Integrated Routing, Scheduling and Power Control in STDMA Wireless Ad-hoc Networks

A Master’s Thesis by
Shao Yu

January 2004

IR–RT–EX–0401
Abstract

This thesis presents several methods for optimizing the performance of wireless ad-hoc networks, integrating routing, scheduling and power control. In order to achieve high throughput, we focus on spatial reuse TDMA (STDMA) scheduling, in which more than one link can be activated simultaneously. We propose optimization models for both fixed and free (multi-path) routing, and optimize performance in terms of throughput, fairness and power consumption.

To solve the optimization problems in a direct (“brute force”) way is computationally very demanding, since the number of variables in the model is exponential with the number of links in the network. To overcome this problem, we consider two heuristic approaches that try to find a small subset of link activations (combinations of links that can be activated in the same timeslot) that lead to a good schedule. The first approach is direct, and is based on insight obtained from trying to optimize the network performance under TDMA scheduling. The other approach is iterative, and requires the repeated solution of a STDMA scheduling problem. In each iteration, a weighted throughput problem is solved to find new link activations to include in the schedule. This last method is inspired by a mathematical programming technique known as column generation. Contrary to classical column generation, however, we do not solve the weighted throughput problem to optimality, but use fast heuristics to find “good” link activations. We propose greedy heuristics both for systems with fixed rate and variable-rate transmissions. We demonstrate that the column generation method with the proposed heuristics produces near-optimal solutions using very reasonable computations.

Finally, we use our approaches to try to gain some insight into the trade-off between throughput and power consumption. We find that link reuse causes our optimized STDMA solutions to have lower power efficiency than the TDMA solution.
Acknowledgements

I am grateful to many people for helping me carry this thesis to completion. First of all, I would like to thank my advisor, Dr. Mikael Johansson, for his scientific guidance, support, feedbacks and valuable discussions, without his helpful guidance, this thesis would have been impossible complete.

I would like to express my gratitude to Dr. Tim Giles for his help and proper arrangement for my thesis.

I would like to thank Miguel Angel Gomez Rodriguez for his good advice and correcting grammar mistakes for my thesis.

I am also thankful to all the other people who helped me on this work.
# Contents

1  INTRODUCTION ............................................................................................................................... 7  
   1.1  A BRIEF REVIEW OF RELATED WORK ................................................................................... 8  

2  NETWORK MODEL .......................................................................................................................... 10  
   2.1  NETWORK TOPOLOGY ................................................................................................................. 10  
   2.2  NETWORK FLOWS AND ROUTING ............................................................................................. 11  
   2.3  RADIO LINK LAYER ..................................................................................................................... 12  
   2.4  SPATIAL REUSED TDMA SCHEDULING ................................................................................... 14  

3  PERFORMANCE MEASURES ............................................................................................................ 15  

4  NETWORK OPTIMIZATION ............................................................................................................... 17  
   4.1  OPTIMIZATION MODEL WITH FIXED ROUTING ......................................................................... 17  
   4.2  OPTIMIZATION MODEL WITH NON-FIXED ROUTING ................................................................. 19  
   4.3  OPTIMIZATION MODEL FOR THROUGHPUT AND POWER CONSUMPTION ............................ 21  

5  CALCULATING CLIQUE SET ............................................................................................................. 22  
   5.1  GREEDY ALGORITHM AND POWER CONTROL ........................................................................ 22  
   5.2  HEURISTIC ALGORITHM FOR WEIGHTED THROUGHPUT OPTIMIZATION ............................. 26  

6  EXAMPLES ........................................................................................................................................ 28  
   6.1  COMPARISON OF DIFFERENT CLIQUE SETS ........................................................................... 30  
   6.2  PERFORMANCE MEASURES OF HEURISTIC ALGORITHM 3 AND 4 .......................................... 32  
   6.3  PERFORMANCE MEASURES OF WIRELESS AD-HOC NETWORK ............................................. 35  
   6.4  RELATION BETWEEN THROUGHPUT AND POWER CONSUMPTION ..................................... 37  
   6.5  COMPUTATIONAL ASPECT ......................................................................................................... 41  

7  CONCLUSION ..................................................................................................................................... 42  

REFERENCES ....................................................................................................................................... 44
1 Introduction

With the fast growth of the mobile market, the competition among telecom operators is crueler than ever. At the same time, consumers have come to expect ‘free’ high rate multimedia services like what they have in Internet. Thus, cost effective network implementations will be one of the key factors to be considered. According to [1], one way to provide cost effective, affordable wireless bandwidth (almost) everywhere is to rely on alternative concepts and architectures that allow simple and cheap deployment of infrastructure, yet still support high data rates. Ad-hoc wireless networks, e.g. Bluetooth, in which cost can be dramatically reduced due to no infrastructure, is one of these low cost networks. In ad-hoc wireless networks, terminals transmit data by other terminals in a store & forward fashion (also called multihop). Hence, the access points are not necessary in the ad-hoc networks; no access points and no transmission lines make implementing cost lower than normal wireless systems. The price paid for this is reduced performance and increased terminal complexity.

In the wireless ad-hoc networks, it is important to make system efficient, thus how to design, implement and optimize wireless ad-hoc networks is a prevalent topic in recent days. In this thesis, we will try to make this system achieve efficiency and fairness by integrating routing, scheduling and power control with reasonable computation, this is a cross-layer optimization problem.

Routing in ad-hoc networks has lots of methods: minimum hop routing, maximum hop routing, minimum energy routing, etc. In this thesis, the routing is not so simple, thus, these routing methods are different with the routing that we want. We need to find out one routing that can satisfy high throughput or low power consumption with proper scheduling. Therefore, a joint optimization is needed because the performance depends on the scheduling as well as the routing.

Scheduling, which will be discussed later in detail, is also important in the ad-hoc network. The simplest schedule is TDMA which means only one link is active in one time slot. But usually we want to increase the throughput of the network, therefore,
we activate more than one link in one time slot, which has been studied in [5, 6]. Different schedules lead to different throughput. Then it is obvious that how to design a proper schedule for ad-hoc networks which can make our objectives (network throughput or power consumption) better is a primary task of our thesis.

Power control is a classical method to improve wireless networks’ performance. We implement it in the ad-hoc network, so that we can have more links, which have the same SIR, activated in the same slot to increase the throughput.

### 1.1 A Brief Review of Related Work

The methods to get high throughput or low power consumption can be divided into heuristic methods and optimization methods:

Heuristic approach, i.e. construct a schedule making sensible but not necessarily optimal decisions along the way [4, 5, 6]. Somarriba and Giles [4] propose one scheduling way that combines traffic, routing and power together. They put the links that are compatible into each time slot until all links have been allocated. And the links with higher relative traffic get higher priority to be put in the slots first, so that they can get higher data rates, otherwise they will become the bottleneck of the system. Furthermore, in their method, centralized power control is used to make the interference lower between transmitting links. The power control enables more links to transmit simultaneously, thus, end-to-end delay is also reduced. This method does not require much computation compared to other methods. It is easy to be implemented, but not optimal. In addition, it considers only two performance measure parameters: delay and throughput, while other parameters, such as fairness and power have not been considered.

