RWTH AACHEN UNIVERSITY

Computer Science 1
Computer Science Department

Thomas Keßelheim

Packet Scheduling with Interference

Diploma Thesis
January 13, 2009

Thesis advisor: Prof. Dr. Berthold Vöcking
Second advisor: Prof. Dr. Petri Mähönen

Erklärung

Hiermit versichere ich, dass ich die Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, 13. Januar 2009

(Thomas Keßelheim)

Abstract

Routing and scheduling in wireless networks is a popular field of research these days. There are many parallels to wired networks but also new aspects such as interference and power control. A common model for these interference constraints is the so-called physical model that is based on the signal to interference plus noise ratio (SINR): For a successful transmission the received signal has to be β-times higher than the signal of other senders plus ambient noise.
A typical problem is to route and schedule packets via multi-hop networks in the shortest possible time. This task is formalized in the cross-layer latency minimization (CLM) problem. It affects three major fields, namely routing, power control and scheduling. Given N source-destination pairs (s_i, t_i), the aim is to find suitable paths and powers for each packet, and a schedule.
We study approximation algorithms for the CLM problem, which we prove to be NP-hard before. When powers are fixed such that the received signal is the same at each receiver we are able to find a randomized $O(\log^2 n)$ whp-approximation. However, we get to know that it is crucial in some instances to choose the right power levels rather than just leaving them fixed. But these instances have a large length diversity Δ compared to the number of packets and networks nodes. Expressed in terms of this length diversity, we find out that the loss is at most $O(\log \Delta \cdot \log^2 n)$ resulting in a total approximation factor of $O(\log \Delta \cdot \log^4 n)$ whp.

Contents

Chapter 1

Introduction

Communication networks have been subject to research for quite a long time up to now. There have also been many interesting results for wired networks such that they are well understood (see e. g. [Sch98]). These days wireless communication becomes more and more important where constraints are different. For example, in wired networks each transmission may use its own link. In contrast, transmissions are not that separated in wireless networks. Instead every transmission is exposed to some interference by other transmissions or ambient noise.
In this thesis we study the algorithmic aspects of these networks. This means our analyses give worst-case guarantees for arbitrary networks in comparison to what theoretically can be achieved.
In literature, many models for the interference constraints have been proposed. Many approaches are based on graphs. This means apart from the graph modelling the network topology there is a second one that defines the interference constraint. An edge between two senders indicates that not both of them can successfully transmit in a single time slot. In contrast, we study a model closer to reality called *SINR model* or *physical model* as stated by Gupta and Kumar [GK00]. It is based on the *signal to interference plus noise ratio (SINR)*: The received signal has to be β-times larger than the interfering signals plus ambient noise, for some constant $\beta > 1$.
The fundamental difference is that interference from all senders has to be summed up in the SINR model. This means in general all other senders have to be considered to state whether a transmission is successful. In contrast, in the graph-based models there is only a binary condition based on the local environment.
We study the traditional end-to-end routing and scheduling problem: Given N pairs (s_i, t_i), the aim is to route one packet for each pair from s_i to t_i in the shortest possible time. It is considered as static which means all requests arise at a certain starting point. In addition, our algorithms are centralized and have full knowledge over the network. Although these limitations narrow the practical use of our results, we get some interesting insight on what future protocol design has to deal with.
Our results are applicable to many different kinds of wireless networks. We do not need to distinguish if there is infrastructure such as access points or not. Furthermore we find our

results for single-hop networks (e. g. cellular networks) first, which we extend to multi-hop networks (e. g. ad-hoc networks) later on.
The networks we consider have in common that nodes are able to make use of power control. This means there may be chosen a different power level for each transmission. We keep the view of slotted protocols. This means time is divided into slots and any direct transmission from one node to another one needs exactly one time slot.

1.1 Formal Description of the Interference Model

We model the network as a metric space (V, d). The set V represents the nodes of the network. The metric d defines the distance $d(u, v)$ between any two nodes $u, v \in V$. Note that in general we do not require the nodes to be located in the plane. However, in Chapter 5 we will study the impact of restricting to this special case.
The metric enables us to model the decay of signals as follows. If a node u transmits a signal with power level p then node v receives it with power $\frac{p}{d(u,v)^\alpha}$. The constant $\alpha > 0$ is called *path-loss exponent*. A signal can be successfully received if its received strength exceeds the *interference* plus ambient noise at least by a factor β. The interference is defined as the received signal strength caused by all other senders transmitting in the same time slot. The constant $\beta > 1$ is called *antenna gain*.
This SINR constraint can formally be defined as follows.

Definition 1.1.1 (SINR condition)**.** *Consider a single time slot in which k packets are transmitted: u_i transmits to v_i using power level $p_i > 0$ for all $i \in [k] := \{1, \ldots, k\}$. Then v_j receives the packet iff the SINR condition*

$$\mathrm{SINR}(v_j) = \frac{\frac{p_j}{d(u_j,v_j)^\alpha}}{Noise + \sum_{i \neq k} \frac{p_i}{d(u_i,v_j)^\alpha}} \geq \beta \tag{1.1}$$

is satisfied.

By this definition it is implicitly guaranteed that a node can either transmit or receive one packet in one time slot.
Furthermore, by scaling powers up, the noise term gets less and less important. It can even be neglected by slightly changing β. Since we do not aim at minimizing energy, noise will be set to 0 in our theoretic analyses throughout this thesis.

1.2 Algorithmic Problems

In our end-to-end scheduling problem, we are given N source-destination pairs (s_i, t_i). The aim is to route a packet from each s_i to t_i such that the *latency* is minimal, which is the overall number of time slots used. In each of these time slots the SINR constraint has to be satisfied.

The most general problem formulation is the cross-layer latency minimization (CLM) problem as stated by Chafekar *et al.*

Definition 1.2.1 (Cross-layer latency minimization problem [CKM+07])**.** *Given N source-destination pairs (s_i, t_i) the aim is to find*

1. *a* routing path P_i *for each packet i from s_i to t_i defining which hops to take,*
2. *a* power assignment p *defining a power for each hop, and*
3. *a* schedule $\mathcal{S}$ *stating which hop has to be done in which time slot*

such that the schedule length $T(\mathcal{S})$ is minimal.

We distinguish two variants: In the *single-hop case* all transmissions are made directly from s_i to t_i. In the *multi-hop case* intermediate nodes may be used to store and forward the packets. In fact, the single-hop case is a special case of the multi-hop case where additionally paths are fixed to length 1.
We will deal with the problem as follows. First, we will consider the single-hop case where the power assignment is fixed. Afterwards we will extend our results to multi-hop schedules with fixed routing paths and power assignments. As a last step we will discuss how to find optimal routing paths and power assignments.
Let us now introduce some more notation on paths. Having a set $\mathcal{P}$ of N paths $P_1, \ldots, P_N$ we denote the number of involved edges on a path P_i by $\ell(P_i)$. Its nodes are referred to as $P_{i,0}, P_{i,1}, P_{i,2}, \ldots, P_{i,\ell(P_i)}$. Furthermore we set D as the maximum path length by $D := \max_{i \in [N]} \ell(P_i)$.
To simplify notation let us define furthermore the set of indices of receivers as $\operatorname{dom}(\mathcal{P}) = \{ (i,j) \in \mathbb{N} \times \mathbb{N} \mid 1 \leq i \leq N, 1 \leq j \leq \ell(P_i) \}$. In case $D = 1$ we identify $\operatorname{dom}(\mathcal{P})$ with $[N]$.
Using these definitions we can formally define a *power assignment* setting a power for each hop.

Definition 1.2.2 (Power Assignment)**.** *Given a set of paths $\mathcal{P}$, a* power assignment p *is a function*

$$p \colon \operatorname{dom}(\mathcal{P}) \to \mathbb{R}_{>0} \ .$$

There are two generic possibilities to assign powers that are of much relevance. On the one hand, we can make all senders use the same transmission power:

$$p(i,j) = 1 \qquad \text{for all } (i,j) \in \operatorname{dom}(\mathcal{P}) \ .$$

This *uniform power assignment* is the only possible choice when we have identical devices without power control.
On the other hand, we can set the powers such that the received power is the same at all receivers:

$$p(i,j) = d(P_{i,j-1}, P_{i,j})^\alpha \qquad \text{for all } (i,j) \in \operatorname{dom}(\mathcal{P}) \ .$$

This power assignment is of special relevance when considering the problem to minimize energy rather than latency: The power used is proportional to the minimum power needed to deal with the ambient noise, which must not be neglected in this case. This is why it is called *linear power assignment.*
Now a schedule stating which hop will be done in which time slot can be formally defined.

Definition 1.2.3 (Schedule)**.** *Given a set of paths $\mathcal{P}$ and a power assignment p a* schedule *$\mathcal{S}$ of length $T(\mathcal{S})$ is a function $\mathcal{S}\colon \operatorname{dom}(\mathcal{P}) \to [T(\mathcal{S})]$ such that*

1. *for all $(i,j) \in \operatorname{dom}(\mathcal{P})$ with $j > 2$ it holds*

$$\mathcal{S}(i,j) > \mathcal{S}(i,j-1) \tag{1.2}$$

   and

2. *for all $(k,\ell) \in \operatorname{dom}(\mathcal{P})$*

$$\frac{p_{k,\ell}}{d(P_{k,\ell-1}, P_{k,\ell})^\alpha} \geq \beta \sum_{\substack{(i,j)\in\mathcal{S}^{-1}\{(k,\ell)\} \\ (i,j)\neq(k,\ell)}} \frac{p_{i,j}}{d(P_{i,j-1}, P_{i,j})^\alpha} \quad . \tag{1.3}$$

In words, Condition 1.2 means each hop has to take place after the packet has reached the starting point of this hop. Of course, this is of no importance in single-hop schedules.
Condition 1.3 simply states the SINR condition (1.1) has to be satisfied for all hops.
By transmitting each hop in its own time slot, a trivial schedule of length $N \cdot D$ can be constructed.

1.3 Comparison to the Protocol Model

As outlined before, there are a number of graph-based interference models. One typical representative is the *protocol model* introduced by Gupta and Kumar in [GK00] and studied further in [KMPS04].

Definition 1.3.1 (Protocol model [GK00])**.** *In the* protocol model *a transmission from u to v is successful if for all transmissions from u' to v' scheduled in the same time slot it holds that*

$$d(u',v) \geq (1+\varepsilon) \cdot d(u,v)$$

where $\varepsilon > 0$ is a model constant.

In general, an optimal schedule can be much shorter when using the SINR condition instead of the protocol model. This was previously shown by Moscibroda *et al.* [MWW06] in theory and also in simulations. Nevertheless in these examples the SINR schedule is not asymptotically shorter but only within a constant. The following example shows that even asymptotically large differences in schedule lengths can occur.

In our instance we have N pairs of senders and receivers (s_i, t_i) which are placed on a line, i. e. $V \subseteq \mathbb{R}$, $d(u,v) = |v-u|$. To simplify notation we set the constant $a = 2^{\alpha+1}\beta + 1$. The i-th sender s_i is located at $-a^{\frac{2i}{\alpha}}$, its receiver t_i is located at $a^{\frac{2i}{\alpha}}$.
First observe that for all $i \in [N]$ receivers t_j with $j < i$ are located between s_i and t_i. In the protocol model this means no two packets can be transmitted together in a single time slot.
In contrast, in the SINR model only a single time slot is needed for all transmissions when the packet from s_i to t_i is transmitted with power a^i.
To show this we fix a receiver t_j, $j \in [N]$. We can now bound the interference caused by senders that are located between s_j and t_j:

$$\sum_{i<j} \frac{p_i}{d(s_i, t_j)^\alpha} \leq \sum_{i<j} \frac{a^i}{a^{2j}} \leq \frac{1}{a-1}\frac{a^j}{a^{2j}} \quad .$$

The interference caused by the outside senders can be bounded similarly:

$$\sum_{i>j} \frac{p_i}{d(s_i, t_j)^\alpha} \leq \sum_{i>j} \frac{a^i}{a^{2i}} \leq \frac{\frac{1}{a^{j+1}}}{1-\frac{1}{a}} = \frac{1}{a-1}\frac{a^j}{a^{2j}} \quad .$$

Thus the total interference is at most

$$2\frac{1}{a-1}\frac{a^j}{a^{2j}} = \frac{1}{2^\alpha \beta}\frac{a^j}{a^{2j}} = \frac{1}{\beta}\frac{p_j}{d(s_j, t_j)^\alpha} \quad .$$

This means the SINR constraint is satisfied, all transmissions can be done in the same time slot.
For this example it is crucial to choose the right power assignment in the SINR model. Similar examples demonstrating the impact of this choice can be found in Section 4.1. Another important factor is the exponential growth of this instances. We will show in Section 5.2 that when distances are not too different the optimal lengths only differ by a constant factor.

