Scheduling in Packet Radio Networks

by

Hisham Ibrahim Kassab

S.M. Electrical Engineering and Computer Science (MIT 1993)

S.B. Electrical Engineering (MIT 1993)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Masters of Science in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1998

Author . . . .
Department of Electrical Engineering and Computer Science
April 30, 1998

Certified by . . . .
Robert Gallager
Professor of Electrical Engineering and Computer Science
Co-director, Laboratory for Information and Decision Systems
Thesis Supervisor

Certified by . . . .
Sanjoy Mitter
Professor of Electrical Engineering
Co-director, Laboratory for Information and Decision Systems
Director, Center for Intelligent Control Systems
Thesis Supervisor

Accepted by . . . .
Robert Freund
Professor of Management Science
Co-director, Operations Research Center

Scheduling in Packet Radio Networks

by

Hisham Ibrahim Kassab

S.M. Electrical Engineering and Computer Science (MIT 1993)

S.B. Electrical Engineering (MIT 1993)

Submitted to the Department of Electrical Engineering and Computer Science
on April 30, 1998, in partial fulfillment of the
requirements for the degree of
Masters of Science in Operations Research

Abstract

A typical wireless communication network for mobile users consists of a centrally-located and stationary base-station with mobile nodes communicating with the base-station. A Packet Radio Network is a wireless network without a base-station, in which nodes are typically mobile. The nodes have equal capabilities, and peer-to-peer multi-hop communication is allowed. However, multi-hop routing in Packet Radio Networks involves a number of issues which interact in a complex fashion. The primary source of this complexity is that unlike point-to-point wireline networks in which each channel is utilized by a single pair of nodes, the radio channel in the packet radio network must be shared by all the nodes. As a result a node cannot transmit and receive simultaneously. Therefore scheduling is required to ensure that no node transmits and receives simultaneously.

We look at scheduling within the larger framework of minimum power routing in packet radio networks. The corresponding mathematical programming problem proves to be difficult, and we investigate a suboptimal approach that simplifies the mathematics, with the hope that it will give us insight into the dynamics of the problem. The suboptimal approach relies on decoupling the routing and scheduling components of the larger problem, solving each one separately. We focus on solving the scheduling component.

Thesis Supervisor: Robert Gallager
Title: Professor of Electrical Engineering and Computer Science
Co-director, Laboratory for Information and Decision Systems

Thesis Supervisor: Sanjoy Mitter
Title: Professor of Electrical Engineering
Co-director, Laboratory for Information and Decision Systems
Director, Center for Intelligent Control Systems

Acknowledgments

I wish to express my deepest appreciation to my research advisor and mentor, Professor Bob Gallager, for his guidance and his help with my research. I also want to thank him for inspiring me to strive to become the best researcher that I can be. I am also grateful to Randy Berry, Steve Finn and Jim Mills for their helpful comments, suggestions and ideas during our group meetings. In addition, I want to thank Professor Sanjoy Mitter for offering to be the OR supervisor for this thesis, as well as Professor Michael Sipser and Professor Madhu Sudan for their help with one of the proofs.

On a more personal note, I wish to thank all my friends at MIT, and my colleagues at LIDS for adding a pleasant social dimension to my experience at MIT. In particular, I want to thank my best friend and the love of my life, Nadine. Her love and support over the last few years have been invaluable; and our partnership has truly enriched my life. Last but not least, I want to thank my brothers Samer and Kamal, my sister Nada, and my parents Ibrahim and Sana for being a constant source of support over the years, and for always being there for me.

The research in this thesis was supported through collaborative participation in the Advanced Telecommunications/Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under Cooperative Agreement DAAL01-96-2-0002.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

A typical wireless communication network for mobile users consists of a centrally-located and stationary base-station with mobile nodes communicating with the base-station. A Packet Radio Network (PRNET) is a wireless network without a base-station, consisting of nodes that are typically mobile. Fig. 1-1 shows an example of a PRNET. What distinguishes Packet Radio Networks (PRNETs) from a typical commercial cellular system is that the communication nodes in PRNETs are equal peers, in the sense that they have equal capabilities. In contrast, in a cellular system the base-station is a much more sophisticated piece of equipment than the cellular phone.

A PRNET has several advantages. It does not require centralized control. It has the potential of fast (and ad-hoc) deployment and set-up of a network. Therefore PRNETs are useful for communication between members of a rescue team, soldiers in an infantry unit, or in any situation where a communication infrastructure is not available.

Each node in a PRNET can receive and transmit radio signals, which implies that nodes can communicate directly, or through intermediate nodes. Looking at Fig. 1-1 again, we see that Node 2 is out of the transmission range of Node 3, and vice versa. However, the two nodes can still communicate through Node 1. For example, Node 3 can send the message (intended for Node 2) to Node 1, and the latter then forwards it to Node 2. This is referred to as multi-hop routing. The number of intermediate

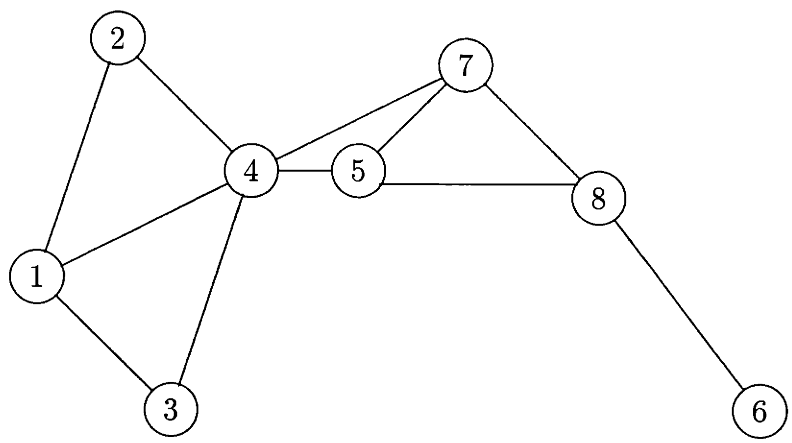

Figure 1-1: An Example of a Packet Radio Network

nodes can be more than one. In addition to allowing communication with out-of-transmission-range nodes, multi-hop routing can lower the network power use, as we will see in Subsection 1.3.5.

1.1 Routing in Packet Radio Networks

In this thesis we focus on scheduling in PRNETs, but we do so within the larger framework of routing in packet radio networks. The function of routing is to guide the data through the communication to its intended destination. The path that the data takes from its source to its destination depends on the criterion used in routing. For example a popular routing criterion in PRNETs is minimum-number-of-hops routing. Another routing criteria is minimum-delay routing. We are interested in minimum-average-network-power routing. The reason for choosing this criterion is that by minimizing the power used in data transmissions, we maximize the lifetimes of the node batteries.

Research in the area of PRNETs has been going on for more than two decades. At a first glance, PRNETs seem to be a simple extension of the technology of packet switched networks. However, two central issues make PRNETs much harder to implement:

1. The nodes in the network share the same communication channel.

2. Packets need to be routed in a highly dynamic network.

In this thesis we will be concerned mostly with the first of these two issues. The fact that the nodes use the same communication channel means that their transmissions interfere with each other. Moreover (and more importantly for us) it also implies that no node can transmit and receive simultaneously. This last constraint is important because it necessitates scheduling in a PRNET. By scheduling we mean designating the time intervals in which a node can transmit.

1.2 Choice of Multiaccess Scheme

The multiaccess scheme is the method by which the nodes share the communication channel. There are three main multiaccess schemes: frequency-division multiaccess (FDMA), time-division multiaccess (TDMA), and code-division multiaccess (CDMA). For information on the first two, see [1], and for information on the third see [2].

Our choice for the multiaccess scheme is a combination of TDMA and CDMA (CDMA/TDMA). TDMA facilitates scheduling as it allows each node to transmit in certain time slots, and receive in the remaining time slots. Within each time slot, we allow several nodes to transmit, so there is still a need to share the channel among those transmitting nodes, and we do so using CDMA. CDMA has several advantages. It allows a node to transmit to several receivers, and allows receivers to receive from several transmitters. In CDMA, when a node reduces its transmitted power, the other nodes benefit from that reduction in the form of lower interference (see [3]). In addition CDMA provides protection against intentional jamming and listening-in. This is advantageous because one of the main uses of PRNETs is in military communications.

1.3 Problem Statement and Model

We consider a wireless network with a set of N mobile nodes. The goal of our larger framework is to find the **minimum average network power** routing for this wireless network, allowing the possibility of **multi-hop** transmissions. Network power refers to the total power used by all the nodes. Each node is powered by a battery, and therefore minimizing average network power (ANP) is important because it helps maximize the lifetime of the batteries. Note that we will not address the issue of comparable levels of power use among individual nodes; rather we will focus on minimizing the overall power use in the network.

1.3.1 Assumption of Acceptable Connection

We assume that a transmitting node can adjust its radiated power level, such that the intended receiving node receives the transmitted signal at the required signal-to-interference ratio (SIR). The required SIR is the ratio that guarantees a specified symbol error rate. In other words, it is the ratio that ensures that the received data is not too garbled. We assume perfect **power control**, which implies that the transmitting node can adjust its radiated power level instantaneously, and has perfect information to adjust its power level appropriately. Power control is important, because whenever a node transmits, it interferes with transmissions from other nodes. If a node transmits at too high a power it will create unnecessarily high interference for the other nodes, which in turn will have to increase their power to overcome the extra interference. The end result will be inefficient use of power, which goes against our goal of minimizing average network power. However, we note that perfect power control is a strong assumption, and is extremely hard to achieve in practice. Indeed the problem of power control in wireless networks is an active area of research.

1.3.2 Network Topology

As mentioned earlier, there are N mobile nodes in the wireless network. We denote the i^{th} node by d_i, where naturally $1 \leq i \leq N$. If d_i sends data bits **directly** to d_j, then we say that d_i is sending bits on the link from d_i to d_j. We will use the shorthand notation $\mathcal{L}(i, j)$ to denote the link from d_i to d_j. Note that $\mathcal{L}(i, j)$ and $\mathcal{L}(j, i)$ are two distinct links.