Optimization approach, the problem of optimal scheduling of transmissions in multi-hop wireless networks has a long history (see, for example, [17, 18]). Recently, several approaches have been proposed for simultaneously optimizing several layers in the networking stack, to further improve network-wide performance [3, 8, 13, 15, 16].
Toumpis and Goldsmith [8] compute the achievable rate regions for small wireless networks under different routing strategies and MAC schemes. The optimization relies on explicit enumeration of all transmission groups, which results in an optimization problem whose number of variables is exponential with the number of links. Consequently, results are only given for small networks.

Cruz and Santhanam [3] consider the problem of minimizing the average power needed to support a set of given end-to-end demands, and propose a computational approach based on Lagrangean duality. Also this method relies on explicit enumeration of all transmission groups, and results have only been presented for a set of small examples. A heuristic approach for iteratively improving the routing is proposed, but no global optimization of scheduling, power control and routing is carried out. Moreover, the performance objective is restricted to minimum average power, while our focus is also on throughput performance and fairness.

Explicit enumeration of all vertices is avoided in approaches that use column generation for solving the network optimization problem [15, 16]. Väbrandt et al. consider the problem of joint transmission scheduling and end-to-end rate selection for maximum throughput and maximum uniform throughput under a specific MAC scheme. Transmission groups are generated by solving an integer programming problem at each iteration. Johansson and Xiao extend the approach to nonlinear performance objectives (necessary for optimizing proportional fairness), arbitrary MAC schemes, and also include the networking layer in the optimization. For the particular MAC schemes considered in [15], transmission groups are also computed using integer programming. Although results are reported for larger networks than in [3, 8], the need to solve integer programming problems restricts the networks under consideration to 100-200 links (10-20 nodes). Neither [16] nor [15] consider performance objectives that account for the power consumption.

In summary, the existing methods for network optimization can only handle relatively small networks, and different approaches have considered only subsets of the issues that could be interesting for the operation of a real network. In our work, we will try to develop methods that optimize routing, power control, transmission
scheduling and end-to-end rate selection; consider throughput, fairness and power objectives; and that can be applied to larger networks than the current approaches (possibly, at a slight decrease in performance from the optimal solution).

In the following chapters, we will introduce ad-hoc networks model first, and then analyze and propose some optimization models for throughput and power consumption. Furthermore, in order to solve these problems, we will propose four heuristic algorithms to get the clique set. Finally we will give some examples for these models and algorithms.

2 Network Model

2.1 Network Topology

We represent an ad-hoc network topology with the standard directed graph model. There are a number of nodes in the network, labeled $n = 1, \ldots, N$, and they can only communicate with each other by wireless medium. Link $(i, j)$ represents the link that can transmit data from node $i$ to node $j$, we label the links with integers $l = 1, \ldots, L$.

We introduce the node-link incidence matrix $A \in \mathbb{R}^{N \times L}$, here, whose entries $A_{nl}$ are defined as:

$$A_{nl} = \begin{cases} 
1 & \text{if link } l \text{ is outgoing from node } n \\
-1 & \text{if link } l \text{ is incoming from node } n \\
0 & \text{otherwise.}
\end{cases}$$
For example, we can represent the network in Figure 1 with the incidence matrix:

$$A = \begin{bmatrix}
1 & -1 & 0 & 0 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & -1 \\
1 & 1 & 0 & 0 & -1 & 0 \\
0 & 0 & 1 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 1 & -1 \\
\end{bmatrix}.$$ 

In the matrix, each column containing one ‘1’ and one ‘-1’ represents one link, because the link must be represented in both start node and end node in the model. On the other hand, each row contains all the links that are outgoing or incoming from the node, 1 entry indicates the outgoing links, while -1 entry indicates the incoming links.

### 2.2 Network Flows and Routing

We use a flow model introduced by Xiao et al. [7] for the routing of data packets across the network. This model describes the average behavior of data transmissions (data rates in bits/second), and ignores packet-level details of transmission protocols and forwarding mechanisms. Each node can send (different) data to many destinations and receive data from many sources. But multicast is not considered. We assume that the data flows are lossless across links, and they satisfy flow conservation law at each node.

We denote $k_{n}^{(d)}$ to be the amount of flow from source node $n$ to destination node $d$, regardless of how the flow is routed. Thus, we can define a source-sink vector $k^{(d)}$, whose $n$th ($n \neq d$) is $k_{n}^{(d)}$. In light of the flow conservation law, we
define the sink flow at the destination as:

\[ k_d^{(d)} = - \sum_{n \neq d} k_n^{(d)} \]

where the summation is over all nodes except for the destination node.

On each link \( l \), we let \( x_l^{(d)} \) be the amount of flow destined for node \( d \), and let \( c_l \) be the capacity of link \( l \), and \( t_l = \sum_d x_l^{(d)} \) be the total amount of traffic on link \( l \), obviously, \( t_l \leq c_l \).

For the routing across single path, we assume that all the routes are labeled by integers \( i = 1, \ldots, R \). Here the only variable is the source vector \( s \), whose component \( s_i \) is the data rate sent through route \( i \). We can then represent the average traffic flow on each link using a link-path incidence matrix \( B \in R^{L \times R} \), \( B_0 = 1 \) when route \( i \) pass through link \( l \), otherwise \( B_0 = 0 \). The total traffic vector is \( t = Bs \) and capacity constraints is simply \( Bs \leq c \).

For data flow across multiple pre-specified paths for each source-destination pair, we can regard different paths between the same pair as different routing when we use link-path incidence matrix. It is clear that the sum of all rates between the same pair is the end-to-end rate of this pair.

The routing also can be optimized simultaneously with other parameters in the network. For example, we can make the total power consumed in the network minimum or total throughput maximum with the optimized routing. This model will be explained in detail in the following chapter.

2.3 Radio Link Layer

In the network, node \( i \) and node \( j \), with \( i, j \in \{1, 2, \ldots, N\} \) are either connected by a link or disconnected, depending on the radio propagation properties of the terrain where the network is deployed. The propagation effect is modeled by the radio propagation losses. We represent the path losses on link \((i, j)\) by \( L_{ij} \). The inverse of
this quantity is commonly referred to as the link path gain, \( G_{ij} = 1/L_{ij} \), and constitutes the elements of the path gain matrix, \( G \). Hence, the received power at node \( j \) when node \( i \) transmits is \( G_{ij}P \). The factors that affect \( G \) include distance, channel fading, shadowing, antenna gain patterns, etc. For example, if the propagation depends on distance mostly, the received power \( P_j \) may be rewritten as \( P_j = P_i (d_{ij})^{-\alpha} \). Here \( d_{ij} \) is the distance between node \( i \) and node \( j \), and \( \alpha \) is the path loss exponent.

However, signals emanating from other transmitters appear to the receiver node \( j \) as interference, and there is background noise to contend with as well. The signal to interference and noise ratio (SINR) for link \( (i, j) \) is defined as

\[
\gamma_{ij} = \frac{G_{ij}P_i}{P_{\text{noise}} + \sum_{k \neq i} G_{ij}P_k}
\]

(1)

Where \( P_{\text{noise}} \) is the background noise power level at node \( j \).