1.4 Related Work

Networks under SINR constraints have been studied widely up to now from several points of view. The first study of SINR scheduling problems on nodes that are arbitrarily – and not randomly – distributed has been presented by Moscibroda and Wattenhofer [MW06]. In contrast to our problem, they aim to schedule all links on a spanning tree. The idea is to have a protocol such that packets between any two nodes are transmitted via a virtual backbone.
This result has been extended by Moscibroda *et al.* [MWZ06] to arbitrary demands, which is in fact the single-hop case of our problem. Their result is an $O(\log^2 N \cdot I_{in})$ algorithm, where I_{in} is a measure of interference in the instance. Unfortunately, I_{in} is no lower bound for the optimal schedule length.

A further extension [MOW07] introduces a new measure of interference χ_ρ called disturbance where $\rho > 0$ is some parameter. The algorithm described achieves a schedule length of $O(\chi_\rho \rho^2 \log N \cdot (\log N + \rho))$. There is still no comparison to the optimal schedule length. This is why both results lack of an approximation factor.
Goussevskaia *et al.* [GMW08] also examine a related problem. In the local broadcasting problem any node in the network intends to transmit a packet to all nodes within its so-called *local broadcasting range*. They describe two distributed algorithms which have a polylogarithmic approximation factor.
Chafekar *et al.* study similar problems. In [CKM+08] they deal with maximizing the throughput in an SINR network. In [CKM+07] they give an approximation algorithm for the CLM problem. It is crucial for their analysis that the instance is located in the plane. This allows to use graph coloring in a similar way to the approaches used in the protocol model. Our algorithms instead work in general metrics while the approximation factor is still better.
Apart from this, Goussevskaia *et al.* [GOW07] prove the scheduling problem in the Euclidean plane with fixed power assignment to be NP-hard. The idea is quite a simple reduction of the Partition problem. However, this result still does not prove the scheduling problem when powers are not fixed to be NP-hard. We will give this result for general metrics.

1.5 Outline of this Thesis

As outlined before, we will prove the NP-hardness of the CLM problem. This will be done in Chapter 2. Afterwards, in Chapter 3, we will find an approximation algorithm for the CLM problem where powers are fixed to the linear power assignment. The effects of this restriction are analyzed in Chapter 4. This allows us to give an approximation factor for our algorithm for the CLM problem. As a last point, we will study a different approach that can be applied when restricting to the Euclidean plane in Chapter 5.

Chapter 2

NP-hardness

As a first step, let us prove the defined problem is NP-hard. Goussevskaia *et al.* [GOW07] already proved the scheduling problem with fixed paths and a fixed power assignment to be NP-hard. Thus we will only have to deal with the CLM problem where powers are subject to optimization.
First, we need to prove two simple bounds.

Lemma 2.0.1. *For $x, y \geq 0$ with $x + y \leq \frac{1}{2}$ it holds that*

$$2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}(x+y) \leq \left[\left(\frac{1}{1-x}\right)^{\alpha} + \left(\frac{1}{1-y}\right)^{\alpha}\right]^{\frac{1}{\alpha}} \leq 2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}(x+y) + \rho(x+y)^2$$

where ρ is a constant defined as follows

$$\rho = (2^{\alpha}+1)^{\frac{1}{\alpha}}\, 2^{\alpha+2}\left(2^{\alpha+1}+\alpha+1\right) \quad .$$

Proof. Let us fix the sum s of x and y, i. e. $0 < s \leq \frac{1}{2}$ and $s = x + y$, and analyze the functions $f, g \colon [0, s] \to \mathbb{R}$ defined by

$$g(x) = \left(\frac{1}{1-x}\right)^{\alpha} + \left(\frac{1}{1-(s-x)}\right)^{\alpha} \qquad f(x) = g(x)^{\frac{1}{\alpha}} \quad .$$

The extremal points of f and g are obviously the same.
Having a look at the derivative of g

$$\frac{\mathrm{d}g}{\mathrm{d}x}(x) = \alpha\left(\frac{1}{1-x}\right)^{\alpha+1} - \alpha\left(\frac{1}{1-(s-x)}\right)^{\alpha+1}$$

we remark that g must have its extremal points at the critical point $x = \frac{1}{2}s$ resp. at the boundary points $x = 0$ and $x = s$. Thus it suffices to have a closer look at the functions at these points.
For the critical point $x = \frac{1}{2}s$ we have:

$$f\left(\frac{1}{2}s\right) = 2^{\frac{1}{\alpha}}\frac{1}{1-\frac{1}{2}s} = 2^{\frac{1}{\alpha}}\sum_{k=0}^{\infty}\left(\frac{1}{2}s\right)^{k} \quad .$$

So:

$$2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}s \leq f\left(\frac{1}{2}s\right) \leq 2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}s + s^2 \ . \tag{2.1}$$

We now consider the boundary points $x = 0$ and $x = s$:
Let us define another function $h\colon \left[0, \frac{1}{2}\right] \to \mathbb{R}$ by

$$h(s) = \left[\left(\frac{1}{1-s}\right)^{\alpha} + 1\right]^{\frac{1}{\alpha}} \ .$$

The first and the second derivative of h are:

$$\frac{\mathrm{d}h}{\mathrm{d}s}(s) = \frac{1}{\alpha}\left[\left(\frac{1}{1-s}\right)^{\alpha} + 1\right]^{\frac{1}{\alpha}-1} \alpha \left(\frac{1}{1-s}\right)^{\alpha+1}$$

$$\frac{\mathrm{d}^2h}{\mathrm{d}s^2}(s) = \left[\left(\frac{1}{1-s}\right)^{\alpha} + 1\right]^{\frac{1}{\alpha}-2} \left(\frac{1}{1-s}\right)^{\alpha+2} \left[2\left(\frac{1}{1-s}\right)^{\alpha} + \alpha + 1\right] \ .$$

By Taylor's theorem it holds that

$$h(s) = h(0) + \frac{\mathrm{d}h}{\mathrm{d}s}(0) \cdot s + R_1(s) = 2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}s + R_1(s) \ .$$

For the remainder term $R_1(s)$ the Lagrange form states there is a ξ, $0 < \xi < s$, such that

$$R_1(s) = \frac{1}{2}\frac{\mathrm{d}^2h}{\mathrm{d}s^2}(\xi) \ \cdot \ s^2 \ .$$

Since $s \leq \frac{1}{2}$ we can conclude

$$0 \leq R_1(s) \leq \rho s^2 \ .$$

In total, this means

$$2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}s \leq h(s) \leq 2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha}-1}s + \rho s^2 \ . \tag{2.2}$$

From Equations 2.1 and 2.2 we can conclude the claim holds. □

We will present a reduction of 3-Partition, which can be regarded as a variant of the Bin Packing Problem in which the weights are much more restricted.

Definition 2.0.2 (3-Partition [GJ79])**.** *Given $B \in \mathbb{N}$, $m \in \mathbb{N}$, $a_1, \ldots, a_{3m} \in \mathbb{N}$, $\frac{B}{4} < a_i < \frac{B}{2}$, $\sum_{i=1}^{3m} a_i = m \cdot B$*
Question: Is there $S_1, \ldots, S_m \subseteq [3m]$ such that $\dot{\bigcup}_j S_j = [3m]$ and $\sum_{i \in S_j} a_i = B$ for all $j \in [m]$?

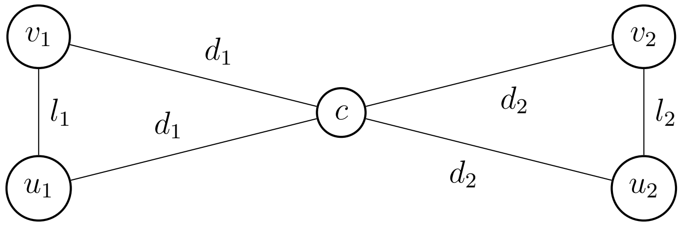

Figure 2.1: A visualization of the scheduling instance network.

For a feasible solution it obviously suffices to have $\sum_{i \in S_j} a_i \leq B$ for all $j \in [m]$ since $\sum_{i=1}^{3m} a_i = m \cdot B$.
To show the CLM problem is strongly NP-hard we reduce it to 3-Partition, which is known to be strongly NP-hard [GJ79].

Theorem 2.0.3. *The single-hop and the multi-hop case of the CLM problem are strongly NP-hard.*

Proof. We will define a reduction of a 3-Partition instance as defined in Definition 2.0.2 to a scheduling instance in which all numbers are bounded polynomially in m.
First define two shortcuts depending on B:

$$A := 2 \left(2^{3-\frac{1}{\alpha}} \rho + \frac{1}{4} \right) B^2$$

and

$$\chi := 2^{\frac{1}{\alpha}-1} + 2^{\frac{1}{\alpha}-3} \frac{B}{A} + \rho \left(\frac{B}{A} \right)^2 = \frac{4A^2 + BA + 2^{3-\frac{1}{\alpha}} \rho B^2}{2^{3-\frac{1}{\alpha}} A^2} \quad .$$

The scheduling instance network is visualized in Figure 2.1. It consists of $2 \cdot 3m + 1$ nodes called $c, u_1, \ldots, u_{3m}, v_1, \ldots, v_{3m}$. There is an edge between u_i and v_i, between u_i and c, and between v_i and c for all $i \in [3m]$. The distances are defined as

$$d(u_i, c) = d(v_i, c) = A - a_i =: d_i \qquad \text{and} \qquad d(u_i, v_i) = \frac{1}{\chi \beta^{\frac{1}{\alpha}}} \left(A - \frac{a_i}{4} \right) =: l_i \quad .$$

There is a communication request between the nodes u_i and v_i for all $i \in [3m]$.
We now claim: There is a schedule of length $m \Leftrightarrow$ There is a solution for 3-Partition.
Note that it does not make sense to do multi-hop scheduling. This is why we only have to consider the single-hop case.

$\Rightarrow$ We have to show that if there is a schedule of length m there is a solution for 3-Partition. Obviously, in any schedule of length m, in each step exactly three transmission requests are operated. Consider such a single step. For the ease of notation let, w. l. o. g., the transmissions take place between u_i and v_i for $i \in \{1, 2, 3\}$. Let furthermore be $p_1 \leq p_2 \leq p_3$.

