On High Spatial Reuse Broadcast Scheduling in STDMA Wireless Ad Hoc Networks

Ashutosh Deepak Gore and Abhay Karandikar
Information Networks Laboratory
Department of Electrical Engineering
Indian Institute of Technology - Bombay
Mumbai 400076 India.
{adgore,karandi}@ee.iitb.ac.in

***Abstract*— Graph-based algorithms for point-to-multipoint broadcast scheduling in Spatial reuse Time Division Multiple Access (STDMA) wireless ad hoc networks often result in a significant number of transmissions having low Signal to Interference and Noise density Ratio (SINR) at intended receivers, leading to low throughput. To overcome this problem, we propose a new algorithm for STDMA broadcast scheduling based on a graph model of the network as well as SINR computations. The performance of our algorithm is evaluated in terms of spatial reuse and computational complexity. Simulation results demonstrate that the proposed algorithm performs significantly better than existing graph-based algorithms.**

I. Introduction

In a wireless ad hoc network, a prevalent scheme for channel spatial reuse is Spatial Time Division Multiple Access (STDMA), in which time is divided into fixed-length slots that are organized cyclically and multiple entities can communicate in the same slot. An STDMA schedule describes the transmission rights for each time slot in such a way that communicating entities assigned to the same slot do not collide. STDMA scheduling algorithms can be categorized into link scheduling and broadcast/node scheduling algorithms [1]. In link scheduling, the transmission right in every slot is assigned to certain source-destination pairs. In broadcast scheduling, the transmission right in every slot is assigned to certain nodes, i.e., there is no apriori binding of transmitter and receiver and the packet transmitted must be received by every neighbor. In this paper, we only consider centralized broadcast scheduling for STDMA networks.

A. Related Work

The concept of STDMA for multihop wireless ad hoc networks was formalized in [2]. A broadcast schedule is typically determined from a graph model of the network [1]. The problem of determining an optimal minimum-length STDMA schedule for a general multihop ad hoc network is NP-complete for both link and broadcast scheduling [1]. In fact, this is closely related to the problem of determining the minimum number of colours to colour all the vertices (or edges) of a graph under certain adjacency constraints. In [1], the authors show that a wireless ad hoc network can be scheduled such that the schedule is bounded by a length proportional to the graph thickness[1] times the optimum number of colours.

However, the above work does not take into account Signal to Interference and Noise density Ratio (SINR) computations when determining an STDMA broadcast schedule. In this paper, we propose a suboptimal algorithm based on the graph model as well as SINR computations. We introduce spatial reuse as a performance metric and demonstrate that the proposed algorithm has low computational complexity and high spatial reuse compared to existing algorithms in the literature.

The rest of the paper is organized as follows. In Section II, we describe our system model, discuss the limitations of graph-based scheduling algorithms and formulate the problem. Section III describes the proposed broadcast scheduling algorithm. The performance of our algorithm is evaluated in Section IV and its computational complexity is derived in Section V. We conclude in Section VI.

II. System Model

Consider an STDMA wireless ad hoc network with N static nodes (wireless routers) in a two-dimensional plane. During a time slot, a node can either transmit, receive or remain idle. We assume homogeneous and backlogged nodes. Let:

$$\begin{aligned} \mathbf{r}_j = (x_j, y_j) &= \text{Cartesian coordinates of the } j^{th} \text{ node} \\ P &= \text{transmission power of every node} \\ N_0 &= \text{thermal noise density} \\ D(j,k) &= \text{Euclidean distance between nodes } j \text{ and } k \end{aligned}$$

We do not consider fading and shadowing effects. The received signal power at a distance D from the transmitter is given by $\frac{P}{D^\alpha}$, where α is the path loss factor.

A broadcast schedule effectively assigns sets of nodes to time slots. Specifically, a broadcast schedule for the STDMA

[1]The thickness of a graph is the minimum number of planar graphs into which the given graph can be partitioned.

