Improved Algorithms for Latency Minimization in Wireless Networks

Alexander Fanghänel*, Thomas Kesselheim*, Berthold Vöcking*

May 26, 2009

Abstract

In the *interference scheduling problem*, one is given a set of n communication requests described by source-destination pairs of nodes from a metric space. The nodes correspond to devices in a wireless network. Each pair must be assigned a power level and a color such that the pairs in each color class can communicate simultaneously at the specified power levels. The feasibility of simultaneous communication within a color class is defined in terms of the Signal to Interference plus Noise Ratio (SINR) that compares the strength of a signal at a receiver to the sum of the strengths of other signals. The objective is to minimize the number of colors as this corresponds to the time needed to schedule all requests.

We introduce an instance-based measure of interference, denoted by I, that enables us to improve on previous results for the interference scheduling problem. We prove upper and lower bounds in terms of I on the number of steps needed for scheduling a set of requests. For general power assignments, we prove a lower bound of $\Omega(I/(\log \Delta \log n))$ steps, where Δ denotes the aspect ratio of the metric. When restricting to the two-dimensional Euclidean space (as previous work) the bound improves to $\Omega(I/\log \Delta)$. Alternatively, when restricting to linear power assignments, the lower bound improves even to $\Omega(I)$. The lower bounds are complemented by an efficient algorithm computing a schedule for linear power assignments using only $\mathcal{O}(I \log n)$ steps. A more sophisticated algorithm computes a schedule using even only $\mathcal{O}(I + \log^2 n)$ steps. For dense instances in the two-dimensional Euclidean space, this gives a constant factor approximation for scheduling under linear power assignments, which shows that the price for using linear (and, hence, energy-efficient) power assignments is bounded by a factor of $\mathcal{O}(\log \Delta)$.

In addition, we extend these results for single-hop scheduling to multi-hop scheduling and combined scheduling and routing problems, where our analysis generalizes previous results towards general metrics and improves on the previous approximation factors.

*Department of Computer Science, RWTH Aachen University, Germany. This work has been supported by the UMIC Research Centre, RWTH Aachen University.

1 Introduction

The media access control (MAC) layer of wireless networks is responsible for scheduling signals taking into account interference caused by concurrent transmissions. Early algorithmic studies of this task were based on graph theoretical vicinity models (see, e.g., [11, 19, 9]). In more recent literature, these studies have been critized to not model interference appropriately as they assume that the interference caused by signals ends abruptly at some boudary (see, e.g.,[15, 16, 4, 5]).

Like the other recent studies mentioned above, we describe interference using the so-called *physical model* in which it is assumed that the strength of a signal fades with the distance from the sender. This fading is described by a *path loss exponent* $\alpha \geq 1$.[1] The strength of a signal sent with some power p received by a node (transceiver) v at distance d from the source of the signal is assumed to be p/d^{α}. The node v can successfully receive the signal if its strength is sufficiently large in comparison to the sum of other signals that are sent simultaneously plus ambient noise, that is, if the *signal to interference plus noise ratio (SINR)* is above some threshold $\beta > 1$, the so-called *gain*.

The *interference scheduling problem* is formally defined as follows. Let V be a set of nodes from a metric space. Let $d(u, v)$ denote the distance between two nodes u and v. One is given a set $\mathcal{R}$ of n requests consisting of pairs $(u_i, v_i) \in V^2$, where u_i is the source and v_i the destination of the signal from the i-th request. For every $i \in [n] := \{1, \ldots, n\}$, one needs to specify a power level $p_i > 0$ and a color $c_i \in [k] := \{1, \ldots, k\}$ such that the *latency*, i. e., the number of colors, k, is minimized and the pairs in each color class satisfy the SINR constraints for all signals: For every $i \in [n]$, it must hold that

$$\frac{p_i}{d(u_i, v_i)^{\alpha}} \geq \beta \left(\sum_{\substack{j \in [n] \setminus \{i\} \\ c_j = c_i}} \frac{p_j}{d(u_j, v_i)^{\alpha}} + \nu \right) ,$$

where $\nu \geq 0$ expresses ambient noise. The so-called *scheduling complexity* of $\mathcal{R}$, as introduced by Moscribroda and Wattenhofer [15], is the minimal number of colors (steps) needed to schedule the requests in $\mathcal{R}$.

In this work, we mostly focus on *linear power assignments*, i.e., for a request pair (u_i, v_i) the power is proportional to $d(u_i, v_i)^{\alpha}$ and, hence, linear in fading. Linear power schemes also have been considered in [2, 21, 4]. Our analysis will show, that one loses only a factor of order $\log \Delta$ due to restricting to this power scheme (where the aspect ratio Δ denotes the ratio between the longest and shortest distance between any two nodes). Let us remark that the dependence on the aspect ratio Δ cannot be avoided using the linear power assignment which, without taking into account other parameters than n, cannot achieve an approximation ratio better than $\Omega(n)$ [15, 7]. Besides leading to good performance results, linear power assignments have the advantage being energy-efficient as the minimal transmission power required to transmit along a distance d is $\Theta(d^{\alpha})$.

1.1 Our contribution

We introduce an instance-based measure of interference that enables us to estimate the scheduling complexity of any set of requests within small factors.

Definition 1 (Measure of Interference). *Let $\mathcal{R} \subseteq V \times V$ be a set of requests. For $w \in V$ define*

$$I_w(\mathcal{R}) = \sum_{(u,v) \in \mathcal{R}} \min \left\{ 1, \frac{d(u,v)^{\alpha}}{d(u,w)^{\alpha}} \right\} .$$

Using this we define the measure of interference induced by the requests $\mathcal{R}$:

$$I = I(\mathcal{R}) = \max_{w \in V} I_w(\mathcal{R}) .$$

We prove upper and lower bounds on the number of steps needed for scheduling $\mathcal{R}$ in terms of I. For general power assignments and general metrics, we prove a lower bound of $\Omega(I/\log \Delta \log n)$ steps. When restricting to the two-dimensional Euclidean space and assuming $\alpha > 2$ the bound improves to $\Omega(I/\log \Delta)$. Alternatively, when restricting to linear power assignments and assuming general metrics, this bound improves even to $\Omega(I)$. The lower bounds are complemented by an efficient algorithm computing a schedule for linear power assignments

[1] It is usually assumed, that α satisfies $2 < \alpha < 5$. Our analysis holds for any constant $\alpha \geq 1$, unless stated otherwise.

using only $\mathcal{O}(I \log n)$ steps. A more sophisticated algorithm computes a schedule using even only $\mathcal{O}(I + \log^2 n)$ steps. This gives a constant factor approximation of the optimal schedule under linear power assignments for *dense* instances, i.e., if $I \geq \log^2 n$. Combining this upper bound for linear power assignments with the lower bound for general power assignments and the two-dimensional Euclidean space shows that the price for using linear, in other words, energy-efficient power assignments is of order $\mathcal{O}(\log \Delta)$.