By SINR constraint it holds

$$\beta \left(\frac{p_2}{(d_1 + d_2)^\alpha} + \frac{p_3}{(d_1 + d_3)^\alpha} \right) \leq \frac{p_1}{l_1^\alpha} \quad .$$

Thus we can conclude

$$\begin{aligned}
&\beta \left(\frac{1}{(d_1 + d_2)^\alpha} + \frac{1}{(d_1 + d_3)^\alpha} \right) \leq \frac{1}{l_1^\alpha} \\
\Rightarrow &\left(\frac{1}{2A - a_1 - a_2} \right)^\alpha + \left(\frac{1}{2A - a_1 - a_3} \right)^\alpha \leq \left(\frac{1}{A - \frac{a_1}{4}} \right)^\alpha \chi^\alpha \\
\Rightarrow &\left[\left(\frac{1}{1 - \frac{a_1}{2A} - \frac{a_2}{2A}} \right)^\alpha + \left(\frac{1}{1 - \frac{a_1}{2A} - \frac{a_3}{2A}} \right)^\alpha \right]^{\frac{1}{\alpha}} \left(A - \frac{a_1}{4} \right) \leq 2A\chi \quad .
\end{aligned}$$

By using Lemma 2.0.1 we have

$$\begin{aligned}
&\left[2^{\frac{1}{\alpha}} + 2^{\frac{1}{\alpha} - 1} \left(\frac{a_1}{A} + \frac{a_2}{2A} + \frac{a_3}{2A} \right) \right] \left(A - \frac{a_1}{4} \right) \leq 2A\chi \\
\Rightarrow &(4A + 2a_1 + a_2 + a_3) \left(A - \frac{a_1}{4} \right) \leq 2^{3 - \frac{1}{\alpha}} A^2 \chi \\
\Rightarrow &4A^2 + (a_1 + a_2 + a_3)A - \frac{B^2}{4} \leq 2^{3 - \frac{1}{\alpha}} A^2 \chi = 4A^2 + BA + 2^{3 - \frac{1}{\alpha}} \rho B^2 \\
\Rightarrow &(a_1 + a_2 + a_3) - \underbrace{\left(2^{3 - \frac{1}{\alpha}} \rho + \frac{1}{4} \right) \frac{B^2}{A}}_{= \frac{1}{2}} \leq B \quad .
\end{aligned}$$

Since a_1, a_2, a_3 and B are integer, it has to be $a_1 + a_2 + a_3 \leq B$. This means the schedule can be transformed to a valid solution of 3-Partition.

$\Leftarrow$ Let there be a solution $S_1, \ldots, S_m$ of 3-Partition. We claim that each set S_i can be regarded as a valid schedule step using the uniform power assignment. For the ease of notation, we we write $S_i = \{1, 2, 3\}$ as well.

We have

$$\begin{aligned}
&\frac{1}{(d_1 + d_2)^\alpha} + \frac{1}{(d_1 + d_3)^\alpha} \\
= &\left(\frac{1}{2A - a_1 - a_2} \right)^\alpha + \left(\frac{1}{2A - a_1 - a_3} \right)^\alpha \\
= &\left(\frac{1}{2A} \right)^\alpha \left[\left(\frac{1}{1 - \frac{a_1}{2A} - \frac{a_2}{2A}} \right)^\alpha + \left(\frac{1}{1 - \frac{a_1}{2A} - \frac{a_3}{2A}} \right)^\alpha \right] \quad .
\end{aligned}$$

Lemma 2.0.1 states this is at most

$$\left(\frac{1}{2A}\right)^{\alpha}\left[2^{\frac{1}{\alpha}}+2^{\frac{1}{\alpha}-1}\left(\frac{a_1}{2A}+\frac{a_2}{2A}+\frac{a_3}{2A}+\frac{a_1}{2A}\right)+\rho\left(\frac{a_1}{2A}+\frac{a_2}{2A}+\frac{a_1}{2A}+\frac{a_3}{2A}\right)^2\right]^{\alpha}\ .$$

Since $a_1+a_2+a_3\leq B$ this is at most

$$\left(\frac{1}{2A}\right)^{\alpha}\left[2^{\frac{1}{\alpha}}+2^{\frac{1}{\alpha}-1}\left(\frac{B}{2A}+\frac{a_1}{2A}\right)+\rho\left(\frac{B}{A}\right)^2\right]^{\alpha}\ .$$

Thus we can bound the interference divided by the signal strength by

$$\begin{aligned}
&\left[\frac{1}{(d_1+d_2)^{\alpha}}+\frac{1}{(d_1+d_3)^{\alpha}}\right]\Big/\frac{1}{l_1^{\alpha}}\\
\leq&\left(\frac{1}{2A}\right)^{\alpha}\left[2^{\frac{1}{\alpha}}+2^{\frac{1}{\alpha}-1}\left(\frac{B}{2A}+\frac{a_1}{2A}\right)+\rho\left(\frac{B}{A}\right)^2\right]^{\alpha}\left(A-\frac{a_1}{4}\right)^{\alpha}\frac{1}{\chi^{\alpha}\beta}\\
=&\frac{1}{2^{\alpha}}\left(2^{\frac{1}{\alpha}}+2^{\frac{1}{\alpha}-1}\left(\frac{B}{2A}+\frac{a_1}{2A}\right)+\rho\left(\frac{B}{A}\right)^2\right.\\
&\qquad\left.-2^{\frac{1}{\alpha}}\frac{a_1}{4A}-2^{\frac{1}{\alpha}-1}\left(\frac{Ba_1}{8A^2}+\frac{a_1}{8A^2}\right)-\frac{\rho B^2a_1}{4A^3}\right)^{\alpha}\frac{1}{\chi^{\alpha}\beta}\\
\leq&\left(2^{\frac{1}{\alpha}-1}+2^{\frac{1}{\alpha}-3}\frac{B}{A}+\rho\left(\frac{B}{A}\right)^2\right)^{\alpha}\frac{1}{\chi^{\alpha}\beta}\\
=&\frac{1}{\beta}\ .
\end{aligned}$$

So the SINR condition is met and the solution for 3-Partition can be regarded as a valid schedule consisting of m steps.

All in all, we can conclude that the CLM problem is NP-hard. It is even strongly NP-hard since 3-Partition is and all numbers are bounded polynomially in m which is a lower bound of the input length. □

We have seen that under the assumption P $\neq$ NP there is no polynomial time algorithm. This is why we will design and analyze approximation algorithms in the next chapters.

Chapter 3

Scheduling in General Metrics

Having proven NP-hardness, we will analyze a randomized approximation algorithm for the CLM problem in this chapter. We will not try to find optimal powers but leave them fixed to the linear power assignment and analyze the effects of this choice in the next chapter.
Under these assumptions we will find a way to estimate the optimal schedule length. Afterwards it is a simple task to find an algorithm for single-hop scheduling. Finally, we will extend this algorithm to multi-hop scheduling and add a path selection algorithm.

3.1 Estimating the Optimal Schedule Length

Given a set of paths and a power assignment, an interesting question is to get upper and lower bounds indicating how long an optimal schedule has to be. In wired networks there is quite a simple, yet surprising answer that was first given by Leighton *et al.* [LMR94]: This optimal schedule has length $\Theta(C + D)$ where C is the *congestion*, which is the maximum number of paths using a certain edge, and D is the *dilation*, which denotes the maximum length of a path.
In wireless networks under SINR constraints it is unfortunately not that easy. Nevertheless, we will find an analogon to the congestion C measuring the interference. It is a lower bound on the optimal schedule length in the linear power assignment and we will use it to give approximation ratios.

Definition 3.1.1 (Measure of Interference I)**.** *For* $(k, l) \in \mathrm{dom}(\mathcal{P})$ *define*

$$I_{k,l} = \sum_{(i,j) \in \mathrm{dom}(\mathcal{P})} \min \left\{ 1, \frac{p_{i,j}}{p_{k,l}} \frac{d(P_{k,l-1}, P_{k,l})^\alpha}{d(P_{i,j-1}, P_{k,l})^\alpha} \right\} \quad .$$

Using this we define the measure of interference:

$$I = \max_{(k,l) \in \mathrm{dom}(\mathcal{P})} I_{k,l} \quad .$$

This measure of interference allows us to make quite a good estimation for the optimal schedule length in the linear power assignment.

Theorem 3.1.2. *Every schedule has length at least* $\max\left\{\frac{I}{\kappa}\frac{q_{\min}}{q_{\max}}, D\right\} = \Omega\left(I\frac{q_{\min}}{q_{\max}} + D\right)$, *where* κ *is a constant and*

$$q_{\max} = \max_{(i,j)\in\mathrm{dom}(\mathcal{P})} \frac{p_{i,j}}{d(P_{i,j-1}, P_{i,j})^\alpha} \qquad \textit{and} \qquad q_{\min} = \min_{(i,j)\in\mathrm{dom}(\mathcal{P})} \frac{p_{i,j}}{d(P_{i,j-1}, P_{i,j})^\alpha} \ .$$

Proof. Any schedule has length at least D since there is a path consisting of D hops which all need a single time slot. It remains to show any schedule has length at least $\frac{I}{\kappa}\frac{q_{\min}}{q_{\max}}$.
For $w \in V$ we define

$$J_w = \sum_{(i,j)\in\mathrm{dom}(\mathcal{P})} \min\left\{\frac{q_{\min}}{q_{\max}}1, \frac{1}{q_{\max}}\frac{p_{i,j}}{d(P_{i,j-1}, w)^\alpha}\right\} \ .$$

It can be easily seen that $\frac{q_{\max}}{q_{\min}} J_w \geq I_{k,l}$ for $w = P_{k,l}$. Thus it obviously suffices to show that $T(\mathcal{S}) \geq \frac{J_w}{\kappa}$ for all $w \in V$. To prove this it suffices to show that in any step J_w decreases by κ at most.
Let us now consider such a single step where transmissions are made from u_i to v_i for $i \in [k]$. Let furthermore be $w \in V$. We have to show:

$$\sum_{i\in[k]} \min\left\{\frac{q_{\min}}{q_{\max}}1, \frac{1}{q_{\max}}\frac{p_i}{d(u_i, w)^\alpha}\right\} \leq \kappa \ .$$

We define $j \in \arg\min_{i\in[k]} d(v_i, w)$, i. e. v_j is the closest (active) receiver from w. This might also be w itself.
We define a set U of indices of "near" senders from w by $U = \{i \in [k] \mid d(u_i, w) \leq \frac{1}{2} d(v_j, w)\}$ (cf. Figure 3.1).
Using the triangle inequality we can conclude for all $i \in U$:

$$d(u_i, v_j) \leq d(u_i, w) + d(w, v_j) \leq \frac{3}{2} d(v_j, w) \ . \tag{3.1}$$

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

$$d(v_j, v) \leq 2d(u_i, v_i) \ . \tag{3.2}$$

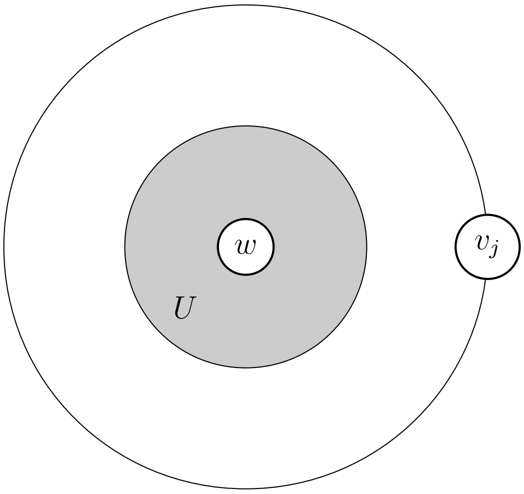

Figure 3.1: The definition of v_j and U. There is no receiver inside the outer circle.