The network topology can be described by a complete symmetric digraph. A complete symmetric digraph has both links $\mathcal{L}(i, j)$ and $\mathcal{L}(j, i)$ for every pair of nodes d_i and d_j. The digraph is complete because any node can adjust its power level until any other node falls into its transmission range. In other words, each link represents potential direct communication between the two nodes. However, if the power required for that communication is too high, then that link will not be used.

1.3.3 End-to-End Sessions

We assume that the network data traffic is in **steady state**, and that there are S end-to-end communication sessions among the N nodes. Associated with Session s ($1 \leq s \leq S$), there is a **required average end-to-end transmission rate** of r^s bits/second. Furthermore, if d_i is the origin node of Session s, and d_j is the destination node of Session s then we define:

- $O(s) = d_i$
- $D(s) = d_j$

1.3.4 The Traffic Vector

The traffic vector summarizes the required steady-state end-to-end data rates. We denote the traffic vector by $\mathbf{r} = [r^1 \, r^2 \ldots r^S]^T$, where r^s is the end-to-end data rate for Session s.

1.3.5 Multi-hop Transmissions

Consider Session s, where $O(s) = d_i$, and $D(s) = d_j$. Session s may be realized by d_i sending Session s's bits to a third node, d_k, and d_k forwarding those bits on $\mathcal{L}(k, j)$.

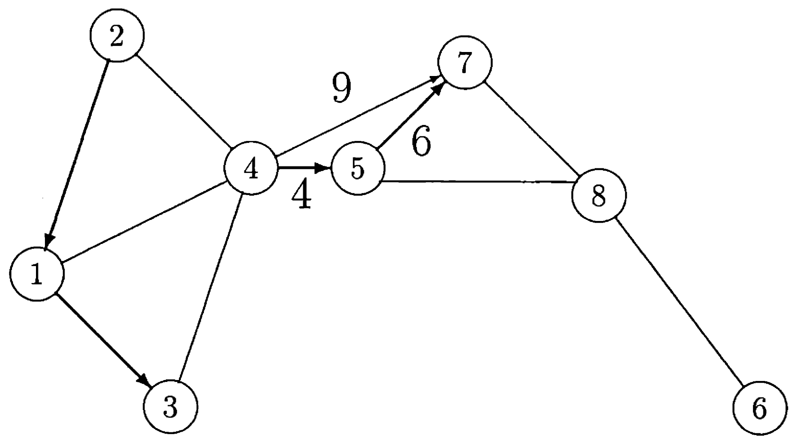

Figure 1-2: The power advantage of multi-hop routing

We refer to such a realization of the session as **multi-hop transmission**. Note that multi-hop transmissions can occur over several intermediate nodes.

Our study of multi-hop transmissions is motivated by our objective to minimize the average network power. The attenuation of radiated power is not linear in the distance; rather it is proportional to a value between the distance squared and the distance to the sixth power (see [4]). In other words, at distance λ from the source, the power of a signal is proportional to $\left(\frac{1}{\lambda}\right)^{\alpha}$, where $2 \leq \alpha \leq 6$. As a result, direct transmission may be more costly (power-wise) to the network as a whole than multi-hop transmissions. We illustrate this point using the following example: consider the PRNET in Fig. 1-2, and suppose we want to send a message from d_4 to d_7. There are two possible paths, namely $d_4 \rightarrow d_5 \rightarrow d_7$ and $d_4 \rightarrow d_7$. Let the distances between these three nodes be as indicated in the figure, and assume the power to be proportional to the square distance, with a proportionality constant k. Then the power associated with $d_4 \rightarrow d_5 \rightarrow d_7$ is $52k$, and that associated with $d_4 \rightarrow d_7$ is $81k$. Therefore in this case multi-hop routing is more efficient.

If bits are transmitted on a link, then that link is called active. In a multi-hop network, several sessions may be using a particular link as a hop. In that case, the

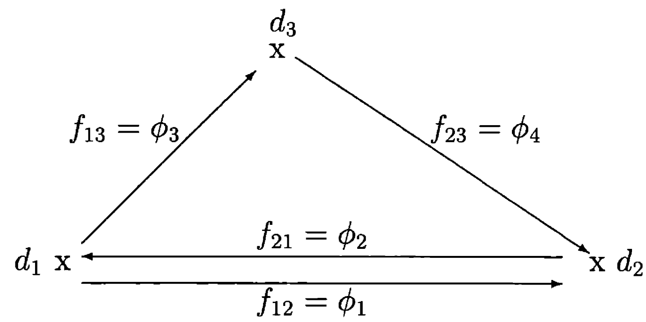

Figure 1-3: Correspondence between f_{ij}'s and ϕ_m's.

link will carry bits belonging to all those sessions. We shall subsequently refer to bits belonging to Session s as Type s bits. In the same vein, let f_{ij} be the total bit-rate on $\mathcal{L}(i,j)$, then $f_{ij} = \sum_{s=1}^{S} x_{ij}^{s}$, where x_{ij}^{s} is the bit-rate of Type s bits on $\mathcal{L}(i,j)$. In some cases it will more convenient for us to use a different notation to describe the total bit-rate on a link. That notation is ϕ_m, with $1 \leq m \leq M$, where M is the total number of active links in the network. We emphasize that there is a one-to-one correspondence between the f_{ij}'s and the ϕ_m's. Fig. 1-3 illustrates the correspondence between the f_{ij}'s and the ϕ_m's for a network with 4 active links.

1.3.6 Frames and Slots

Recall that TDMA is part of our multi-access scheme. Therefore, data is sent in successive frames. Each frame contains K time slots. In each slot a subset of the nodes transmit, and a different mutually exclusive subset of nodes receive. The same slot (i.e. with the same combination of transmitters and receivers) may occur more than once in the frame.

Figure 1-4 shows two successive and identical frames. Each frame consists of 3 slots. The length of each slot is l, and the length of a frame is L.

We use the term **frame-average** to refer to averages taken over the frame length. For example, the frame-average power of a node is the total energy radiated by that node over the length of the frame, divided by L, the length of the frame. Similarly, the term **slot-average** refers to averages taken over the slot length. In addition to

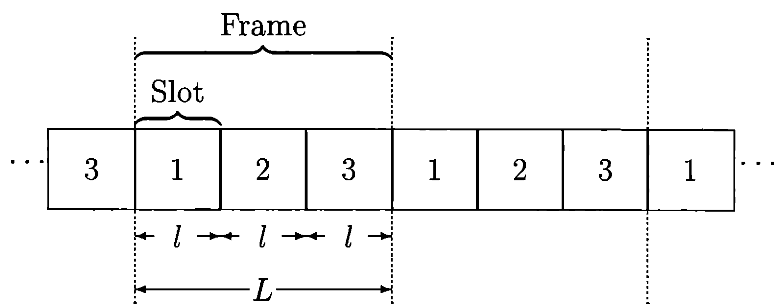

Figure 1-4: Illustration of Frames and Slots

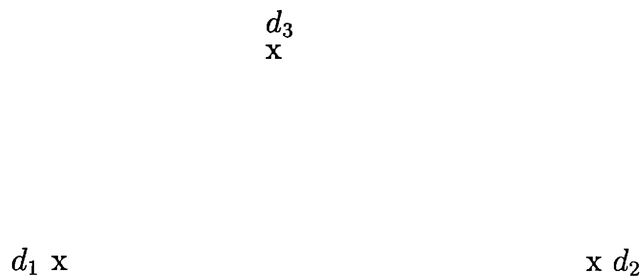

Figure 1-5: Simple Network with Three Nodes

slot-average and frame-average powers, we will also encounter the terms slot-average and frame-average bit rates.

At this point we need to modify our definition of the parameters f_{ij} and x^s_{ij} to $f_{ij}[k]$ and $x^s_{ij}[k]$ respectively. $f_{ij}[k]$ is the total slot-average bit-rate on $\mathcal{L}(i,j)$ in Slot k, and $x^s_{ij}[k]$ is the slot-average bit-rate of Type s bits on $\mathcal{L}(i,j)$ in Slot k.

1.3.7 Simple Example

In this subsection, we provide a simple example to illustrate the concepts and definitions discussed so far. Consider the simple network in Figure 1-5. The network contains three nodes d_1, d_2, and d_3. By assumption, data traffic is in steady-state, and the network uses the CDMA/TDMA multi-access scheme.

Suppose there are two ongoing end-to-end sessions in the network in Figure 1-5:

1. Session 1: $O(1) = d_1$, $D(1) = d_2$, and $r^1 = 10$ bits/second.

$d_1 \to d_2$	$d_1 \to d_2$	$d_2 \to d_3$

Figure 1-6: Frame for the Simple Network Example

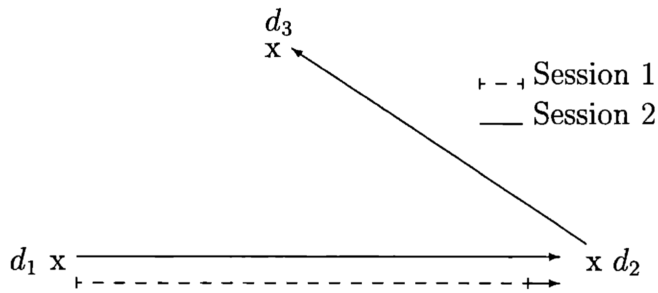

Figure 1-7: Paths of Session 1 and Session 2.

2. Session 2: $O(2) = d_1$, $D(2) = d_3$, and $r^2 = 20$ bits/second.

Let the slot length be $l = 2$ seconds. Recall that $x_{ij}^s[k]$ is the slot-average rate of Type s bits on $\mathcal{L}(i, j)$ in Slot k. Suppose we have the following transmissions in the network:

- $x_{12}^1[1] = 30$ bits/second (Session 1 bits from d_1 to d_2 in Slot 1.)
- $x_{12}^2[1] = 45$ bits/second (Relay Session 2 traffic through d_2.)
- $x_{12}^2[2] = 15$ bits/second
- $x_{23}^2[3] = 60$ bits/second (Forward Session 2 traffic to d_3.)