We further extend our results towards multi-hop scheduling and routing. In the multi-hop scheduling problem, a request is defined by a sequence of pairs, so-called *paths*, rather than a single pair of nodes. Along each of these paths, one should forward a signal from the first to the last node on the path. Let D denote the maximum number of hops on each of these paths, the so-called *dilation*. Generalizing, the lower bounds from the single-hop to the multi-hop problem, shows that one needs at least $\Omega(I/_{\log \Delta \log n} + D)$ steps, for general power assignments, $\Omega(I/_{\log \Delta} + D)$ for the Euclidean space, and $\Omega(I + D)$ steps, for linear power assignments. We show how to extend our second algorithm for the single-hop scheduling to the multi-hop case, where it produces a schedule of at most $\mathcal{O}(I + D \cdot \log^2 n)$ steps.

Our results for multi-hop scheduling reminds of the $\mathcal{O}(\text{congestion} + \text{dilation})$-type results that have been shown previously for routing in wired networks, see, e.g. [12, 13, 1, 20]. In fact, this previous work was the inspiration to search for an instance-based density measure that allows to derive lower bounds for the scheduling complexity in wireless networks like the congestion in wired networks. At this point, let us remark that, unlike the congestion, our interference measure I does not trivially give a lower bound on the number of steps needed for scheduling a set of requests but it requires a careful analysis as also the upper bound does.

Finally, we extend our result to combined multi-hop routing and scheduling. Now requests are again defined by pairs of nodes. The problem is to find source-destination paths for all requests and to compute a power assignment and a schedule delivering all packets using as few steps as possible. Combining our multi-hop scheduling algorithm with a linear programming approach for computing paths that minimize the term $\max\{I, D\}$ gives an $\mathcal{O}(\log \Delta \log^3 n)$-approximation for the combined routing and scheduling problem in general metrics. In the two-dimensional Euclidean space the approximation factor is $\mathcal{O}(\log \Delta \log^2 n)$. This generalizes the results from Chafekar et al. [4] (cf. Section 1.2) towards general metrics and improves on their approximation factors.

1.2 Related Work

The first theoretical studies about interference scheduling in the physical model focus on topologies generated by placing nodes randomly in two-dimensional Euclidean space, see, e.g., [8, 3, 10].

The study of interference scheduling with respect to arbitrary topologies has been initiated by Moscibroda and Wattenhofer [15]. They present the first analysis of the interference scheduling problem. However, they do not handle general request sets but only specific kinds of sets. In particular, they study the question of how many time slots are needed to schedule a set of communication requests ensuring strong connectivity among n points placed arbitrarily in two-dimenisonal Euclidean space. On the one hand, they prove that there are configurations requiring $\Omega(n)$ time slots using either uniform or linear power assignments, when not taking other parameters, like the aspect ratio Δ, into account. On the other hand, they show that $\mathcal{O}(\log^4 n)$ time slots are sufficient to ensure strong connectivity when choosing the right power assignment.

This result has been extended by Moscibroda et al. [16] to arbitrary demands. Their result is an $\mathcal{O}(\log^2 n \cdot I_{\text{in}})$ algorithm, where I_{in} is a certain interference measure. This result enables them to improve the bound for strong connectivity from $\mathcal{O}(\log^4 n)$ to $\mathcal{O}(\log^3 n)$. Unfortunately, I_{in} is no lower bound for the optimal schedule length. Thus, it does not give any approximation guarantee for general request sets since there is no comparison between I_{in} and the optimal schedule length.

In [14], another measure of interference χ_ρ called disturbance is introduced where $\rho > 0$ is a parameter. The algorithm described achieves a schedule length of $\mathcal{O}(\chi_\rho \rho^2 \log n \cdot (\log n + \rho))$. Unfortunately, also this result does not yield a comparison to the optimal schedule length.

Fanghänel et al. [7] deal with directed and undirected request sets. For the directed case they extend the results of Moscibroda and Wattenhofer by showing that any power assignment that is *oblivous*, i. e., the transmission power is based only on the distance between the sender and the receiver, cannot be bounded in an useful manner without taking into account metric properties like the aspect ratio Δ. For the undirected case they prove the *square-root power assignment* to be an $\mathcal{O}(\log^{3.5+\alpha} n)$-approximation. However, neither is this power scheme energy efficient, nor can their constructive results be generalized towards the multi-hop case with standard techniques, as there is no measure of interference given that is a lower bound for the optimal schedule.

Chafekar et al. [4] study the combined routing and multi-hop version of the interference scheduling problem. It is crucial for their analysis to deal with two-dimensional Euclidean instances and $\alpha > 2$. This allows to use graph coloring in a similar way to the approaches used in the graph-theoretical vicinity models. Our approach instead works in general metrics taking the non-locality of the SINR constraint into account. In their analysis the considered power assignment is restricted, that is, it is assumed that power levels must be chosen from a specified interval $[p_{\min}, p_{\max}]$. It yields a schedule using $\mathcal{O}(\mathrm{opt}' \cdot \log^2 n \log \Delta \log^2 \Gamma)$ time slots where opt' denotes the minimal number of time slots needed for a schedule with slightly smaller power range $[p_{\min}, (1-\epsilon)p_{\max}]$ and Γ denotes the ratio between $p_{\max}$ and $p_{\min}$.

2 Introducing a Measure of Interference

In this section we justify the choice of our measure of interference, as it yields lower bounds for the optimal schedule length under both arbitrary and linear power assignments. In Section 2.1 we show, that the length T of an optimal schedule using a linear power assignment is lower bounded by $\Omega(I)$. In Section 2.2 we show, that a lower bound for the length of an optimal schedule under an arbitrary power assignment is $\Omega(I/\log\Delta\cdot\log n)$ in general metrics and $\Omega(I/\log\Delta)$ in the two-dimensional Euclidean space for $\alpha > 2$.