Putting together 3.1 and 3.2 we get

$$d(u_i, v_j) \leq 3d(u_i, v_i) \ .$$

Thus it holds

$$\frac{q_{\min}}{q_{\max}} |U \backslash \{j\}| \leq \frac{1}{q_{\max}} \sum_{\substack{i \in U \\ i \neq j}} \frac{p_i}{d(u_i, v_i)^\alpha} \leq \frac{1}{q_{\max}} \sum_{\substack{i \in U \\ i \neq j}} \frac{p_i}{\frac{1}{3^\alpha} d(u_i, v_j)^\alpha} \leq \frac{3^\alpha}{q_{\max} \beta} \frac{p_j}{d(u_j, v_j)^\alpha} \leq \frac{3^\alpha}{\beta} \ .$$

For all $i \in [k] \backslash U$ 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 $i \in [k] \backslash U$:

$$\frac{1}{q_{\max}} \sum_{\substack{i \in [k] \backslash U \\ i \neq j}} \frac{p_i}{d(u_i, w)^\alpha} \leq \frac{1}{q_{\max}} \sum_{\substack{i \in [k] \backslash U \\ i \neq j}} \frac{p_i}{\frac{1}{3^\alpha} d(u_i, v_j)^\alpha} \leq \frac{3^\alpha}{\beta} \ .$$

Summing up all $i \in [k]$ gives

$$\begin{aligned} \sum_{i \in [k]} \min \left\{ \frac{q_{\min}}{q_{\max}}, \frac{1}{q_{\max}} \frac{p_i}{d(u_i, w)^\alpha} \right\} &\leq \frac{q_{\min}}{q_{\max}} |U \backslash \{j\}| + \sum_{\substack{i \in [k] \backslash U \\ i \neq j}} \frac{1}{q_{\max}} \frac{p_i}{d(u_i, w)^\alpha} + 1 \\ &\leq \frac{3^\alpha}{\beta} + \frac{3^\alpha}{\beta} + 1 \end{aligned}$$

$$\leq \frac{2 \cdot 3^\alpha}{\beta} + 1 \ .$$

Thus we have shown the claim for $\kappa = \frac{2 \cdot 3^\alpha}{\beta} + 1$. □

Since having $q_{\max} = q_{\min}$ in the linear power assignment, we have proven a lower bound of $\Omega(I + D)$ for the linear power assignment. In the remaining part of this chapter we will complete this result by an algorithm generating a schedule of length $O(I + D \log^2 \hat{n})$.

3.2 A Randomized Algorithm for Single-hop Scheduling

Using the measure of interference defined in the previous section, we can construct quite simple randomized algorithms for the single-hop case and also for the multi-hop case. Please note that these algorithms work with any power assignment. However, their performance guarantees are always given in terms of I, which is a lower bound on the optimal schedule length for the linear power assignment only.

Algorithm 1 A simple single-hop algorithm

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

Algorithm 1 is a simple waiting-for-success algorithm where all packets use the same probability of transmission at any time.

Lemma 3.2.1. *Algorithm 1 generates to a schedule of length at most $O(I \log N)$ whp.*

Proof. First consider a fixed packet k in a single step of the algorithm. Let X_i, $i \in [N]$, be the 0/1 random variable that packet i is chosen to be transmitted in this step.
Let Z be the interference at receiver t_k divided by its signal strength. Since we have $\beta > 1$ the transmission cannot be successful anymore when $Z \geq 1$. Thus we may cut off the influence by a single sender at 1.

$$Z = \sum_{\substack{i=1 \\ i \neq k}}^{N} \min \left\{ 1, \frac{d(s_k, t_k)^\alpha}{p_k} \frac{p_i}{d(s_i, t_k)^\alpha} \right\} X_i$$

Its expectation value is:

$$\mathrm{E}\, Z = \sum_{\substack{i=1 \\ i \neq k}}^{N} \min \left\{ 1, \frac{d(s_k, t_k)^\alpha}{p_k} \frac{p_i}{d(s_i, t_k)^\alpha} \right\} \mathrm{E}\, X_i$$

$$= \left(\sum_{\substack{i=1 \\ i \neq k}}^{N} \min \left\{ 1, \frac{d(s_k, t_k)^\alpha}{p_k} \frac{p_i}{d(s_i, t_k)^\alpha} \right\} \right) \frac{1}{2\beta I} \leq \frac{1}{2\beta}$$

So by Markov's inequality the probability that this packet cannot be transmitted successfully given $X_k = 1$ is

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

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

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

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 leads to an overall bound that one of N packets is not successfully transmitted in $(k_0 + 1)4\beta I \ln N = O(I \log N)$ independent repeats:

$$\begin{aligned} &\Pr[T(\mathcal{S}) > (k_0 + 1)4\beta I \ln N] \\ &\leq \sum_{i=1}^{N} \Pr[\text{Packet } i \text{ needs more that } (k_0 + 1)4\beta I \ln N \text{ attempts}] \\ &\leq N \cdot N^{-(k_0+1)} = N^{-k_0} \end{aligned}$$

This means all senders are successful within $O(I \log N)$ steps whp. □

For the analysis of a more sophisticated algorithm, we need a Chernoff bound in which random variables do not have to be independent.

Lemma 3.2.2. *Let $X_1, \ldots, X_n$ be 0/1 random variables for which hold that there is a $p \in [0, 1]$ such that for all $k \in [n]$ and all $\mathcal{I} \subseteq [k - 1]$*

$$\Pr\left[X_k = 1 \;\middle|\; \bigwedge_{i \in \mathcal{I}} X_i = 1, \bigwedge_{i \in [k-1] \setminus \mathcal{I}} X_i = 0 \right] \leq p \ . \tag{3.3}$$

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

$$\Pr\left[\sum w_i X_i \geq (1 + \delta)\mu \right] \leq \left(\frac{e^\delta}{(1 + \delta)^{(1+\delta)}} \right)^\mu$$

holds.

Setting $1+\delta = \gamma \geq 2e$ this implies

$$\Pr\left[\sum w_i X_i \geq \gamma\mu\right] \leq 2^{-\gamma\mu} \ .$$

Proof. To show this Chernoff bound, a standard proof [Rag88] can be adapted. The only step that needs to be changed is that still

$$\mathrm{E}[e^{tX}] \leq \prod_{i=1}^{n} \left(pe^{tw_i} + 1 - p\right)$$

although random variables are no more independent. No other step makes use of the independence.
Let us first prove that for all $k \in \{0, \dots n-1\}$ and for all $\mathcal{J} \subseteq [k]$

$$\sum_{\mathcal{I} \subseteq [n]\setminus[k]} \left(\prod_{i\in\mathcal{I}} e^{tw_i}\right) \cdot \Pr\left[\bigwedge_{i\in\mathcal{I}} X_i = 1, \bigwedge_{i\in([n]\setminus[k])\setminus\mathcal{I}} X_i = 0 \,\middle|\, \bigwedge_{i\in\mathcal{J}} X_i = 1, \bigwedge_{i\in[k]\setminus\mathcal{J}} X_i = 0\right]$$
$$\leq \sum_{\mathcal{I} \subseteq [n]\setminus[k]} \left(\prod_{i\in\mathcal{I}} pe^{tw_i}\right) \left(\prod_{i\in([n]\setminus[k])\setminus\mathcal{I}} (1-p)\right) \ . \quad (3.4)$$

by reverse induction.
For $k = n-1$ this is a direct conclusion from Condition 3.3, which is satisfied by assumption. Now let us assume Equation 3.4 has already been shown for $k+1$ and we have to prove it for k.
We split up the sum on the left side depending on whether $k+1 \in \mathcal{I}$ or not:

$$e^{tw_{k+1}} \sum_{\mathcal{I} \subseteq [n]\setminus[k+1]} \left(\prod_{i\in\mathcal{I}} e^{tw_i}\right)$$
$$\cdot \Pr\left[X_{k+1} = 1, \bigwedge_{i\in\mathcal{I}} X_i = 1, \bigwedge_{i\in([n]\setminus[k+1])\setminus\mathcal{I}} X_i = 0 \,\middle|\, \bigwedge_{i\in\mathcal{J}} X_i = 1, \bigwedge_{i\in[k]\setminus\mathcal{J}} X_i = 0\right]$$
$$+ \sum_{\mathcal{I} \subseteq [n]\setminus[k+1]} \left(\prod_{i\in\mathcal{I}} e^{tw_i}\right)$$
$$\cdot \Pr\left[X_{k+1} = 0, \bigwedge_{i\in\mathcal{I}} X_i = 1, \bigwedge_{i\in([n]\setminus[k+1])\setminus\mathcal{I}} X_i = 0 \,\middle|\, \bigwedge_{i\in\mathcal{J}} X_i = 1, \bigwedge_{i\in[k]\setminus\mathcal{J}} X_i = 0\right] \ .$$

Using the definition of conditional probability is the same as

$$e^{tw_{k+1}} \Pr[\, X_{k+1} = 1\,] \sum_{\mathcal{I} \subseteq [n]\setminus[k+1]} \left(\prod_{i\in\mathcal{I}} e^{tw_i}\right)$$

$$\cdot \Pr\left[\bigwedge_{i \in \mathcal{I}} X_i = 1, \bigwedge_{i \in ([n] \setminus [k+1]) \setminus \mathcal{I}} X_i = 0 \;\middle|\; X_{k+1} = 1, \bigwedge_{i \in \mathcal{J}} X_i = 1, \bigwedge_{i \in [k] \setminus \mathcal{J}} X_i = 0 \right]$$

$$+ \Pr[\, X_{k+1} = 0 \,] \sum_{\mathcal{I} \subseteq [n] \setminus [k+1]} \left(\prod_{i \in \mathcal{I}} e^{t w_i} \right)$$

$$\cdot \Pr\left[\bigwedge_{i \in \mathcal{I}} X_i = 1, \bigwedge_{i \in ([n] \setminus [k+1]) \setminus \mathcal{I}} X_i = 0 \;\middle|\; X_{k+1} = 0, \bigwedge_{i \in \mathcal{J}} X_i = 1, \bigwedge_{i \in [k] \setminus \mathcal{J}} X_i = 0 \right] .$$

Applying the induction hypothesis, we can bound this by

$$\left(e^{t w_{k+1}} \Pr[\, X_{k+1} = 1 \,] + \Pr[\, X_{k+1} = 0 \,] \right) \sum_{\mathcal{I} \subseteq [n] \setminus [k+1]} \left(\prod_{i \in \mathcal{I}} p e^{t w_i} \right) \left(\prod_{i \in ([n] \setminus [k+1]) \setminus \mathcal{I}} (1-p) \right) .$$

As we have $e^{t w_{k+1}} \Pr[\, X_{k+1} = 1 \,] + \Pr[\, X_{k+1} = 0 \,] \leq e^{t w_{k+1}} p + (1-p)$, this is at most:

$$\left(e^{t w_{k+1}} p + (1-p) \right) \sum_{\mathcal{I} \subseteq [n] \setminus [k+1]} \left(\prod_{i \in \mathcal{I}} p e^{t w_i} \right) \left(\prod_{i \in ([n] \setminus [k+1]) \setminus \mathcal{I}} (1-p) \right)$$

$$= \sum_{\mathcal{I} \subseteq [n] \setminus [k]} \left(\prod_{i \in \mathcal{I}} p e^{t w_i} \right) \left(\prod_{i \in ([n] \setminus [k]) \setminus \mathcal{I}} (1-p) \right) .$$

Thus we have successfully proven Equation 3.4 for all $k \in \{0, \ldots, n-1\}$.
Obviously, the expectation of e^{tX} can be calculated as follows:

$$\mathrm{E}[e^{tX}] = \sum_{\mathcal{I} \subseteq [n]} \left(\prod_{i \in \mathcal{I}} e^{t w_i} \right) \cdot \Pr\left[\bigwedge_{i \in \mathcal{I}} X_i = 1, \bigwedge_{i \in [n] \setminus \mathcal{I}} X_i = 0 \right] .$$

Using the above result with $k = 0$ we can bound it by

$$\sum_{\mathcal{I} \subseteq [n]} \left(\prod_{i \in \mathcal{I}} p e^{t w_i} \right) \left(\prod_{i \in [n] \setminus \mathcal{I}} (1-p) \right) .$$

It can easily be seen that this is exactly

$$\prod_{i=1}^{n} \left(p e^{t w_i} + 1 - p \right) .$$

Thus we have filled the gap. □

Algorithm 2 An $O(I + \log^2 N)$ whp algorithm

```
while I^curr ≥ log N do
  J := I^curr
  while I^curr ≥ J/2 do
    if packet i has not been successfully transmitted then
      assign a delay 1 ≤ δ_i ≤ 16eβJ i. u. r.
      try transmission after waiting the delay
    end if
  end while
end while
execute algorithm 1
```

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 how many transmissions have already been successful. Let I^{curr} denote the measure of interference that is only caused by the remaining transmissions to be made. The delay maximum is reduced to its half when $I^{\text{curr}} = \frac{I}{2}$.
We first observe that the inner *while* loop, which reduces I to $\frac{I}{2}$ generates $O(I)$ steps whp.