The frame length is $L = 3 \times 2 = 6$ seconds. Fig. 1-6 shows the active links in each slot. Note that the first slot and the second are identical from the perspective of which links are active (i.e. they have identical transmission matrices). Fig. 1-7 shows the paths of the two sessions. Session 2 is realized by multi-hopping. The frame-average bit-rate for Session 1 is the number of bits delivered in one frame divided by the frame length which is equal to $\frac{30 \times 2}{6} = 10$ bits/second. Similarly the frame-average bit-rate for Session 2 is $\frac{60 \times 2}{6} = 20$ bits/second. Therefore we clearly satisfy the required session rates r^1 and r^2, with the routing described above.

Finally, note that Session 2 is spread over several slots, i.e. $x_{12}^2[1]$, $x_{12}^2[2]$ and $x_{12}^2[3]$. We may want to spread the traffic on a link over several slots if the capacity of the channel cannot accommodate the required slot-average bit-rate, if we were to transmit in one slot.

1.4 Thesis Outline

As mentioned earlier, we want to investigate scheduling in PRNETs within the larger framework of minimum-average-network-power routing in PRNETs. Accordingly, we devote Chapter 2 to the formulation of the minimum-average-network-power problem. The problem is formulated in the form of a mathematical programming problem, which in turn is shown to be difficult to solve. In Chapter 3 we propose a suboptimal approach to the routing problem. The suboptimal approach involves the sub-problem of scheduling, to which most of Chapter 3 is devoted. Finally, we make our conclusions in Chapter 4.

Chapter 2

Initial Approach: Direct Optimization

The problem of finding minimum average network power (ANP) routing involves finding the optimal number of slots in a frame, the optimal allocation of traffic among the slots, and the optimal paths for the different sessions.

In this section we formulate the optimization problem, whose solution is the minimum ANP routing solution. More accurately, we will present two formulations of the optimization problem. The first is more intuitive than the second, but it suffers from the lack of a unique optimal solution as we will discuss in Section 2.1. The second, presented in Section 2.2, has a more cumbersome formulation, however its optimal solution will be more useful to us.

We emphasize that both formulations find the optimal routing for a snapshot (in time) of the network, which implies that the nodes are treated as stationary nodes, and the required end-to-end rates are treated as steady-state rates. It is unlikely that the solution found through the problem formulations in this section will be used in real life networks. First the formulations assume global knowledge of the network, and centralized control. In addition the problem formulated in this section is so complex that is unlikely it can be solved in real time.

What we hope to get out of the solution is insight into what constitutes optimal routing. Hopefully we can use that insight to construct fast good heuristic methods.

2.1 Mathematical Formulation 1

The direct objective in this formulation is to find the routing resulting in the lowest frame-average network power. This is done by finding the optimal frame-average powers for different frame lengths (in terms of the number of slots in the frame), and then finding the overall optimal solution among them.

We start by defining the parameters in this formulation. Let

- K be the number of slots in the frame.
- l be the length of a slot.
- $x_{ij}^{s}[k]$ be the slot-average bit-rate of Type s bits transmitted on $\mathcal{L}(i,j)$ in Slot k.
- $\mathbf{F}[k] = [f_{ij}[k]]$, where $f_{ij}[k]$ is the total slot-average bit-rate of bits transmitted on $\mathcal{L}(i,j)$ in Slot k.
- $\mathbf{P}[k] = [P_i[k]]$, where $P_i[k]$ is the slot-average power radiated by d_i in Slot k.
- r^s be the required average bit-rate for Session s.
- $y_i[k]$ be a binary logical variable, used as a tool (in conjunction with M below) to ensure no simultaneous reception and transmission by the same node. It is equal to 1 if d_i is transmitting in Slot k and 0 otherwise.
- M be a number larger than any feasible rate of transmission.
- $\mathcal{Q}_1(.)$ be a function mapping the matrix $\mathbf{F}[k]$ to the vector $\mathbf{P}[k]$, in such a way that the sum of the powers in the vector $\mathbf{P}[k]$ is minimized, while maintaining acceptably small error probabilities in the transmission of $\mathbf{F}[k]$.

The optimization problem is formulated as follows:

$$\underbrace{\min_{K\in\mathcal{Z}^+}}_{\text{minimize over pos. integers}} \quad (\min \frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{N} P_i[k]) \tag{2.1}$$

such that, for i,$j = 1,\ldots,N$; $k = 1,\ldots,K$; and $s = 1,\ldots,S$

$$x^s_{ij}[k] \begin{cases} \geq & 0, i \neq j \\ = & 0, i = j \quad \text{No self} - \text{transmission} \end{cases} \tag{2.2}$$

and such that, for i,$j = 1,\ldots,N$; and $k = 1,\ldots,K$

$$f_{ij}[k] \quad = \quad \sum_{s=1}^{S} x^s_{ij}[k] \tag{2.3}$$

and such that, for $i = 1,\ldots,N$; and $k = 1,\ldots,K$

$$\left.\begin{array}{lcl} \sum_{j=1}^{N} f_{ij}[k] & \leq & My_i[k] \\ \sum_{j=1}^{N} f_{ji}[k] & \leq & M(1 - y_i[k]) \end{array}\right\} \text{(No simultaneous transmit/receive)} \tag{2.4}$$

$$y_i[k] \in \{0, 1\} \tag{2.5}$$

and such that, for $k = 1,\ldots,K$,

$$\mathbf{P}[k] \quad = \quad \mathcal{Q}_1(\mathbf{F}[k]) \tag{2.6}$$

and such that, for $i = 1,\ldots,N$; and $s = 1,\ldots,S$

$$\sum_{k=1}^{K} (\underbrace{\sum_{j=1}^{N} x^s_{ij}[k]}_{\text{Flow out}} - \underbrace{\sum_{j=1}^{N} x^s_{ji}[k]}_{\text{Flow in}}) \quad = \quad \begin{cases} Kr^s \text{ if } d_i = O(s) \\ -Kr^s \text{ if } d_i = D(s) \\ 0 \text{ otherwise} \end{cases} \tag{2.7}$$

2.1.1 Lack of a Unique Optimal Solution

The above mathematical programming problem does not have a unique optimal solution. Suppose we find that K^* is the length of a frame (in slots) that achieves minimum ANP routing. Then we can construct a frame of length $2K^*$, that also achieves minimum ANP routing. This is done by duplicating each slot in the original frame (the one of length K^*), to create the new frame of length $2K^*$. Indeed for every multiple of K^*, there is an associated frame with that length, that also achieves minimum ANP routing.

We need to find another formulation to the optimization problem, because the solution to this optimization problem does not provide much insight.

2.2 Mathematical Formulation 2

Unfortunately, the formulation of the new optimization problem presented in this section, requires us to temporarily depart from some of the ideas that we thus far established.

First, in this formulation, we no longer restrict the slots to have a fixed length. On the other hand, a slot may only appear once in a frame. In other words, a frame may not contain two slots with the same combination of transmitters and receivers. Finally, we fix the frame length, L, to be equal to 1 second, and require it to include all $q(N)$ different slots (see Eq. 3.7). $q(N)$ represents the total number of possible transmitter/receiver combinations in a network with N nodes.

The optimal solution therefore will consist of a frame with $q(N)$ slots. Slot k will have length $l^*[k]$. Note that the optimal length of a slot might be 0. Since the length of the frame is fixed to be 1, $l^*[k]$ represents the fraction of time that slot occupies in the frame.

When we revert to the scheme with equal-length slots, and variable-length frame, we allow multiple occurrences of the same slot in the frame. The number of identical slots divided by the total number of slots in a frame is then equal to the fraction given by the optimal solution to the optimization problem presented here.

As with the previous section, we start by defining the parameters in the optimization problem. Let

- K be the total number of slots in the frame. K is equal to the total number of different combinations of transmitters and receivers (i.e. $q(N)$).
- $l[k]$ $(1 \leq k \leq K)$ be the length of Slot k. $0 \leq l[k] \leq 1$, and $\sum_{k=1}^{K} l[k] = L = 1$.

- $x_{ij}^s[k]$ be the slot-average rate (bits/sec) of transmission of bits of Type s on $\mathcal{L}(i,j)$ in Slot k.

- $\mathbf{F}[k] = f_{ij}[k]$, where $f_{ij}[k]$ is the slot-average rate (bits/sec) of transmission of bits of all types on $\mathcal{L}(i,j)$ in Slot k.

- $\mathbf{P}[k] = [P_i[k]]$, where $P_i[k]$ is the slot-average power radiated by d_i in Slot k.

- $P_{\text{net}}[k]$ is the total slot-average power radiated by the network in Slot k. $P_{\text{net}}[k] = \sum_{i=1}^{N} P_i[k]$.

- $\mathcal{Q}_1(.)$ be a function mapping the matrix $\mathbf{F}[k]$ to the vector $\mathbf{P}[k]$, in such a way that the sum of the powers in the vector $\mathbf{P}[k]$ is minimized, while maintaining acceptably small error probabilities in the transmission of $\mathbf{F}[k]$.

The problem can now be formulated as:

$$\min \sum_{k=1}^{K} \sum_{i=1}^{N} P_i[k] l[k] = \sum_{k=1}^{K} P_{\text{net}}[k] l[k] \tag{2.8}$$

such that,

$$\sum_{k=1}^{K} l[k] = 1 \tag{2.9}$$

and such that, for i, $j = 1,\ldots,N$; $k = 1,\ldots,K$; and $s = 1,\ldots,S$,

$$x_{ij}^s[k] \begin{cases} \geq & 0, \text{ if } a_{ij}[k] = 1 \\ = & 0, \text{ if } a_{ij}[k] = 0 \end{cases} \tag{2.10}$$

and such that, for i, $j = 1,\ldots,N$ and $k = 1,\ldots,K$

$$f_{ij}[k] = \sum_{s=1}^{S} x_{ij}^s[k] \tag{2.11}$$

and such that, for $k = 1,\ldots,K$,

$$\mathbf{P}[k] = \mathcal{Q}_1(\mathbf{F}[k]) \tag{2.12}$$

$$l[k] \geq 0 \tag{2.13}$$

and such that, for $i = 1,\ldots,N$; and $s = 1,\ldots,S$

$$\underbrace{\sum_{k=1}^{K}\sum_{j=1}^{N} x_{ij}^{s}[k]l[k]}_{\text{Flow out}} - \underbrace{\sum_{k=1}^{K}\sum_{j=1}^{N} x_{ji}^{s}[k]l[k]}_{\text{Flow in}} = \begin{cases} r^s \text{ if } d_i = O(s) \\ -r^s \text{ if } d_i = D(s) \\ 0 \text{ otherwise} \end{cases} \tag{2.14}$$

2.2.1 Solving Mathematical Formulation 2

The above mathematical formulation is a multi-commodity flow problem. Solving it would be a difficult task for a number of reasons. The number of variables is proportional to the number of different combinations of transmitters and receivers. As we will see in the next chapter this number grows exponentially with the number of nodes in the network, rendering the optimization problem hard to solve by virtue of the number of variables involved. In addition, the function $\mathcal{Q}_1(\mathbf{F}[k])$ is not known. Recall that $\mathcal{Q}_1$ finds the required average-powers of the nodes to achieve the rates in $\mathbf{F}[k]$, minimizing the sum of those powers while maintaining acceptably low error probabilities. This is an unsolved problem in information theory (The Interference Channel Problem). In practice (e.g. spread spectrum cellular networks), a suboptimal solution is implemented. The suboptimal solution insures that the signal-to-interference ratio for each received signal is above a certain ratio (see [2]). Not knowing $\mathcal{Q}_1(\mathbf{F}[k])$ is problematic for us, since if we are to have any hope of solving the optimization problem above (optimally), we will need that function.

