Report 3 on Project 15 – Capacity, Coverage and Connectivity of Wireless Networks

Group 20: Xinzhi Zou, Shaofeng Zou, Qian Cai, Chenchen Fan

***Abstract*—In this report, we present our study on the capacity of arbitrary wireless networks. Based on the previous study on the capacity of wireless networks, we choose the arbitrary wireless network as a typical research aspect. First, we briefly recall the previous work. Then we introduce some related work and results, after which three aspects on the capacity scaling of arbitrary wireless networks-routing scheme, scheduling algorithm and power control-will be discussed respectively. We focus on the routing scheme and scheduling algorithm, and conduct simulation on scheduling algorithm to get better understanding. In the end, we conclude the effort in the whole semester and express our thoughts on this course.**

***Index Terms*—capacity, arbitrary networks, cooperative communication, scheduling algorithm, power control**

I. Introduction

In previous work,our group read the recommended papers on the course website, and then some more research papers published recent years to get a full perspective on the related subsection and the prevailing trend. The capacity of wireless networks was first studied in a landmark seminal work by Gupta and Kumar [1]. It has sparked a growing amount of interest in the understanding of the fundamental capacity limits of wireless ad hoc networks. This kind of work includes consideration on some fundamental characteristics such as mobility [2] [3], and research on different types of wireless network such as hybrid wireless networks [4]. There are some more different approaches to increase the network throughput, such as the scheduling on the MAC layer, route selection on the routing layer, different packet forwarding methods listed as unicast, multicast and broadcast, and power control on the physical layer. Based on these understanding, we focus our study on the capacity of arbitrary wireless networks, about which very little is understood when compared with the random wireless networks and the worst-case networks.

This report is organized as follows. The next section we introduce two interference models widely used on the capacity scaling of wireless networks, and the following study is mainly based on the physical model. In section III we present some related work respectively on communication schemes, scheduling algorithm and power control, which are all aimed at improving the performance of arbitrary wireless networks. In section IV, we describe two communication schemes - the hierarchical relaying scheme and the cooperative multi-hop scheme. In section V, we go on to present two scheduling algorithm, based on which we discuss our simulation work. And in section VI we present one approach on power control to maximize the capacity in arbitrary wireless networks. In the last section we conclude the effort in the whole semester and express our thoughts on this course.

II. Two Models

In the arbitrary setting nodes are arbitrarily located in a disk of unit area in the plane. Each node has an arbitrarily chosen destination to which it wishes to send traffic at an arbitrary rate; thus the traffic pattern is arbitrary. Each node can choose an arbitrary range or power level for each transmission.

An important issue when studying the capacity for wireless networks is how to model interference. To describe when a transmission is received successfully by its intended recipient, two possible models for successful reception of a transmission over one hop, called *the Protocol Model* and *the Physical Model*, are described below [1]. Let X_i denote the location of a node; we will also use X_i to refer to the node itself.

1) The Protocol Model: Suppose node X_i transmits over the mth subchannel to a node X_j. Then this transmission is successfully received by node X_j if

$$|X_k - X_j| \geq (1 + \Delta)|X_i - X_j| \tag{1}$$

for every other node X_k simultaneously transmitting over the same subchannel.

The quantity $\Delta > 0$ models situations where a guard zone is specified by the protocol to prevent a neighboring node from transmitting on the same subchannel at the same time. It also allows for imprecision in the achieved range of transmissions.

Another model which is more related to physical layer considerations is

2) The Physical Model: Let $X_k; k \in \Gamma$ be the subset of nodes simultaneously transmitting at the time instant over a certain subchannel. Let P_k be the power lever chosen by node X_k, for $k \in \Gamma$. Then the transmission from a node $X_i, i \in \Gamma$, is successfully received by a node X_j if

$$\frac{\frac{P_i}{|X_i - X_j|^\alpha}}{N + \Sigma_{k\in\Gamma, k\neq i} \frac{P_k}{|X_i - X_j|^\alpha}} \geq \beta \tag{2}$$

This models a situation where a minimum signal-to-interference ratio (SIR) of β is necessary for successful receptions, the ambient noise power level is N , and signal power decays with distance r as $\frac{1}{r_\alpha}$. The assumption $\alpha > 2$ is the usual model outside a small neighborhood of the transmitter.

The protocol model typically defines a set of interference edges, containing pairs of nodes within a certain distance to each other, thus modeling interference as a binary and a local measure. Such models serve as a useful abstraction of wireless networks; they facilitate the process of designing protocols and proving their efficiency, but are subject to several limitations. Although the interference of a single far-away transmitter can be relatively small, the accumulated interference of several such nodes can be sufficiently high to corrupt a transmission. Therefore protocols based on localized interference models that simply ignore interference beyond a certain range are not guaranteed to work in a real scenario.

The physical model offers a more realistic representation of wireless communication. A signal is received successfully if the SINR, the ratio of the received signal strength to the sum of the interference caused by all other nodes sending simultaneously plus noise, is above a hardware-defined threshold. This definition of a successful transmission, as opposed to the protocol model definition, accounts also for interference generated by transmitters located far away.