Lemma 3.2.3. *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 $1 - N^{-k_0}$.*

Proof. We first consider a single iteration of this loop. All senders are taking part as if they had not yet been successful.
We observe if the senders of a set S are transmitting and there is a collision for packet i we have

$$\sum_{\substack{j \in S \\ j < i}} \min \left\{ 1, \frac{d(s_i,t_i)^\alpha}{p_i} \frac{p_j}{d(s_j,t_i)^\alpha} \right\} > \frac{1}{2\beta} \quad \text{or} \quad \sum_{\substack{j \in S \\ j > i}} \min \left\{ 1, \frac{d(s_i,t_i)^\alpha}{p_i} \frac{p_j}{d(s_j,t_i)^\alpha} \right\} > \frac{1}{2\beta}.$$

In the first case let be $Y_i^< = 1$, in the second one $Y_i^> = 1$.
We now show that the random variables $Y_1^<, \ldots, Y_N^<$ fulfill Equation 3.3 with $p = \frac{1}{8e}$. Let us fix $k \in [N]$ and $\mathcal{I} \subseteq [k-1]$.
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 $Y_1^<, \ldots, Y_{k-1}^<$.
This means there is a subset $M \subseteq [16e\beta J]^{k-1}$ of delay values such that $\bigwedge_{i \in \mathcal{I}} Y_i^< = 1, \bigwedge_{i \in [k-1] \setminus \mathcal{I}} Y_i^< = 0$ iff $(\delta_1, \ldots, \delta_{k-1}) \in M$.
Let now be 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 $(a_1, \ldots a_{k-1}) \in [16e\beta J]^{k-1}$:

$$\mathrm{E}[X_i \mid \delta_1 = a_1, \ldots, \delta_{k-1} = a_{k-1}] = \frac{1}{16e\beta J} \quad .$$

Define furthermore

$$Z_k^< = \sum_{i=1}^{k-1} \min\left\{1, \frac{d(s_k,t_k)^\alpha}{p_k}\frac{p_i}{d(s_i,t_j)^\alpha}\right\} X_i$$

with $\mathrm{E}[Z_k^< | \delta_1 = a_1, \ldots, \delta_{k-1} = a_{k-1}] \leq \frac{1}{16e\beta}$.
Now it holds that

$$\begin{aligned}
\Pr[\, Y_k^< = 1 \mid \delta_1 = a_1, \ldots, \delta_{j-1} = a_{k-1} \,] &= \Pr\left[Z_k^< > \frac{1}{2\beta} \,\middle|\, \delta_1 = a_1, \ldots, \delta_{k-1} = a_{k-1} \right] \\
&\leq 2\beta\, \mathrm{E}[Z_k^< | \delta_1 = a_1, \ldots, \delta_{k-1} = a_{k-1}] \\
&= \frac{1}{8e} = p \enspace .
\end{aligned}$$

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

$$\begin{aligned}
&\Pr\left[Y_k^< = 1 \,\middle|\, \bigwedge_{i \in \mathcal{I}} Y_i^< = 1, \bigwedge_{i \in [k-1] \setminus \mathcal{I}} Y_i^< = 0 \right] \\
= &\sum_{(a_1,\ldots,a_{k-1}) \in M} \Pr\left[\delta_1 = a_1, \ldots \delta_{k-1} = a_{k-1} \,\middle|\, \bigwedge_{i \in \mathcal{I}} Y_i^< = 1, \bigwedge_{i \in [k-1] \setminus \mathcal{I}} Y_i^< = 0 \right] \\
&\quad \cdot \Pr[\, Y_k^< = 1 \mid \delta_1 = a_1, \ldots, \delta_{k-1} = a_{k-1} \,] \\
\leq &\; p \enspace .
\end{aligned}$$

Thus we may apply Lemma 3.2.2 on $I_j^<$ defined as follows:

$$I_j^< = \sum_{\substack{i=1 \\ i \neq j}}^{N} \min\left\{1, \frac{d(s_j,t_j)^\alpha}{p_j}\frac{p_i}{d(s_i,t_j)^\alpha}\right\} Y_i^< \enspace .$$

This random variable indicates the remaining measure of interference that is caused by these collisions.
Setting $\gamma = 2e$, $\mu = \frac{J}{8e}$ Lemma 3.2.2 states:

$$\Pr\left[I_j^< \geq \frac{J}{4} \right] \leq 2^{-\frac{J}{4}} \leq N^{-1} \enspace .$$

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

$$\Pr\left[I_j^< \geq \frac{J}{4} \right] \leq N^{-(k_0+2)} \enspace .$$

For reasons of symmetry 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 $I_j^{\text{curr}} \leq I_j^< + I_j^>$. For the remaining measure of interference $I^{\text{curr}} = \max_{j \in [N]} I_j^{\text{curr}}$ we can conclude:

$$\begin{aligned}
\Pr\left[I^{\text{curr}} \geq \frac{J}{2} \right] &\leq \sum_{j \in [N]} \Pr\left[I_j^{\text{curr}} \geq \frac{J}{2} \right] \\
&\leq \sum_{j \in [N]} \Pr\left[I_j^< \geq \frac{J}{4} \text{ or } I_j^< \geq \frac{J}{4} \right] \\
&\leq N \left(N^{-(k_0+2)} + N^{-(k_0+2)} \right) \\
&\leq N^{-k_0} \quad .
\end{aligned}$$

This proves the claim. □

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

Theorem 3.2.4. *Algorithm 2 generates a schedule of length at most $O(I + \log^2 N)$ steps whp.*

Proof. Let T_k be the number of scheduling steps generated in the k-th execution of the outer *while* loop. As shown in the previous lemma, it holds that

$$\Pr\left[T_k \geq (k_0 + 3)\, 16 e \beta \frac{1}{2^{k-1}} I \right] \leq \frac{1}{N^{k_0+1}} \quad .$$

Let furthermore U denote the number of scheduling steps generated in the execution of Algorithm 1. As shown in Lemma 3.2.1, it holds that

$$\Pr[\, U \geq (k_0 + 2)\, 4 \beta \ln N \log N \,] \leq \frac{1}{N^{k_0+1}} \quad .$$

Thus the total number of steps generated in the *while* loops $\sum_k T_k + U$ can be estimated by

$$\begin{aligned}
&\Pr\left[\sum_k T_k + U \geq (k_0 + 3)\, 32 e \beta I + (k_0 + 2)\, 4 \beta \ln N \log N \right] \\
&\leq \Pr\left[\bigvee_k T_k \geq (k_0 + 3)\, 16 e \beta \frac{1}{2^{k-1}} I \vee U \geq (k_0 + 2)\, 4 \beta \ln N \log N \right] \\
&\leq \sum_k \Pr\left[T_k \geq (k_0 + 3)\, 16 e \beta \frac{1}{2^{k-1}} I \right] + \Pr[\, U \geq (k_0 + 2)\, 4 \beta \ln N \log N \,] \\
&\leq \sum_k \frac{1}{N^{k_0+1}} + \frac{1}{N^{k_0+1}} \\
&\leq (\log N + 1) \frac{1}{N^{k_0+1}}
\end{aligned}$$

$$\leq \frac{1}{N^{k_0}} \ .$$

This means the total number of steps is at most $(k_0 + 3)\, 32e\beta I + (k_0 + 2)\, 4\beta \ln N \log N = O(I + \log^2 N)$ with probability at least $1 - \frac{1}{N^{k_0}}$. □

3.3 Extensions for Multi-hop Scheduling

Up to now, the described algorithms only take care of single-hop scheduling with a fixed power assignment. As a first generalization, we extend the results to multi-hop schedules in this section. This means we route along fixed paths first and approximate optimal paths afterwards. To simplify the proofs we introduce $\hat{n} = |V|^2 \cdot N$ as an upper bound for the maximum number of reasonable hops. Obviously, it is still polynomially bounded in the input size.

3.3.1 Multi-hop Scheduling with Fixed Paths

In our first approach, we leave paths fixed. Thus the considered scheduling problem is more or less the same as considered before apart from some dependencies such that one communication hop has to be done after another one has already taken place. With Algorithm 3 we succeed in re-using single-hop algorithms for this problem. The idea is to simply assign a delay to each packet. By this shift, a number of time frames is created. The measure of interference I is sufficiently balanced between these time frames and thus the single-hop algorithm $\mathcal{A}$ can be used to schedule the time frame to generate feasible time slots since there is at most one hop of a packet assigned to each time frame.

Algorithm 3 Fixed path multi-hop scheduling

```
for all i ∈ [N] do
   assign a delay 1 ≤ δ_i ≤ 2eI / log^ψ n̂ i. u. r.
end for
for all 1 ≤ t ≤ 2eI / log^ψ n̂ + D do
   execute A on all hops (i, j) with δ_i + j = t
end for
```

Theorem 3.3.1. *Let $\mathcal{A}$ be a single-hop algorithm generating a schedule of length $O(I + \log^\psi N)$, $\psi \geq 1$, whp. Then the schedule generated by Algorithm 3 has length $O(I + D \log^\psi \hat{n})$ whp.*

Proof. For $(k, l) \in \mathrm{dom}(\mathcal{P})$ let $I_{k,l}^{(t)}$ be the random variable of I caused by all hops assigned to time frame t. Let T_t denote the schedule length that is used to schedule the t-th time frame.

Let be $X_{i,j,t} = 1$ iff $\delta_i + j = t$. Then we have

$$I_{k,l}^{(t)} = \sum_{(i,j) \in \operatorname{dom}(\mathcal{P})} \min \left\{ 1, \frac{d(P_{i,j-1}, P_{i,j})^\alpha}{d(P_{i,j-1}, P_{k,l})^\alpha} \right\} X_{i,j,t} \quad .$$

This means

$$\operatorname{E} I_{k,l}^{(t)} = \sum_{(i,j) \in \operatorname{dom}(\mathcal{P})} \min \left\{ 1, \frac{p_{i,j}}{p_{k,l}} \frac{d(P_{i,j-1}, P_{i,j})^\alpha}{d(P_{i,j-1}, P_{k,l})^\alpha} \right\} \frac{\log^\psi \hat{n}}{2eI} \leq \frac{\log^\psi \hat{n}}{2e} \quad .$$

For fixed t the random variables $X_{i,j,t}$ are *negatively associated* as defined by Dubhashi and Ranjan [DR98]. So a Chernoff bound is applicable: for all $k_2 \geq 1$ it holds that

$$\Pr\left[I_{k,l}^{(t)} \geq k_2 \log^\psi \hat{n} \right] \leq 2^{-k_2 \log^\psi \hat{n}} \leq 2^{-k_2 \log \hat{n}} = \hat{n}^{-k_2} \quad .$$

By definition of $\mathcal{A}$ for all constants k_1 and k_2 there is a constant k_0 such that:

$$\Pr\left[T_t \geq k_0 k_2 \log^\psi \hat{n} \;\middle|\; \max_{(k,l) \in \operatorname{dom}(\mathcal{P})} I_{k,l}^{(t)} \leq k_2 \log^\psi \hat{n} \right] \leq \frac{1}{\hat{n}^{k_1}} \quad .$$

We can now estimate the overall schedule length:

$$\begin{aligned}
&\Pr\left[\sum_t T_t \geq k_0 k_2 (I + D \log^\psi \hat{n}) \right] \\
\leq &\sum_t \Pr\left[T_t \geq k_0 k_2 \log^\psi \hat{n} \right] \\
\leq &\sum_t \left(\frac{1}{\hat{n}^{k_1}} + \sum_{(k,l) \in \operatorname{dom}(\mathcal{P})} \Pr\left[I_{k,l}^{(t)} \leq k_2 \log^\psi \hat{n} \right] \right) \\
\leq &\left(\frac{2eI}{\log^\psi \hat{n}} + D \right) \left(\frac{1}{\hat{n}^{k_1}} + \hat{n} \cdot \frac{1}{\hat{n}^{k_2}} \right) \\
\leq &\frac{1}{\hat{n}^{k_3}} \qquad \text{for some constant } k_3 \text{ depending on } k_1 \text{ and } k_2 \quad .
\end{aligned}$$

This means we have a schedule length of $O(I + D \log^\psi \hat{n})$ whp for fixed path scheduling in the linear power assignment. □

In total, this means by combining Algorithms 2 and 3 we get an $O(I + D \log^2 \hat{n})$ whp algorithm.

