{cases}$$

This definition is essentially equivalent to the definition of that of $f_\ell(\ell')$ of [11] and of $w(\ell, \ell')$ of [14] (also see Appendix B). Those definitions are presented in terms of distances. The reason why we choose to define $f_\ell(\ell')$ in terms of affectances here, instead of distances, is that affectances (or their SINR equivalents) can be measured by the link receivers and thus used as a selection criteria. For a set X , define $f_\ell(X) = \sum_{\ell' \in X} f_\ell(\ell')$ and $f_X(\ell') = \sum_{\ell \in X} f_\ell(\ell')$.

Recall that the input set \mathcal{T} is $O(1)$ -sparse, which is of crucial importance. Consider once again the execution of the algorithm for phase i . Let \mathcal{T}'_{i-1} be the selected set at the end of phase $i-1$. As before, let Q be the links considered in phase i and Q^* be the links that succeeded in that phase. Since \mathcal{T} is $O(1)$ -sparse, so is Q .

Lemma 19. *Let Q' be the subset of links ℓ in Q with $f_{\mathcal{T}'}(\ell) \leq \gamma_2^2 \cdot \tau/8$. Then, $\mathbb{E}(|Q^*|) = \Omega(|Q'|)$.*

Proof. Consider any link $\ell \in Q'$. We shall show below that $\mathbb{P}(\ell \in Q^*) = \Omega(1)$, which implies the Lemma.

In the first slot, ℓ transmits with probability p . We claim that:

Claim 8.4. $\mathbb{P}(a_T^{\mathcal{L}}(\ell) \leq \tau/8) \geq \frac{1}{2}$, where T are the links in Q transmitting.

Proof. Let ρ be such that length class in phase i covers lengths in $[\rho, 2\rho)$. Since Q is $O(1)$ -sparse, it follows that balls of radius ρ contain only a constant number of nodes that have links in Q . The claim now follows from arguments essentially identical to those in Lemma 5, after setting the probability p sufficiently small. \square

Since $\ell \in Q'$, we see that $a_{T'_{i-1}}^{\mathcal{L}}(\ell) \leq \tau/8$, by the definition of Q' . Thus, if $a_T^{\mathcal{L}}(\ell) \leq \tau/8$, then $a_{T \cup T'_{i-1}}^{\mathcal{L}}(\ell) \leq \tau/4$, and the transmission is recorded as a success. Thus, ℓ transmits and is recorded as a success with probability $\frac{1}{2}p$. In other words,

$$(4) \quad \mathbb{P}(\ell \in \tilde{Q}) \geq \frac{1}{2}p.$$

Now, condition on ℓ being in \tilde{Q} . Then ℓ_d transmits with probability $\gamma_2 p$. The following claim can be proven using Claim 8.3 and is similar to Claim 8.4.

Claim 8.5. $\mathbb{P}(a_{T_d}^{\mathcal{L}}(\ell_d) \leq \frac{\gamma_2 \tau}{8}) \geq \frac{1}{2}$, where $T_d \subseteq \tilde{Q}_d$ are the (dual) links transmitting in this slot.

Following a argument similar to the one used for the first slot, we see that in the second slot, such a transmission is recorded as a success as well.

Thus, $\mathbb{P}(\ell \in Q^* | \ell \in \tilde{Q}) \geq \frac{1}{2}\gamma_2 p$. Combining this with Eqn. 4, we get $\mathbb{P}(\ell \in Q^*) \geq \frac{1}{4}\gamma_2 p^2 = \Omega(1)$, completing the proof of the Lemma. \square

This leads to the desired bound on the size of \mathcal{T}' .

Theorem 20. The set \mathcal{T}' chosen by the algorithm satisfies $\mathbb{E}(|\mathcal{T}'|) = \Omega(|\mathcal{T}(M)|)$.

Proof. By Thm. 9, there exists a set $O \subseteq \mathcal{T}$ such that O is feasible and $|O| = \Omega(|\mathcal{T}|)$. Thus, it suffices to show that $\mathbb{E}(|\mathcal{T}'|) = \Omega(|O|)$ for any feasible set O .