network is denoted by $\Omega(C, \mathcal{B}_1, \cdots, \mathcal{B}_C)$, where

$$\begin{aligned} C &= \text{number of slots in the broadcast schedule} \\ \mathcal{B}_i &= \text{set of broadcast transmissions in the } i^{th} \text{ slot} \\ &:= \{t_{i,1} \to \{r_{i,1,1}, r_{i,1,2}, \ldots, r_{i,1,\eta(t_{i,1})}\}, \cdots, \\ &\quad\ t_{i,M_i} \to \{r_{i,M_i,1}, r_{i,M_i,2}, \ldots, r_{i,M_i,\eta(t_{i,M_i})}\}\} \end{aligned}$$

where $t_{i,j} \to \{r_{i,j,1}, \ldots, r_{i,j,\eta(t_{i,j})}\}$ denotes a point-to-multipoint transmission of the same packet from node $t_{i,j}$ to all its neighbors[2] $\{r_{i,j,1}, \ldots, r_{i,j,\eta(t_{i,j})}\}$ in the i^{th} slot and $\eta(t_{i,j})$ denotes the number of neighbors of node $t_{i,j}$. Note that M_i denotes the number of concurrent transmissions in the i^{th} slot and $t_{i,j}, r_{i,j,k} \in \{1, \ldots, N\}$. The SINR at receiver $r_{i,j,k}$ is given by

$$\text{SINR}_{r_{i,j,k}} = \frac{\frac{P}{D^\alpha(t_{i,j}, r_{i,j,k})}}{N_0 + \sum_{\substack{l=1 \\ l \neq j}}^{M_i} \frac{P}{D^\alpha(t_{i,l}, r_{i,j,k})}} \tag{1}$$

We define the signal to noise ratio (SNR) at receiver $r_{i,j,k}$ by

$$\text{SNR}_{r_{i,j,k}} := \frac{P}{N_0 D^\alpha(t_{i,j}, r_{i,j,k})} \tag{2}$$

A. Physical and Protocol Interference Models

According to the *physical interference model* [3], the unicast transmission $t_{i,j} \to r_{i,j,k}$ is successful if and only if (iff) the SINR at receiver $r_{i,j,k}$ is greater than or equal to a certain threshold γ_c, termed as the communication threshold.

$$\frac{\frac{P}{D^\alpha(t_{i,j}, r_{i,j,k})}}{N_0 + \sum_{\substack{l=1 \\ l \neq j}}^{M_i} \frac{P}{D^\alpha(t_{i,l}, r_{i,j,k})}} \geqslant \gamma_c \tag{3}$$

According to the *protocol interference model* [3], $t_{i,j} \to r_{i,j,k}$ is successful if:

1) the SNR at receiver $r_{i,j,k}$ is no less than the communication threshold γ_c. From (2), this translates to

$$D(t_{i,j}, r_{i,j,k}) \leqslant \left(\frac{P}{N_0 \gamma_c}\right)^{\frac{1}{\alpha}} =: R_c \tag{4}$$

where R_c is termed as communication range.

2) the signal from any unintended transmitter $t_{i,l}$ is received at $r_{i,j,k}$ with SNR less than a certain threshold γ_i, termed as the interference threshold. Equivalently

$$D(t_{i,l}, r_{i,j,k}) \geqslant \left(\frac{P}{N_0 \gamma_i}\right)^{\frac{1}{\alpha}} =: R_i \ \forall \ l \neq j \tag{5}$$

where R_i is termed as interference range. Note that $0 < \gamma_i < \gamma_c$, thus $R_i > R_c$.

The physical model of our system is denoted by $\Phi(N, (\mathbf{r}_1, \ldots, \mathbf{r}_N), P, \gamma_c, \gamma_i, \alpha, N_0)$.

A schedule $\Omega(\cdot)$ is *feasible* if it satisfies the following:

1) Operational constraint: A node cannot transmit and receive in the same time slot. Also, a node cannot receive from multiple transmitters in the same time slot.

2) Communication range constraints:
   a) Every receiver is within the communication range of its intended transmitter.

$$D(t_{i,j}, r_{i,j,k}) \leqslant R_c \tag{6}$$

   b) Every receiver is outside the communication range of its non-intended transmitters.

$$D(t_{i,l}, r_{i,j,k}) > R_c \ \forall \ l \neq j \tag{7}$$

If node b is within node a's communication range, then b is defined as a *neighbor* of a, since b can decode a's packet correctly (subject to (3)). Note that if node b is outside node a's communication range, then it can never decode a's packet correctly (from (3)). A schedule $\Omega(\cdot)$ is *exhaustive* if every two nodes c, d who are neighbors of each other are included in the schedule twice, once with c being the transmitter and d being a receiver, and vice versa.