III. Related Work

A. On Communication Scheme

The problem of scaling the capacity of wireless networks has received a considerable amount of attention. One stream of work [5] [6] [7] has progressively broadened the conditions on the channel model and the communication model, under which multi-hop communication is order optimal. Specifically, with a power loss of r_{-a} for signals sent over distance r, it has been established that under high signal attenuation $\alpha > 3$ and random node placement, the best achievable per-node rate for random source-destination pairing scales essentially like $\Theta n^{-1/2}$ and that this scaling is achievable with multi-hop communication. Another stream of work [8] [9] [10] has proposed progressively refined multi-user cooperative schemes, which have been shown to significantly out-perform multi-hop communication in certain environments. It has been shown [8] that for $\alpha \in (2, 3]$, the best achievable per-node rate for random source-destination pairing scales as $O(n^{1-\alpha/2+\epsilon})$ and cooperative communication achieves a per-node rate of $\Omega(n^{1-\alpha/2-\epsilon})$ (here, $\epsilon > 0$ is an arbitrary but fixed constant). That is, cooperative communication is essentially order optimal in the attenuation regime $\alpha \in (2, 3]$.

B. On Scheduling Algorithm

Throughput capacity of randomly deployed wireless networks has been intensely studied from the information theory perspective, such as in [11]. However, in practical, networks with heterogenous topologies may be more common than randomly-deployed networks.

Scheduling algorithms in graph-based models usually employ some sort of matching or coloring, and have been widely studied, such as in [12], but the inefficiency of graph-based scheduling protocols in the SINR model is well documented and has been shown theoretically and experimentally, [13].

In [14], a greedy scheduling algorithm with approximation ratio of $O(n^{1-2/(\psi(\alpha)+\epsilon)}(\log n)^{2/(\psi(\alpha)+\epsilon)})$, where $\psi(\alpha)$ is a constant that depends on the path-loss exponent α, is proposed. This result, however, holds only under the assumption that nodes are distributed uniformly at random in a square of unit area. In [15], an algorithm with a factor $O(g(L))$ approximation guarantee in arbitrary topologies, where $g(L)$ is the so called diversity of the network, is proposed. The diversity depends on the topology of the network and captures the variation in the lengths of the links to be scheduled. The problem is that the diversity of a network can be as large as n. In [16], an algorithm with approximation guarantee of $O(\log \Delta)$ was proposed, where Δ is the ratio between the maximum and the minimum distances between nodes. This parameter can be arbitrarily large (note that $g(L) \geq \log \Delta$).

C. On Power Control

In [17], Saraydar et al. look at a game-theoretic algorithm for choosing powers on the uplink of a single cell wireless system. In [18], Stolyar and Viswanathan study fractional frequency reuse algorithms for joint channel assignment and power control in cellular OFDM systems and provide a game theoretic algorithm that always leads to a stable solution. In [19], Bahl et al. provide distributed algorithms inspired by game theory for the problem of sizing cells and assigning users to base stations. Goussevskaia et al. [15] show NP-hardness and provide $O(\log d_{max})$ approximation algorithms to maximize the capacity. They also consider a related objective of minimizing the number of "rounds" required to serve all connections.

IV. Cooperative Communication Scheme

The characterization of the scaling of networks capacity as a function of the path-loss exponent α mentioned in the last paragraph depends critically on the regularity induced with high probability by placing the nodes uniformly at random. However, a wireless network encountered in practice might not exhibit this amount of regularity. A novel cooperative communication scheme is present in [20]. The cooperative communication scheme is essentially order optimal for any such arbitrary network with $\alpha \in (2, 3]$. The situation is quite different for large path-loss exponents $\alpha > 3$. In this regime the scaling of capacity depends crucially on the regularity of the node placement, and multi-hop communication may not be order optimal. In the paper mentioned above, a communication schemes that smoothly "interpolate" between cooperative communication and multi-hop communication is presented. The amount of interpolation between the cooperative and multi-hop schemes depends on the level of regularity of the underlying node placement. This schemes achieves best capacity for all $\alpha > 3$ under adversarial node placement.

A. Hierarchical relaying scheme for $\alpha \in (2, 3]$

The construction of Hierarchical relaying scheme is as follows. Consider n nodes $V(n)$ placed arbitrarily on the square region $A(n)$ with a minimum separation r_{min}. Divide $A(n)$ into squarelets of equal size. Call a squarelet dense, if it contains a number of nodes proportional to its area. For each source-destination pair, choose such a dense squarelet as a relay, over which it will transmit information.