We assume that the data rate on the link between node \( i \) and node \( j \) depends on the signal-to-interference ratio \( \gamma_{ij} \) on the link, which is an interference limited network. One possible relationship between SIR and rate would be Shannon capacity equation, \( R_{ij} = W \log_2 (1 + \gamma_{ij}) \). In practice, however, the relation between \( \gamma_{ij} \) and the link rate depends on the particular radio technology and MAC scheme employed. In this thesis, we focus on two particular models: the first model uses an ultra-wideband assumption of linear relationship between SIR and rate,

\[
R_{ij} = W \gamma_{ij}.
\]

The other model tries to mimic the operation of 802.11b, where all links that can sustain a given SIR transmit at a constant rate, we call it base rate or zero model,

\[
R_{ij} = \begin{cases} W\gamma_0 & \text{if } \gamma_{ij} \geq \gamma_0 \\ 0 & \text{otherwise} \end{cases}
\]

For both models, we make the following additional assumptions: antennas are omnidirectional, and nodes can only transmit or receive data from one other node at a
time; all transmitters use the same frequency band; the maximum instantaneous power is limited, \(0 \leq P_i \leq P_{\text{max}}\); the network topology and fading is assumed static (i.e., mobility is not considered, and the link gain matrix is assumed to be constant).

### 2.4 Spatial Reused TDMA Scheduling

As mentioned in [5], one of the important design issues in wireless ad-hoc network is Medium Access Control (MAC), i.e. how to avoid or resolve conflicts due to simultaneously transmitting radio units. The traditional approach to multi-access in packet radio systems is to use contention-based protocols, such as carrier sense multiple access (CSMA). This has clear advantages when the traffic is unpredictable, but contention based medium access methods are inherently inappropriate for providing QoS guarantees due to hidden terminal problem, and the solution of hidden terminal problem makes delay guarantees difficult. However, one of the most important QoS parameters in many applications that are specifically sensitive to the MAC is delay guarantees.

To guarantee delay in ad-hoc network, the approach of time division multiple access (TDMA) is introduced, i.e. the time is divided into time slots and each user receives its own time slot. Unfortunately, in sparsely connected networks, this approach is usually inefficient. But if the different links are separated so far that they do not interfere each other too much, that means they can reach a SIR above the reliable communication threshold, then the signals can be transmitted in these links at the same time, which is called spatial reused TDMA. Due to STDMA, we can achieve both high efficiency and delay guarantees.

STDMA schedule \(S\) is defined as the clique set \(Y\) and time percentage vector \(p_t\), for \(t = 1, 2, ..., T\), where \(T\) is the period of the schedule. Clique means a group of links which can be transmitted simultaneously, and a clique set is a group of cliques. The entry \(p_t\) of \(p\) is the time percentage value of slot \(t\) duration in one STDMA
frame and $\sum p_i = 1$. A STDMA schedule describes the transmission rights for each time slot, and the aim of the schedule is to optimize the network parameters such as network throughput. A schedule is called conflict-free if this is the case for all time slots in the schedule.

In figure 2, we show an example of STDMA schedule with a simple 4-node network. We can see from this network, communication between node 1 and node 4 must be relayed by node 2 or node 3. We assume that link (1, 2) and link (3, 4) or link (1, 3) and link (2, 4) could be transmitted simultaneously due to far distance separations. Thus, we can design a two-time slot STDMA schedule that link (1, 2) and link (3, 4) are transmitted simultaneously in timeslot 1, and link (1, 3) and link (2, 4) are transmitted simultaneously in timeslot 2, which means that clique set has two clique: clique 1 (link (1, 3), link (2, 4)) and clique 2(link (1, 2), link (3, 4)), and time percentage vector $p$ is $(p_1, p_2)$. Obviously, if we allocate time percentage vector $p$ to cliques properly, this schedule will use radio resource efficiently. That is why we are interested in STDMA scheduling.

\begin{figure}
\centering
\includegraphics[width=0.3\textwidth]{4-node-sample-network.png}
\caption{4-node sample network}
\end{figure}

In above example, we assign all transmission right to the links. This is called link assignment. If we assign all transmission right to the nodes, this is called node assignment. In this thesis, we only consider link assignment.

### 3 Performance Measures

When we evaluate the performance of wireless ad-hoc network routing or scheduling algorithm, there are several measurable parameters that should be concerned about:
1. Capacity. One aim of network optimization is to maximize the capacity of network (network throughput). We measure the ad-hoc network total capacity by the sum of all end-to-end rates $s$ between source and destination which is also called network throughput.

2. Fairness. To maximize the capacity of network we can get the highest efficiency, while it may cause unfairness. The extreme example is that with the largest network throughput all the end-to-end rates are on the highest efficient route, while other routes can not carry any data, which is totally unfair. Therefore, fairness is an important performance measure for ad-hoc network, but obviously, there exists a trade-off between fairness and efficiency. Max-min fairness has been used in some wireless multihop networks, which is often viewed as extreme fairness, i.e. as found by [12], max-min fairness leads to equal rates of all flows regardless of the network topology, routing and power constraints. This means that all rates equal to the worst flow, making the network very inefficient. To reduce the tension between efficiency and fairness, proportional fairness is introduced, which is obtained by maximizing the sum of the logarithm of the achieved rates over all source-destination pairs $\sum \log(s_i)$ [19]. It is shown in [12] that the proportional fairness can get a robust trade-off between fairness and efficiency.

3. Power is one of the most important resources in wireless network. For a performance measure parameter, the aim is to minimize power consumption in the network. Proper power consumption will help to reduce the total interference and increase the capacity of the network, especially in the ad-hoc network, and it will also increase the network life. Power control is used to get the lowest amount of power each node need to achieve the SINR target. Thus, if we consider power in the wireless network, power control is a very important tool to reach our aim.
4 Network Optimization

4.1 Optimization Model with Fixed Routing

Considering the ad-hoc network model described above, and the rate of link \((i,j)\) is \(R_{ij} = W\gamma_{ij}\), where \(W\) is the bandwidth of the signal and \(\gamma_{ij}\) is the SINR of link \((i,j)\), we assume that the power consumption of each node in each time slot is known, thus given the clique set \(Y_t\), we can get the rate set \(C_t, t = 1,2,\ldots,T\) that has the same size as clique set \(Y_t\), the entry \(C_t\) of rate set is the link rate vector in the time slot \(i\). Thus the average link rate vector \(R_{\text{avg}} = Cp\), where \(\sum_t p_t = 1\). The average achievable rate region is the convex hull of the instantaneous achievable rate region. From above, we know that the total traffic vector is \(t = Bs\) and capacity constraints is simply \(Bs \leq c\), where \(B\) is a link-path incidence matrix \(B \in R^{k \times l}\), and \(s\) is the end-to-end rate vector. In this case, we can set link capacity \(c = R_{\text{avg}} = Cp\), thus, \(Bs \leq Cp\). We suppose that our object is to maximize a concave end-to-end rates function, and then we have the following generic formulation of the optimization problem:

\[
\begin{align*}
\text{Maximize} & \quad f(s) \\
\text{Subject to} & \quad Bs \leq Cp \\
& \quad \sum p = 1 \\
& \quad 0 \leq p, s \geq 0 \\
& \quad 0 \leq \text{power}_i \leq P_{\text{max}}
\end{align*}
\]

where \(\text{power} \in R^{N \times T}\) is a matrix whose entry \(\text{power}_{ij}\) is the power consumed by node \(i\) in timeslot \(j\).
Since the constraints define a convex set and we assume that the objective function is concave and strictly increasing. Therefore, this optimization problem is a convex optimization problem.

The result includes not only end-to-end rate vector $s$, but also time percentage vector $p$, thus, we have got one STDMA schedule over one clique set $Y_i$, since time percentage vector $p$ and clique sets $Y_i$ together represent one STDMA schedule.