2.1 A Comparison to the Optimal Schedule Using Linear Power Assignments

Theorem 1. *Let T be the minimum schedule length for a set of requests $\mathcal{R}$ in a linear power assignment. Then we have $T = \Omega(I)$.*

Proof. Let there be a schedule of length T when using a linear power assignment. Then there exist sets of requests $\mathcal{R}_1, \ldots, \mathcal{R}_T$ each of which satisfies the SINR constraint for the linear power assignment. I is subadditive, i. e., we have $I\left(\bigcup_{t=1}^{T} \mathcal{R}_t\right) \leq \sum_{t=1}^{T} I(\mathcal{R}_t)$. Thus it suffices to show that $I(\mathcal{R}_t) = \mathcal{O}(1)$ for such a set.

Let $\mathcal{R}_t = \{(u_1, v_1), \ldots, (u_{\bar{n}}, v_{\bar{n}})\}$. Let furthermore be $w \in V$. The node w does not necessarily act as a receiver v_i in this request set $\mathcal{R}_t$. This is why we define v_j as the closest (active) receiver from w, i. e. $j \in \arg\min_{i\in[\bar{n}]} d(v_i, w)$. This node might also be w itself.

To bound the measure of interference, we distinguish between two kinds of requests. We define a set U of indices of requests whose senders u_i lie within a distance of at most $\frac{1}{2}d(v_j, w)$ from w, i. e. $U = \{i \in [\bar{n}] \mid d(u_i, w) \leq \frac{1}{2}d(v_j, w)\}$. Using the triangle inequality we can conclude for all $i \in U$:

$$d(u_i, v_j) \leq d(u_i, w) + d(w, v_j) \leq \frac{3}{2}d(v_j, w) \quad . \tag{1}$$

In addition, we have

$$\begin{aligned} d(v_j, w) &\leq d(v_i, w) && \text{since } v_j \text{ is the closest receiver} \\ &\leq d(v_i, u_i) + d(u_i, w) && \text{by triangle inequality} \\ &\leq d(v_i, u_i) + \frac{1}{2}d(v_j, w) && \text{by definition of } U \ . \end{aligned}$$

This implies

$$d(v_j, w) \leq 2d(u_i, v_i) \quad . \tag{2}$$

Combining Equation 1 and Equation 2 we get $d(u_i, v_j) \leq 3d(u_i, v_i)$. Thus it holds

$$|U \setminus \{j\}| = \sum_{\substack{i\in U\\ i\neq j}} \frac{d(u_i, v_i)^\alpha}{d(u_i, v_i)^\alpha} \leq \sum_{\substack{i\in U\\ i\neq j}} \frac{d(u_i, v_i)^\alpha}{\frac{1}{3^\alpha}d(u_i, v_j)^\alpha} \leq \frac{3^\alpha}{\beta} \quad .$$

For all $i \in [\bar{n}] \setminus U$ it holds that

$$\begin{aligned} d(u_i, v_j) &\leq d(u_i, w) + d(w, v_j) && \text{by triangle inequality} \\ &\leq d(u_i, w) + 2d(u_i, w) && \text{by definition of } U \\ &= 3d(u_i, w) \ . \end{aligned}$$

Now, we can sum up all $i \in [\bar{n}] \setminus U$:

$$\sum_{\substack{i\in[\bar{n}]\setminus U\\ i\neq j}} \frac{d(u_i, v_i)^\alpha}{d(u_i, w)^\alpha} \leq \sum_{\substack{i\in[\bar{n}]\setminus U\\ i\neq j}} \frac{d(u_i, v_i)^\alpha}{\frac{1}{3^\alpha}d(u_i, v_j)^\alpha} \leq \frac{3^\alpha}{\beta} \quad .$$

Summing up all $i \in [\bar{n}]$ gives

$$I_w(\mathcal{R}_t) = \sum_{i \in [\bar{n}]} \min \left\{ 1, \frac{d(u_i, v_i)^\alpha}{d(u_i, w)^\alpha} \right\} \leq |U \setminus \{j\}| + \sum_{\substack{i \in [\bar{n}] \setminus U \\ i \neq j}} \frac{d(u_i, v_i)^\alpha}{d(u_i, w)^\alpha} + 1 \leq \frac{2 \cdot 3^\alpha}{\beta} + 1 = \mathcal{O}(1) \ .$$

□

2.2 A Comparison to the Optimal Schedule

Theorem 2. *Let T denote the optimal schedule length using any power assignment. Then we have $T = \Omega\left({}^{I}\!/\!_{\log \Delta \cdot \log n} \right)$.*

Proof. To prove this theorem, we use a similar technique as in the proof of Theorem 1. However, we have to deal with an unknown power assignment. Since there is a schedule of length T in this power assignment, there exist sets of requests $\mathcal{R}_1, \ldots, \mathcal{R}_T$ each of which satisfies the SINR constraint for this power assignment. We divide such a set $\mathcal{R}_t$ into $\log \Delta$ classes $C_{t,j} = \{(u,v) \in \mathcal{R}_t \mid 2^{j-1} d_{\min} \leq d(u,v) \leq 2^j d_{\min}\}$, where $d_{\min} = \min_{(u,v) \in \mathcal{R}} d(u,v)$. Again, by using the subadditivity of I, it suffices to show that $I(C_{t,j}) = \mathcal{O}(\log n)$ for such a class. Fix $C_{t,j}$ and let $C_{t,j} = \{(u_1, v_1), \ldots, (u_{\bar{n}}, v_{\bar{n}})\}$. Further, for clarity we write $L = 2^{j-1} d_{\min}$.

As an important fact we can bound the number of requests whose senders are located around a node within a distance of at most ℓ.