2.3 Convexity of Power

Even though we do not have an explicit expression for $\mathcal{Q}_1(\mathbf{F}[k])$, we show in this section that it is at least convex. We start by giving the definitions for convex sets and convex functions.

Definition 1 *A set* $\mathbf{X}$ *in* $\mathcal{R}^n$ *is said to be convex if for each* $\mathbf{x}_1$, $\mathbf{x}_2 \in \mathbf{X}$, *and for each* $\beta \in [0,1]$, $\beta\mathbf{x}_1 + (1-\beta)\mathbf{x}_2 \in \mathbf{X}$.

Definition 2 *Let* $f : \mathbf{X} \rightarrow \mathcal{R}$*, where* $\mathbf{X}$ *is a nonempty convex set in* $\mathcal{R}^n$*. The function* f *is said to be convex on* $\mathbf{X}$ *if for each* $\mathbf{x}_1, \mathbf{x}_2 \in \mathbf{X}$*, and for each* $\beta \in (0,1)$
$f(\beta\mathbf{x}_1 + (1-\beta)\mathbf{x}_2) \leq \beta f(\mathbf{x}_1) + (1-\beta) f(\mathbf{x}_2)$

Let $\underline{\phi}$ be a vector in $\mathcal{R}^M$, where M is the number of active links in the network. The m^{th} element of $\underline{\phi}$ is the average bit-rate along the m^{th} active link. That is, if link m goes from d_i to d_j, then $\phi_m = f_{ij}$. Here, the issue of slots is not relevant since we are simply studying the relationship between flow and power for a given set of active links. A vector $\underline{\phi}$ is achievable, if there exists a transmission strategy such that the rates indicated in the vector can be achieved with arbitrarily small probabilities of error.

Theorem 1 *Let* $\mathbf{\Phi}$*, a set in* $\mathcal{R}^M$*, be the set of achievable network rate vectors.* $\mathbf{\Phi}$ *is a convex set.*

Proof: Suppose we have two achievable rate vectors $\underline{\phi}_1$ and $\underline{\phi}_2$. Consider a time interval of length T. For any $\beta \in [0,1]$, let $t_1 = \beta T$. If for the first t_1 seconds the network transmits at network rate vector $\underline{\phi}_1$ and for the last $T - t_1$ seconds at rate vector $\underline{\phi}_2$,[1] then the following rate vector $\frac{t_1\underline{\phi}_1 + (T-t_1)\underline{\phi}_2}{T} = \beta\underline{\phi}_1 + (1-\beta)\underline{\phi}_2$ is achieved. Therefore $\beta\underline{\phi}_1 + (1-\beta)\underline{\phi}_2 \in \mathbf{\Phi}$. This implies that for each $\underline{\phi}_1, \underline{\phi}_2 \in \mathbf{\Phi}$, and for each $\beta \in [0,1]$, $\beta\underline{\phi}_1 + (1-\beta)\underline{\phi}_2 \in \mathbf{\Phi}$. Therefore, $\mathbf{\Phi}$ is a convex set.

Theorem 2 *Let* $\mathbf{\Phi}$ *be a nonempty convex set in* $\mathcal{R}^M$*. Let* $P_{\text{net}}(\underline{\phi})$ *be the minimum network power required to achieve* $\underline{\phi}$*, where* $\underline{\phi} \in \mathbf{\Phi}$*.* $P_{\text{net}}(\underline{\phi})$ *is convex.*

Proof: For each pair $\underline{\phi}_1, \underline{\phi}_2 \in \mathbf{\Phi}$ and for each $\beta \in (0,1)$, let $\underline{\phi}_0 = \beta\underline{\phi}_1 + (1-\beta)\underline{\phi}_2$. Now consider a time interval of length T seconds long. Let $t_1 = \beta T$. If we transmit at network rate vector $\underline{\phi}_1$ for the first t_1 seconds, and at network rate vector $\underline{\phi}_2$ for the last $T - t_1$ seconds,[2] then the average network rate vector will be $\frac{t_1\underline{\phi}_1 + (T-t_1)\underline{\phi}_2}{T} = \beta\underline{\phi}_1 + (1-\beta)\underline{\phi}_2 = \underline{\phi}_0$. Furthermore, the average power over the interval of length T will be

[1] We assume that t_1 and $T - t_1$ are sufficiently long to achieve those rate vectors.
[2] Again we assume that t_1 and $T - t_1$ are sufficiently long to achieve those rate vectors.

$\frac{t_1 P_{\text{net}}(\underline{\phi}_1)+(T-t_1)P_{\text{net}}(\underline{\phi}_2)}{T} = \beta P_{\text{net}}(\underline{\phi}_1)+(1-\beta)P_{\text{net}}(\underline{\phi}_2)$. This implies that the network can achieve the network rate vector $\underline{\phi}_0$ with average power of $\beta P_{\text{net}}(\underline{\phi}_1)+(1-\beta)P_{\text{net}}(\underline{\phi}_2)$. By definition $P_{\text{net}}(\underline{\phi})$ is the minimum average power required to achieve the network rate vector $\underline{\phi}$. This implies that $P_{\text{net}}(\underline{\phi}_0) \leq \beta P_{\text{net}}(\underline{\phi}_1) + (1-\beta)P_{\text{net}}(\underline{\phi}_2)$ Therefore, $P_{\text{net}}(\underline{\phi})$ is convex.

2.4 Conclusion

In this chapter, we investigated the possibility of solving the minimum ANP routing problem using direct optimization. However it seems that this path is a blind alley. One of the main problems is the exponential growth in the number of variables. This growth comes about from the enumeration of all possible scheduling combinations in a PRNET. The other main problem is that we cannot express network power as a function of the link flows.

What we learn from mathematical programming formulations in this chapter is that the solution has two components: scheduling and routing. The routing component dictates the path(s) the data follows, and the scheduling component determines the number of slots in a frame, and the transmit/receive configuration in each slot. We note that the scheduling and routing components depend on each other (in other words they are intertwined).

In the next chapter we consider a suboptimal approach, in which routing and scheduling are assumed to be independent, with the hope that optimality is sacrificed for insight.

Chapter 3

Scheduling

As mentioned at the end the last chapter, we are considering a suboptimal approach in which we separate the routing component from the scheduling component in the hope to gain insight into the dynamics of the problem. The main focus of this chapter is the scheduling component, but first we briefly discuss the routing component.

3.1 The Routing Component

Consider the network in Figure 3-1. We assume that somehow a "length" is associated with each arc. We envision this "length" to be an empirical function of the power used on that link, the average interference experienced by the receiving node, the congestion on that link and possibly other factors. The task becomes that of finding the shortest path (for a source-destination pair of nodes) in terms of those lengths. The shortest path problem is well-understood, and indeed the distributed Bellman-Ford algorithm can be applied (provided arc lengths not change too rapidly), which is very useful since we have a distributed network.

3.2 The Scheduling Component

Suppose the solution to the routing component dictates that the links indicated in Figure 3-2 must be active within the span of one frame (without considering the

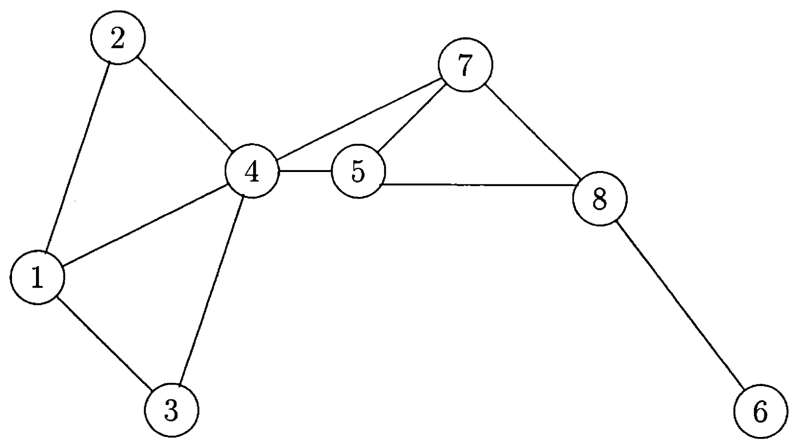

Figure 3-1: Routing in a Packet Radio Network

simultaneous transmit/receive constraint). Generally, some links need to be active more frequently than others, but assume for now that all links are equally active. The task becomes that of scheduling transmissions such that no node is transmitting and receiving in the same slot. One feasible frame consists of the four slots shown in Fig. 3-3, Fig. 3-4, Fig. 3-5 and Fig. 3-6.

3.2.1 The Transmission Matrix

Each time slot k in the CDMA/TDMA scheme has an associated transmission matrix. We denote the k^{th} such transmission matrix by the 0-1 matrix $\mathbf{A}[k] = [a_{ij}[k]]$, where $a_{ij}[k]$ is 1 if d_i is transmitting to d_j in the k^{th} slot, and 0 otherwise.