The above optimization problem is very general, and can be used to get many end-to-end rates objectives, especially the measure parameters we mentioned above, such as throughput, transport capacity, max-min fairness, and proportional fairness.

**Maximum throughput:**

Given the ad-hoc network model above, it is natural to maximize the total end-to-end network rates (network throughput). Let $w \in R^T_+$ be a vector, when all $w_j=1$, $w^T s$ represent the sum of all end-to-end rates. Then the maximum throughput problem can be formulated as:

\[
\text{Maximize} \quad w^T s \\
\text{Subject to} \quad \text{constraints in (2)}
\]

where $w_j=1$.

In this case, the efficiency is the highest but totally unfair. It is because the scheme allocates all time percentage to the slot that contains the highest efficiency routing, while the weak routes have low opportunities to transmit.

**Maximum equal-rate allocation (the max-min fairness approach in [12]):**

Fairness is another performance parameter of the network. For the max-min fairness, all rates are the same [12], thus, it is the absolute fairness. To reach max-min fairness, the optimization objective function tries to maximize equal rate allocation. Thus the problem can be formulated as:

\[
\text{Maximize} \quad w^T s
\]
\[ Bs \leq Cp \]
\[ \sum p = 1 \]

Subject to
\[ p \geq 0, s \geq 0 \]
\[ 0 \leq power_{ij} \leq P_{\text{max}} \]
\[ s_i = s_j \quad i \neq j \]

where \( w_i = 1 \).

Obviously, the model causes low efficiency compared to the model which does not consider fairness.

**Proportional fairness:**

Proportional fairness is a tradeoff between efficiency and fairness, which is the sum of the logarithm of the achieved rates over all source-destination pairs. This problem can be formulated as:

\[ \text{Maximize} \quad \sum \log(s_i) \]

Subject to constraints in (2).

As shown in [12], proportional fairness allows the network to be with good fairness as well as high efficiency.

One extension of this model is in the multi-path situation. Multi-path means the source node can send data to the destination node through different paths at the same time. If we regard the different paths between the source and destination as different routes, then they may have different rates over these routes, where the total rate is the sum of the rates over all the routes between source and destination. Thus we can let \( s_{il} \) represent the end-to-end rate of the \( i \) th pair over its \( l \)th path, then the total rate in \( i \) th pair \( s_i = \sum s_{il} \).

### 4.2 Optimization Model with Non-fixed Routing

We assumed that in model (2) the routing was fixed, and then we optimized the scheduling. However, we can also make the scheduling and routing optimal simultaneously.
Following the method introduced by Xiao et al. [7], the problem can be seen as a simultaneously routing, resource allocation and scheduling (SRRAS) problem. As we know, \( k_n^{(d)} \) denotes the non-negative amount of flow injected into the network at node \( n \) (the source) and destined for node \( d \) (the destination), and \( x_l^{(d)} \) is the amount of flow destined for node \( d \) on link \( l \). At each node \( n \), components of the flow vector and the source-destination vector with the same destination \( d \) satisfy the following flow conservation law:

\[
\sum_{d} x_l^{(d)} = k_n^{(d)}
\]

where \( l \) are the links that are incoming or outgoing from node \( n \). The flow conservation law across the whole network can be compactly written as

\[
Ax^{(d)} = k^{(d)}
\]

where \( A \) is the node-link incidence matrix.

We know that link capacity \( c = C_p \sum_i p_i = 1 \), and obviously for each link \( l \),

\[
\sum_d x_l^{(d)} \leq C_i p.
\]

Where \( C_l \) represents the rate vector of link \( l \).

To summarize, we can get a model for the non-fixed routing optimization model:

Maximize \( f(k) \)

\[
Ax^{(d)} = k^{(d)}
\]

\[
\sum x^{(d)} \leq C_p
\]

Subject to

\[
\sum p = 1 \quad \quad (3)
\]

\[
p \geq 0, k^{(d)} \geq 0, x^{(d)} \geq 0
\]

\[
0 \leq power_{ij} \leq P_{max}
\]

The model above is very general, which assumes that the flow is from every node to every other node. And the upbound number of variables is \( L*N+N*N+T \), where \( L \) is the number of links, \( N \) is the number of nodes and \( T \) is the number of slots. Therefore, the computation is not small in large networks. Fortunately, the nodes that are transmitting simultaneously are not so many in real networks, which make this
4.3 Optimization Model for Throughput and Power Consumption

So far, we have got optimization model for fixed-routing and non-fixed routing. Yet we have not considered another important performance measure parameter: power. In order to minimize total power consumption in our model, we rewrite model (3) as below:

Maximize \( \lambda f(k) - a(1 - \lambda)(\sum power_g)p \)

Subject to

\[
Ax^{(d)} = k^{(d)} \\
\sum x^{(d)} \leq C_p \\
\sum p = 1 \\
p \geq 0, k^{(d)} \geq 0, x^{(d)} \geq 0, 0 \leq \lambda \leq 1 \\
0 \leq power_g \leq P_{\text{max}}
\]

where \( \sum power_g \in R^+ \) is the power allocation vector that contains the total power consumed in each time slot. \( a \) is a constant to normalize power allocation vector, since the value of power is so small compared to the throughput value. From this model, we can find minimum power and maximum throughput at the same time integrating routing and scheduling. Different \( \lambda \) results in different optimal values. If \( \lambda \) equals to 1, then this model is the same as model (3), and if \( \lambda \) is 0, the model tries to minimize power consumption. Usually we do not expect throughput to be 0, so we can set \( \lambda \) between 0 and 1 to satisfy our demand.

Although these models we mentioned above can get optimal throughput, power consumption, routing and scheduling, unfortunately, these models are the NP hard problems, thus it is hard to find an efficient solution to solve it. But if we can get clique set and power allocation matrix first, the model will becomes LP model, and the solution may be possible and efficient. In the following chapter, we will propose four heuristic algorithms to calculate clique set and make results as large as possible.
with reasonable computation.

5 Calculating Clique Set

One problem that we encounter in efficiently solving optimization models is to find out how many link combinations should be included in the clique set. We assume that the number of cliques is $M$, and the link capacity is represented as a convex combination of $C_m$, $m = 1, 2, \ldots, M$. The upper bound on $M$ is $2^L$. If we apply a combination to get the link capacity directly, the result of scheduling would be optimal, but huge computation makes this solution impractical in large network. If the $M$ is lower than $2^L$, the schedule may be not optimal, but the reduced computation is a big advantage for being implemented in the real world.

For the models mentioned above and as shown in [14] by Oh and Wasserman, if the linear rate model is used in the optimization model, the optimal policy is for transmitters to either use maximum power or to stay silent which is one of the radio link models we have mentioned in section 2.3. But sometimes, we do not need very high data rate over certain links, which would also save power consumption. For this purpose, centralized power control (SIR balancing [2, 4]) can be implemented in the wireless ad-hoc network.

5.1 Greedy Algorithm and Power Control

There are some heuristic methods to get clique set. The method we introduced here is one greedy method combined with centralized power control (SIR balancing).

The transmitted power clearly affects the link signal quality and the interference environment in ad-hoc wireless network. Proper power control can increase the number of links that can transmit simultaneously, and also can increase the capacity of the system. Normally we don’t need very high link capacity. If there is no power control in the wireless networks, then certain amount of power will be wasted. In this case, power control is also necessary. In the wireless ad-hoc network, both centralized
and distributed power control can be used as those in the cellular network [4, 13]. In this thesis, we only implement centralized power control to get proper clique set, although the distributed power control is a good alternative way.