Fact 3. *For all $w \in V$, $\ell \geq L$ we have for $K_\ell(w) = \{i \in [\bar{n}] \mid d(u_i, w) \leq \ell\}$:*

$$|K_\ell(w)| \leq \frac{1}{\beta} \left(\frac{4\ell}{L} \right)^\alpha + 1 \ .$$

Proof. Let p be the power assignment that allows all requests to be served in a single time slot. Let furthermore be (u_k, v_k) be the request with $k \in K_L(w)$ that is transmitted with minimal power p_k. As the SINR condition is satisfied for request (u_k, v_k), we get:

$$\frac{1}{\beta} \frac{p_k}{d(u_k, v_k)^\alpha} \geq \sum_{\substack{i \in K_\ell(w) \\ i \neq k}} \frac{p_i}{d(u_i, v_k)^\alpha} \geq \sum_{\substack{i \in K_\ell(w) \\ i \neq k}} \frac{p_i}{(2\ell + 2L)^\alpha} \geq \frac{(|K_\ell(w)| - 1) \cdot p_k}{(2\ell + 2L)^\alpha} \ .$$

So:

$$|K_\ell(w)| - 1 \leq \frac{1}{\beta} \left(\frac{2\ell + 2L}{d(u_k, v_k)} \right)^\alpha \leq \frac{1}{\beta} \left(\frac{4\ell}{L} \right)^\alpha \ .$$

□

To prove $I(C_{t,j}) = \mathcal{O}(\log n)$, let w be a vertex such that $I_w(C_{t,j}) = I(C_{t,j})$. W.l.o.g. let $u_1, \ldots, u_{\bar{n}}$ be ordered by increasing distance to w. Note that for all $\ell > 0$ we have $K_\ell(w) = \{1, \ldots, x\}$ for some $x \in \mathbb{N}$ by this definition.

For $k \leq \log \bar{n} + 1$ let be $R_k = [2^k] \setminus [2^{k-1}]$. Furthermore, let ℓ_k be defined as $\ell_k = \min_{i \in R_k} d(u_i, w)$. For $I(C_{t,j})$ follows from these definitions:

$$I(C_{t,j}) = \sum_{i=1}^{\bar{n}} \min \left\{ 1, \frac{d(u_i, v_i)^\alpha}{d(u_i, w)^\alpha} \right\} \leq \sum_{k=1}^{\log \bar{n} + 1} \sum_{i \in R_k} \frac{d(u_i, v_i)^\alpha}{d(u_i, w)^\alpha} + \sum_{i \in K_L(w)} 1 \leq (2L)^\alpha \sum_{k=1}^{\log \bar{n} + 1} \frac{|R_k|}{\ell_k^\alpha} + |K_L(w)|$$

As the distances are increasing, we have $\ell_k \geq d(u_i, w)$ for all $i \leq 2^{k-1}$. In other words: $[2^{k-1}] \subseteq K_{\ell_k}(w)$.

Since we add up the interference induced by requests from $K_L(w)$ separately, we may assume $\ell_k \geq L$ for all k and thus apply Fact 3 on $|K_{\ell_k}(w)|$, which gives

$$2^{k-1} = |[2^{k-1}]| \leq |K_{\ell_k}(w)| \leq \left(\frac{4\ell_k}{L} \right)^\alpha + 1 \ .$$

Consequently, we have

$$\ell_k^\alpha \geq (2^{k-1} - 1) \left(\frac{L}{4} \right)^\alpha \ .$$

Using the above results for ℓ_k^α and $|K_L(w)|$ we can bound $I(C_{t,j})$ by:

$$(2L)^\alpha \sum_{k=1}^{\log \bar{n}+1} \frac{2^{k-1}}{(2^{k-1}-1)\left(\frac{L}{4}\right)^\alpha} + \left(\frac{4^\alpha}{\beta}+1\right) \leq 8^\alpha \sum_{k=1}^{\log \bar{n}+1} 2 + \frac{4^\alpha}{\beta} + 1 = \mathcal{O}(\log n)$$

□

In previous work, the instances often are restricted to the Euclidean plane and α is required to be strictly greater than 2. Under these assumptions we can use geometric arguments to get an even better bound of $\Omega(I/\log\Delta)$ on the optimal schedule length, as we show in the following.

Theorem 4. *Let the instance be located in the Euclidean plane and let $\alpha > 2$. Then we have $T = \Omega\left(I/\log\Delta\right)$, where T denotes the optimal schedule length using any power assignment.*

Proof. Again, we divide the requests into $\log\Delta \cdot T$ classes $C_{t,i}$. This time, we have to prove $I(C_{t,i}) = \mathcal{O}(1)$. Let us remark that in the Euclidean plane a ring of inner radius $L \cdot r$ and width L can be covered by $8(r+1)$ circles of radius L. If x is the center of such a circle, we get from Fact 3 that $|K_L(x)| \leq \frac{4^\alpha}{\beta}$. Thus we have $|K_{L(r+1)}(w) \setminus K_{Lr}(w)| \leq 8(r+1)\frac{4^\alpha}{\beta} \leq 16r\frac{4^\alpha}{\beta} = r\frac{4^{\alpha+2}}{\beta}$ for $r \geq 1$. We can write $I_w(C_{t,j}) = I(C_{t,j})$ as

$$I(C_{t,j}) \leq \sum_{r=1}^{\infty} |K_{L(r+1)}(w) \setminus K_{Lr}(w)| \cdot \frac{(2L)^\alpha}{(Lr)^\alpha} + |K_L(w)| \ .$$

Using the above result we get:

$$2^\alpha \frac{4^{\alpha+2}}{\beta} \sum_{r=1}^{\infty} r^{1-\alpha} + \frac{4^\alpha}{\beta} \leq \frac{4^\alpha}{\beta}\left(2^\alpha 4^2 \frac{\alpha-1}{\alpha-2} + 1\right) = \mathcal{O}(1) \ .$$

□

In total we proved several bounds on the measure of interference that allow comparisons to the scheduling complexity. To complete these results, we will present a single-hop algorithm that generates a schedule of length $\mathcal{O}(I + \log^2 n)$ whp in the next section and extend this to multi-hop scheduling afterwards.

3 Single-Hop Scheduling

The measure of interference enables us to design randomized algorithms using linear power assignments, i. e., the power for the transmission from u to v is $c \cdot d(u,v)^\alpha$ for some fixed $c \geq \beta\nu$. As a key fact, we can simplify the SINR constaint in this setting as follows. If $\mathcal{R}$ is a set of requests that can be scheduled in one time slot, we have for all nodes v' with $(u',v') \in \mathcal{R}$

$$\sum_{\substack{(u,v)\in\mathcal{R} \\ (u,v)\neq(u',v')}} \frac{c \cdot d(u,v)^\alpha}{d(u,v')^\alpha} \leq \frac{c}{\beta} - \nu \ .$$

Since $\beta > 1$ we can write equivalently

$$I_{v'}(\mathcal{R}) = \sum_{(u,v)\in\mathcal{R}} \min\left\{1, \frac{d(u,v)^\alpha}{d(u,v')^\alpha}\right\} \leq \frac{1}{\beta} - \frac{\nu}{c} \ . \tag{3}$$

For simplicity of notation we replace $\frac{1}{\beta} - \frac{\nu}{c}$ by $\frac{1}{\beta'}$ in the following proofs.