Suppose, for example, we have a network of three nodes (d_1, d_2, and d_3), where in the first slot d_1 transmits to both d_2 and d_3, and in the second slot d_2 and d_3 both transmit to d_1. Then,

$$\mathbf{A}[1] = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.1}$$

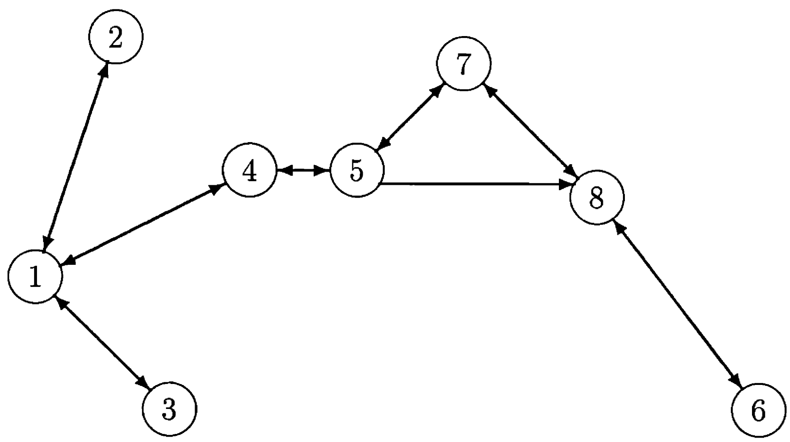

Figure 3-2: Active links in a frame as dictated by routing component.

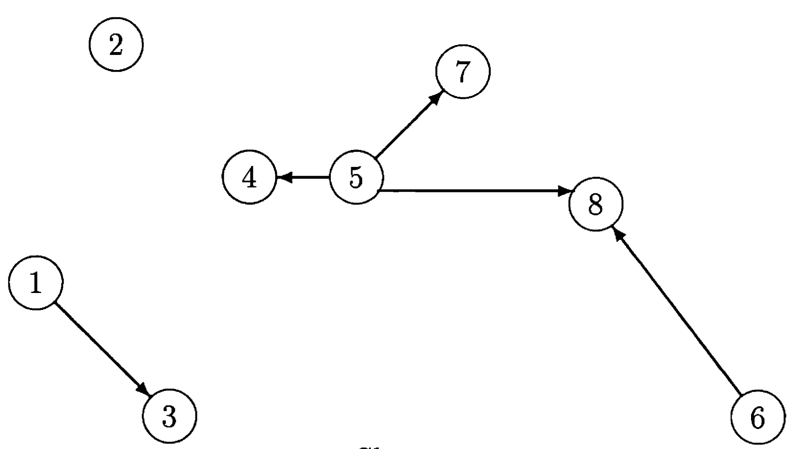

Figure 3-3: Active links in Slot 1.

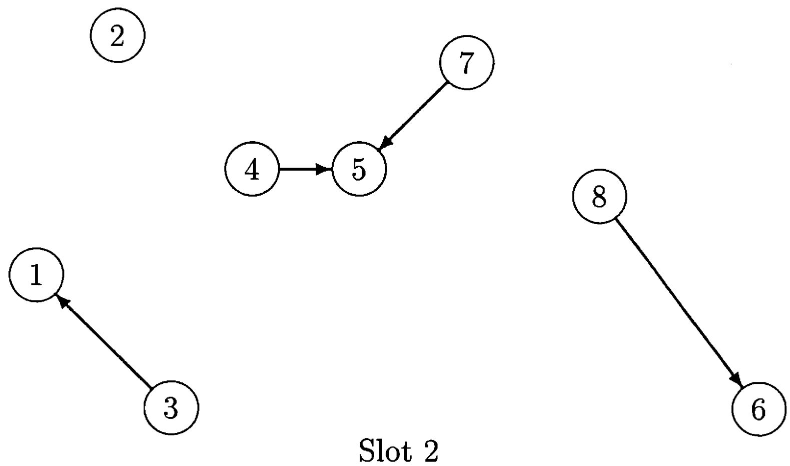

Figure 3-4: Active links in Slot 2.

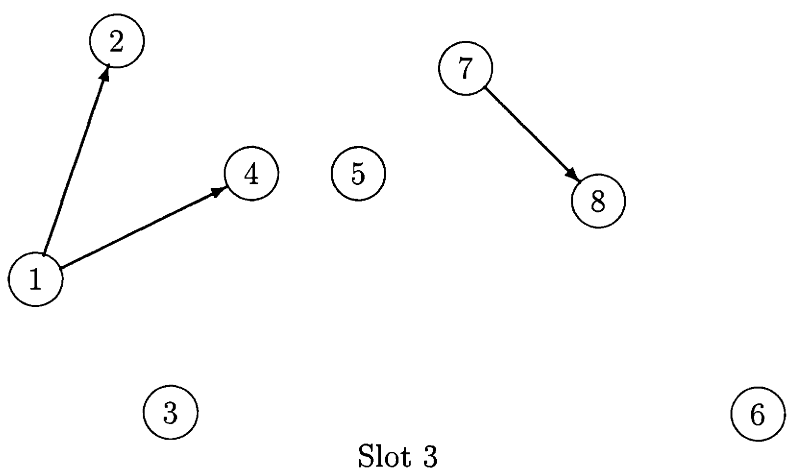

Figure 3-5: Active links in Slot 3.

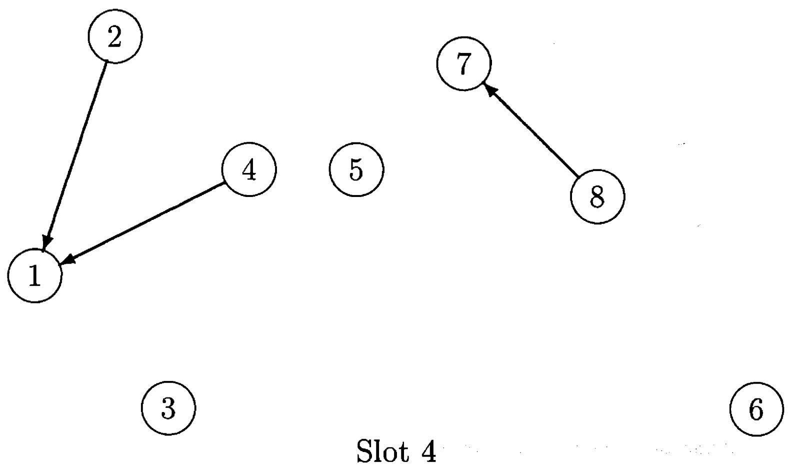

Figure 3-6: Active links in Slot 4.

and

$$\mathbf{A}[2] = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{3.2}$$

Not every 0-1 matrix is an admissible transmission matrix. For example the following matrix

$$\mathbf{A}[k] = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.3}$$

is not admissible because $a_{11}[k] = 1$ implying d_1 is transmitting to itself, which is nonsensical. Furthermore, according to this matrix, d_2 is receiving and transmitting simultaneously, which is not possible. Below we give the definition of an admissible transmission matrix:

Definition 3 *Let* $\mathbf{A}[k]$ *be an* $N \times N$ 0-1 *matrix.* $\mathbf{A}[k]$ *is an* **admissible transmission matrix** *if and only if in the transmission pattern described by* $\mathbf{A}[k]$*,*

1. *There is at least one transmission. Equivalently,* $\mathbf{A}[k] \neq \mathbf{0}$.

2. *No node is both transmitting and receiving.*

3. No node is transmitting to itself. Equivalently, $a_{ii} = 0$ *for* $1 \leq i \leq N$.

The theorem below provides an easy test for the admissibility of a transmission matrix:

Theorem 3 *Let* $\mathbf{A}[k]$ *be an* $N \times N$ 0-1 *matrix.* $\mathbf{A}[k]$ *is an* **admissible transmission matrix** *if and only if* $\mathbf{A}[k]\mathbf{A}[k] = \mathbf{0}$ *and* $\mathbf{A}[k] \neq \mathbf{0}$.

Proof: If $\mathbf{A}[k]$ is an admissible transmission matrix, then $\mathbf{A}[k] \neq \mathbf{0}$. Further, let $\mathbf{v}_r[k]$ be a row vector whose i^{th} element is the sum of the entries in the i^{th} column of $\mathbf{A}[k]$. If the i^{th} element of $\mathbf{v}_r[k]$ is nonzero, then d_i is receiving in Slot k. Similarly, let $\mathbf{v}_c[k]$ be a column vector, whose i^{th} element is the sum of the entries in the i^{th} row of $\mathbf{A}[k]$. If the i^{th} element of $\mathbf{v}_c[k]$ is nonzero, then d_i is transmitting in Slot k. If $\mathbf{A}[k]$ is an admissible transmission matrix, then the i^{th} element of $\mathbf{v}_c[k]$ and the i^{th} element of $\mathbf{v}_r[k]$ cannot both be nonzero. This implies that $\mathbf{v}_r[k]\mathbf{v}_c[k] = 0$. This can be rewritten as:

$$(\mathbf{u}^{\mathrm{T}}\mathbf{A}[k])(\mathbf{A}[k]\mathbf{u}) = 0 \tag{3.4}$$

where $\mathbf{u} = [1\,1 \ldots 1\,1]^{\mathrm{T}}$.

$$\Rightarrow \mathbf{u}^{\mathrm{T}}(\mathbf{A}[k]\mathbf{A}[k])\mathbf{u} = 0 \tag{3.5}$$

$$\Rightarrow \mathbf{A}[k]\mathbf{A}[k] = \mathbf{0} \tag{3.6}$$

which follows since $\mathbf{A}[k]$ is non-negative. Therefore, if $\mathbf{A}[k]$ is an admissible transmission matrix, then $\mathbf{A}[k]\mathbf{A}[k] = \mathbf{0}$, and $\mathbf{A}[k] \neq \mathbf{0}$.