As mentioned in [4], we describe all transmitter powers of the nodes in the network by the following power vector notation:

$$P = [P_1, P_2, ..., P_N]^T$$  \hfill (5)

This vector has to be non-negative, i.e. $P \geq 0$.

We define $\gamma_0$ as the threshold of the communication possibility. If the link SINR is larger than $\gamma_0$, then the communication is possible, otherwise, this link can not be created. We know that equation (1) can be rewritten as

$$\gamma_{ij} = \frac{P_i}{P_{\text{noise}} + \sum_{k \neq i} G_{ij} P_k} \geq \gamma_0$$ \hfill (6)

Let us introduce the $N \times N$ matrix $H$ such that

$$H_{ik} = \begin{cases} 0 & \text{if } i = k \\ \frac{G_{ij}}{G_{ij} \gamma_0} & \text{if } i \neq k \\ \end{cases}$$ \hfill (7)

Furthermore, we define the $N \times 1$ vector $\eta$ whose entry $\eta_i = \gamma_0 (P_{\text{noise}} / G_{ij})$, then rewrite the equation (6) using (7) in a matrix form results in (8)

$$AP \geq \eta$$ \hfill (8)

Where $A = I - H$ and $I$ denotes the identity matrix. From (8), it can be easily seen that the power vector $P$ could be computed by the following expression

$$P = (I - H)^{-1} \eta \leq P_{\text{max}}$$ \hfill (9)

If any elements of $P > P_{\text{max}}$, then the clique is not feasible.

After SIR balancing, we will find out that all the rates over the links are the same, thus, this radio link rate model is called as base rate or zero model that we have introduced in section 2.3, and this base rate is based on $\gamma_0$.

We assume that there are L links in the ad-hoc network, and all SINRs of these
links are over $\gamma_0$ when each of them is transmitting alone with maximum power. The order of links is the same as that in the node-link incidence matrix $A$, and it is a random sequence. We activate links in order of the links vector under SIR balancing to get one new clique. This is repeated in the rest of the links to get all the other cliques.

The main steps of this greedy algorithm to get clique set are described as below:

Algorithm 1:
- Beginning of loop 1.
  - Repeat until all the links are in the clique set.
  - Link group= all the links that are not in the clique set.
  - $i=1$, vector clique= the first link of the link group.
  - Beginning of loop 2.
    - If the number of links in the link group =1.
      - Add link group as clique into clique set.
    - Else.
      - $i = i + 1$.
      - Assume that the links in the clique transmit simultaneously with link $i$.
      - SIR balancing (or calculating SIR vector of clique).
      - If all transmitting powers $\leq P_{\text{max}}$ (or all SIRs $\geq$ SIR threshold).
        - Add link $i$ into clique.
    - End.
  - End.
  - Repeat until link $i$ is the last link in the link group.
  - End of loop 2.
  - Add clique into clique set.
- End of loop 1.

This greedy algorithm tries to maximize the number of links in every loop by
using power control, and guarantees that all the links are included in the clique set. Although it is not an optimal method for scheduling, it provides one way to get clique set quickly. By combining this algorithm with the optimization models which were mentioned above, we can get the optimization objectives (capacity, fairness, etc.) over the clique set from greedy algorithm. It is clear that the number of clique in the clique set is no more than L, thus the computation of getting the optimization objectives is much smaller than the case that all the combinations of links are in the clique set. With this greedy algorithm, it is possible to calculate the optimization model in the large ad-hoc network quickly, although scheduling result are not optimal.

If we only want to maximize the throughput, then the nodes either transmit with maximum power or stay silent. For this case, we just replace power control with the calculation of the clique SINR vector in the algorithm, and if all the values of this vector are not less than the SINR threshold, then the clique satisfy our demand.

The greedy algorithm mentioned above shows us that it is possible to get clique set without much computation, while the schedule is not optimal. Our aim is to get an efficient scheduling, as well as to have a reasonable computation.

If we apply the above greedy algorithm, the two kinds of links (the links that need high capacity and the links that need low capacity) may exist in the same clique. Therefore, when we optimize the scheduling, these two kinds of links will be optimized without any difference, which makes the system ineffective. In the improved algorithm, we must separate these two kinds of links in different cliques to make the system effective, which is the basic idea of this algorithm.

Let us consider the optimization model (3) with non-fixed routing. We use TDMA clique set here, which means each clique only contains one link. Thus, in the clique set, there are L cliques. We try to optimize the throughput with proportional fairness. From the result we get the percentage vector p. If \( p_i > p_j \), we can say that link i is more important than link j, because it has been used for longer time than link j to be activated in one TDMA frame.

Our aim is to separate the links that need large time percentage allocation and the
links that need small time percentage allocation in different cliques. Therefore, we can use percentage vector as link weight. We sort the link sequence by weight, then the links that need large time percentage allocation are in the previous part of the sequence, and the links that need small time percentage allocation are in the latter part of the sequence. Then we can separate them easily by following this sequence using greedy algorithm.

Above all, the modified algorithm can be described as below:

Algorithm 2:
- Form clique set by TDMA.
- Calculate throughput with proportional fairness using TDMA clique set, and get the optimized percentage vector $p$.
- Sort links in order of decreasing value of $p$.
- Apply greedy Algorithm 1 with sorted links to get clique set.
- Optimize problem again using the clique set above.

With this improved algorithm, we can get more effective clique set than that just from greedy algorithm, thus, the schedule will also be more efficient. The reason why we use proportional fairness is that it is a good trade-off between efficiency and fairness, thus every useful link can get the right to be activated, and contain different importance.

If we only want to maximize throughput in which case node transmits with maximum power or stay silent, the improved algorithm is the same as in base rate or zero case.

5.2 Heuristic Algorithm for Weighted Throughput Optimization

As shown in [15] by Johansson and Xiao, the optimal clique to introduce into the clique set is one that solves the weighted throughput problem

$$\text{Maximize } \sum_i \lambda_i \varphi_i(y_i)$$

(10)
subject to antenna and transmit power constraints. Here, $\varphi_l(\gamma_l)$ is the transmission rate over link $l$ as function of $\gamma_l$, the SINR value on the same link. The weights $\lambda_l$ are the Lagrange multipliers for the capacity constraints of link $l$ obtained by solving the optimization problem (3).

Unfortunately, the weighted throughput problem turns out to be NP hard for the radio link models that we consider, which limits the networks that can be solved using the method presented in [15]. Rather than solving the weighted throughput problem to optimality (using a global optimization method such as branch-and-bound), we will derive heuristic methods to solve the problem for the two radio link models described above (Chapter 2.3).

For the case that all the rates over active links are the same and the rates of the other links are 0, we suggest a heuristic algorithm that tries to activate links in order of decreasing value of $\lambda_l$ under SIR balancing and add this clique to the clique set.

Algorithm 3:
- Form clique set by TDMA.
- Beginning of loop.
  - Until the optimized throughput value is stable.
  - Optimize throughput using current clique set, and get Lagrange multipliers vector $\lambda_l$.
  - Activate links in order of decreasing value of $\lambda_l$ under SIR balancing to get new clique.
  - Add new clique to clique set.
- End of loop.

Here, the optimized throughput is stable if the increased optimized throughput is less than a certain value over several iterations.