If we assume for the moment that all the nodes within the same relay squarelet could cooperate then we would have a multiple access channel (MAC) between the source nodes and the relay squarelet, where each of the source nodes has one transmit antenna, and the relay squarelet (acting as one node) has many receive antennas. Between the relay squarelet and the destination nodes, we would have a broadcast channel (BC), where each destination node has one receive antenna, and the relay squarelet (acting again as one node) has many transmit antennas. The cooperation gain from using this kind of scheme arises from the use of multiple antennas for these multiple access and broadcast channels. To calculated

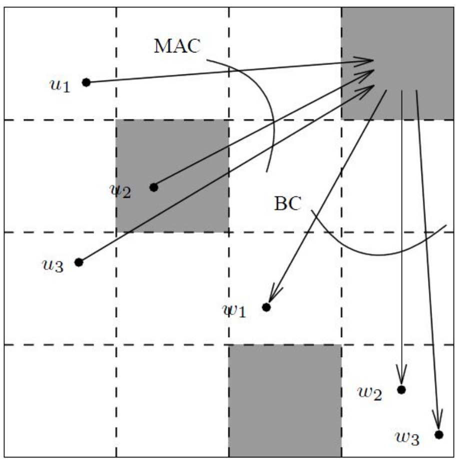

Fig. 1. Sketch of one level of the hierarchical relaying scheme

the achievable rate per node, instead of computing the rate achieved by hierarchical relaying, it will be convenient to instead analyze its inverse, i.e., the time utilized for transmission of a single message bit from each source to its destination under a permutation traffic matrix $\lambda(n)$. With the transmission time of MAC phase and BC phase, it is shown that the per-node rate of the hierarchical relaying scheme is lower bounded as

$$\begin{aligned} \rho^{HR}(n) &\geq b(n)n^{1-\alpha/2} \\ b(n) &\geq n^{-O(\log^{\delta-1/2}(n))} \end{aligned} \quad (3)$$

B. Cooperative multi-hop scheme for $\alpha > 3$

A node placement $V(n)$ is -regular at resolution $d(n)$ if every square $[id(n),(i+1)d(n)] \times [jd(n),(j+1)d(n)]$ for some $i,j \in N$contains at least $d^2(n)$ nodes. Given such a node placement $V(n)$, divide it into squares of sidelength $d(n)$. Consider four adjacent squares, combined into a bigger square of sidelength $2d(n)$ which contains at least $4d2(n)$ nodes. Within this bigger square at a per-node rate is

$$b(n)(d^2(n))^{1-\alpha/2} = b(n)d^{2-\alpha}(n) \quad (4)$$

where $b(n)$ is essentially of order $n^{-\log^{-1/2}(n)}$. When implemented properly, the edge between squarelets has a capacity of

$$d^2(n)b(n)d^{2-\alpha}(n) = b(n)d^{4-\alpha}(n) \quad (5)$$

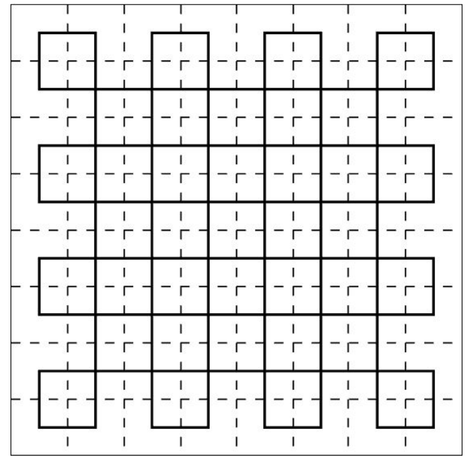

Fig. 2. Communication graph (in bold) resulting from the construction of the cooperative multi-hop scheme

To send a message from a source node in $V(n)$ to its destination node, first locate the squares of sidelength $d(n)$ they are located in. Then route the message over the edges of the communication graph constructed above in a multi-hop fashion. By the construction of the communication graph, each such edge is implemented using the hierarchical relaying scheme. In other words, we perform multihop communication over distance $\sqrt{n}$ with hop length $d(n)$, and each such hop is implemented using hierarchical relaying over distance $d(n)$. Since each edge in the communication graph has a capacity of $b(n)d^{4-\alpha}(n)$ and has to support roughly $n^{1/2}d(n)$ source-destination pairs, we obtain a per-node rate of

$$\rho^{CMH}(n) \geq b(n)d^{4-\alpha}(n)n^{-1/2}d^{-1}(n) = b(n)d^{3-\alpha}(n)n^{-1/2} \quad (6)$$

per source-destination pair.

V. Scheduling Algorithm and Simulation

A. Model for Scheduling

The scheduling problem can be formulated as follows [21]. Given a set of links $L = \{l_1, ..., l_n\}$,where each link l_v represents a communication request from a sender s_v to a receiver r_v, two objectives can be defined: (1)maximizing the number of links scheduled concurrently in one time-slot,and (2)schedule all the requests in as few time-slots as possible. We assume that each link has a unit-traffic demand, and model the case of non-unit traffic demand by replicating each link x times, where x is the demand on the link. All nodes are positioned in Euclidean space. The distance between two nodes s_v, r_w is denoted by $d_{vw} = d(s_v, r_w)$. The length of link l_v is denoted by d_{vv} .

We use the physical interferene model. In this model, a node r_v successfully receives a message from a sender s_v if and only if the following condition holds:

$$\frac{\frac{P}{d_{vv}^{\alpha}}}{\sum_{l_w \in S_t l_v} \frac{P}{d_{vv}^{\alpha}} + N} \quad (7)$$

Where P is the power level of the transmission, $\alpha > 2$ is the path-loss exponent, $\beta > 1$ denotes the minimum SINR required for a message to be successfully received, N is the ambient noise, and S_t is the set of concurrently scheduled links in slot t.