Thm. 1 of [14] shows that for a feasible link set R and any link ℓ ,

$$(5) \quad f_\ell(R) = O(1).$$

Consider the set $R = O \setminus \mathcal{T}'$. We divide R further into two subsets: $R_1 = \{\ell' \in R : f_{\mathcal{T}'}(\ell') > \gamma_2^2 \tau / 8\}$ and $R_2 = R \setminus R_1$. Summing Eqn. 5 for all $\ell \in \mathcal{T}'$,

$$(6) \quad f_{\mathcal{T}'}(R) = O(|\mathcal{T}'|).$$

By definition of R_1 , $f_{\mathcal{T}'}(R_1) > \frac{1}{8}\gamma_2^2 \tau |R_1|$. Assume first that $|R_1| \geq |R|/2$. Then, we get, $f_{\mathcal{T}'}(R_1) > \frac{1}{16}\gamma_2^2 \tau |R|$, which combined with Eqn. 6 gives, $|\mathcal{T}'| = \Omega(f_{\mathcal{T}'}(R)) \geq f_{\mathcal{T}'}(R_1) = \Omega(|R|)$. Since $|O| \leq |\mathcal{T}'| + |R|$, this clearly implies that $|\mathcal{T}'| = \Omega(|O|)$. Otherwise assume, $|R_1| < |R|/2$ and thus $|R_2| > |R|/2$. But Lemma 19 implies that $\Omega(|R_2|)$ links were chosen by the algorithm (in expectation), from which $\mathbb{E}(|\mathcal{T}'|) = \Omega(|O|)$ follows. \square

8.2.3. Computing the power assignment. So far we have dealt with the *selection* of a large set of feasible links. Once the link set \mathcal{T}' is identified, we must select the power assignments for this set. Given a set of links that are known to be feasible, there exists a large body of work proposing algorithms that converge to a power assignment making the assignment feasible. For example, two recent ones are [17] and [2]. Using such an algorithm as a black box, we can find the appropriate power assignment.

Theorem 21. *There exists a distributed algorithm that connects the nodes in $O(\log n)$ slots. Assuming that there exists an algorithm to find the power assignment for a feasible set in time η , this algorithm completes in time $O(\log n(\log \Delta \cdot \log n + \eta))$.*

As an example, if we select the algorithm from [17], η can be bounded by $O(\log \Delta(\log n + \log \log \Delta))$. This proves the first part of Thm. 4.

9. CONCLUSIONS

Our distributed algorithms have efficiency and effectiveness that appear to be close to best possible. An interesting direction would be to treat dynamic situations, including asynchronous node wakeup, node and link failures, and mobility.