3.3.2 Finding Optimal Paths

So far the paths taken by the packets are fixed. In this section we will find a way to approximate optimal paths when using the linear power assignment. For this purpose we will adapt an approach first used by Srinivasan and Teo [ST97] for wired networks. Chafekar *et al.* [CKM+07] also use it as a part of their CLM algorithm. We solve an *Integer Linear Program* (ILP) approximatively by using relaxation and randomized rounding.
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 $N_{\text{in}}(v)$ resp. $N_{\text{out}}(v)$ denote the incoming resp. outgoing edges from v.
Minimize w subject to:

$$\forall i \in [N] \qquad \sum_{e \in N_{\text{out}}(s_i)} y(i,e) - \sum_{e \in N_{\text{in}}(s_i)} y(i,e) = 1 \tag{3.5a}$$

$$\forall i \in [N], v \in V \backslash \{s_i, t_i\} \qquad \sum_{e \in N_{\text{out}}(v)} y(i,e) - \sum_{e \in N_{\text{in}}(v)} y(i,e) = 0 \tag{3.5b}$$

$$\forall i \in [N] \qquad \sum_{e \in E} y(i,e) \leq w \tag{3.5c}$$

$$\forall i \in [N], v \in V \qquad \sum_{e'=(u',v')} y(i,e') \min\left\{1, \frac{d(u',v')^\alpha}{d(u',v)^\alpha}\right\} \leq w \tag{3.5d}$$

$$\forall i \in [N], e \in E \qquad y(i,e) \in \{0,1\} \tag{3.5e}$$

This ILP is designed to minimize $w = \max\{I, D\}$ as follows. Condition 3.5d ensures that $I \leq w$ whereas condition 3.5c ensures $D \leq w$. By leaving out condition 3.5e, this ILP can be relaxed to an LP which then describes a multi-commodity flow problem. Algorithm 4 solves this LP and uses the result to approximate a solution of the ILP applying the technique of randomized rounding [RT87] .

Algorithm 4 Finding optimal paths

```
solve the relaxed ILP 3.5
for all i ∈ [N] do
    decompose the resulting flow for commodity i to paths P^1, ..., P^k with flows f_1, ..., f_k
    remove paths longer that 2w and scale the flows along the other paths proportionally
    select one of the remaining paths at random with probability f_i
end for
```

Theorem 3.3.2. *Let I^* and D^* be the values such that $I+D$ is minimal. Then Algorithm 4 calculates paths such that $I = O(I^* \log \hat{n})$ whp and $D \leq 2D^*$.*

Proof. We fix $i \in [N]$. Observe that

$$\sum_{j:\ell(P^j) \leq 2w} f_j \geq \frac{1}{2}$$

because otherwise condition 3.5c could not be satisfied.
This means by leaving out all paths longer than $2w$ and scaling the flows along the other paths proportionally all flows are doubled at most. Therefore conditions 3.5c and 3.5d hold after substituting w by $2w$.
Since all remaining paths are not longer than $2w$, it is ensured that $D \leq 2w$. Thus only I will increase depending on which paths are selected at random. Let $I_{k,l}$ denote the random variable of measure of interference at link (k, l).
Its expectation value is obviously

$$\mathrm{E}[I_{k,l}] \leq 2w \ .$$

By applying a Chernoff bound, we get

$$\Pr[\, I_{k,l} \geq (k_0 + 1) 2e \log \hat{n} \cdot 2w \,] \leq 2^{-(k_0+1)2e \log \hat{n} \cdot 2w} \leq \hat{n}^{-(k_0+1)} \ .$$

For the maximum value I we have

$$\Pr[\, I \geq (k_0 + 1) 2e \log \hat{n} \cdot 2w \,] \leq \hat{n}^{-k_0} \ .$$

In total this means: if I^* and D^* are the values such that $I + D$ is minimal – which is the optimal solution for the ILP 3.5 – we found out how to calculate paths such that $I = O(I^* \log \hat{n})$ whp and $D \leq 2D^*$. □

Thus we have found a path selection algorithm for the linear power assignment. Using the scheduling algorithms we can now compose an approximation algorithm for the CLM problem restricted to the linear power assignment.

3.4 Summary

Let us summarize what we have found in this chapter for the CLM problem. We left the powers fixed to the ones given by the linear power assignment. On the one hand, we found a way to estimate the optimal schedule length when paths are fixed by $\Omega(I + D)$ where I is an analogon to the congestion C measuring the interference. On the other hand, using Algorithm 3 together with Algorithm 2 we get an $O(I + D \log^2 \hat{n})$ whp schedule. In combination with our path selection, Algorithm 4, we get a schedule of length $O(I^* \log \hat{n} + D^* \log^2 \hat{n})$ whp. In other words, we have found an $O(\log^2 \hat{n})$ whp approximation algorithm.

Chapter 4

Different Power Assignments

Up to now, we did not take the power assignment into consideration but left it fixed such that all schedules had to use the linear power assignment. Although it is quite common to use the linear power assignment (see e. g. [CKM+07]) there has been no study on the direct effect of this choice. Even if the linear power assignment has the advantage of using the minimum power needed to deal with ambient noise, we have to analyze the impact of this restriction on the optimal schedule length.

4.1 Lower Bounds

Both of the generic power assignments considered so far, the linear as well as the uniform power assignment, have in common that each transmission power only depends on the distance between the sender and the receiver and not on the other transmissions to be done. We call this behaviour *oblivious*.

Definition 4.1.1 (Oblivious Power Assignment)**.** *A power assignment p is called oblivious iff there is a function $f : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ such that $p_{i,j} = f(d(P_{i,j-1}, P_{i,j}))$, i. e. the assigned power only depends on the distance between sender and receiver.*

Of course, one could imagine to use more complicated functions than the linear or the constant function. In this section, we will see that in general any choice of a function can have a very bad effect on the optimal schedule length.

Theorem 4.1.2. *Let $f : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ be an oblivious power assignment function. Then there exists a family of instances on a line that needs $\Omega(N)$ steps when using the powers defined by f whereas all senders can transmit in a single step using a different power assignment.*

Proof. We consider the family of instances illustrated in Figure 4.1.
Formally, this kind of instance can be defined by $s_1, t_1, \ldots, s_N, t_N \in \mathbb{R}$ such that

$$s_i = \begin{cases} 0 & \text{if } i = 1 \\ t_{i-1} + y_i & \text{otherwise} \end{cases} \qquad \text{and} \qquad t_i = s_i + x_i.$$

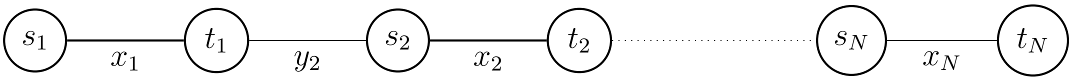

Figure 4.1: A visualization of the instances. x_i and y_i are chosen depending on f.

We fix the distance x_1 to 1. To simplify notation, let us furthermore define $\kappa := 2\beta + 1$ as a constant.
We now define the distances x_i and y_i between the nodes recursively depending on the function f. We can distinguish three cases as an arbitrary function $f\colon \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ obviously satisfies at least one of these conditions:

(a) For every $x_0 > 0$ and every $c > 0$ there is a value $x > x_0$ such that $f(x) > c$.

(b) There is a value $b > 0$ such that for all $x_0 > 0$ there is an $x > x_0$ with $\frac{b}{2} \leq f(x) \leq b$.

(c) For every $x_0 > 0$ and every $c > 0$ there is a value $x > x_0$ such that $f(x) < c$.

In case (a) we set:

$$y_i = \kappa^{2/\alpha}(x_{i-1} + y_{i-1}).$$

Given $x_1, \ldots, x_{i-1}$ and y_i, we choose x_i such that $x_i \geq y_i$ and

$$f(x_i) \geq y_i^\alpha \frac{f(x_j)}{x_j^\alpha} \qquad \text{for all } j < i.$$

By this construction it is ensured that a link k is exposed to high interference by links with larger indices. To show this, let S be a set of indices of packets that can be transmitted together in one step; $k = \min S$.
For $i \in S \backslash \{k\}$ it holds that

$$d(s_i, t_k) = \sum_{j=k+1}^{i-1} x_j + \sum_{j=k+1}^{i} y_j \leq 2 \sum_{j=k}^{i} y_j \leq 2 \sum_{j=k}^{i} \frac{1}{\kappa^{2(i-j)/\alpha}} y_i \leq \frac{2}{1 - \kappa^{-2/\alpha}} y_i.$$

Since all signals in S can be transmitted in one step the SINR condition is satisfied for packet k:

$$\beta \sum_{i \in S \backslash \{k\}} \frac{p_i}{d(s_i, t_k)^\alpha} \leq \frac{p_k}{d(s_k, t_k)^\alpha} = \frac{f(x_k)}{x_k^\alpha}.$$

Putting this together, we get

$$\frac{1}{\beta} \frac{f(x_k)}{x_k^\alpha} \geq \sum_{i \in S \backslash \{k\}} \frac{p_i}{d(s_i, t_k)^\alpha} \geq \sum_{i \in S \backslash \{k\}} \frac{y_i^\alpha \frac{f(x_k)}{x_k^\alpha}}{(\frac{1}{1 - \kappa^{-2/\alpha}} y_i)^\alpha} = \frac{1}{2^\alpha} (|S| - 1)(1 - \kappa^{-2/\alpha})^\alpha \frac{f(x_k)}{x_k^\alpha}.$$

In case (b) we choose $x_i = y_i$ such that $x_{i+1} \geq \kappa^{2/\alpha} x_i$ and $\frac{b}{2} \leq f(x_i) \leq b$ for all $i \in [N]$. Again, let S be a set of packets that can be transmitted together in one step; this time let be $k = \max S$.
For $i \in S$ it holds that

$$d(s_i, t_k) = \sum_{j=i}^{k} x_j + \sum_{j=i+1}^{k} y_j \leq 2 \sum_{j=i}^{k} x_j \leq 2 \sum_{j=i}^{k} \frac{1}{\kappa^{2(k-j)/\alpha}} x_k \leq \frac{2}{1 - \kappa^{-2/\alpha}} x_k.$$

The SINR condition states for packet k

$$\beta \sum_{i \in S \setminus \{k\}} \frac{p_i}{d(s_i, t_k)^\alpha} \leq \frac{p_k}{d(s_k, t_k)^\alpha} \leq \frac{b}{x_k^\alpha}.$$

Thus:

$$\frac{1}{\beta} \frac{b}{x_k^\alpha} \geq \sum_{i \in S \setminus \{k\}} \frac{p_i}{d(s_i, t_k)^\alpha} \geq \sum_{i \in S \setminus \{k\}} \frac{\frac{1}{2} b}{\left(\frac{1}{1-\kappa^{-2/\alpha}} x_k\right)^\alpha} = \frac{1}{2^\alpha} (|S| - 1)(1 - \kappa^{-2/\alpha})^\alpha \frac{\frac{1}{2} b}{(x_k)^\alpha}.$$