Now, if $\mathbf{A}[k] \neq \mathbf{0}$ then the first condition in Definition 3 is directly satisfied. Further, if $\mathbf{A}[k]\mathbf{A}[k] = \mathbf{0}$ then every row in $\mathbf{A}[k]$ is orthogonal to every column. In particular the product of the i^{th} row and the i^{th} column is 0, for $1 \leq i \leq N$. This implies that a_{ii}^2+ other nonnegative terms $= 0$. Therefore $a_{ii} = 0$, for $1 \leq i \leq N$, thus satisfying the third condition in Definition 3. Finally, if $\mathbf{A}[k]\mathbf{A}[k] = \mathbf{0}$ then $\mathbf{v}_r[k]\mathbf{v}_c[k] = 0$. This in turn implies that no node is both transmitting and receiving in Slot k, which satisfies the second condition in Definition 3.

Q.E.D.

N	$q(N)$
2	2
3	12
4	86
5	840
6	11642
7	227892
$\vdots$	$\vdots$

Table 3.1: Growth of the number of admissible matrices with N.

It is of interest to know the number of distinct admissible transmission matrices there are for a network of N nodes (see Subsection 2.2). Suppose there are i active receivers. Then we have at least 1 and at most $(N-i)$ active transmitters among the remaining nodes. For each of the active receivers, there are $(2^{(N-i)}-1)$ possible combinations of transmitters sending to that receiver. In other words, for each active receiver there are $(2^{(N-i)}-1)$ possible ways of being an active receiver, which implies that for the i active receivers there are $(2^{(N-i)}-1)^i$ ways. There are $\begin{pmatrix} N \\ i \end{pmatrix}$ ways of choosing i active receivers among N nodes. Therefore the total number of admissible transmission matrices for an N-node network, $q(N)$, is given by

$$q(N) = \sum_{i=1}^{N-1} \begin{pmatrix} N \\ i \end{pmatrix} [2^{(N-i)} - 1]^i \qquad (3.7)$$

Table 3.1 shows how the number of admissible matrices grows with N, the number of nodes in the network.

For a network with 3 nodes, d_1, d_2 and d_3, the number of distinct admissible matrices is $q(3) = 3 \times 3 + 3 \times 1 = 12$ (see Eq. 3.7). These matrices are:

1. $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_1 \rightarrow d_2\}$.

2. $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_1 \to d_3\}$.

3. $\begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_1 \to d_2, d_1 \to d_3\}$.

4. $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_1 \to d_3, d_2 \to d_3\}$.

5. $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ which corresponds to $\{d_1 \to d_2, d_3 \to d_2\}$.

6. $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_2 \to d_1\}$.

7. $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_2 \to d_3\}$.

8. $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_2 \to d_1, d_2 \to d_3\}$.

9. $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_2 \to d_1, d_3 \to d_1\}$.

10. $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$ which corresponds to $\{d_3 \to d_1\}$.

11. $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ which corresponds to $\{d_3 \to d_2\}$.

12. $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$ which corresponds to $\{d_3 \to d_1, d_3 \to d_2\}$.

Note that the transmission matrix

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

is a superset (i.e. contains) the two matrices

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

and

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

This is significant, because in the mathematical formulation in Section 2.2 we argued that the number of slots in the frame is equal to the total number of combinations of transmitters and receivers, which is exactly equal to the number of admissible transmission matrices. An alternative approach would be to set the number of slots in the frame equal to the number of supersets. Then within each slot, the algorithm would be allowed to divide the slot in mini-slots, where each mini-slot is characterized by a subset of the superset. Dividing a slot into mini-slots would occur if for example power is minimized in that slot by a time-sharing transmission scheme. The number of supersets in a network of N nodes is

$$p(N) = \sum_{i=1}^{N-1} \binom{N}{i} \tag{3.8}$$

$p(N)$ still grows exponentially, but not as rapidly as $q(N)$.

3.2.2 Minimal Decomposition

Revisiting the problem of scheduling the transmissions in Figure 3-2, we see that the active links in that network correspond to the following (non-admissible) transmission matrix:

$$\mathbf{S} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

We implemented those active links over four slots. Algebraically, what we did was decompose $\mathbf{S}$ into four admissible transmission matrices:

$$\begin{bmatrix} 0&1&1&1&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 1&0&0&0&1&0&0&0\\ 0&0&0&1&0&0&1&1\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&1&0&0&1\\ 0&0&0&0&0&1&1&0 \end{bmatrix} = \begin{bmatrix} 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&1&1\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix} + \begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0 \end{bmatrix}$$

$$+ \begin{bmatrix} 0&1&0&1&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0 \end{bmatrix} + \begin{bmatrix} 0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0 \end{bmatrix}$$

It turns out we can decompose $\mathbf{S}$ into a smaller number of admissible transmission matrices:

$$\begin{bmatrix} 0&1&1&1&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 1&0&0&0&1&0&0&0\\ 0&0&0&1&0&0&1&1\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&1&0&0&1\\ 0&0&0&0&0&1&1&0 \end{bmatrix} = \begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&1&0&0&1\\ 0&0&0&0&0&0&0&0 \end{bmatrix} + \begin{bmatrix} 0&1&1&1&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&1&1\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}$$

$$
+ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}
$$

It is of interest to decompose **S** into the smallest number of admissible transmission matrices possible. The smaller the number of matrices we decompose into, the shorter the frame length, as each admissible transmission matrix is associated with a slot in the frame. Typically, the shorter the frame length the smaller and the more controllable the delay of an end-to-end session.

Unfortunately, the problem of minimal decomposition appears to be an NP-Hard problem, as we show in the next subsection.

3.2.3 Proving NP-Hardness of Minimal Decomposition

A coloring of a (undirected) graph G is an assignment of colors to the vertices of the graph, such that no two vertices joined by an arc have the same color. Two non-adjacent vertices (i.e. not linked by an arc) may have the same color. The minimum number of colors required to color the graph is called the chromatic number of the graph and is denoted by $\mathcal{X}(G)$.

It is known that finding the chromatic number for an undirected graph G is an NP-Hard problem (see [9]). In [8], an extension to that result is given in the following theorem:

Theorem 4 *Unless NP $\subseteq$ ZPP*[1]*, it is intractable*[2] *to approximate* $\mathcal{X}(G)$ *to within*

[1] ZPP denotes the class of languages decidable by a random expected polynomial-time algorithm that makes no errors.

[2] According to [9], a problem is intractable "it is so hard that no polynomial time algorithm can possibly solve it."

$N^{1-\epsilon}$ *for any constant* $\epsilon > 0$.[3]

where N is the number of nodes in the graph.

We shall refer to the problem of approximating the chromatic number to within $N^{1-\epsilon}$ for any constant $\epsilon > 0$ as the Chromatic Approximation problem. The Chromatic Approximation problem is shown to be NP-Hard in [8][4]. We note that traditionally an NP-Hard problem is said to be solvable in polynomial time unless P = NP. In [8] this condition is modified slightly to "unless NP $\subseteq$ ZPP".

The problem of Minimal Decomposition is shown to be NP-Hard through the traditional method of transforming a known NP-Hard problem (in this case, the Chromatic Approximation problem) to the problem of interest (in this case, the Minimal Decomposition problem). This transformation is done in the following manner: let G be an undirected graph, whose chromatic number we want to approximate to within $N^{1-\epsilon}$. We transform G into a directed graph G' by replacing each undirected arc with two directed edges (in opposite directions) as shown in Figure 3-7. Note that this transformation can be done in polynomial time.

Next we show how the solutions to Chromatic Approximation and Minimal Decomposition are related. Given $\mathcal{X}(G)$, we can find a feasible decomposition for G': let the colors of G be $1, 2, \ldots, \mathcal{X}(G)$. Now we label every node in G' by the binary representation of its color in G. The number of digits of each label is $\lceil \log_2 \mathcal{X}(G) \rceil$. We can construct a schedule of length $2\lceil \log_2 \mathcal{X}(G) \rceil$, whereby in the first slot, any node whose label's first digit is 0 may transmit to any node whose label's first digit is 1. In the second slot, we reverse the direction and allow nodes with first digit 1 to transmit to nodes with first digit 0. In the third and fourth slot we repeat the process for the second digit in the labels. We continue this process for all digits. Because the binary labels are derived from the node colors (in G), any two adjacent nodes will differ in at least one digit of their respective binary labels. Therefore, each directed arc in G' can be placed in at least one slot of the $2\lceil \log_2 \mathcal{X}(G) \rceil$ slots.

[3] "An algorithm approximates a function $f(x)$ within a ratio $q(x)$, if on any input x the algorithm outputs two numbers $l(x)$ and $u(x)$ such that $l(x) \leq f(x) \leq u(x)$, and $u(x)/l(x) \leq q(x)$" ([8]).

[4] Although Theorem 4 does not specifically classify the Chromatic Approximation problem as an NP-Hard problem, this classification is stated elsewhere in [8]

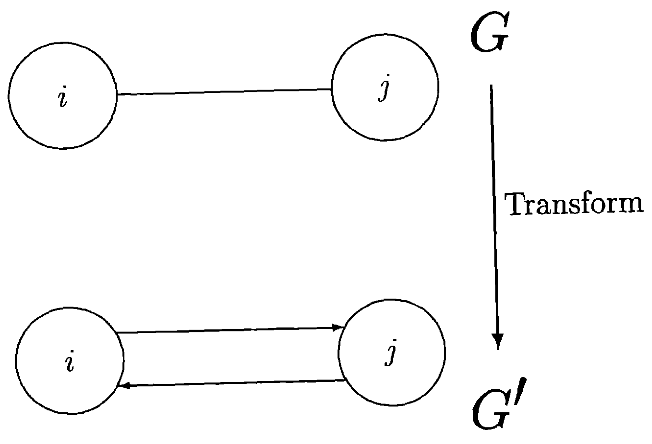

Figure 3-7: Transforming an undirected edge to two directed edges.