3.1 A Basic Algorithm

The idea of our basic algorithm (Algorithm 1) is that each sender decides randomly in each time slot if it tries to transmit until it is successful. The probability of transmission is set to $\frac{1}{2\beta' I}$ and is not changed throughout the process.

Theorem 5. *Algorithm 1 generates a schedule of length at most $\mathcal{O}(I \log n)$ whp.*

```
while packet has not been successfully transmitted do
    try transmitting with probability 1/(2β'I)
end
```

Algorithm 1: A simple single-hop algorithm

Proof. Let us first consider the probability of success for a fixed request (u_k, v_k) in a single step of the algorithm. Let X_i, $i \in [n]$, be the 0/1 random variable indicating if sender u_i tries to transmit in this step. Assume a sender u_k tries to transmit in this step, i. e. $X_k = 1$. To make this attempt successful, the SINR constraint (Equation 3) has to be satisfied. We can express this event as $Z \leq 1/\beta'$ where Z is defined by

$$Z = \sum_{\substack{i \in [n] \\ i \neq k}} \min \left\{ 1, \frac{d(u_i, v_i)^\alpha}{d(u_i, v_k)^\alpha} \right\} X_i \ .$$

We have $\mathbf{E}[Z] \leq 1/2\beta'$ and thus we can use Markov's inequality to bound the probability that this packet cannot be transmitted successfully by

$$\mathbf{Pr}\left[Z \geq \frac{1}{\beta'} \right] \leq \mathbf{Pr}\left[Z \geq 2\mathbf{E}[Z] \right] \leq \frac{1}{2} \ .$$

To make the transmission successful the two events $X_k = 1$ and $Z \leq 1/\beta'$ have to occur. Since they are independent it holds that

$$\mathbf{Pr}\left[X_k = 1, Z \leq \frac{1}{\beta'} \right] = \mathbf{Pr}\left[X_k = 1 \right] \cdot \mathbf{Pr}\left[Z \leq \frac{1}{\beta'} \right] \geq \frac{1}{2\beta' I} \left(1 - \frac{1}{2} \right) = \frac{1}{4\beta' I} \ .$$

The probability for packet k not to be successfully transmitted in $(k_0 + 1) 4\beta' I \ln n$ independent repeats of such a step is therefore at most

$$\left(1 - \frac{1}{4\beta' I} \right)^{(k_0+1)4\beta' I \ln n} \leq e^{-(k_0+1)\ln n} = n^{-(k_0+1)} \ .$$

Applying a union bound we get an overall bound on the probability that one of n packets is not successfully transmitted in these independent repeats by n^{-k_0}. This means all senders are successful within $\mathcal{O}(I \log n)$ steps whp. $\square$

3.2 A More Sophisticated Algorithm

An obvious disadvantage of the basic algorithm is that the probability of transmission stays the same throughout the process. To improve it, one idea could be to increase the probability of transmission after some transmissions have successfully taken place. This why we need the following weighted Chernoff bound that can deal with dependent random variables.

Lemma 6. *Let $X_1, \ldots, X_n$ be 0/1 random variables for which there is a $p \in [0, 1]$ such that for all $k \in [n]$ and all $a_1, \ldots, a_{k-1} \in \{0, 1\}$*

$$\mathbf{Pr}\left[X_k = 1 \mid X_1 = a_1, \ldots X_{k-1} = a_{k-1} \right] \leq p \ . \tag{4}$$

Let furthermore $w_1, \ldots, w_n$ be reals in $(0, 1]$ and $\mu \geq p \sum w_i$. Then the weighted Chernoff bound

$$\mathbf{Pr}\left[\sum_{i=1}^{n} w_i X_i \geq (1 + \delta)\mu \right] \leq \left(\frac{e^\delta}{(1 + \delta)^{(1+\delta)}} \right)^\mu$$

holds.

Proof (Sketch). To show this bound, a standard proof for the weighted Chernoff bound [17] can be adapted. By using the definition of expectation and repeatedly applying Equation 4, one can show that

$$\mathbf{E}\left[e^{tX} \right] \leq \prod_{i=1}^{n} \left(p e^{t w_i} + 1 - p \right) \ ,$$

although random variables are no more independent. In the original proof no other step makes use of the independence.

$\square$

We can now use this bound to analyze Algorithm 2. This algorithm assigns random delays to all packets. The maximum delay is decreased depending on I^{curr}, which denotes the measure of interference that is induced by the requests that have not been scheduled at this point.

while $I^{curr} \geq \log n$ **do**
  $J := I^{\text{curr}}$
  while $I^{curr} \geq \frac{J}{2}$ **do**
    if *packet* i *has not been successfully transmitted* **then**
      assign a delay $1 \leq \delta_i \leq 16e\beta' J$ i. u. r.
      try transmission after waiting the delay
    end
  end
end
execute algorithm Algorithm 1

Algorithm 2: An $\mathcal{O}(I + \log^2 n)$ whp algorithm

The algorithm works as follows: During one iteraton of the outer *while* loop by repeatedly assigning random delays to the packets the measure of interference is reduced to a half of its initial value. This is repeated until we have $I^{\text{curr}} < \log n$ and the basic algorithm is applied.

Our first observation is that reducing I^{curr} by factor 2 takes $\mathcal{O}(I^{\text{curr}})$ scheduling steps whp.

Lemma 7. *During one iteration of the outer* while *loop, the inner* while *loop of Algorithm 2 is executed at most* $k_0 + 2$ *times with probability at least* $1 - n^{-k_0}$ *for all constants* k_0*.*

Proof. Let us first consider a single iteration of this loop. We assume all senders are taking part as if none has been successful during this iteration of the outer *while* loop yet. We only benefit from any previous success.

Observe, if the senders of a set S are transmitting and there is a collision for packet i we have

$$\sum_{\substack{j \in S \\ j < i}} \min\left\{1, \frac{d(u_j, v_j)^\alpha}{d(u_j, v_i)^\alpha}\right\} > \frac{1}{2\beta'} \quad \text{or} \quad \sum_{\substack{j \in S \\ j > i}} \min\left\{1, \frac{d(u_j, v_j)^\alpha}{d(u_j, v_i)^\alpha}\right\} > \frac{1}{2\beta'}.$$

In the first case let $Y_i^< = 1$, in the second one $Y_i^> = 1$. We now show that the random variables $Y_1^<, \ldots, Y_n^<$ fulfill Equation 4 with probability $p = \frac{1}{8e}$. Let us fix $k \in [n]$ and $a_1, \ldots, a_{k-1} \in \{0, 1\}$. We have to show $\mathbf{Pr}\left[Y_k^< = 1 \mid Y_1^< = a_1, \ldots, Y_{k-1}^< = a_{k-1}\right] \leq p$.