B. Graph-Based Scheduling

Broadcast schedules are typically designed by modeling the STDMA network $\Phi(\cdot)$ by a directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ is the set of edges. Let $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$, where vertex v_j represents the j^{th} node in $\Phi(\cdot)$. In general, $\mathcal{E} = \mathcal{E}_c \uplus \mathcal{E}_i$, where $\mathcal{E}_c$ and $\mathcal{E}_i$ denote the set of communication and interference edges respectively. If node k is node j's neighbor, then there is a communication edge from v_j to v_k, denoted by $v_j \xrightarrow{c} v_k$. If node k is outside node j's communication range but within its interference range, then there is an interference edge from v_j to v_k, denoted by $v_j \xrightarrow{i} v_k$. Thus, the mapping from $\Phi(\cdot)$ to $\mathcal{G}(\cdot)$ can be described as follows:

$$\begin{aligned} D(j,k) \leqslant R_c \quad &\Rightarrow \quad v_j \xrightarrow{c} v_k \in \mathcal{E}_c \text{ and } v_k \xrightarrow{c} v_j \in \mathcal{E}_c \\ R_c < D(j,k) \leqslant R_i \quad &\Rightarrow \quad v_j \xrightarrow{i} v_k \in \mathcal{E}_i \text{ and } v_k \xrightarrow{i} v_j \in \mathcal{E}_i \end{aligned}$$

The subgraph $\mathcal{G}_c(\mathcal{V}, \mathcal{E}_c)$ consisting of communication edges only is termed as the *communication graph*.

An STDMA broadcast schedule is equivalent to assigning a unique colour to every vertex in the graph, such that nodes with the same colour transmit simultaneously in a particular time slot, subject to: Any two vertices v_i, v_j can be coloured the same iff:

i) edge $v_i \xrightarrow{c} v_j \notin \mathcal{E}_c$ and edge $v_j \xrightarrow{c} v_i \notin \mathcal{E}_c$, i.e., there is no *primary vertex conflict*, and
ii) there is no vertex v_k such that $v_i \xrightarrow{c} v_k \in \mathcal{E}_c$ and $v_j \xrightarrow{c} v_k \in \mathcal{E}_c$, i.e., there is no *secondary vertex conflict*.

These criteria are based on the operational constraint.

Graph-Based scheduling algorithms utilize various graph colouring methodologies to obtain a non-conflicting schedule, i.e., a schedule devoid of primary and secondary vertex conflicts. To maximize the throughput of an STDMA network, graph-based scheduling algorithms seek to minimize the total number of colours used to colour all the vertices of $\mathcal{G}(\cdot)$.

[2] The set of neighbors of a given node depends on the geographical locations of the nodes and will be made precise in Section II-A.

C. Limitations of Graph-Based Algorithms

Observe that Criteria i) and ii) are not sufficient to guarantee that the resulting schedule $\Omega(\cdot)$ is conflict-free. Due to hard-thresholding based on communication and interference radii, graph-based scheduling algorithms can lead to high cumulative interference at a receiver [4] [5]. This is because the SINR at receiver $r_{i,j,k}$ decreases with an increase in M_i, while R_c and R_i have been defined for a single transmission only. For example, consider Fig. 1 with six labeled nodes whose

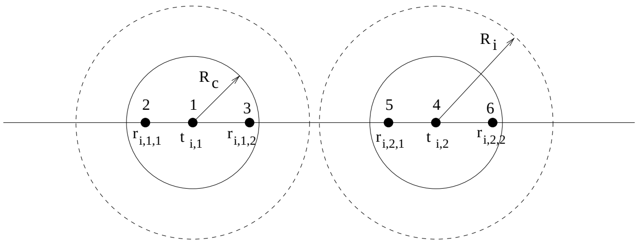

Fig. 1. Graph-Based algorithms can lead to high cumulative interference.

coordinates are $1 \equiv (0,0)$, $2 \equiv (-80,0)$, $3 \equiv (90,0)$, $4 \equiv (280,0)$, $5 \equiv (200,0)$ and $6 \equiv (370,0)$. The system parameters are $P = 10$ mW, $\alpha = 4$, $N_0 = -90$ dBm, $\gamma_c = 20$ dB and $\gamma_i = 10$ dB, which yields $R_c = 100$ m and $R_i = 177.8$ m. A graph-based scheduling algorithm will typically schedule the transmissions $1 \rightarrow \{2,3\}$, and $4 \rightarrow \{5,6\}$ in the same time slot, say the i^{th} time slot, since the resulting graph colouring is devoid of primary and secondary vertex conflicts. However, our computations show that the SINRs at receivers $r_{i,1,1}$, $r_{i,1,2}$, $r_{i,2,1}$ and $r_{i,2,2}$ are 21.85 dB, 12.45 dB, 15.27 dB and 19.97 dB respectively. From the physical interference model, the transmission $t_{i,1} \rightarrow r_{i,1,1}$ is successful, while the transmissions $t_{i,1} \rightarrow r_{i,1,2}$, $t_{i,2} \rightarrow r_{i,2,1}$ and $t_{i,2} \rightarrow r_{i,2,2}$ are unsuccessful. This leads to low throughput.