In case (c) we choose $x_i = y_i$ such that $x_{i+1} \geq \kappa^{2/\alpha} x_i$ and $f(x_{i+1}) \leq f(x_i)$ for all $i \in [N]$. Let S be a set of packets that can be transmitted together in one step; $k = \max S$.
The SINR condition is satisfied for packet k:

$$\beta \sum_{i \in S \setminus \{k\}} \frac{p_i}{d(s_i, t_k)^\alpha} \leq \frac{p_k}{d(s_k, t_k)^\alpha} \leq \frac{f(x_k)}{x_k^\alpha}$$

Now, we have:

$$\frac{1}{\beta} \frac{f(x_k)}{x_k^\alpha} \geq \sum_{i \in S \setminus \{k\}} \frac{p_i}{d(s_i, t_k)^\alpha} \geq \sum_{i \in S \setminus \{k\}} \frac{f(x_k)}{\left(\frac{2}{1-\kappa^{-2/\alpha}} x_k\right)^\alpha} = \frac{1}{2^\alpha} (|S| - 1)(1 - \kappa^{-2/\alpha})^\alpha \frac{f(x_k)}{(x_k)^\alpha}$$

In all three cases, we have $|S| \leq \frac{2 \cdot 2^\alpha}{\beta(1-\kappa^{-2/\alpha})^\alpha} + 1$. This means only a constant number of packets can be transmitted in one step and therefore $\Omega(N)$ steps are needed when using $p_i = f(d(s_i, t_i))$. In contrast, there is a power assignment, $p_i = \kappa^i$, which needs only one step in each of these instances. Note that in all instances it holds that $y_i \leq x_i$ and $y_{i+1} \geq \kappa^{2/\alpha} x_i$.
Let us first bound the interference on a receiver t_j by senders s_i with index $i < j$:

$$\sum_{i<j} \frac{p_i}{d(s_i, t_j)^\alpha} \leq \sum_{i<j} \frac{p_i}{x_j^\alpha} = \frac{1}{x_j^\alpha} \sum_{i<j} \kappa^i \leq \frac{1}{\kappa - 1} \frac{\kappa^j}{x_j^\alpha} \quad .$$

The interference by the remaining senders:

$$\sum_{i>j} \frac{p_i}{d(s_i, t_j)^\alpha} \leq \sum_{i>j} \frac{p_i}{\kappa^{2(i-j)} x_i^\alpha} \leq \frac{\kappa^{2j}}{x_i^\alpha} \sum_{i>j} \frac{1}{\kappa^i} \leq \frac{\kappa^{2j}}{x_i^\alpha} \frac{\frac{1}{\kappa^{j+1}}}{1 - \frac{1}{\kappa}} = \frac{1}{\kappa - 1} \frac{\kappa^j}{x_j^\alpha} \quad .$$

Summed up we get the total interference on a receiver t_j:

$$\sum_{i \neq j} \frac{p_i}{d(s_i, t_j)^\alpha} \leq 2 \frac{1}{\kappa - 1} \frac{\kappa^j}{x_j^\alpha} = \frac{1}{\beta} \frac{p_j}{d(s_j, t_j)^\alpha} \quad .$$

We can see that the SINR condition is satisfied. This means all packets can be transmitted in a single time slot using this power assignment. □

In total, we see that particularly the linear and the uniform power assignment can have a the worst possible performance that is even fulfilled by trivial schedules.

4.2 Upper Bounds

Having a closer look at the instances used as counter-examples in the last proof, it shows that the distances differ very much. The smallest value is 1 whereas the largest is $\Omega(2^N)$. In this section we will show that this large growth has to occur in worst-case examples. To express this difference, we introduce the ratio between the longest and the shortest distance between two nodes which act as a pair of sender and receiver for a hop:

$$\Delta = \frac{\max_{(u,v) \in E} d(u, v)}{\min_{(u,v) \in E} d(u, v)} \quad .$$

We furthermore define the set of indices of senders around a node with distance at most r

$$K_r(v) = \{ \, i \in [N] \mid d(s_i, v) \leq r \, \} \quad .$$

Our analysis will be non-constructive in the following way: We assume to know an optimal schedule $\mathcal{S}$ (using any power assignment) and to split it up. Then we consider the single time slots separately. Each of these time slots corresponds to an instance for which there is a schedule of length 1. We only need to bound the number of time slots needed for scheduling such a *one-step instance.*
A central result is that we can bound $|K_r(v)|$ in a one-step instance for every node v when $\Delta \leq 2$.

Lemma 4.2.1. *Given a single-hop instance such that all packets may be transmitted in a single time slot using any power assignment.*
If there is an $L > 0$ *such that* $L \leq d(s_i, t_i) \leq 2L$ *for all* $i \in [N]$*, then it holds for all* $v \in V$*,* $\ell \geq L$*:*

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

Proof. Let p be the optimum power assignment. Let furthermore be $i \in \arg\min_{j \in K_\ell(v)} p_j$. As the SINR condition is satisfied for packet i, we get:

$$\frac{(|K_\ell(v)| - 1) \cdot p_i}{(2\ell + 2L)^\alpha} \leq \sum_{j \in K \setminus \{i\}} \frac{p_j}{(2\ell + 2L)^\alpha} \leq \sum_{j \in K \setminus \{i\}} \frac{p_j}{d(s_j, t_i)^\alpha} \leq \frac{1}{\beta} \frac{p_i}{d(s_i, t_i)^\alpha} \quad .$$

So:

$$|K_\ell(v)| - 1 \leq \frac{1}{\beta} \left(\frac{2\ell + 2L}{d(s_i, t_i)} \right)^\alpha \leq \frac{1}{\beta} \left(\frac{4\ell}{L} \right)^\alpha \quad .$$

□

We use this bound to get a bound on the measure of interference I in a one-step instance with $\Delta \leq 2$.

Lemma 4.2.2. *Given a single-hop instance such that all packets may be transmitted in a single time slot using any power assignment.*
If there is an $L > 0$ such that $L \leq d(s_i, t_i) \leq 2L$ for all $i \in [N]$, then the same instance using the power assignment $p_i = d(s_i, t_i)^\gamma$, $\gamma \in \mathbb{R}$, has a measure of interference of

$$I \leq 4 \cdot 8^\alpha (\log N + 1) + \frac{4^\alpha}{\beta} + 1 \quad .$$

Proof. For the ease of notation we change the indices. Let t_0 be the vertex where I attains its maximum value. Let $s_1, \ldots, s_{N-1}$ be ordered by increasing distance to t_0. Note that for all $\ell > 0$ we have $K_\ell(t_0) \setminus \{0\} = \{1, \ldots, m\}$ for some $m \in \mathbb{N}$ by this definition.
For $k \leq \log N + 1$ let be $R_k = [2^k] \setminus [2^{k-1}]$. Furthermore, let ℓ_k be defined as $\ell_k = \min_{j \in R_k} d(s_j, t_0)$.
As the distances are increasing, we have $\ell_k \geq d(s_j, t_0)$ for all $j \leq 2^{k-1}$. In other words: $[2^{k-1}] \subseteq K_{\ell_k}(t_0)$.
Since we will add up the interference by $K_L(t_0)$ separately, we may assume $\ell_k \geq L$ for all k and thus apply Lemma 4.2.1 on $|K_{\ell_k}(t_0)|$:

$$2^{k-1} = |[2^{k-1}]| \leq |K_{\ell_k}(t_0)| \leq \left(\frac{4\ell_k}{L} \right)^\alpha + 1$$

This means:

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

Using this result we can bound I. As outlined before, we add up the interference by $K_L(t_0)$ as well.

$$I = \sum_{i=1}^{N} \min \left\{ 1, \frac{d(s_0, t_0)^{\alpha - \gamma} d(s_i, t_i)^\gamma}{d(s_i, t_0)^\alpha} \right\}$$

$$
\begin{aligned}
&\leq \sum_{k=1}^{\log N+1} \sum_{i \in R_k} \frac{d(s_0,t_0)^{\alpha-\gamma} d(s_i,t_i)^{\gamma}}{d(s_i,t_0)^{\alpha}} + \sum_{i \in K_L(t_0)} 1 \\
&\leq (2L)^{\alpha} \sum_{k=1}^{\log N+1} \frac{|R_k|}{\ell_k^{\alpha}} + |K_L(t_0)| \\
&\leq (2L)^{\alpha} \sum_{k=1}^{\log N+1} \frac{2^k}{(2^{k-1}-1)\left(\frac{L}{4}\right)^{\alpha}} + \left(\frac{4^{\alpha}}{\beta} + 1\right) \\
&\leq 8^{\alpha} \sum_{k=1}^{\log N+1} 4 + \frac{4^{\alpha}}{\beta} + 1 \\
&\leq 4\, 8^{\alpha}(\log N + 1) + \frac{4^{\alpha}}{\beta} + 1
\end{aligned}
$$

□

Having found a bound on the measure of interference I for such an instance, we can now use Algorithm 2 to generate a new schedule that uses the uniform resp. the linear power assignment.

Theorem 4.2.3. *Given an instance that can be scheduled in T steps using the optimal power assignment. The same instance can be scheduled in $O(\log \Delta \cdot \log^2 N \cdot T)$ steps using the uniform or the linear power assignment.*

Proof. Given a schedule $\mathcal{S}$, we define a schedule $\mathcal{S}'$ of length $O(\log \Delta \cdot \log^2 N \cdot T(\mathcal{S}))$ for the linear or the uniform power assignment. Each step of $\mathcal{S}$ is split up to $O(\log \Delta \cdot \log^2 N)$ steps in $\mathcal{S}'$ as follows.
Divide the hops scheduled in one step into classes $C_0, \ldots, C_{\log \Delta}$. C_i contains all hops of distance 2^i to 2^{i+1}. Now each class C_i corresponds to an instance suitable for Lemma 4.2.2. Applying Algorithm 2 we find out there is a schedule of length $O(\log^2 N)$ for each class. Let $\mathcal{S}'$ be the concatenation of these schedules. The total length is obviously $O(\log \Delta \cdot \log^2 N \cdot T)$. □

We see that the loss of performance is at most a factor of $O(\log \Delta \cdot \log^2 N)$. Compared to the examples in the previous section we see this upper bound is not yet tight and, in fact, we will find a better one when restricting to the Euclidean plane in Chapter 5.

4.3 Summary

Our first observation in this chapter was that fixing to the linear power assignment can lead to a very poor schedule. Thus algorithms using it cannot have good approximation factors in terms of N. However, we found out that worst-case instances have to have an "unnatural" structure. The ratio Δ between shortest and longest distance between two

nodes has to be large in these instance as we found out the loss is at most a factor of $O(\log \Delta \cdot \log^2 N)$.
Let us combine this result with the ones in Chapter 3. There, we found an $O(\log^2 \hat{n})$ whp approximation algorithm for the CLM problem compared to the optimum under the linear power assignment. Since we can at most lose another factor of $O(\log \Delta \cdot \log^2 N)$ by this restriction, we get a total approximation factor of $O(\log \Delta \cdot \log^4 \hat{n})$ whp.

Chapter 5

Scheduling in the Euclidean Plane

As already pointed out, the networks so far were based on an arbitrary metric, which is only restricted by the positive definiteness, symmetry, and the triangle inequality. In most practical scenarios, however, all nodes of the network are located in a plane. This is why in almost all previous work only networks in the Euclidean plane have been analyzed.
In this chapter we will get some more insight in which way this restriction changes the results. In fact, it enables us to use a completely different approach for scheduling that is much closer to common approaches in graph-based models. This is why the protocol model will be studied again under new assumptions.
As a further, technical limitation we require the path-loss exponent α to be strictly greater than 2 in this chapter.

5.1 A Scheduling Algorithm

Let us first observe an important fact for designing a different scheduling approach. The idea will be to create a situation where space between senders is large enough, which is expressed in $d_{\max} = \max_{i \in [N]} d(s_i, t_i)$. We will prove a condition sufficient for feasible schedules formally for bounded power assignments as follows.