Let K^* the number of slots in the minimal decomposition of the transformed graph G'. Since $2\lceil\log_2 \mathcal{X}(G)\rceil$ is the number of slots in a feasible decomposition,

$$\begin{aligned} K^* &\leq 2\lceil\log_2 \mathcal{X}(G)\rceil \\ &\leq 2(\log_2 \mathcal{X}(G) + 1) \end{aligned} \tag{3.9}$$

We can also use the solution to minimal decomposition to upper-bound the chromatic number of G: given G, transform it to G' and find the latter's minimal decomposition. The minimal decomposition consists of K^* slots. Now, we label each node with a K^*-digit binary label. If a node is transmitting in the k^{th} slot (where $1 \leq k \leq K^*$), then we set the k^{th} digit of its label equal to 1, otherwise the k^{th} digit is set equal to 0. Since each node transmits to each of its adjacent nodes in exactly one slot, adjacent nodes will have different labels. Thus we have found a coloring for G with 2^{K^*} colors. This implies,

$$\mathcal{X}(G) \leq 2^{K^*} \tag{3.10}$$

From Eq. 3.9 and Eq. 3.10 we get

$$2^{K^*/2-1} \le \mathcal{X}(G) \le 2^{K^*} \tag{3.11}$$

Since the chromatic number cannot exceed the number of nodes in the graph, we can modify Eq. 3.11 to

$$2^{K^*/2-1} \le \mathcal{X}(G) \le \min(2^{K^*}, N) \tag{3.12}$$

where N is the total number of nodes in G. If $N \ge 2^{K^*}$, then $\mathcal{X}(G)$ will be approximated to within

$$\begin{aligned} \frac{2^{K^*}}{2^{(K^*/2)-1}} &= 2 \cdot 2^{K^*/2} \\ &\le 2\sqrt{N} \end{aligned}$$

If $N < 2^{K^*}$ then $\mathcal{X}(G)$ will be approximated to within

$$\begin{aligned} \frac{N}{2^{K^*/2-1}} &= \frac{2N}{2^{K^*/2}} \\ &< 2\sqrt{N} \end{aligned}$$

Therefore, using the minimal decomposition solution we can approximate $\mathcal{X}(G)$ to within $2\sqrt{N}$. In order to use Theorem 4 we need to get the ratio in the form of $N^{1-\epsilon}$. We note that for $N \ge 5$ and $\epsilon = 0.01$, $2\sqrt{N} \le N^{1-\epsilon}$. Therefore, using the minimal decomposition solution we can approximate $\mathcal{X}(G)$ to within $N^{1-\epsilon}$, where $\epsilon = 0.01$. [5]

To summarize, we have shown that the problem of Chromatic Approximation[6] can be solved by reducing it (in polynomial time) to the Minimal Decomposition problem, and solving the latter problem. The solution to Minimal Decomposition corresponds

[5] For graphs with 4 or fewer nodes, we solve for $\mathcal{X}(G)$ exactly, by enumerating of all possible solutions. This gives us an approximation ratio of 1 which is still less than $N^{1-\epsilon}$. The time required to enumerate all possible solutions for graphs with 4 or fewer nodes is bounded by a constant.

[6] More accurately, Chromatic Approximation with $\epsilon = 0.01$

to a solution to Chromatic Approximation by virtue of the following inequalities,

$$2^{K^*/2-1} \leq \mathcal{X}(G) \leq \min(2^{K^*}, N)$$

and

$$\frac{\min(2^{K^*}, N)}{2^{K^*/2-1}} \leq N^{1-\epsilon}$$

where in this case $\epsilon = 0.01$. Since the NP-Hard problem of Chromatic Approximation (with $\epsilon = 0.01$) reduces to the Minimal Decomposition problem in polynomial time, we conclude that Minimal Decomposition is also NP-Hard.[7]

3.2.4 Evaluation of Suboptimal Approach

We resorted to the suboptimal approach of decoupling scheduling from routing because direct optimization (i.e. solving the mathematical programming problem in Chapter 2) is extremely difficult to solve. The hope was that the suboptimal approach can provide some insight into the dynamics of the solution, and/or help develop fast heuristic methods for solving the routing problem. Unfortunately, the minimal decomposition problem is NP-Hard, implying that it is intractable to find a the optimal solution. Another shortcoming of the suboptimal approach is the assumption that the routing component can ignore scheduling, when in all likelihood, the router depends heavily on the schedule in order to determine the arc "lengths" mentioned in Section 3.1.

An alternative approach is to apply a non-optimal pre-determined schedule to the network. The routing algorithm is then restricted to consider only the paths that are permitted by the pre-determined schedule. In the next two subsections, we investigate two different schemes for pre-determined scheduling.

[7] Another way of looking at this is as follows: if Minimal Decomposition is solvable in polynomial time, then Chromatic Approximation will also be solvable in polynomial time. However, since Chromatic Approximation is NP-Hard, we believe it is not solvable in polynomial time, which in turn implies that Minimal Decomposition is not solvable in polynomial time either i.e. it too is NP-Hard.

3.2.5 Pre-Determined Scheduling: Node Coloring

The simplest scheme for pre-determined scheduling is to color half the nodes white and the other half black. The frame then will consist of two slots. In the first slot the white nodes transmit to the black slots, and in the second slot the blacks transmit to the whites.

The fact that the frame consists of only two slots should be appealing to us. However, there is a price for the short frame length. Two nodes with the same color cannot communicate directly with each other. This is not favorable since if two nodes with the same color are stranded, then there will be no way for them to communicate (directly or indirectly).

3.2.6 Pre-Determined Scheduling: Logarithmic Labeling

Suppose we label each node with a binary label. Each binary label is a binary string of length $\lceil \log_2 N \rceil$, where N as you recall is the number of nodes in the network. Now, in the first slot we allow each node with a label $0*****$ ($*$ implies "0 or 1") to potentially transmit to all the nodes with labels $1*****$. In the second slot, we reverse the direction, and allow nodes with labels $1*****$ to transmit to nodes with labels $0*****$. The scheduling proceeds as shown in Fig. 3-8. The length of the frame in this scheme is $2\lceil \log_2 N \rceil$. Since each node's label differs from another's in at least one digit, each node will be able transmit to every other node (not necessarily simultaneously), at least once in a frame. Note that Node 000000 can transmit to Node 111111 in six of the twelve frame slots.

We can generalize this approach to label nodes using base m instead of base 2 (what we did above). Let K be the length of the frame, then by similar analysis to what we did above,

$$K = m\lceil \log_m N \rceil \tag{3.13}$$

We now see whether base-2 or binary labeling is optimal in terms of the resulting frame length. To simplify the mathematics, we temporarily ignore the integrality

Slot #	Transmissions												
1	0	*	*	*	*	*	→	1	*	*	*	*	*
2	1	*	*	*	*	*	→	0	*	*	*	*	*
3	*	0	*	*	*	*	→	*	1	*	*	*	*
4	*	1	*	*	*	*	→	*	0	*	*	*	*
5	*	*	0	*	*	*	→	*	*	1	*	*	*
6	*	*	1	*	*	*	→	*	*	0	*	*	*
.							.						
.							.						
.							.						
12	*	*	*	*	*	1	→	*	*	*	*	*	0

Figure 3-8: Scheduling using binary labeling

constraint and say

$$K = m \log_m N$$

Then,

$$\frac{dK}{dm} = \frac{\ln N}{\ln m} - m \frac{\ln N}{(\ln m)^2} \frac{1}{m} = \frac{\ln N}{\ln m} - \frac{\ln N}{(\ln m)^2} \tag{3.14}$$

Setting the first derivative to zero, we get $m^* = e$. We check the sign of the first derivative,

$$\frac{d^2K}{dm^2} = \frac{\ln N}{m(\ln m)^2} \left(\frac{2}{\ln m} - 1 \right) \tag{3.15}$$

For $m^* = e$, the second derivative is positive, which implies that we have a minimum. However, the base needs to be an integer, therefore the optimal base is either 2 or 3. In Fig. 3-9 we plot $2 \log_2 N$ and $3 \log_3 N$. From that plot, it seems that base 3 is optimal. But a more careful analysis is needed, since the number of slots in the frame is really $m \lceil \log_m n \rceil$. As Figure 3-10 and Figure 3-11 indicate, neither base 2 nor base 3 labeling dominates the other, even if we allow the number of nodes in the network to grow to up to 2000000, which is too large for the packet radio networks we have in mind. Therefore, for our purposes, it is fair to say that both of them are optimal.

We are curious to see how large the packet radio network must get, before ternary labeling completely dominates binary labeling. We want

$$3\lceil \log_3 N \rceil \leq 2\lceil \log_2 N \rceil$$

The worst-case scenario (in terms of the ceiling functions) occurs when

$$\lceil \log_2 N \rceil = \log_2 N$$

and

$$\lceil \log_3 N \rceil = \log_3 N + 1 - \epsilon$$

for arbitrarily small ϵ. We rewrite the above inequality as

$$3(\log_N + 1) \leq 2 \log_2 N$$

to which the solution is $N \geq 2.65 \times 10^8$. Therefore, for networks with more than $N \geq 2.65 \times 10^8$ nodes, we are guaranteed that ternary labeling dominates binary labeling. However, as we indicated earlier, this size is too large for the kind of networks we are considering.

3.3 Conclusion

In this chapter we focused on scheduling in PRNETs. We first considered minimal decomposition scheduling, which would construct the smallest possible frame, given a set of paths dictated by the router. In a mobile network, the topology of the network is constantly changing. This implies that data routes are also constantly changing, which would necessitate resolving the minimal decomposition problem many times. Therefore, even if minimal decomposition were not NP-Hard, it would be a difficult task to find a distributed algorithm that constantly updates the minimal decomposition solution.