Since the delays δ_i are drawn independently they can be considered as if they were drawn one after the other in order $\delta_1, \delta_2, \ldots$. Then the value of $Y_i^<$ would already be determined after drawing δ_i by definition. In other words: The values of $\delta_1, \ldots, \delta_{k-1}$ already determine the values of $Y_1^<, \ldots, Y_{k-1}^<$. It follows that there is a subset $M \subseteq [16e\beta' J]^{k-1}$ of delay values such that $Y_1^< = a_1, \ldots, Y_{k-1}^< = a_{k-1}$ iff $(\delta_1, \ldots, \delta_{k-1}) \in M$.

Now let X_i be a 0/1 random variable for $i \in [k-1]$ such that $X_i = 1$ iff $\delta_i = \delta_k$. We can observe that we have for all $(b_1, \ldots, b_{k-1}) \in [16e\beta' J]^{k-1}$:

$$\mathbf{E}\left[X_i \mid \delta_1 = b_1, \ldots, \delta_{k-1} = b_{k-1}\right] = \frac{1}{16e\beta' J} \quad .$$

Define furthermore

$$Z_k^< = \sum_{i=1}^{k-1} \min\left\{1, \frac{d(u_i, v_i)^\alpha}{d(u_i, v_k)^\alpha}\right\} X_i$$

with $\mathbf{E}\left[Z_k^< \mid \delta_1 = b_1, \ldots, \delta_{k-1} = b_{k-1}\right] \leq \frac{1}{16e\beta'}$. Now it holds that

$$\begin{aligned}
\mathbf{Pr}\left[Y_k^< = 1 \mid \delta_1 = b_1, \ldots, \delta_{j-1} = b_{k-1}\right] &= \mathbf{Pr}\left[Z_k^< > \frac{1}{2\beta'} \,\middle|\, \delta_1 = b_1, \ldots, \delta_{k-1} = b_{k-1}\right] \\
&\leq 2\beta' \mathbf{E}\left[Z_k^< \mid \delta_1 = b_1, \ldots, \delta_{k-1} = b_{k-1}\right] \\
&= \frac{1}{8e} = p \quad .
\end{aligned}$$

We can now apply the so-called law of alternatives:

$$\begin{aligned}
& \mathbf{Pr}\left[Y_k^< = 1 \mid Y_1^< = a_1, \ldots, Y_{k-1}^< = a_{k-1}\right] \\
= & \sum_{(b_1,\ldots,b_{k-1})\in M} \mathbf{Pr}\left[\delta_1 = b_1, \ldots \delta_{k-1} = b_{k-1} \mid Y_1^< = a_1, \ldots, Y_{k-1}^< = a_{k-1}\right] \\
& \quad \cdot \mathbf{Pr}\left[Y_k^< = 1 \mid \delta_1 = b_1, \ldots, \delta_{k-1} = b_{k-1}\right] \\
\leq & \ p \ .
\end{aligned}$$

Thus we may apply Lemma 6 on $I_w^<$ defined as follows:

$$I_w^< = \sum_{i=1}^n \min\left\{1, \frac{d(u_i,v_i)^\alpha}{d(u_i,w)^\alpha}\right\} Y_i^< \ .$$

This random variable indicates the remaining measure of interference that is caused by these collisions. Setting $\delta = 2e - 1$ and $\mu = \frac{J}{8e}$ Lemma 6 states

$$\mathbf{Pr}\left[I_w^< \geq \frac{J}{4}\right] \leq 2^{-\frac{J}{4}} \leq n^{-1} \ .$$

Let us now consider the situation after $k_0 + 2$ iterations of the inner *while* loop. Since these are independent repeats we have

$$\mathbf{Pr}\left[I_w^< \geq \frac{J}{4}\right] \leq n^{-(k_0+2)} \ .$$

With a symmetric argument this also applies to $I_j^>$. For a sender that has not been successful we have $Z_j^< + Z_j^> \geq 1$. This means we have the bound $I_w^{\text{curr}} \leq I_w^< + I_w^>$. For the remaining measure of interference $I^{\text{curr}} = \max_{w\in V} I_w^{\text{curr}}$ we can conclude

$$\begin{aligned}
\mathbf{Pr}\left[I^{\text{curr}} \geq \frac{J}{2}\right] & \leq \sum_{w\in V} \mathbf{Pr}\left[I_w^{\text{curr}} \geq \frac{J}{2}\right] \\
& \leq \sum_{w\in V} \mathbf{Pr}\left[I_w^< \geq \frac{J}{4} \text{ or } I_w^< \geq \frac{J}{4}\right] \\
& \leq n\left(n^{-(k_0+2)} + n^{-(k_0+2)}\right) \\
& \leq n^{-k_0} \ .
\end{aligned}$$

□

Using this lemma, we can add up all numbers of steps that are generated in the *while* loops.

Theorem 8. *Algorithm 2 generates a schedule of length at most $\mathcal{O}(I + \log^2 n)$ steps whp.*

Proof. Let T_k denote the number of scheduling steps generated in the k-th execution of the outer *while* loop. In the previous lemma we showed $T_k = \mathcal{O}(I/2^k)$ whp. This means the *while* loops generate in total $\sum_k T_k = \mathcal{O}(I)$ step whp. The basic algorithm generates at most an additional $\mathcal{O}(\log^2 n)$ steps whp which concludes the proof. □

In sufficiently dense instances, i. e., $I \geq \log^2 n$, this algorithm yields a constant-factor approximation for the optimal schedule compared to the linear power assignment with high probability. Compared to the optimal power assignment the approximation factor is $\mathcal{O}(\log\Delta \cdot \log n)$ whp for general metrics resp. $\mathcal{O}(\log\Delta)$ for the two-dimensional Euclidean plane.