If node transmits with maximum power or stay silent that can maximize the throughput in our optimization models, we propose a different greedy algorithm with the one for the SIR balancing case, since the rate is different for the different link
reuse for the same link. We evaluate the benefit of adding each of the non-active links to the clique, add the link that causes the highest increase in weighted throughput, and repeat it. When the loop is finished, we add this clique to the clique set, and optimize the throughput.

Algorithm 4:

- Form clique set by TDMA.
- Beginning of loop 1.
  - Until the optimized throughput value is stable.
  - Optimize throughput using current clique set, and get Lagrange multipliers vector \( \lambda_i \).
  - No links are active in the new clique.
- Beginning of loop 2.
  - Until no increase in weighted throughput.
  - Evaluate the benefit of adding each of non-active links to the new clique, and then add the link that causes the highest increase in weighted throughput to the new clique.
- End of loop 2.
- Add new clique to clique set.
- End of loop 1.

6 Examples

In this section, we will implement our optimization model and scheduling methods in the sample networks. Through these examples, we try to investigate the performance of the optimization model with different clique sets, the objectives (throughput and fairness) that we get from optimization model, and the optimized relation between network throughput and power consumption.

Here we use the same network environment as in [8], since the sample network environment in [8] is used to get capacity regions of wireless ad-hoc networks. The sample network has a random topology obtained by uniform and independent nodes in
the box \( \{10m \leq x \leq 10m, 10m \leq y \leq 10m\} \). The power gains between nodes \( N_i \) and \( N_j \) are given by \( G_{ij} = KS_{ij} \left( \frac{d_0}{d_{ij}} \right)^\alpha \), where \( d_{ij} \) is the distance between the nodes, \( K \) and \( d_0 \) are normalization constants set to \( K = 10^{-6} \) and \( d_0 = 10m \) respectively, the path loss exponent \( \alpha \) is set to \( \alpha = 4 \), and the shadowing factors \( S_{ij} = S_{ji} \) are random, independent, and identically generated from a lognormal distribution with a mean of 0 \( db \) and variance \( \sigma = 8db \) (so \( S_{ij} = 10^{\frac{N_i}{10}} \) and \( N_j \) is Gaussian with expectation \( EN_{ij} = 0 \) and standard deviation \( \sigma_{N_j} = 8 \)). The maximum transmitted power is \( P_{\text{max}} = 0.1W \). All receivers are subject to AWGN noise with the same power spectral density \( \eta = 10^{-10}W/Hz \) and bandwidth \( W = 10^6 Hz \), \( P_{\text{noise}} = \eta W \). We assume that the node can transmit data to another node if the distance between these two nodes is below one constant value, and the node can reach any other nodes by multihop. In the examples below, we will consider two cases respectively, case 1: base rate or zero over the links, and case 2: maximum power transmission over the active links.

<table>
<thead>
<tr>
<th>Description</th>
<th>Routing</th>
<th>Heuristic algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>Base rate or zero over links</td>
<td>Free</td>
</tr>
<tr>
<td>Case 2</td>
<td>Maximum power transmitting over active links</td>
<td>Free</td>
</tr>
<tr>
<td>Sample network 1</td>
<td>20 random nodes, 106 links</td>
<td>Free</td>
</tr>
<tr>
<td>Sample network 2</td>
<td>10 random nodes, 46 links</td>
<td>Free</td>
</tr>
</tbody>
</table>

Table 1 Different cases and networks in the examples
6.1 Comparison of Different Clique Sets

We select pairs randomly in the sample network 1 which consists of 20 random nodes and 106 links and we set all the links’ SINR threshold $\gamma = -2.6dB$, for both cases. We calculate the throughput (no fairness) with increasing pairs from 1 to 15. The results are shown in Figure 4 and 5. We find out that Algorithm 3 and Algorithm 4 always can get the highest network throughput amongst these five clique sets. Algorithm 3 and 4 (80 iterations) add suboptimal clique to the previous clique set for improving the result in every iteration, thus, the clique set is closer and closer to optimal clique set, which makes the throughput higher than with other algorithms.
The throughput gain got by Algorithm 2 is larger than Algorithm 1, since the clique set by Algorithm 2 is more efficient than Algorithm 1. TDMA always get the lowest throughput, because of no link reuse in each time slot. In the case 2, the throughput by Algorithm 1 does not increase much compared to the throughput by TDMA, it may be because the link reuse also increases the interference level in the network which reduces the throughput increase from link reuse, while in the case 1, all the rates over active links is the same, which means the increased interference level will not decrease the data rate over the active links.
6.2 Performance Measures of Heuristic Algorithm 3 and 4

Since Algorithm 3 and 4 perform better in terms of throughput, we try to investigate the performance of Algorithm 3 and 4 more in detail. Here, we implement these two algorithms in one small network (sample network 2) that consists of 10 random nodes and 46 links, in the sample network 2, the other network parameters are the same as that of sample network 1 except the SIR threshold which is now $\gamma_0 = -33dB$. We assume that there are 3 source-destination pairs in the sample network 2: pair 1 (-9.63, 8.34) to (7.82, -6.47), pair 2 (-0.87, 8.70) to (6.42, -1.79), pair 3 (-5.37, 5.83) to (9.00, 2.30). We set the number of iterations to 80, the results are shown in Figure 7 and 8.

![Sample network 2](image)

**Figure 6 Sample network 2**

The results got by direct way which tries all possible link combinations are also shown in Figure 7 and 8 for comparison reasons and these results are the optimal results of the problems, but obviously, the computation of this method is huge. As mentioned in [15], the optimal clique to introduce into the clique set is the one that solves the weighted throughput problem (10) subject to antenna and transmit power constraints. Therefore, in the Figure 7 and 8, we also implement the method that adds
clique to current clique set by solving problem (10) directly. This method has a very good convergence performance which is shown in [15] and should be better than Algorithm 3 and 4. Thus, we apply it to measure performance of our heuristic Algorithm 3 and 4,

![Convergence performance of Algorithm 3](image1)

**Figure 7** Convergence performance of Algorithm 3 (case 1)

![Convergence performance of Algorithm 4](image2)

**Figure 8** Convergence performance of Algorithm 4 (case 2)

but obviously this method also costs high computation. Here, let’s call this method Algorithm 5. We find out that the throughput by Algorithm 3 and 4 converge fast to high throughput, but the convergence speed is a little slower than with Algorithm 5. In Figure 7, the throughputs of Algorithm 3 only iterate 11 times to be converged to
optimal result. Even though, it is still slower than that of algorithm 5 which costs only 9 times. The difference between results from Algorithm 3 and 5 is very small, but Algorithm 5 costs high computation. Algorithm 4 also makes the result converge to a throughput which is very close to the optimal result within 17 iterations. It seems that the result can not converge to optimal throughput within 80 iterations, while Algorithm 5 in Figure 8 can converge to optimal result within 21 iterations. Since the throughput that Algorithm 4 converges to is not much less than the optimal result, the convergence performance is also very good. After all, both of these two algorithms show very good convergence performance. In order to investigate how close the results by Algorithm 3 and 4 are to optimal result by direct method, we implement Algorithm 3 and 4 in 100 random 10-node networks. In each network, we select 4 source-destination pairs randomly, and calculate throughput by iterating 50 times. Finally we compare this throughput with optimal result which is reached by direct method.