We also proposed two pre-determined scheduling schemes, and it seems that the

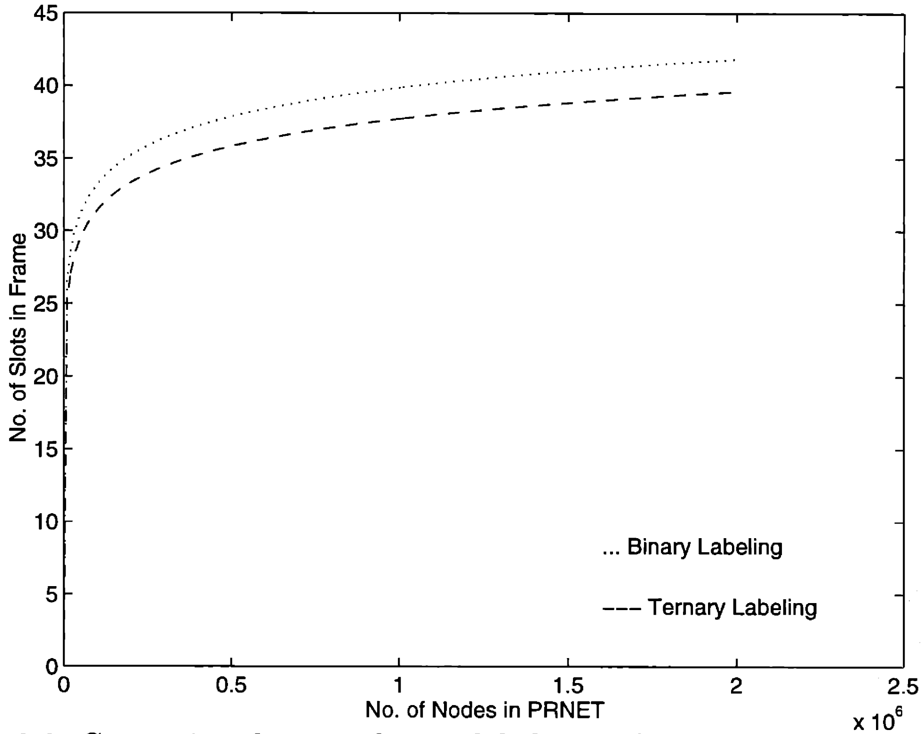

Figure 3-9: Comparison between binary labeling and ternary labeling, allowing fractional number of slots.

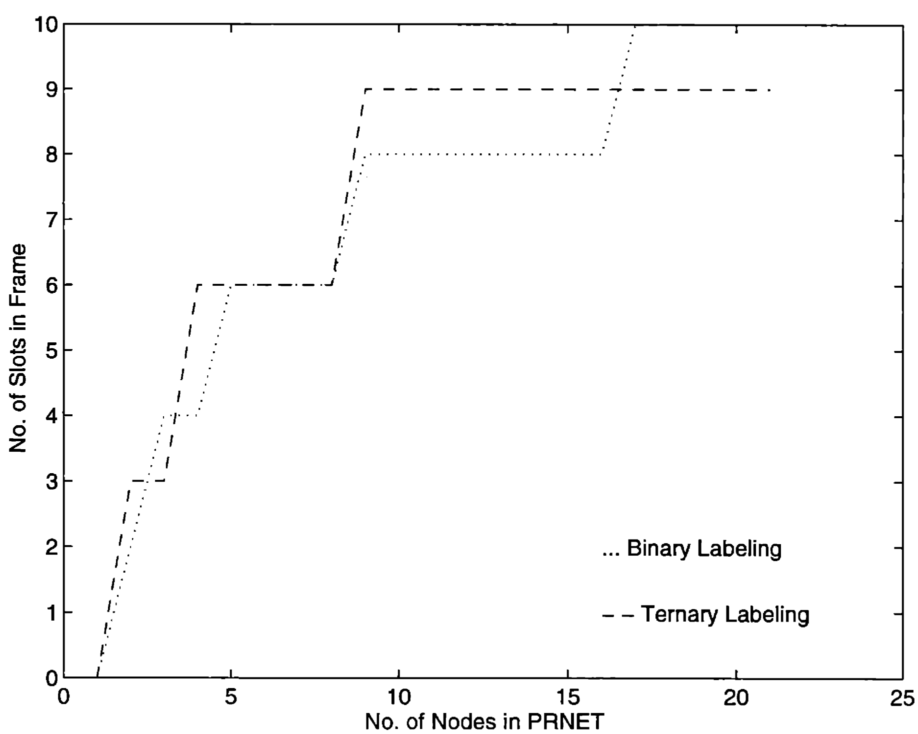

Figure 3-10: Comparison between binary labeling and ternary labeling for PRNETs with up to 20 nodes, restricting number of slots to be integral.

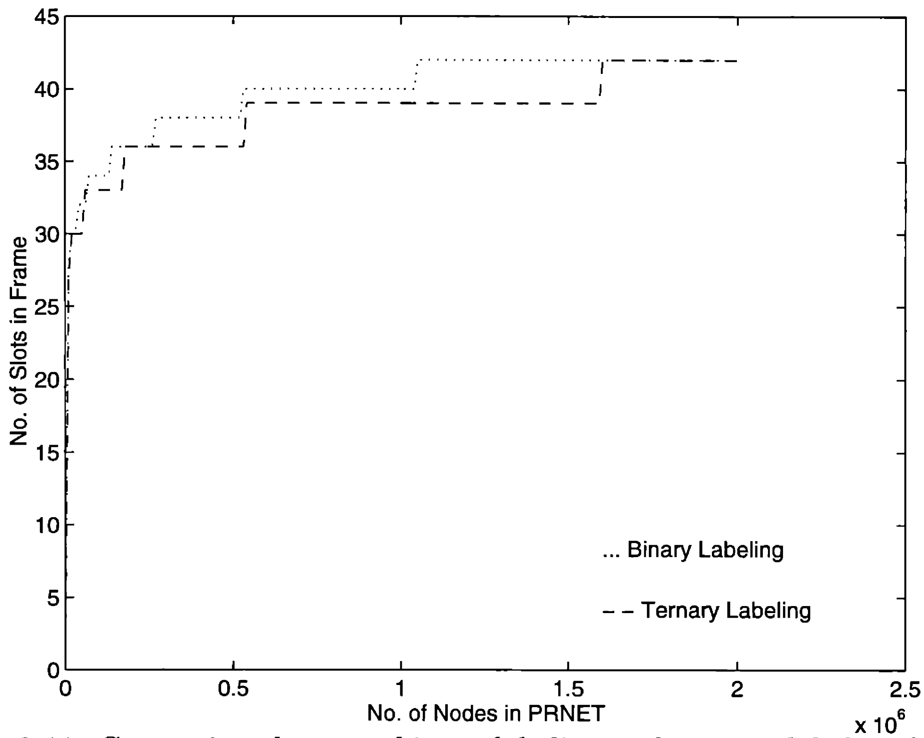

Figure 3-11: Comparison between binary labeling and ternary labeling for PRNETs with up to 2000000 nodes, restricting number of slots to be integral.

logarithmic labeling scheme suits our purposes better, since it allows each node to communicate directly with each other node. Moreover, certain links are active in more than one slot, thus giving the router more freedom in determining the routes. In addition, in logarithmic labeling the frame length grows only logarithmically with the size of the network.

Chapter 4

Conclusion

Any PRNET routing algorithm (regardless of the optimality criterion) must deal with the constraint that no node can transmit and receive simultaneously. Therefore, embedded in the routing algorithm there is a scheduling component; and hence optimal routing involves optimizing scheduling according to some criterion or another. In this thesis we investigated optimizing scheduling according to the criterion of minimal decomposition, which turned out to be an NP-Hard problem. Due to the combinatorial nature of scheduling, one would guess that other optimality criteria (for scheduling) also lead to NP-Hard problems. Even if they do not, or if we apply greedy algorithms to solve the scheduling problem suboptimally, we will still have a problem. This problem stems from the fact that we have a distributed network, which implies that many messages between the nodes will be required to converge to the solution. Furthermore, the network is mobile, which implies that rapid messages between the nodes are required to keep the solution up-to-date. As a result, a potentially significant fraction of the network capacity might be used to implement scheduling, instead of being used to transfer data.

The complexity of optimal dynamic scheduling, and the fact that its implementation can potentially use up a significant fraction of the network capacity, leads us to favor pre-determined scheduling. However, static pre-determined scheduling can be costly in the long run, because its performance depends on the changing topology of the network. A better approach would be to implement a quasi-static pre-determined scheduling. In quasi-static pre-determined scheduling, the schedule is updated at the rate that significant topological changes occur. Between updates the pre-determined scheduling remains fixed. We investigated two pre-determined scheduling schemes, vertex coloring and logarithmic labeling, and concluded that logarithmic labeling is more favorable.

As for future research, we need to investigate the issue of updating the pre-determined schedule. We also propose investigating scheduling schemes for PRNETs where the number of neighbors that a node has is limited. This limit may result from technological constraints, or as [6] suggests may be desired to reduce interference in the network. Finally, we need to find a routing algorithm that goes on top of whichever scheduling scheme we decide on. Although we lose (routing) optimality by fixing the scheduling first, and then routing accordingly, it is probably worth paying such a price for the reduced complexity of the overall minimum ANP routing problem.

Bibliography

[1] D. Bertsekas and R. Gallager, "Data Networks." Prentice Hall, New Jersey, 1992.

[2] A. J. Viterbi, "CDMA Principles of Spread Spectrum Communication." Addison-Wesley Publishing Company, Massachusetts, 1995.

[3] R. D. Yates, "A Framework for Uplink Power Control in Cellular Radio Systems." *IEEE Journal on Selected Areas in Communications*, Vol. 13, No. 7, pp. 1341-1348, September 1995.

[4] W. C. Y. Lee, "Mobile Communications Engineering." McGraw-Hill, New York, 1982.

[5] M. Bazaraa, H. Sherali and C. M. Shetty, "Nonlinear Programming Theory and Algorithms." John Wiley & Sons, New York, 1979.

[6] L. Kleinrock and J.A. Silvester, "Optimal Transmission Radii in Packet Radio Networks or Why six is a Magic Number," *Proc. National Telecommunication Conference*, Birmingham, AL, December 1978.

[7] C. Chartrand and L. Lesniak, "Graphs & Digraphs." Chapman & Hall, London, 1996.

[8] U. Feige and J. Kilian, "Zero Knowledge and the Chromatic Number," in *Proceedings Eleventh Annual IEEE Conference on Computational Complexity*, pp. 278-287, 1996.

[9] M. R. Garey and D. S. Johnson, "Computers and Intractability A guide to The Theory of NP-Completeness." W.H. Freeman & Company, San Fransisco, 1979.

[10] B. Leiner, D. L. Nielson and F. A. Tobagi, "Issues in Packet Radio Network Design," *Proc. IEEE*, Vol. 75, No. 1, January 1987.

[11] M. R. Garey and D. S. Johnson, "The Complexity of Near-Optimal Graph Coloring," *Journal of the Association for Computing Machinery*, Vol. 23, No. 1, pp. 43-49, January 1976.