REFERENCES

- [1] C. Bettstetter. On the connectivity of ad hoc networks. *The Computer Journal*, 47(4):432–447, 2004.
- [2] J. Dams, M. Hoefer, and T. Kesselheim. Convergence time of power-control dynamics. In *ICALP (2)*, pages 637–649, 2011.
- [3] D. Dimarogonas and K. Johansson. Bounded control of network connectivity in multi-agent systems. *Control Theory & Applications*, 4(8):1330 – 1338, 2010.
- [4] A. Fanghänel, T. Kesselheim, H. Räcke, and B. Vöcking. Oblivious interference scheduling. In *PODC*, pages 220–229, August 2009.
- [5] O. Goussevskaia, M. M. Halldórsson, R. Wattenhofer, and E. Welzl. Capacity of Arbitrary Wireless Networks. In *Infocom*, pages 1872–1880, April 2009.
- [6] O. Goussevskaia, T. Moscibroda, and R. Wattenhofer. Local Broadcasting in the Physical Interference Model. In *DialM-POMC*, August 2008.
- [7] P. Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Trans. Information Theory*, 46(2):388–404, 2000.
- [8] M. M. Halldórsson. Wireless scheduling with power control. *ACM Trans. Algorithms*, 2012. To appear. See also <http://arxiv.org/abs/1010.3427>.
- [9] M. M. Halldórsson and P. Mitra. Nearly optimal bounds for distributed wireless scheduling in the SINR model. In *ICALP*, 2011.
- [10] M. M. Halldórsson and P. Mitra. Wireless Capacity with Oblivious Power in General Metrics. In *SODA*, 2011.
- [11] M. M. Halldórsson and P. Mitra. Wireless Connectivity and Capacity. In *SODA*, 2012.
- [12] M. M. Halldórsson and R. Wattenhofer. Wireless Communication is in APX. In *ICALP*, pages 525–536, July 2009.
- [13] C.-F. Huang, Y.-C. Tseng, and H.-L. Wu. Distributed protocols for ensuring both coverage and connectivity of a wireless sensor network. *ACM Trans. Sen. Netw.*, 3, March 2007.
- [14] T. Kesselheim. A Constant-Factor Approximation for Wireless Capacity Maximization with Power Control in the SINR Model. In *SODA*, 2011.
- [15] T. Kesselheim and B. Vöcking. Distributed contention resolution in wireless networks. In *DISC*, pages 163–178, August 2010.
- [16] H. Li, Q.-S. Hua, C. Wu, and F. C.-M. Lau. Minimum-latency aggregation scheduling in wireless sensor networks under physical interference model. In *MSWiM*, pages 360–367, 2010.
- [17] Z. Lotker, M. Parter, D. Peleg, and Y. A. Pignolet. Distributed power control in the SINR model. In *Infocom*, pages 2525–2533, 2011.
- [18] R. Maheshwari, S. Jain, and S. R. Das. A measurement study of interference modeling and scheduling in low-power wireless networks. In *SenSys*, pages 141–154, 2008.
- [19] J. Meng and M. Egerstedt. Connectedness preserving distributed coordination control over dynamic graphs. In *Proc. of American Control Conference*, pages 3591–3596, 2005.
- [20] T. Moscibroda. The worst-case capacity of wireless sensor networks. In *IPSN*, pages 1–10, 2007.
- [21] T. Moscibroda and R. Wattenhofer. The Complexity of Connectivity in Wireless Networks. In *Infocom*, 2006.
- [22] T. Moscibroda, R. Wattenhofer, and Y. Weber. Protocol Design Beyond Graph-Based Models. In *Hotnets*, November 2006.
- [23] T. Moscibroda, R. Wattenhofer, and A. Zollinger. Topology control meets SINR: The scheduling complexity of arbitrary topologies. In *MOBIHOC*, pages 310–321, 2006.
- [24] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *Infocom*, pages 404 – 413, 2000.

- [25] V. Rodoplu and T. H. Meng. Minimum energy mobile wireless networks. *IEEE JSAC*, 17(8):1333–1344, 1999.
- [26] C. Scheideler, A. W. Richa, and P. Santi. An $O(\log n)$ dominating set protocol for wireless ad-hoc networks under the physical interference model. In *MobiHoc*, pages 91–100, 2008.
- [27] R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. In *Infocom*, pages 1388–1397, 2001.
- [28] M. M. Zavlanos and G. J. Pappas. Distributed connectivity control of mobile networks. In *Proc. of American Control Conference*, pages 3591–3596, 2007.

APPENDIX A. MISSING PROOFS

Proof of Thm. 13

Proof. Recall that M is the set of nodes of degree at most $\rho = \frac{160}{p^2}$ in \mathcal{T} . For sets X and Y , let $\mathcal{E}(X, Y)$ be the number of links with senders in X and receivers in Y . We claim that setting $\mathcal{T}(M) = \mathcal{E}(M, M)$ fulfills the properties claimed in the theorem.

The $O(1)$ -sparsity follows by noting that the nodes in M have degree $O(1)$; the proof of Lemma 10 can be followed verbatim using the constant-degree bound instead of the $O(\log n)$ -bound employed there.

Thus, what remains to be proven is that $\mathbb{E}(|\mathcal{E}(M, M)|) = \Omega(n) = \Omega(|\mathcal{T}|)$. Let $M' = P \setminus M$ (recall that P is the set of all nodes). Since \mathcal{T} is a tree, $|\mathcal{T}| = n - 1$. Then, since the number of unique links adjacent to M is at least $\frac{1}{2}M\rho$, it is easily computed that $|M'| \leq \frac{2n}{\rho}$ and thus $|M| \geq n(1 - \frac{2}{\rho})$. We show in Lemma 22 below that $\mathbb{E}(|\mathcal{E}(M', P)|) \leq \frac{n}{e^9}$. Note that since \mathcal{T} is a connected tree, $|\mathcal{E}(M, P)| \geq |M| - 1$. Thus,