![Figure 9 Relative performance losses distribution of Algorithm 3](image)

**Figure 9 Relative performance losses distribution of Algorithm 3 (case 1)**

We assume that the SIR threshold in all the networks is -33dB, and we measure the relative performance losses of heuristic algorithm 3 and 4 using loss percentage. We define loss percentage as \( \frac{opt\_true - opt\_34}{opt\_true} \) where opt\_true is true
optimality, and \( opt_34 \) is the result achieved by Algorithm 3 or 4. Figure 9 and 10 are the histograms which show the distribution of loss percentage.

![Figure 10 Relative performance losses distribution of Algorithm 4](image)

In Figure 9, 92% of loss percentage are less than 1.4% and in Figure 10, 89% loss percentages are less than 2.3% after 50 iterations, only a few loss percentages are larger than 5% in both of these two figures. If we iterate more than 50 times, it is possible that the loss percentages which is more than 5% will be reduced. In a word, in most sample networks, the results of Algorithm 3 and 4 are very close to optimal results, while the number of iterations is only 50. Thus, we can also regard the stable result by Algorithm 3 and 4 as near-optimal result in most network cases.

As shown in the examples above, Algorithm 3 and 4 are heuristic algorithms, thus, the computation is much less than both Algorithm 5 and the direct method which have to try all possible link combinations, while the convergence result is close to optimal result in a finite number of iteration. All in all, if we consider the convergence performance and the computation complexity together, these two algorithms perform excellently to reach near-optimal results.

### 6.3 Performance Measures of Wireless Ad-hoc Network

As we know, there are several measure parameters we should concern in the wireless
ad-hoc network: throughput, fairness, and power. We focus on throughput and fairness here with the optimization model (3) in which the routing is not fixed. We assume that there are 3 source-destination pairs in the sample network 1: pair 1 (-9.73, -7.75) to (9.04, 9.64), pair 2 (-7.62, 8.54) to (8.94, -8.05), pair 3 (-6.13, 2.19) to (8.47, -3.33) and $\gamma_0 = -2.6 dB$. We use standard deviation of all rates to decide which result can get more fairness. The results are shown below:

Case 1:

<table>
<thead>
<tr>
<th></th>
<th>No fairness</th>
<th>Max-min fairness</th>
<th>Proportional fairness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network throughput</td>
<td>833</td>
<td>681</td>
<td>694</td>
</tr>
<tr>
<td>Single pair rates</td>
<td>0, 0, 833</td>
<td>227, 227, 227</td>
<td>208, 208, 277</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>481</td>
<td>0</td>
<td>40</td>
</tr>
</tbody>
</table>

Table 2: Results with TDMA clique set (bps)

<table>
<thead>
<tr>
<th></th>
<th>No fairness</th>
<th>Max-min fairness</th>
<th>Proportional fairness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network throughput</td>
<td>3750</td>
<td>3368</td>
<td>3680</td>
</tr>
<tr>
<td>Single pair rates</td>
<td>866, 594, 2288</td>
<td>1123, 1123, 1123</td>
<td>1023, 1283, 1372</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>909</td>
<td>0</td>
<td>181</td>
</tr>
</tbody>
</table>

Table 3: Results with clique set from algorithm 3(bps), iteration times: 80

Case 2:

<table>
<thead>
<tr>
<th></th>
<th>No fairness</th>
<th>Max-min fairness</th>
<th>Proportional fairness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network throughput</td>
<td>6003</td>
<td>3854</td>
<td>4179</td>
</tr>
<tr>
<td>Single pair rates</td>
<td>0, 0, 6003</td>
<td>1185, 1185, 1185</td>
<td>1124, 1057, 1998</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>3466</td>
<td>0</td>
<td>525</td>
</tr>
</tbody>
</table>

Table 4: Results with TDMA clique set (bps)

<table>
<thead>
<tr>
<th></th>
<th>No fairness</th>
<th>Max-min fairness</th>
<th>Proportional fairness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network throughput</td>
<td>17500</td>
<td>15477</td>
<td>16800</td>
</tr>
<tr>
<td>Single pair rates</td>
<td>3619, 205, 13674</td>
<td>5159, 5159, 5159</td>
<td>6285, 4435, 6078</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>7002</td>
<td>0</td>
<td>1113</td>
</tr>
</tbody>
</table>

Table 5: Results with clique set from algorithm 4(bps), iteration times: 80

As we can see from these tables above, if we apply TDMA clique set in both of the two cases, the result without fairness will cause all flows passed on the most efficient pair, while the other pairs can not get any rates. In order to allocate some flows on the other pairs, we can optimize throughput with max-min fairness or
proportional fairness, although some throughput will be lost after optimization. From Table 2 or Table 4, it seems that the result with proportional fairness is a good trade-off between efficiency and fairness.

In Table 3 and 5, we find out that the objective function without fairness can get certain degree of fairness in the result, because the links belong to different pairs are included in the same clique. Therefore, not only the most efficient pair but also other pairs can get data flows. The result with proportional fairness is still a good trade-off between efficiency and fairness, which is larger than the result with max-min fairness, and still maintains much fairness.

6.4 Relation between Throughput and Power Consumption

Through model (4), we can minimize the total power in the network and maximize the throughput with this power, thus, it exists the region of power and throughput.

Figure 11 Optimized throughput and power with TDMA clique set (case 1)

Figure 12 Optimized throughput and power with clique set from Algorithm 1 (case 1)
Figure 13 Optimized throughput and power with clique set from Algorithm 2 (case 1)

Figure 14 Optimized throughput and power with clique set from Algorithm 3 (case 1)

Figure 15 Optimized throughput and power with TDMA clique set (case 2)
Here, we set the constant $a = 10^4$ in the model (4), and use the same network and pairs as those in section 6.3, then calculate throughput (no fairness) and power consumption by changing $\lambda$ from 0 to 1. The results are shown in Figure 11-18.

Since the minimum power consumption and maximum throughput are always the objectives of the network design, in the model (4), if we set one $\lambda$, we can get minimum power and maximum throughput. Therefore, we can regard the result, such as that in Figure 11 to 18, as the network design criteria. For example, if we need total power consumption no more than 0.2W in one frame when we use clique set from Algorithm 2 to optimize this network, then for the whole network the maximum throughput we can reach is about 1800bps in case 1 (see Figure 13). Through this method, we can decide if 0.2W is enough for our demand. It is also a good way to see the trade-off between power and throughput.
In Figure 14 and 18, the plots have a little difference with other plots, and it may be because at some points in Figure 14 and 18, the convergence speed is a little bit slower than at other points, which make the value at this point probably smaller than at other previous points.

Another useful result from model (4) is power efficiency. We define power efficiency as throughput divided by power consumption. From the result above, we get power efficiency figures shown in Figure 19 and 20.
We find out that TDMA clique set has the highest power efficiency among these five clique sets in Figure 19 and 20, probably because it has no link reuse. In the TDMA clique, there is only one node consuming power in one time slot, while in the link reuse case, several nodes may be transmitting data simultaneously in one time slot. This result shows that TDMA is a good clique set for the network in which low power consumption is more important than high throughput. But if the network need high throughput, which also means high data traffic, the STDMA approach would be a better solution. The other three clique sets have almost the same power efficiencies in case 1, however, in case 2, Algorithm 4 performs better than Algorithm 1 and 2. It is probably because in case 2, if too many links are reused in one clique, the weighted throughput may be reduced due to the increased interference level. It makes this clique with high link reuse not a good choice for being added to the clique set. It also means that high network power consumption probably does not always help to increase the throughput.