Here, we assume that all noes transmit with the same power level P. Nevertheless, this analysis holds in case nodes transmit with different but fixed power levels, provided that either the ratio P_{max}/P_{min} between the maximum and the minimum power levels is bounded by a constant, or there are only a constant number of possible power levels.

We use the notation $P_{vv} = P/d_{vv}^{\alpha}$ to denote the power received by receiver r_v from its intended sender s_v, and $I_{vw} = P/d_{vw}^{\alpha}$ to denote the interference received by receiver r_w from a concurrently scheduled sender s_v.

B. Algorithm for Scheduling

In order to solve the minimum-length scheduling problem, a "master-slave" approximation strategy is used, where the "slave" problem is the one-slot scheduling. The one-slot scheduling problem is a maximization problem that, given an input set of links L, has the objective to maximize the number of links to be scheduled successfully in a single time-slot. Firstly we show that one-slot scheduling algorithm has constant approximation guarantee. Thereafter we show that by iteratively computing constant approximations of maximum one-slot schedules, we can obtain a factor $O(\log n)$ for the overall minimum-length scheduling problem.

Some definitions are made first. The $relative interference(RI)$ of a link l_u is the increase caused by l_u in the inverse of the SINR at l_v, namely $RI_u(v) = I_{uv}/P_{vv}$. The affectedness of link l_v,caused by a set S of links, is the sum of the relative interferences of the links in S on l_v, as well as the effect of noise, scaled by β, or

$$\begin{aligned} a_s(l_v) &= \beta(\frac{N}{P_{vv}} + \sum_{l_u \in S} RI_u(v)) \\ &= \beta \frac{\sum_{l_u \in S} I_u(v) + N}{P_{vv}} \end{aligned} \tag{8}$$

1) Algorithm 1

TABLE I
ALGORITHM 1

Algorithm 1 One-Slot Scheduling Algorithm
1:input: Set of links $L = l_1, ..., l_n$;
2:output: One-slot schedule S;
3:Set c according to (12);
4:repeat
5: Add the shortest link $l_v \in L$ to S;
6: Delete $l_u \in L$, where $d_{uv} = d(s_u, r_v) \leq c \cdot d_{vv}$;
7: Delete $l_w \in L$, where $a_S(l_w) \geq 2/3$;
8:until $L = \emptyset$
9:return S;

The one-slot scheduling algorithm (for a description in pseduo-code see Algorithm 1) greedily schedules links in increasing order of length, i.e., "strong" links are scheduled first. After a link l_v is added to the solution S, its "safety" is guaranteed in two steps. First(line 6 of Algorithm 1), all links l_u(remaining in L) whose senders are within the radius $c \cdot d_{vv}$ of the receiver r_v are removed from L (c is a constant always bigger than 2, and is defined in (3)(12)). Second (line 7 of Algorithm 1), all links l_w, whose affectedness $a_S(l_w)$ rose to or above a threshold of 2/3, are removed. This process is repeated until all links in L have been either scheduled or deleted. The strength of this simple algorithm lies in the combination of elimination steps in lines 6 and 7, which ensures that the greedily constructed solution does not lose its feasibility after addition of new links. And we only prove that the obtained schedule is correct.

2) Correctness of one-slot scheduling

In this section we prove that the solution S obtained in Algorithm 1 is correct, i.e., all selected links can be scheduled concurrently without collisions.

Lemma 1: Algorithm 1 produces a valid solution.

Proof: Let S_v^- be the set of links longer than l_v,i.e., those added after l_v. When a link l_v is added to the solution, its affectedness is less than 2/3, since it has not been deleted in the previous step. Therefore, the interference caused on l_v by concurrently scheduled shorter links (plus the ambient noise N) is $a_{S_v^-}(l_v) < 2/3$. It remains to show that S_v^+ affects l_v by at most 1/3.

First observation is that, by the first elimination criterion of the algorithm, discs D_w of radius $c \cdot d_{ww}$ around each receiver $r_w \in S_v^+$ do not contain any sender $s_z \neq s_w$. Using this fact and the triangular inequality, we can lower bound the distance between any two senders$(s_w, s_z) \in S_v^+$ as $d(s_w, s_z) \geq d((r_w, s_z) - d_{ww} \geq c \cdot d_{ww} - d_{ww} = d_{ww}(c-1) \geq d_{vv}(c-1)$. Therefore discs D_w of radius $d_{vv}(c-1)/2$ around senders in S_v^+ do not intersect.