Hence, graph-based scheduling algorithms do not maximize the throughput of an STDMA network.

D. Problem Formulation

We propose a new suboptimal algorithm for STDMA broadcast scheduling based on the physical interference model.

To evaluate the performance of our algorithm and compare it with existing suboptimal STDMA broadcast scheduling algorithms, we define the following metric: spatial reuse. Consider the STDMA broadcast schedule $\Omega(\cdot)$ for the network $\Phi(\cdot)$. Under the physical interference model, the point-to-point transmission $t_{i,j} \rightarrow r_{i,j,k}$ is successful iff (3) is satisfied. The *spatial reuse* of the schedule $\Omega(\cdot)$ is defined as the average number of successful point-to-multipoint transmissions per time slot in the STDMA schedule. Thus

$$\text{Spatial Reuse} = \frac{\sum_{i=1}^{C} \sum_{j=1}^{M_i} \frac{\sum_{k=1}^{\eta(t_{i,j})} I(\text{SINR}_{r_{i,j,k}} \geqslant \gamma_c)}{\eta(t_{i,j})}}{C} \quad (8)$$

where $I(A)$ denote the indicator function for event A, i.e., $I(A) = 1$ if event A occurs; $I(A) = 0$ if event A does not occur. Note that a high value of spatial reuse[3] directly translates to high long-term network throughput.

We seek a low complexity STDMA broadcast scheduling algorithm with spatial reuse reasonably greater than unity. We only consider STDMA schedules which are feasible and exhaustive.

III. SINR-Based Broadcast Scheduling Algorithm

Our proposed SINR-based broadcast scheduling algorithm is called MaxAverageSINRSchedule, which considers the communication graph $\mathcal{G}_c(\mathcal{V}, \mathcal{E}_c)$ and is described in Algorithm 1. In Phase 1 (Line 3), we label all the vertices randomly[4]. Specifically, if $\mathcal{G}_c(\cdot)$ has v vertices, we perform a random permutation of the sequence $(1, 2, \ldots, v)$ and assign these labels to vertices with indices $1, 2, \ldots, v$ respectively. In Phase 2 (Lines 4-7), the vertices are examined in increasing order by label[5] and the MaxAverageSINRColour function is used to assign a colour to the vertex under consideration.

The MaxAverageSINRColour function is explained in Algorithm 2. It begins by discarding all colours that conflict with u, the vertex under consideration. Among the set of non-conflicting colours $\mathcal{C}_{nc}$, it chooses that colour for u which results in the maximum value of average SINR at the neighbors of u. Intuitively, this average SINR is also a measure of the average distance of every neighbor of u from all co-coloured transmitters. The higher the average SINR, the higher is this average distance. We choose the colour which results in the maximum average SINR at the neighbors of u, so that the additional interference at the neighbors of all co-coloured transmitters is kept low.

Algorithm 1 MaxAverageSINRSchedule

```
1: input: Physical network Φ(·), communication graph G_c(·)
2: output: A colouring C : V → {1, 2, ...}
3: label the vertices of G_c randomly
4: for j ← 1 to n do
5:     let u be such that L(u) = j
6:     C(u) ← MaxAverageSINRColour(u)
7: end for
```

IV. Performance Results

A. Simulation Model

In our simulation experiments, the location of every node is generated randomly in a circular region of radius R. If (X_j, Y_j) are the Cartesian coordinates of the j^{th} node, then $X_j \sim U[-R, R]$ and $Y_j \sim U[-R, R]$ subject to $X_j^2 + Y_j^2 \leqslant R^2$. Equivalently, if (R_j, Θ_j) are the polar coordinates of the j^{th} node, then $R_j^2 \sim U[0, R^2]$ and $\Theta_j \sim U[0, 2\pi]$. Using (4) and (5), we compute R_c and R_i, and then map the STDMA

[3] Note that spatial reuse in our system model is analogous to spectral efficiency in digital communication systems.