Lemma 5.1.1. *Let be* $\Gamma \geq 1$. *We consider a single-hop scheduling instance in the Euclidean plane such that*

$$K_{3a}(s_i) = \{i\} \qquad \textit{for } a = \max\left\{ \left(\frac{12\Gamma}{\alpha - 2} \beta \right)^{\frac{1}{\alpha}} d_{\max}, d_{\max} \right\} .$$

Then all packets can be transmitted in a single time slot using any power assignment p *in which holds* $p_{\max} \leq \Gamma \cdot p_{\min}$ *for*

$$p_{\max} = \max_{i \in [N]} p_i \qquad \textit{and} \qquad p_{\min} = \min_{i \in [N]} p_i .$$

Proof. Let us first bound the number of senders s_i with $r < d(s_i, t_j) \leq r + 1$ for each receiver t_j. These senders lie on the ring with inside radius ar and outside radius $a(r+1)$ that is centered at t_j.
For two senders s_i and s_j we have $d(s_i, s_j) \geq 3a$. Thus disks of radius a centered at the nodes s_i do not overlap. Each of them covers an area of πa^2; the entire ring covers an area of $\pi a^2(r+1)^2 - \pi a^2 r^2$.
Since at least a quarter of each disk lies inside the ring we have:

$$|K_{a(r+1)}(t_j) \backslash K_{ar}(t_j)| \leq \frac{1}{\frac{1}{4}\pi a^2} \pi a^2 \left((r+1)^2 - r^2\right) = 4(2r+1) \leq 12r \ .$$

The distance from t_j to all senders is at least $2a$. Thus we can bound the interference on t_j by

$$\sum_{r=2}^{\infty} |K_{a(r+1)}(t_j) \backslash K_{ar}(t_j)| \frac{p_{\max}}{(ar)^\alpha} \leq \frac{12\Gamma p_{\min}}{a^\alpha} \sum_{r=2}^{\infty} r^{1-\alpha} \leq \frac{12\Gamma p_{\min}}{a^\alpha} \frac{1}{\alpha - 2} \leq \frac{1}{\beta} \frac{p_{\min}}{d_{\max}^\alpha} \ .$$

This means the SINR constraint is satisfied. □

This is an interesting observation: We see that the SINR condition can be replaced by a local binary one. This enables us to construct a different kind of algorithm. To make a schedule feasible we only have to solve this constraint which can be regarded as a graph coloring problem. Algorithm 5 gives a formalization for single-hop scheduling.

Algorithm 5 A simple algorithm in the plane for single-hop scheduling

```
C_j := { i ∈ [N] | 2^j ≤ d(s_i, t_i) ≤ 2^{j+1} }
for all 0 ≤ j ≤ log Δ do
   define the graph H = (C_j, E') where E' = { (i, j) ∈ C_j^2 | d(s_i, s_j) ≤ 3a }
   color H with deg(H) + 1 colors[1]
   construct a schedule such that all senders of the same color transmit at a time
end for
```

Theorem 5.1.2. *Algorithm 5 generates a correct schedule of length $O(\log \Delta \cdot T)$ for the linear and the uniform power assignment where T is the optimal schedule length using any power assignment.*

Proof. Let us remark first the resulting schedule is feasible as figured out in Lemma 5.1.1. So it remains to show the schedule's length is $O(\log \Delta \cdot T)$. Let there be an optimal schedule of length T consisting of steps $S_1, \ldots, S_T$. Then we can apply Lemma 4.2.1 to all sets $K_{3a}(t_i) \cap S_t \cap C_j$. For $\Gamma = 1$ resp. $\Gamma = 2^\alpha$ we get:

$$|K_{3a}(t_i) \cap S_t \cap C_j| \leq \frac{1}{\beta} \left(\frac{4 \cdot 3a}{2^j}\right)^\alpha = \frac{12}{\beta} \max\left\{ \left(\frac{12\Gamma}{\alpha - 2}\beta\right)^{\frac{1}{\alpha}}, 1 \right\} = O(1) \ .$$

[1] $\deg(H)$ denotes the maximum degree in the graph H.

We can conclude that $\deg(i) = |K_{3a}(t_i) \cap C_j| = O(T)$ for all $i \in [N], 0 \leq j \leq \log \Delta$. Thus we have $\deg(H) = O(T)$ resulting in a schedule of a total length of $O(\log \Delta \cdot T)$. □

Obviously, this approach can be extended to multi-hop scheduling by the techniques used in Section 3.3 taking $\max_{v \in V} |K_{3a}(v)|$ as a measure of interference.
Another important fact is that we have compared the schedule length to the optimal schedule using any power assignment. This extends our previous result of Section 4.2: we can only lose a factor of $O(\log \Delta)$ when using the uniform or the linear power assignment instead of the optimal one in the Euclidean plane. This means our results in Section 4.1 are tight.

5.2 Equivalence of the Protocol Model and the SINR Model

The techniques used in the previous section remind of the ones used for graph-based models. Indeed, there is a close relationship between the protocol model and the SINR model restricted to small ratios Δ between the shortest and longest distance between two nodes in the Euclidean plane.

Theorem 5.2.1 (Equivalence of protocol model and SINR model)**.** *Consider a network in the plane with $L \leq d(s_i, t_i) \leq 2L$ for all $i \in [N]$. Let $T_{protocol}$ resp. T_{SINR} be the optimal schedule length in the protocol model resp. in the SINR model.*
Then there exist constants c_1 and c_2 such that

(a) $T_{SINR} \leq c_1 \cdot T_{protocol}$ *and*

(b) $T_{protocol} \leq c_2 \cdot T_{SINR}$.

Proof. To prove claim (a), we split a schedule in the protocol model into its single steps. For each of these steps we prove there is a schedule of constant length in the SINR model using the uniform power assignment. To use a graph coloring like in Algorithm 5 we only have to show $|K_{3a}(s_i)|$ is constant.
We have

$$K_{(1+\varepsilon)d(s_i,t_i)}(t_i) = \{s_i\} \ .$$

This means

$$d(s_i, s_j) \geq L\varepsilon \ .$$

For $K_{3a}(s_i)$ we can conclude:

$$|K_{3a}(s_i)| \leq \frac{1}{\frac{1}{4}\pi(L\varepsilon)^2}\pi(3a)^2 = O(1) \ .$$

Now it remains to prove claim (b). Given a one-step instance in the SINR model Lemma 4.2.1 states

$$|K_{(3+\varepsilon)2L}(t_k)| = O(1) \ .$$

From the sender's point of view this is

$$|K_{(2+\varepsilon)2L}(s_k)| = O(1) \ .$$

This means there is a coloring $\mathcal{C}$ of the graph $H = ([N], E')$ with $E' = \{(i, j) \in [N]^2 \mid d(s_i, s_j) \leq 2L\}$ using a constant number of colors such that

$$K_{(2+\varepsilon)2L}(s_k) \cap \mathcal{C}(k) = \{k\} \ .$$

As a result we have

$$K_{(1+\varepsilon)d(s_k,t_k)}(t_k) \cap \mathcal{C}(k) \subseteq K_{(1+\varepsilon)2L}(t_k) \cap \mathcal{C}(s_k) = \emptyset \ .$$

So the coloring corresponds to a schedule in the protocol model of constant length. □

Of course, this theorem can be generalized for arbitrary Δ by splitting up the senders into $\log \Delta$ classes. We can draw as a conclusion that the SINR model and the protocol model are quite similar when restricted to "natural" settings.

Chapter 6

Conclusions and Further Work

For the linear power assignment, we have found quite an interesting algorithm for the CLM problem with an approximation factor of $O(\log^2 N)$ whp. But in addition, we have shown that the linear power assignment can perform very poorly. However, no other oblivious power assignment can be better.

As a first solution to this dilemma, we have found out that a large growth in distances has to occur in worst-case examples for the linear and the uniform power assignment. In the Euclidean plane we have found a bound of $O(\log \Delta)$. This means that an exponential growth has to take place. This, in contrast, is quite unrealistic within practical networks. On the other hand, we have also found out that schedule lengths in the SINR model only differ by a factor of $O(\log \Delta)$ from the lengths in the protocol model in the Euclidean plane. Altogether, this means to profit from the SINR model and its shorter schedule compared to the ones of the protocol model, it is crucial to find optimal power assignments. This still remains an open problem. In fact, the power assignment has not yet been optimized at all up to now.

Furthermore we have proven the CLM problem to be NP-hard when powers are subject to optimization. Unfortunately, the constructed instances use high dimensional metrics. Up to now there does not seem to be any proof of the NP-hardness of the CLM problem restricted to the plane.

Partly, our algorithms are designed in a distributed manner. After all, none of them can be implemented in a distributed setting this way. This would be vital to benefit from our results in practice.

In this thesis we have focused on minimizing latency. But apart from the CLM problem, for example, it is of much practical relevance how much energy is consumed since mobile devices have very limited capacities. As already pointed out the linear power assignment is optimal with respect to energy consumption when the routing paths are fixed. Thus many of the techniques we applied could also be useful for studying these problems.

A last point to be mentioned is that in practice scheduling is not static. For wired networks there are models of dynamic scheduling dealing with adversarial or stochastic packet injection. It is definitely an interesting question whether these results can be transferred to the SINR model as well.

Bibliography

[CKM+07] 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 *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pages 110–119, New York, NY, USA, 2007.

[CKM+08] 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. *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 1166–1174, April 2008.

[DR98] Devdatt P. Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *Random Structures and Algorithms*, 13(2):99–124, 1998.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*, pages 96–107. W. H. Freeman & Co., New York, NY, USA, 1979.

[GK00] Piyush Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46:388–404, 2000.

[GMW08] Olga Goussevskaia, Thomas Moscibroda, and Roger Wattenhofer. Local broadcasting in the physical interference model. In *DIAL M-POMC '08: Proceedings of the fifth international workshop on Foundations of mobile computing*, pages 35–44, New York, NY, USA, 2008.

[GOW07] Olga Goussevskaia, Yvonne Anne Oswald, and Roger Wattenhofer. Complexity in geometric SINR. In *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pages 100–109, New York, NY, USA, 2007.

[KMPS04] V. S. Anil Kumar, Madhav V. Marathe, Srinivasan Parthasarathy, and Aravind Srinivasan. End-to-end packet-scheduling in wireless ad-hoc networks. In *SODA '04: Proceedings of the fifteenth annual ACM-SIAM symposium on*

Discrete algorithms, pages 1021–1030, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics.

[LMR94] F. T. Leighton, Bruce M. Maggs, and Satish B. Rao. Packet routing and job-shop scheduling in O(congestion+dilation) steps. *Combinatorica*, 1994.

[MOW07] Thomas Moscibroda, Yvonne Anne Oswald, and Roger Wattenhofer. How optimal are wireless scheduling protocols? In *Proc. of the 26 thAnnual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM)*, 2007.

[MW06] Thomas Moscibroda and Roger Wattenhofer. The complexity of connectivity in wireless networks. 2006.

[MWW06] Thomas Moscibroda, Roger Wattenhofer, and Yves Weber. Protocol design beyond graph-based models. In *In Proceedings of the 5th ACM SIGCOMM Workshop on Hot Topics in Networks (HotNets) 2006.*, 2006.

[MWZ06] Thomas Moscibroda, Roger Wattenhofer, and Aaron Zollinger. Topology control meets sinr: the scheduling complexity of arbitrary topologies. In *MobiHoc '06: Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, pages 310–321, New York, NY, USA, 2006.

[Rag88] Prabhakar Raghavan. Probabilistic construction of deterministic algorithms: approximating packing integer programs. *Journal of Computer and System Sciences*, 37(2):130–143, 1988.

[RT87] Prabhakar Raghavan and Clark D. Tompson. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374, 1987.

[Sch98] Christian Scheideler. *Universal Routing Strategies for Interconnection Networks*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1998.

[ST97] 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.