Next, we partition the sender set in S_v^+ into concentric rings $Ring_k$ of width $c \cdot d_{vv}$ around the receiver r_v. Each ring $Ring_k$ contains all senders $s_w \in S_v^+$ for which $k(c \cdot d_{vv}) \leq d_{wv} \leq (k+1)(c \cdot d_{vv})$. We know that the first ring $Ring_0$ does not contain any sender. Consider all senders $s_w \in Ring_k$ for some integer $k > 0$. All discs of radius $d_{vv}(c-1)/2$ around each s_w must be located entirely in an extended ring $Ring_k$ of area

$$\begin{aligned} A(Ring_k) =& [(d_{vv}(k+1)c + d_{vv}(c-1)/2)^2 - \\ & (d_{vv}kc - d_{vv}(c-1)/2)^2]\pi \\ =& (2k1)d_{vv}^2 c(2c-1)\pi. \end{aligned} \tag{9}$$

Since discs D_w of area $A(D_w) \geq (d_{vv}(c - 1/2)^2\pi$ around senders in S_v^+ do not intersect, and the minimum distance between r_v and $s_w \in Ring_k, k > 0 is k(c \cdot d_{vv})$, we can use an area argument to bound the number of senders inside each ring. The total interference coming from ring $Ring_k, k \geq 1$ is then bounded by

$$\begin{aligned} I_{Ring_k}(l_v) &\le \sum_{s_w \in Ring_k} I_{s_w}(l_v) \\ &\le \frac{A(Ring_k}{A(D_w} \cdot \frac{P}{(kcd_{vv})^\alpha} \\ &\le \frac{1}{k^{\alpha-1}} \cdot \frac{P}{d_{vv}^\alpha} \frac{2^5 3}{c^\alpha}. \end{aligned} \tag{10}$$

where the last inequality holds since $k \ge 1 \Rightarrow 2k+1 \le 3k$ and $c \ge 2 \Rightarrow c-1 \ge c/2$. Summing up the interferences over all rings yields.

$$\begin{aligned} I_{S_v^+}(l_v) &< \sum_{k=1}^{\infty} I_{Ringk}(l_v) \\ &\le \sum_{k=1}^{\infty} \frac{1}{k^{\alpha-1}} \cdot \frac{P}{d_{vv}^\alpha} \frac{2^5 3}{c^\alpha} \\ &< \frac{\alpha-1}{\alpha-2} \cdot \frac{P}{d_{vv}^\alpha} \frac{2^5 3}{c^\alpha}, \end{aligned} \tag{11}$$

where the last inequality holds since $\alpha > 2$. This results in affectedness

$$\begin{aligned} a_{S_v^+}(l_v) = \frac{\beta I_{S_v^+}(l_v)}{P_v(v)} < \frac{\alpha-1}{\alpha-2} \cdot \frac{2^5 3\beta}{c^\alpha} \le 1/3, where \\ c = max(2, (253^2 \beta \frac{\alpha-1}{\alpha-2})^{\frac{1}{\alpha}}). \end{aligned} \tag{12}$$

We have shown that $\forall l_v \in S, a_S(l_v) \le 2/3 + 1/3 = 1$, which means that $SINR(l_v) \ge \beta$ for every scheduled link. This concludes the proof of the lemmas.

3) Algorithm 2

For algorithm 2, we apply one-slot scheduling algorithm to derive a minimum-length schedule.The minimum-length scheduling algorithm (for a description in pseudo-code see Algorithm 2) consists in iteratively computing a one-slot schedule using Algorithm 1. Each one-slot solution is scheduled in a separate slot, and the remaining links are repeatedly used as input to Algorithm 1. The procedure continues until all links in L have been scheduled.

TABLE II
ALGORITHM 2

Algorithm 2 Multi-Slot Scheduling Algorithm
1:input: Set of links $L = l_1, ..., l_n$;
2:output: One-slot schedule S;
3:t:=0;
4:repeat
5: $S_t := OneSlotSchedule(L)$;
6: $L := L \setminus S_t$;
7: $t := t + 1$;
8:until $L = \emptyset$
9:return S;

C. Our Work on Simulation

In this section, we present our simulation result on algorithm 2 by repeat algorithm 1. From the algorithm we can see that, in (12) a threshold 2/3 is made, there is no evidence shows that 2/3 is the best threshold for the algorithm 2. So, we make a further study based on the algorithm 1. We vary the threshold from 0 to 1 by 0.01 each step, and do a simulation according to algorithm 2. Then we get the number of time-slots need for scheduling all the transmission requests. From

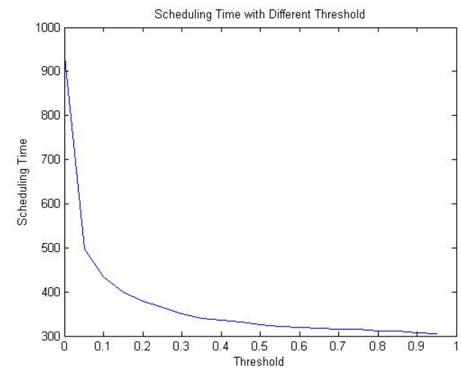

Fig. 3. time-slots – threshold

the simulation results shown in the figure, we can see that for a fixed set of transmission requests, number of time-slots need for transmission decrease with the increase of the threshold. While the speed it decrease is becoming slower as the threshold goes near to 1. Also, from the definition before, the threshold should not be too close to 1, for too near to 1, will make little links remains in step 6 for algorithm 1. So, combined with the two characters, 2/3 is proper, but we can not guarantee it is the best.