6.5 Computational Aspect

We note that the scheduling by heuristic algorithms is not optimal, and if we want to get optimal scheduling directly, we have to try all possible link combinations in the clique set. Thus, the computation will be of exponential complexity $2^L$, where $L$ is the number of links, which is only feasible for very small network. While with Algorithm 1 and 2 used in the examples, the maximum possible link combination is N, which makes the optimization model feasible for large network in reality. For the Algorithm 2, there are two iterations, because the TDMA scheduling should be done first. Thus, the computation will be larger than for Algorithm 1, which is the cost of using Algorithm 2 to reach higher throughput. Algorithm 3 and 4 reduce considerably the computation to get near-optimal result due to good convergence performance, thus, it is possible to reach the near-optimal result with reasonable computation. In the section 6.1, it costs 80 iterations for Algorithm 3 and 4 to reach near-optimal result, 2 iterations for Algorithm 2 to reach result, and 1 iteration for Algorithm 1. Our original
optimization model is the NP hard problem, but it is proved that the optimal result can be obtained if the node transmits with maximum power or stays silent. Thus, it makes our models become LP models, and furthermore, these optimization problems are convex optimization problems that can be efficiently solved by recently developed Interior point method [11].

7 Conclusion

We have developed an integrated routing, scheduling and power control policy for wireless ad-hoc network that maximizes end-to-end rates or throughput, and minimizes the power consumption based on certain scheduling. If the routing is fixed, we can use model (2, page 17) to optimize throughput, and if routing is not decided before optimization, we can use model (3, page 20) to get the results. The optimal result would be obtained by nodes that transmit with maximum power or stay silent. We also consider the case in which we apply SIR balancing in the network with the same SIR threshold. The objectives that we can optimize include efficiency and fairness, and we investigate throughput with no fairness, max-min fairness and proportional fairness in the sample network 1. We find that if we do not consider fairness, the throughput is the maximum, while the throughput with proportional fairness is a good trade-off between efficiency and fairness.

Compared with previous works, our method has less computation because we use the greedy algorithm to get clique set which reduces the computation complexity from $2^L$ to L, where L is the number of links. We also propose one improved greedy algorithm to calculate clique set. By this algorithm, we can improve network throughput significantly compared with the greedy algorithm. It is because we find that after we have optimized the proportional throughput, we can know which link is more important by the percentage vector, which is part of the result. In sample network 1, we can see the throughputs with Algorithm 1 and 2 are higher than with TDMA.

It is not rational to get optimal result by trying all possible link combinations
directly in the large network due to high computation. In order to get a result as high as possible without much computation, we propose two heuristic iterative algorithms which use weighted throughput for different cases. It is shown in the examples that these two algorithms perform very well, and can converge fast to near-optimal value with finite iteration.

We also found that the relation between maximum throughput and minimum power consumption can be obtained by optimization model (4). The figures from the result of model (4) provide us one method to design one ad-hoc network that satisfies the power or throughput demand. We also notice that the TDMA clique set is proper for the small traffic case, since it has higher power efficiency, while STDMA is better for high traffic routing.

<table>
<thead>
<tr>
<th></th>
<th>TDMA</th>
<th>Algorithm 1 and 2</th>
<th>Algorithm 3</th>
<th>Algorithm 4</th>
<th>Direct method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation</td>
<td>small</td>
<td>small</td>
<td>medium</td>
<td>medium</td>
<td>huge</td>
</tr>
<tr>
<td>Throughput</td>
<td>small</td>
<td>medium</td>
<td>near-optimal</td>
<td>near-optimal</td>
<td>optimal</td>
</tr>
<tr>
<td></td>
<td>result</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power efficiency</td>
<td>high</td>
<td>low</td>
<td>low</td>
<td>medium</td>
<td>unknown</td>
</tr>
</tbody>
</table>

Table 6 Comparison of different algorithms for calculating clique set (case1&2)

In conclusion, as shown in Table 6, we if we want to get high throughput integrating routing and scheduling in large wireless ad-hoc network without much computation, we can implement Algorithm 1 or 2. While if we need near-optimal result with reasonable computation, we can apply Algorithm 3 or 4 to get clique set. However, if we consider power efficiency, the TDMA clique set has more advantage than all other clique sets.

In this thesis, we only consider two radio link rate models and one specific example network. In the future, our models and algorithms can be studied in the more radio link rate models and more types of networks. For example, we can try to use directional antenna in the network and these algorithms also can be tested in CDMA networks. In these different cases, to see the performance of our models and algorithms would be interesting.
References


Organization and structure of the report

Your thesis is well presented with nice plots and clarifying tables. You also have a good structure, but as a little comment, I would recommend you to have the review of the related work section a little bit earlier in the report, to create a subsection in the examples section with the system model, to create a future work section and to start each chapter in a new page.

Literature review

I think you have done a deep study on the previous work related to several topics interesting for your work achieving a good background before starting working on the thesis. Really good literature review work.

Used methods

In your thesis you have considered many parameters and conditions, and combined them properly. You have done a wide study within ad-hoc wireless networks considering fixed and not fixed routing, scheduling and power control in order to maximize throughput and minimize power consumption.

You have studied several optimization models depending on what you wanted to optimize and the characteristics of the network: optimize throughput for fixed/non-fixed routing, transport capacity, to have fairness (max-min or proportional), optimize throughput and power consumption, etc.

Relevance and meanings of conclusions

From my point of view I would say that your results are quite relevant due to the introduction of those 4 algorithms which reduce considerably the computation cost. Another important fact is that your results group together lots of parameters which were not considered in previous studies.

I want to emphasize the good idea to show in a table which would be the most suitable algorithm to use depending on the necessities.

Language and graphical presentation of the results

You have some language mistakes which I have tried to correct and that you can take a look in the corrected report that I attach to this one. It was quite clear what you wanted to do but sometimes the explanation was a little bit complicated.

The graphics are quite good, showing interesting results in a few plots, although some of the curves could be a little bit smoother. A good point is that you did not fill the report with useless graphics, but concentrated on the most important ones.
Other comments and questions

1) Graphics and tables
In figure 5 (p.31), could you explain why there is a large throughput increase for the TDMA case when going from 9 to 10 pairs?
In figure 9 and 10, could you explain briefly what those plots show?
In figure 11 to 18, why are there 2 curves for the lambda plot?
Could you explain the strange shapes of the curves in figures 14, 15 and 18?
In the tables that you show in p.36, do you know why you get such a large standard deviation when there is no fairness?

2) Other questions
When you talk about ad-hoc networks, do you only consider multihop networks?
(In page 17, could you explain what you mean by convex hull of the instantaneous achievable rate regions? And in page 18 – convex optimization problem?)
Did you expect that the optimized STDMA solutions would have lower power efficiency than the TDMA solution?

3) Recommendations
It would be a good idea to give one example in section 2.4 to see clearly what p, Y, etc are.

4) Other comments
There are some things that should be referenced:
   p.15 – Σ log(si) for proportional fairness
   p.21 – λf(k) – a(1-λ)(2 powerij)p
There are some things repeated:
   p.21, 25, 26, 29
In 6.1 you should say how many iterations you are considering.
In figures 4 and 5 the x-axis legend should be: Source-destination number of pairs.
It should be clearly explained that Algorithm 3 and 4 cannot be compared in the same figure.

Overall conclusion about the report

Although there are some language mistakes the work done is wide and very interesting, arriving to some important algorithms which group a lot of conditions and consider many parameters. Good work.