Algorithm 1 can be implemented in a distributed way losing a factor $\log n$ in the following way. In contrast to the centralized problem, the nodes do not know the correct value of I, thus, they do not know their transmission probability. Now in the distributed setting the algorithm processes in each *while* iteration $\log n$ steps, where in each of these steps the transmission probability is halfed, that is, starting by $1/2\beta'$ down to $1/2\beta' n$.

Algorithm 2 can be modified analogeously, leading to a schedule of length $O(\log n \cdot (I + \log^2 n))$ whp.

4 Extensions for Multi-hop Scheduling and Routing

The multi-hop variant of the interference scheduling problem was first stated by Chafekar et al. [4] as *Cross-Layer Latency Minimization* (CLM). Given m source destination pairs (s_i, t_i), the objective is to find paths from s_i to t_i to send the packets along, powers for each transmission and a schedule assigning the hops to time slots. In this section we will present how the measure of interference introduced in Section 2 and the single-hop algorithms from Section 3 can be extended to multi-hop scheduling.

4.1 Multi-hop Scheduling with Fixed Paths

Let us first consider the paths to be fixed. In this case the task is to schedule a set of requests $\mathcal{R}$ consisting of n pairs of nodes that lie on paths, respecting dependencies such that one request may not be served before the ones lying earlier on the path have been served. Obviously, the bounds on the measure of interference proven in Section 2 still hold. However, we additionally express these dependencies in the dilation D, which is the maximum path length. Of course, any schedule using an arbitrary power assignment has length at least D.

In a naive approach to solve this problem we could regard the multi-hop problem as a concatenation of D single-hop problems and schedule each of them separately. This schedule has a length of $\mathcal{O}((I + \log^2 n)D)$ steps whp. Algorithm 3 extends this idea by assigning a random delay to each packet. This technique has also been applied for scheduling in wired networks, e.g., by Leighton et al. [13].

By this shift, a number of time frames is created and to each of them a set of requests $\mathcal{R}_i$ is assigned. Due to the random delay the measure of interference $I(\mathcal{R}_i)$ is sufficiently balanced between those time frames. As different hops that lie on the same path are assigned to different time frames, our single-hop algorithm can be used to generate a schedule for each time frame.

forall $i \in [m]$ **do**
  assign a delay $1 \leq \delta_i \leq \frac{2eI}{\log^2 n}$ i. u. r.
end
forall $1 \leq t \leq \frac{2eI}{\log^2 n} + D$ **do**
  execute Algorithm 2 on all hops (i, j) with $\delta_i + j = t$
end

Algorithm 3: Fixed path multi-hop scheduling

Theorem 9. *The schedule generated by Algorithm 3 has length $\mathcal{O}(I + D \log^2 n)$ whp.*

Proof. Let $I_w(\mathcal{R}_t)$ be the random variable of I caused by all requests assigned to time frame t. Let $P_{i,j}$ denote the i-th node on the j-th path. Let $X_{i,j,t}$ be a 0/1 random variable such that $X_{i,j,t} = 1$ iff $\delta_i + j = t$. Then we have

$$I_w(\mathcal{R}_t) = \sum_{i,j} \min \left\{ 1, \frac{d(P_{i,j-1}, P_{i,j})^\alpha}{d(P_{i,j-1}, w)^\alpha} \right\} X_{i,j,t} \quad .$$

As we have $\mathbf{Pr}\left[X_{i,j,t} = 1\right] = {\log^2 n}/{2eI}$, we can bound the expectation by $\mathbf{E}\left[I_w(\mathcal{R}_t)\right] \leq {\log^2 n}/{2e}$. For fixed t the random variables $X_{i,j,t}$ are *negatively associated* as defined by Dubhashi and Ranjan [6]. So a Chernoff bound is applicable: for all $k_2 \geq 1$ it holds that

$$\mathbf{Pr}\left[I_w(\mathcal{R}_i) \geq k_2 \log^2 n\right] \leq 2^{-k_2 \log^2 n} \leq 2^{-k_2 \log n} = n^{-k_2} \quad .$$

Let T_t denote the schedule length that is used by Algorithm 2 to schedule $\mathcal{R}_t$. We proved in Theorem 8 that for all constants k_1 and k_2 there is a constant k_0 such that

$$\mathbf{Pr}\left[T_t \geq k_0 k_2 \log^2 n \;\middle|\; \max_{w \in V} I_w(\mathcal{R}_t) \leq k_2 \log^2 n \right] \leq \frac{1}{n^{k_1}} \quad .$$

Applying a union bound we get the probability that none of the ${2eI}/{\log^2 n} + D \leq n$ random variables T_t exeeds $k_0 k_2 \log^2 n$. In total, Algorithm 3 generates a schedule length of $\mathcal{O}(I + D \log^2 n)$ whp. ☐

4.2 Finding Optimal Paths (Routing)

To find optimal paths an approach first used by Srinivasan and Teo for wired networks [20], solving an *Integer Linear Program* (ILP) approximately by using relaxation and randomized rounding, can be adapted. Chafekar et al. [4] also use it as a part of their CLM algorithm.

First, let us formalize the problem of finding paths such that $\max\{I, D\}$ is minimal as ILP. We introduce a set of edges $E \subseteq V \times V$ which describes the set of links that may be used. Let furthermore $N_{\mathrm{in}}(v)$ resp. $N_{\mathrm{out}}(v)$ denote the incoming resp. outgoing edges from v.

Minimize w subject to:

$$\forall i \in [m] \qquad \sum_{e \in N_{\mathrm{out}}(s_i)} y(i,e) - \sum_{e \in N_{\mathrm{in}}(s_i)} y(i,e) = 1 \tag{5a}$$

$$\forall i \in [m], v \in V \setminus \{s_i, t_i\} \qquad \sum_{e \in N_{\mathrm{out}}(v)} y(i,e) - \sum_{e \in N_{\mathrm{in}}(v)} y(i,e) = 0 \tag{5b}$$

$$\forall i \in [m] \qquad \sum_{e \in E} y(i,e) \leq w \tag{5c}$$

$$\forall i \in [m], v \in V \qquad \sum_{e'=(u',v')} y(i,e') \min\left\{1, \frac{d(u',v')^\alpha}{d(u',v)^\alpha}\right\} \leq w \tag{5d}$$

$$\forall i \in [m], e \in E \qquad y(i,e) \in \{0,1\} \tag{5e}$$

This ILP is designed to minimize $w = \max\{I, D\}$ as follows. Condition 5d ensures that $I \leq w$ whereas Condition 5c ensures $D \leq w$. By leaving out Condition 5e, this ILP can be relaxed to an LP which then describes a multi-commodity flow problem.