VI. Power Control

In Matthew Andrews and Michael Dinitz 's work [22], the problem of maximizing the number of supported connections in arbitrary wireless network where a transmission is supported if and only if the signal-to-interference-plus-noise ratio at the receiver is greater than some threshold. Their aim is to choose transmission powers for each connection so as to maximize the number of connections for which this threshold is met.

Maximizing the transmission capacity in wireless networks as been studied in many contexts. Typically this work can be partitioned along two axes. On one axis, two typical models are used to model channel conditions. The simplest case is the unit-disk graph (UDG) model in hich transmissions interfere if and only if they are within distance. A more complex model is the SINR model in which each transmission is given a power and we assume a distance-dependent path loss. A transmission is deemed to be successful if the signal-to-interference-plus-noise-ratio (SINR) is more than some specified threshold. On the other axis is the structure of the networks that are being considered. One option is to look at random networks under a certain distribution of node placements and transmitter-receiver pairings. In this case the typical goal is to calculate the expected capacity of the system and examine how it changes as the density of the network increases. Another option is to simply look at a worst-case topology. In this case it makes no sense to consider some notion of average capacity since that could depend greatly on what the topology looks like, and Matthew Andrews and Michael Dinitz are interested in the complexity of calculating the optimum capacity and in determining how close they can come to optimality via efficient algorithms.

Here are some details about their model, and their results.

A. Basic model

They consider a set of n connections in the plane. Each connection i has a transmitter t_i and a receiver r_i. They let $d(u, v)$ be the Euclidean distance between two points u and v.They use d_i to denote $d(t_i, r_i)$ and refer to it as the distance of connection i. Suppose that a node u is transmitting with power p. They assume that for some parameters d_0 and α the received signal at another point v is given by $p \cdot min\{(\frac{d_0}{d(u,v)})^\alpha, 1\}$. The $min\{(\frac{d_0}{d(u,v)})^\alpha, 1\}$ is the path loss between u and v and denote it by $g(u, v)$.

They make the assumption that for any connection i the distance $d(t_i, r_i)$ is either 0 or else lies between d_{min} and d_{max} for some parameterd_{min} and $d_{max} \geq d_0$. The running times and the performance guarantees of many of their algorithms will depend on the ratio $\frac{d_{max}}{d_{min}}$

Let p_i be the power used by connection i (which can be zero). They assume that there is a maximum power p_{max} with which any node can transmit. The signal received at a receiver r_i is given by $p_i g(t_i, r_i)$ and the interference heard from the other connection is $\sum_{j\neq i} p_j g(t_j, r_i)$. The background noise level W and so the signal-to-interference-plus-noise-ratio (SINR) is $\frac{p_i g(t_i,r_i)}{W+\sum_{j\neq i} p_j g(t_j,r_i)}$. They assume that each connection is for a single application type such as Voice-over-IP for which there is a fixed signal-to-noise requirement that we denote by τ. Their aim is to maximize the number of satisfied connections. And they refer to this problem as MAX-CONNECTIONS and denote the maximum achievable value by OPT.

Results from the basic model

(1). There first result is a hardness result.MAX-CONNECTIONS is NP-hard and obtaining a polynomial-time exact algorithm is not achievable.

(2). Given that the problem is NP-hard, they turn their attention to approximation algorithms for MAX-CONNECTIONS. Their first algorithm runs in polynomial time and gives an $O(log d_{max})$ approximation. For the case of zero background noise they describe a second algorithm that gives an $O(1)$ approximation in time $n^{O(d_{max}^2)}$.

(3). The approximation algorithms they presented in their article part 3 are centralized. Although this might be appropriate in a situation where they are given a network configuration and they wish to analyze the capacity, centralized algorithms are unlikely to be useful if they wish to optimize capacity as a network evolves. Distributed algorithms are much more likely to be useful. They consider the extreme case of completely decentralized algorithms that do not exchange any information but instead selfishly maximize their payoffs in a game that they design in which a strategy is a transmit power.They first show that their game does not always have a pure Nash equilibrium. On the other hand, They show that in any mixed Nash equilibrium (of which there is always at least one) the expected number of connections that are supported is always within a $O(d_{max}^{2\alpha})$ factor of OPT. Thus if a pure Nash does exist it is close to optimal.

B. Extended model

They also give some ways to extend their model together with some results that they can obtain when these new features are introduced.

(1). The first extension is to assume that each connection i has a weight w_i and the goal is to maximize the weighted total of supported connections. In this model we can slightly modify the proof from the basic model to obtain a similar$O(\log d_{max})$ approximation algorithm.

(2). Another extension is to assume that there are multiple carriers in the system that do not interfere. Each connection must be assigned to a separate carrier. These carriers might be different channels in an 802.11 system or they might be different frequency bands in an OFDM system such as 3GPPsLong-Term Evolution (LTE) standard. In this case they have three decisions to make,namely which connections should be supported, which powers should they be assigned, and which channels should they be assigned. They remark that the third problem can be thought of as providing a frequency reuse pattern for the connections. In this model all of the results from the basic model continue to hold other than losing another constant factor independent of the number of carriers.