$$\begin{aligned} \mathbb{E}(|\mathcal{E}(M, M)|) &\geq \mathbb{E}(|\mathcal{E}(M, P)|) - \mathbb{E}(|\mathcal{E}(M, M')|) \\ &\geq \mathbb{E}(|M|) - 1 - \mathbb{E}(|\mathcal{E}(M', P)|) \\ &\geq n \left(1 - \frac{2}{\rho}\right) - \frac{n}{e^9} = \Omega(n), \end{aligned}$$

which implies the theorem. □

Lemma 22. $\mathbb{E}(|\mathcal{E}(M', P)|) \leq \frac{n}{e^9}$.

Proof. Recall that by Thm. 7, $\mathbb{P}(\deg(u) \geq d) \leq e^{-\frac{p^2 d}{8}}$, where $\deg(u)$ is the degree of u . This implies that $\mathbb{P}(\deg(u) \in [d, 2d]) \leq e^{-\frac{p^2 d}{8}}$. Since $\rho = \frac{160}{p^2}$, we can verify using basic calculus that $e^{p^2 \rho 2^t / 8} \geq \rho^2 2^{2t+2}$, for all t . Using this bound, we get,

$$\begin{aligned} \mathbb{E}(|\mathcal{E}(M', P)|) &\leq n \sum_{t=0}^{\infty} \mathbb{P}(\deg(u) \in [\rho 2^t, \rho 2^{t+1})) \rho 2^{t+1} \\ &\leq n \sum_{t=0}^{\infty} e^{-\frac{p^2 \rho 2^t}{8}} \rho 2^{t+1} \leq n \sum_{t=0}^{\infty} e^{-\frac{p^2 \rho 2^t}{16}} \leq n \sum_{t=0}^{\infty} e^{-10 \cdot 2^t} \\ &= \frac{n}{e^{10}} + n \sum_{t=1}^{\infty} \frac{1}{e^{10 \cdot 2^t}} \leq \frac{n}{e^{10}} + \frac{n}{e^{10}} \sum_{t=1}^{\infty} \frac{1}{e^{2^t}} \leq \frac{2n}{e^{10}} \leq \frac{n}{e^9}. \end{aligned}$$

□

Proof of Lemma 14

Proof. The proof of this Lemma follows ideas from [8] and [10]. We need to relate the idea of sparsity to the idea of “independence” used in [8].

We say that a set of links is q -independent if any two of them, $\ell = (x, y)$ and $\ell' = (x', y')$, satisfy the constraint $d(x, y') \cdot d(y, x') \geq q^2 d(x, y) \cdot d(x', y')$.

We claim,

Claim A.1. *Let C be a sufficiently large constant. Let Q be a C -independent set, and for any link ℓ in \mathcal{T}' , let Q^ℓ be the links in Q longer than ℓ . Then, $a_\ell(Q^\ell) + a_{Q^\ell}(\ell) = O(\Upsilon)$.*

Proof. Partition Q^ℓ into two sets: Q_l^ℓ , with links length at least $d(x, y) \cdot 2(2\beta n)^{2/\alpha}$, and Q_s^ℓ , with the remaining links. It follows from [8, Lemma 4.4] that $a_{Q_l^\ell}^\mathcal{M}(\ell) + a_\ell^\mathcal{M}(Q_l^\ell) = O(\log \log \Delta)$. On the other hand, Q_s^ℓ can be partitioned into $O(\log n)$ length classes. For such sets, it is known [8] that C -independence, for some constant C , implies feasibility. Let Z be such a set. By Lemma 7 of [15], $a_Z^\mathcal{M}(\ell) = O(1)$. Since Z belongs to a single length class, it is also possible to show (following arguments similar to [15]) that $a_\ell^\mathcal{M}(Z) = O(1)$. Thus, $a_{Q_s^\ell}^\mathcal{M}(\ell) + a_\ell^\mathcal{M}(Q_s^\ell) = O(\log n)$, summing over the $O(\log n)$ such Z 's. The claim follows. \square