[4] Randomized algorithms are known to outperform deterministic algorithms, esp. when the characteristics of the input are not known apriori [6].

[5] In essence, the vertices are scanned in a random order, since labeling is random.

Algorithm 2 integer MaxAverageSINRColour(u)

input: Physical network $\Phi(\cdot)$, communication graph $\mathcal{G}_c(\cdot)$
output: A non-conflicting colour
$\mathcal{C} \leftarrow$ set of existing colours
$\mathcal{C}_p \leftarrow \{C(x) : x$ is coloured and is a neighbor of $u\}$
$\mathcal{C}_s \leftarrow \{C(x) : x$ is coloured and is two hops away from $u\}$
$\mathcal{C}_{nc} = \mathcal{C} \setminus \{\mathcal{C}_p \cup \mathcal{C}_s\}$
if $\mathcal{C}_{nc} \neq \phi$ **then**
  $r \leftarrow$ colour in $\mathcal{C}_{nc}$ which results in maximum average SINR at neighbors of u
  if maximum average SINR $\geqslant \gamma_c$ **then**
    return r
  end if
end if
return $|\mathcal{C}| + 1$

network $\Phi(\cdot)$ to the two-tier graph $\mathcal{G}(\mathcal{V}, \mathcal{E}_c \uplus \mathcal{E}_i)$. Once the broadcast schedule is computed by every algorithm, the spatial reuse is computed using (8). We use two sets of prototypical values of system parameters in wireless networks [7] and describe them in Section IV-B. For a given set of system parameters, we calculate the spatial reuse by averaging this quantity over 1000 randomly generated networks. Keeping all other parameters fixed, we observe the effect of increasing the number of nodes on the spatial reuse.

In our experiments, we compare the performance of the following algorithms:

1) BroadcastSchedule [1] (BS)
2) MaxAverageSINRSchedule (MASS)

B. Performance Comparison

In our first experiment (Experiment 1), we assume that $R = 500$ m, $P = 10$ mW, $\alpha = 4$, $N_0 = -90$ dBm, $\gamma_c = 20$ dB and $\gamma_i = 10$ dB. Thus, $R_c = 100$ m and $R_i = 177.8$ m. We vary the number of nodes from 30 to 110 in steps of 5. Figure 2 plots the spatial reuse vs. number of nodes for both the algorithms.

In our second experiment (Experiment 2), we assume that $R = 700$ m, $P = 15$ mW, $\alpha = 4$, $N_0 = -85$ dBm, $\gamma_c = 15$ dB and $\gamma_i = 7$ dB. Thus, $R_c = 110.7$ m and $R_i = 175.4$ m. We vary the number of nodes from 70 to 150 in steps of 5. Figure 3 plots the spatial reuse vs. number of nodes for both the algorithms.

From Figures 2 and 3, we observe that spatial reuse increases with the number of nodes for both the algorithms. The MASS algorithm consistently yields higher spatial reuse compared to BS. The spatial reuse of MASS is 9-20% higher than BS in Expt. 1 and 3-5% higher in Expt. 2. This improvement in performance translates to substantially higher long-term network throughput.

V. Analytical Result

In this section, we derive an upper bound on the running time (computational) complexity of our algorithm. Let v denote the number of vertices of the communication graph $\mathcal{G}_c(\mathcal{V}, \mathcal{E}_c)$.

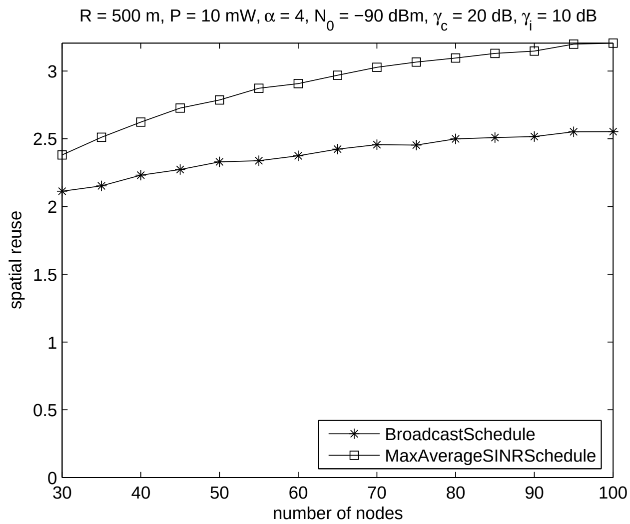

Fig. 2. Spatial reuse vs. number of nodes for Experiment 1.