VII. Conclusion

During the whole semester, our group have gone through three steps to accomplish the project. In report 1, we presented our preliminary research work on the subject. We focused on the existing results of previous studies carried out by famed professors, such as P. R. Kumar and Xiang-Yang Li and also introduced some generalization based on the papers we read and information we collected from the Internet. In this step, our main purpose was to get a general idea about the subject, since by then we only began to study the course “wireless communication principle and application” and it was even not easy for us to understand some basic concept.

In report 2 and report 3, as we studied on the project more widely and deeply, we gradually found some proper methods. As in report 2, we read the recommended papers in the course website, and then some more research papers published recent years to get a full perspective on the related subsection and the prevailing trend. We generalized several main aspects popularly discussed in this area, such different communication schemes, the trade-off between mobility and delay and specific types of wireless networks.

In the last step of our work, we concentrated our effort on the capacity of arbitrary wireless network, just as shown in this report. Based on the previous understanding, we conducted our work more efficiently and we simulated some of the result mentioned in papers as verification which demonstrated our understanding. Although this report is focused on one specific type of wireless network, the related work gave us much more global view on the whole area.

At last, we would like to express our appreciation to Prof. Wang for providing this opportunity to develop our potential. Generally speaking, our work is still limited and the whole process is time-consuming and torturous, but we find our interests and see our potential.

REFERENCES

[1] P. Gupta and P. Kumar, "The capacity of wireless networks," IEEE Trans. Inform. Theory, vol. 46, no. 2, pp. 388C404, Mar. 2000.
[2] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks, IEEE/ACM Transaction on Networking, zbol. 10, no. 4, pp. 477-486, 2002.
[3] G. Sharma, R. R. Mazumdar and N. B. Shroff, "Delay and capacity trade-offs in mobile ad hoc networks: A global perspective," IEEE/ACM Transactions on Networking, vol. 15, no. 5, pp. 981-992, 2007.
[4] Lap Kong Law; Krishnamurthy, S.V.; Faloutsos, M. "Capacity of Hybrid Cellular-Ad Hoc Data Networks" INFOCOM 2008. Page(s): 1606 - 1614
[5] L. Xie and P. R. Kumar. "A network information theory for wireless communication: Scaling laws and optimal operation". IEEE Transactions on Information Theory, 50(5):748C767, May 2004.
[6] F. Xue, L. Xie, and P. R. Kumar. "The transport capacity of wireless networks over fading channels". IEEE Transactions on Information Theory, 51(3):834C847, March 2005.
[7] L. Xie and P. R. Kumar. "On the path-loss attenuation regime for positive cost and linear scaling of transport capacity in wireless networks". IEEE Transactions on Information Theory, 52(6):2313C2328, June 2006.
[8] A. Ozgur, O. Leveque, and D. Tse. "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks". IEEE Transactions on Information Theory, 53(10):3549-3572, October 2007.
[9] G. Kramer, M. Gastpar, and P. Gupta. "Cooperative strategies and capacity theorems for relay networks". IEEE Transactions on Information Theory, 51(9):3037C3063, September 2005.
[10] S. Aeron and V. Saligrama. "Wireless ad hoc networks: Strategies and scaling laws for the fixed SNR regime". IEEE Transactions on Information Theory, 53(6):2044C2059, June 2007.
[11] U. Kozat and L. Tassiulas. "Throughput capacity of random ad hoc networks with infrastructure support". In Mobicom, 2003.
[12] C. Joo, X. Lin, and N. Shroff. "Understanding the Capacity Region of the Greedy Maximal Scheduling Algorithm in Multi-Hop Wireless
[13] T. Moscibroda, R. Wattenhofer, and Y. Weber. "Protocol Design Beyond Graph-Based Models". In Hotnets, November 2006 Networks". In Infocom, 2008.
[14] G. Brar, D. Blough, and P. Santi. "Computationally Efficient Scheduling with the Physical Interference Model for Throughput Improvement in Wireless Mesh Networks". In Mobicom, 2006.
[15] O. Goussevskaia, Y. Oswald and R. Wattenhofer. "Complexity in geometric SINR". In Proceedings of MobiHoc 07, September 2007.
[16] D. Chafekar, V. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan. "Approximation Algorithms for Computing Capacity of Wireless Networks with SINR Constraints". In Infocom, 2008.
[17] C. Saraydar, N. Mandayam and D. Goodman. "Efficient power control via pricing in wireless data networks". IEEE Transactions on Communications,
[18] A. Stolyar and H. Viswanathan. "Self-organizing dynamic fractional frequency reuse in OFDMA systems". In Infocom, 2008
[19] P. Bahl, M. Hajiaghayi, K. Jain, V. Mirrokni, L. Qiu and A. Saberi. "Cell breathing in wireless LANs: Algorithms and evaluation". IEEE Transactions on Mobile Computing 6(2):164C178, 2007.
[20] Niesen, U.; Gupta, P.; Shah, D.; "On Capacity Scaling in Arbitrary Wireless Networks", IEEE Transactions onInformation Theory in 2009, Page(s): 3959 - 3982
[21] O. Goussevskaia, R. Wattenhofer, M. M. Halldorsson and E. Welzl, "Capacity of arbitrary wireless networks", In IEEE INFOCOM, 2009.
[22] Andrews, M.; Dinitz, M. "Maximizing Capacity in Arbitrary Wireless Networks in the SINR Model: Complexity and Game Theory", INFOCOM 2009, IEEE, Page(s): 1332 - 1340

APPENDIX

Group Members

Xinzhi Zou. Group leader. Efforts on time arrangement, reading papers, topic analysis and writing project reports.