By Lemma 23 below, we know that \mathcal{T}' can be partitioned into a constant number of C -independent sets. Let Q_1, Q_2, \dots, Q_t be a partition of L' into t different C -independent sets. For a link ℓ , let $Q_i^\ell = \{\ell' \in Q_i : \ell' \geq \ell\}$. Then,

$$\begin{aligned} a_{\mathcal{T}'}^\mathcal{M}(\mathcal{T}') &\leq \sum_{\ell=(x,y) \in \mathcal{T}'} \sum_{i=1}^t a_\ell^\mathcal{M}(Q_i^\ell) + a_{Q_i^\ell}^\mathcal{M}(\ell) \\ &= t|\mathcal{T}'|O(\Upsilon) = O(|\mathcal{T}'|\Upsilon), \end{aligned}$$

since $t = O(1)$. \square

Lemma 23. *\mathcal{T}' can be partitioned into a constant number of C -independent sets.*

Proof. Consider any link ℓ . We claim that there are $O(1)$ links ℓ' at least as long as ℓ such that ℓ and ℓ' are not C -independent. This claim proves the lemma by the following algorithm. Sort the links in an ascending order of their length, breaking ties arbitrarily.

Now consider the graph on links where there is an edge between links if they are not C -independent.

By the claim, all links have $O(1)$ edges to links after them in the ascending order. Such a graph is $O(1)$ -colorable, where each color represents an independent set in graph theoretic sense, and thus a C -independent set according to our definition.

Now we prove the claim. Recall that \mathcal{T}' is γ_3 -sparse for some constant γ_3 . Consider the link $\ell = (u, v)$ and a ball of radius $(2C)^2 \cdot d(u, v)$ around u . By a basic geometric argument, this ball can be covered by $O(1)$ balls of radius $d(u, v)/8$. By the definition of sparsity, there can be at most γ_3 links of length $d(u, v)$ or higher that have one endpoint in each of the smaller balls. Thus, the larger ball also contains only $O(1)$ such links. We now claim that all other links, i.e., $\ell' = (u', v')$ such that $\min(d(u', u), d(v', u)) \geq (2C)^2 \cdot d(u, v)$ are such that ℓ and ℓ' are C -independent. First, assume that $d(u', v) \geq \frac{1}{4}d(u', v')$. Then $d(u', v) \cdot d(u, v') \geq \frac{1}{4}d(u', v') \cdot (2C)^2 \cdot d(u, v) = C^2 d(u', v') d(u, v)$ which implies C -independence. On the other hand, if $d(u', v) < \frac{1}{4}d(u', v')$, then $d(u, v') \geq d(u', v') - d(u', v) - d(u, v) \geq d(u', v') - \frac{5}{4}d(u', v) \geq \frac{11}{16}d(u', v')$, from which C -independence follows by similar computations. \square

APPENDIX B. A SHORT PRIMER ON SPARSITY, AMENABILITY AND FEASIBILITY

In [11], a set of links L was defined to be η -**amenable** if the following holds: for any link ℓ (ℓ not necessarily a member of L), $\sum_{\ell' \in L} f_\ell(\ell') \leq \eta$, for a function f (see Eqn. 8.2.2), for some η . Actually, in [11], η is implicitly considered to be a constant, and just the term **amenable** is used.

The definition extends naturally to arbitrary η . It was shown in [14] that an η -**amenable** set L has a feasible subset of size $\Omega\left(\frac{1}{\eta}|L|\right)$.

Now the final ingredient needed is to tie sparsity to feasibility (and thus get Thm. 9). We claim that sparsity as defined in this paper implies amenability. This is implicit in [11]. Specifically, in proving the main Lemma 4 of [11], it is first shown that the structure in question (which happens to be a Minimum Spanning Tree on the set of nodes) is $O(1)$ -sparse (Lemma 5) and then this is used to show that the structure is amenable (which then implies a large feasible subset).

APPENDIX C. A NOTE ON THE APPROXIMATION FACTOR FOR DISTRIBUTED SCHEDULING

If acknowledgments have to be explicitly implemented, the algorithm of [15, 9] produces a schedule length of $O((T + T') \cdot \log n)$, where T is the optimal schedule for the input link set, and T' is the optimal schedule for the dual set of the input set, which may be larger than $O(T \log n)$. For our instance, this problem simply is not relevant. The constructed link set \mathcal{T} *is its own dual*, and thus a $O(\log n)$ -approximation factor can be safely asserted.