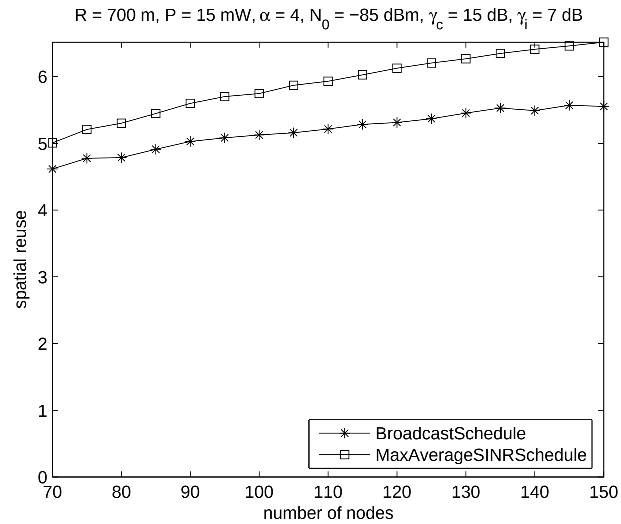

Fig. 3. Spatial reuse vs. number of nodes for Experiment 2.

Theorem 1: The running time of MaxAverageSINRSchedule is $O(v^2)$.

Proof: Assuming that an element can be chosen randomly and uniformly from a finite set in unit time (Chapter 1, [6]), the running time of Phase 1 can be shown to be $O(v)$. In Phase 2, the vertex under consideration is assigned a colour using MaxAverageSINRColour. The worst-case size of the set of colours to be examined $|\mathcal{C}_{nc} \cup \mathcal{C}_p \cup \mathcal{C}_s|$ is $O(v)$. With a careful implementation, MaxAverageSINRColour runs in time proportional to $|\mathcal{C}_{nc}|$, i.e., $O(v)$. Thus, the running time of Phase 2 is $O(v^2)$. Finally, the overall running time of MaxAverageSINRSchedule is $O(v^2)$. ■

VI. Discussion

In this paper, we have developed a broadcast scheduling algorithm for STDMA multihop wireless ad hoc networks under the physical interference model, namely MaxAverageSINRSchedule. The performance of our algorithm is superior to existing graph-based algorithms. A practical experimental modeling shows that, on an average, our algorithm achieves 15% higher spatial reuse than the BroadcastSchedule algorithm [1]. Since schedules are constructed offline only once and then used by the network for a long period of time, this improvement in performance directly translates to higher long-term network throughput. Also, the computational complexity of MaxAverageSINRSchedule is comparable to the computational complexity of BroadcastSchedule. Therefore, MaxAverageSINRSchedule is a good candidate for efficient SINR-based STDMA broadcast scheduling algorithms.

It would be interesting to apply techniques like simulated annealing, genetic algorithms and neural networks to determine SINR-compliant STDMA broadcast schedules.

References

[1] S. Ramanathan and E. L. Lloyd, "Scheduling Algorithms for Multihop Radio Networks," *IEEE/ACM Trans. Networking*, vol. 1, no. 2, pp. 166–177, April 1993.

[2] R. Nelson and L. Kleinrock, "Spatial TDMA: A Collision-Free Multihop Channel Access Protocol," *IEEE Trans. Commun.*, vol. 33, no. 9, pp. 934–944, September 1985.

[3] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. Inform. Theory*, vol. 46, pp. 388–404, March 2000.

[4] J. Grönkvist and A. Hansson, "Comparison Between Graph-Based and Interference-Based STDMA Scheduling," in *ACM MOBIHOC*, October 2001.

[5] A. Behzad and I. Rubin, "On the Performance of Graph-based Scheduling Algorithms for Packet Radio Networks," in *IEEE GLOBECOM 2003*, vol. 6, December 2003, pp. 3432 – 3436.

[6] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge University Press, 1995.

[7] T.-S. Kim, H. Lim, and J. C. Hou, "Improving Spatial Reuse through Tuning Transmit Power, Carrier Sense Threshold, and Data Rate in Multihop Wireless Networks," in *ACM MobiCom 2006*, Los Angeles, CA, September 2006, pp. 366–377.