Shaofeng Zou. Efforts on reading papers, simulation and writing reports.

Qian Cai. Efforts on basic understanding of the project and mainly on four lab reports.

Chenchen Fan. Efforts on basic understanding of the project and Latex usage.

Simulation Code

```
for i=1:1000
a1=fix(rand(1,2)*1000);
a2=fix(rand(1,2)*1000);
a3=a1+fix(rand(1,2)*14.14*2-14.14);
a4=a2+fix(rand(1,2)*14.14*2-14.14);
link(i,:,:)=[a1,a2;a3,a4;];
end
for vvv=0.001:0.09:0.999
t(((vvv-0.001)/0.05)+1)=wireless(link,vvv);
end

function [ret]=delete_elem(in1,in2)
for i=1:size(in1,1)
    if(all(all(in2==in1(i,:,:))))
        if(i==1)
            ret=in1(2:size(in1,1),:,:);
            break
        elseif(i==size(in1,1))
            ret=in1(1:(size(in1,1)-1),:,:);
            break
        else
            ret=vertcat(in1(1:i-1,:,:),in1(i:size(in1,1),:,:));
        end
    else
        ret=in1;
    end
end

function [t]=wireless(link,vvv)
t=0;%time for all transmissions

p=1;%assume p=1
alpha=3;beta=2;
c=(96/vvv*beta*2)^(1/3);%constant c
% 1000*1000
% for i=1:1000
%     a1=fix(rand(1,2)*1000);
%     a2=fix(rand(1,2)*1000);
%     a3=a1+fix(rand(1,2)*14.14*2-14.14);
%     a4=a2+fix(rand(1,2)*14.14*2-14.14);
%     link(i,:,:)=[a1,a2;a3,a4;];
% end
for i=1:1000
    dis(i)=cal_distance(link(i,1,:),link(i,2,:));
end
[temp1,temp2]=sort(dis);
for i=1:1000
    link_l(i,:,:)=link(temp2(i),:,:);
    dis_l(i)=dis(temp2(i));
end;
link_ll=link_l;%two L,ll stands for the L in the paper and L means the unarranged connection.


while(size(link_l,1)~=0),
S_link=0;
i=1;
S_I=0;
while(size(link_ll,1)~=0),
   if(i==1)
       S_link=link_ll(1,:,:);
       V_link=link_ll(1,:,:);
       i=i+1;
       link_l=link_l(2:size(link_l,1),:,:);
       link_ll=link_ll(2:size(link_ll,1),:,:);
   else
       S_link=vertcat(S_link,link_ll(1,:,:));
```

```
        V_link=link_ll(1,:,:);
        link_l=delete_elem(link_l,V_link);
        link_ll=link_ll(2:size(link_ll,1),:,:);
        i=i+1;
    end
%     for m=1:(size(S_link,1)-1)
%         S_I=S_I+p/(cal_distance(S_link(m,1,:),V_link(1,2,:))^alpha);
%     end
    for m=1:size(link_ll,1)
        if(cal_distance(link_ll(m,1,:),V_link(1,2,:)) < c*cal_distance(V_link(1,1,:),V_link(1,2,:)))
            if(m==1)
                link_ll=link_ll(2:size(link_ll,1),:,:);
            elseif(m==size(link_ll,1))
                link_ll=link_ll(1:(size(link_ll,1)-1),:,:);
            else
                link_ll=vertcat(link_ll(1:m-1,:,:),link_ll(m+1:size(link_ll,1),:,:));
            end

        end
        if(m>size(link_ll,1))
            break;
        end
        for p=1:size(S_link,1)
            S_I=S_I+p/(cal_distance(S_link(p,1,:),link_ll(1,2,:))^alpha);
        end

        if( (S_I)*beta/( p/(cal_distance(link_ll(1,1,:),link_ll(1,2,:)))^alpha) > vvv )
            if(m==1)
                link_ll=link_ll(2:size(link_ll,1),:,:);
            elseif(m==size(link_ll,1))
                link_ll=link_ll(1:(size(link_ll,1)-1),:,:);
            else
                link_ll=vertcat(link_ll(1:m-1,:,:),link_ll(m+1:size(link_ll,1),:,:));
            end

        end
        if m>=size(link_ll,1)
            break;
        end

    end


end
link_ll=link_l;
size(link_ll,1);
t=t+1;
end
t
```