This LP can be solved in polynomial time. Afterwards we can use the LP result to approximate a solution of the ILP, by selecting paths of length at most $2w$ and applying the technique of randomized rounding [18]. In a simple analysis we find out the following. If I^* and D^* are the values such that $\max\{I, D\}$ is minimal – which is the optimal solution for the ILP – we calculate paths such that $I = \mathcal{O}(I^* \log n)$ whp and $D \leq 2D^*$ this way.

4.3 Consequences for the CLM Problem

Let us combine our results to get an approximation algorithm for the CLM problem as stated by Chafekar et al. [4]. Assume there is an optimal choice of paths, powers and a schedule such that the latency is T. Let the measure of interference caused by these paths be denoted by $I^\dagger$ and their dilation by $D^\dagger$. In Section 2 we showed that it holds $I^\dagger = \mathcal{O}(\log \Delta \cdot \log n \cdot T)$. Obviously $D^\dagger = \mathcal{O}(T)$ holds, too.

If I^* and D^* are the values such that $\max\{I, D\}$ is minimal, our path selection algorithm chooses paths such that $I = \mathcal{O}(I^* \log n)$ whp and $D = \mathcal{O}(D^*)$. A schedule by Algorithm 3 using these paths has length $\mathcal{O}(I + D \log^2 n) = \mathcal{O}(I^* \log n + D^* \log^2 n) = \mathcal{O}((I^\dagger + D^\dagger) \log^2 n) = \mathcal{O}(\log \Delta \cdot \log^3 n \cdot T)$ whp. Thus we reached an approximation factor of $\mathcal{O}(\log \Delta \cdot \log^3 n)$ whp. For instances restricted to the Euclidean plane, we even get an approximation factor for $\mathcal{O}(\log \Delta \cdot \log^2 n)$ whp.

References

[1] Friedhelm Meyer auf der Heide and Berthold Vöcking. A packet routing protocol for arbitrary networks. In *Proceedings of the 12th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 291–302, 1995.

[2] Suman Banerjee and Archan Misra. Minimum energy paths for reliable communication in multi-hop wireless networks. In *Proceedings of the 3rd ACM International Symposium Mobile Ad-Hoc Networking and Computing (MOBIHOC)*, pages 146–156, 2002.

[3] Nikhil Bansal and Zhen Liu. Capacity, delay and mobility in wireless ad-hoc networks. *Conference of the IEEE Communications Society (INFOCOM)*, 2:1553–1563, 2003.

[4] Deepti Chafekar, V. S. Anil Kumar, Madhav V. Marathe, Srinivasan Parthasarathy, and Aravind Srinivasan. Cross-layer latency minimization in wireless networks with SINR constraints. In *Proceedings of the 8th ACM International Symposium Mobile Ad-Hoc Networking and Computing (MOBIHOC)*, pages 110–119, 2007.

[5] Deepti Chafekar, V. S. Anil Kumar, Madhav V. Marathe, Srinivasan Parthasarathy, and Aravind Srinivasan. Approximation algorithms for computing capacity of wireless networks with SINR constraints. In *Proceedings of the 27th Conference of the IEEE Communications Society (INFOCOM)*, pages 1166–1174, 2008.

[6] Devdatt P. Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *Random Structures and Algorithms*, 13(2):99–124, 1998.

[7] Alexander Fanghänel, Thomas Keßelheim, Harald Räcke, and Berthold Vöcking. Oblivious interference scheduling. In *Proceedings of the 28thAnnual ACM Symposium on Principles of Distributed Computing (PODC)*, 2009.

[8] Piyush Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46:388–404, 2000.

[9] Shripad Thite Hari Balakrishnan, Christopher L. Barrett, V. S. Anil Kumar, Madhav V. Marathe. The distance-2 matching problem and its relationship to the MAC-layer capacity of ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1069–1079, 2004.

[10] Ulas C. Kozat and Leandros Tassiulas. Network layer support for service discovery in mobile ad hoc networks. In *Proceedings of the 22nd Conference of the IEEE Communications Society (INFOCOM)*, pages 1965–1975, 2003.

[11] Sven O. Krumke, Madhav V. Marathe, and S. S. Ravi. Models and approximation algorithms for channel assignment in radio networks. *Wireless Networks*, 7(6):575–584, 2001.

[12] F. T. Leighton, Bruce M. Maggs, Abhiram G. Ranade, and Satish B. Rao. Randomized routing and sorting on fixed-connection networks. *Journal of Algorithms*, 1994.

[13] F. T. Leighton, Bruce M. Maggs, and Satish B. Rao. Packet routing and job-shop scheduling in O(congestion+dilation) steps. *Combinatorica*, 1994.

[14] Thomas Moscibroda, Yvonne Anne Oswald, and Roger Wattenhofer. How optimal are wireless scheduling protocols? In *Proceedings of the 26th Conference of the IEEE Communications Society (INFOCOM)*, pages 1433–1441, 2007.

[15] Thomas Moscibroda and Roger Wattenhofer. The complexity of connectivity in wireless networks. In *Proceedings of the 25th Conference of the IEEE Communications Society (INFOCOM)*, pages 1–13, 2006.

[16] Thomas Moscibroda, Roger Wattenhofer, and Aaron Zollinger. Topology control meets SINR: The scheduling complexity of arbitrary topologies. In *Proceedings of the 7th ACM International Symposium Mobile Ad-Hoc Networking and Computing (MOBIHOC)*, pages 310–321, 2006.

[17] Prabhakar Raghavan. Probabilistic construction of deterministic algorithms: approximating packing integer programs. *Journal of Computer and System Sciences*, 37(2):130–143, 1988.

[18] Prabhakar Raghavan and Clark D. Tompson. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374, 1987.

[19] Subramanian Ramanathan and Errol L. Lloyd. Scheduling algorithms for multi-hop radio networks. *ACM SIGCOMM Computer Communication Review*, 22(4):211–222, 1992.

[20] Aravind Srinivasan and Chung-Piaw Teo. A constant-factor approximation algorithm for packet routing, and balancing local vs. global criteria. In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 636–643, New York, NY, USA, 1997.

[21] Jeffrey E. Wieselthier, Gam D. Nguyen, and Anthony Ephremides. Energy-efficient broadcast and multicast trees in wireless networks. *Mobile Networks and Applications (MONET)*, 7(6):481–492, 2002.