Introduction

Wireless interference affects network capacity

Throughput = 2 Mbps

Throughput = 1 Mbps
Assumptions

• Fluid model of data transmission

• Data transmissions can be finely scheduled by an omniscient central entity
  - The derived network capacity is under optimal scheduling and optimal routing
  - Applications
    • Assess the efficiency of the existing network protocols
    • Help network provision (e.g., what-if analysis)
Interference Models

• Protocol model
  - Transmission is successful if $d(i,j) \leq R(i)$ and any node $k$ with $d(k,j) \leq R'(k)$ is not transmitting
  - Binary interference model

• Physical model
  - Transmission is successful if $\text{SNR}(i,j) \geq \text{threshold}$
  - Non-binary interference model
Overview of Our Framework

1. Model the problem as a standard network flow problem
   - Described as a linear program
Step 1: Network Flow Model

• Create a connectivity graph
  - Each vertex represents a wireless node
  - Draw a directed edge from vertex A to vertex B if B is within range of A

• Write a linear program that solves the basic MAXFLOW problem on this connectivity graph

• Several generalizations possible
  - Discussed later in the talk.
Example: Network Flow Model

Connectivity Graph

A (Sender) --- B --- C (Receiver)

Link capacity = 1

Linear Program:
Maximize Flow out of A
Subject to:
1. Flow on any link can not exceed 1
2. At node B, Flow in == Flow out.

Answer: 1 (Link 1, Link 2)
Overview of Our Framework

1. Model the problem as a standard network flow problem
   • Described as a linear program

2. Represent interference among wireless links using a conflict graph
Step 2: Model Interference using Conflict Graph

• A conflict graph that shows which wireless links interfere with each other

• Represent each link in the connectivity graph by a vertex in the conflict graph

• Draw an edge between two vertices if the wireless links interfere with each other

• Several generalizations possible
  - Discussed later in the talk.
Example: Conflict Graph

Connectivity Graph

Conflict Graph
Overview of Our Framework

1. Model the problem as a standard network flow problem
   • Described as a linear program

2. Represent interference among wireless links using a conflict graph

3. Derive constraints on utilization of wireless links using cliques in the conflict graph
   • Augment the linear program to obtain upper bound on optimal throughput
Step 3: Clique Constraints

- At most one of the vertices in a clique can be active at any given instant
  - Total utilization of links belonging to a clique is $\leq 100\%$

- MAXFLOW LP can be augmented with these clique constraints to get a better upper bound

- Speed-up convergence: consider maximal cliques in the conflict graph
  - A maximal clique is a clique to which we cannot add any more vertices
Example: Clique Constraints

Linear Program:

Maximize Flow out of A

Subject to:
1. Flow on any link can not exceed 1 * link utilization
2. At node B, Flow in == Flow out.
3. Sum of utilizations of links 1, 2, 3 and 4 (a clique) can not exceed 100%

Answer = 0.5 (Link1, Link 2)
Properties of Clique Constraints

• Finding all cliques can take exponential time
  - Moreover, finding all cliques does not guarantee optimal solution (due to odd holes and odd anti-holes)

• The upper bound is monotonically non-increasing as we find and add new cliques
  - As we add each clique, the link utilizations are constrained further

• More computing time can provide better solution
Overview of Our Framework

1. Model the problem as a standard network flow problem
   - Described as a linear program
2. Represent interference among wireless links using a conflict graph
3. Derive constraints on utilization of wireless links using cliques in the conflict graph
   - Augment the linear program to obtain upper bound on optimal throughput
4. Derive constraints on utilization of wireless links using independent sets in the conflict graph
   - Augment the linear program to obtain lower bound on optimal throughput
Step 4: Independent Set Constraints

• All links belonging to an independent set can be active at the same time
• No two independent sets are active at the same time
• MAXFLOW LP can be augmented with constraints derived from independent sets to get a lower bound
• Speed up convergence: consider maximal independent sets in the conflict graph
  – An independent set to which we cannot add any nodes
Linear Program:

Maximize Flow out of A

Subject to:

1. Flow on any link can not exceed 1 * link utilization
2. At node B, Flow in == Flow out.
3. Sum of utilizations of all independent sets can not exceed 100%
4. Utilization of a link can not exceed the sum of utilization of independent sets it belongs to.

Answer = 0.5 (Link1, Link2)
Properties of Independent Set Constraints

• Lower bound is always feasible
  - LP also outputs a transmission schedule

• Finding all independent sets can take exponential time
  - If we do find all independent sets, the resulting lower bound is guaranteed to be optimal

• Lower bound is monotonically non-decreasing as we find and add more independent sets
  - More computing time provides better answers

• If upper and lower bounds converge, optimality is guaranteed
Putting It All Together

1. Model the problem as a standard network flow problem
   • Described as a linear program
2. Represent interference among wireless links using a conflict graph
3. Derive constraints on utilization of wireless links using cliques in the conflict graph
   • Augment the linear program to obtain upper bound on optimal throughput
4. Derive constraints on utilization of wireless links using independent sets in the conflict graph
   • Augment the linear program to obtain lower bound on optimal throughput

Iterate over steps 3 and 4 to find progressively tighten bounds on optimal throughput
Putting It All Together (Cont.)

Houses talk to immediate neighbors, all links are capacity 1, 802.11-like MAC, Multipath routing
What-if Analysis

Houses talk to immediate neighbors, all links are capacity 1, 802.11-like MAC, Multipath routing

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Aggregate Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.5</td>
</tr>
<tr>
<td>Double range</td>
<td>0.5</td>
</tr>
<tr>
<td>Two ITAPs</td>
<td>1</td>
</tr>
<tr>
<td>Two Radios</td>
<td>1</td>
</tr>
</tbody>
</table>
Physical Interference

- Represent wireless links as vertices in conflict graphs
- Directed conflict graph
- Weight on edge X→Y represents the fraction of the maximum permissible noise at the receiver of link Y when link X is active
- Schedulable sets instead of independent sets
- Non-schedulable sets instead of cliques
Other Generalizations

• **Multiple senders and/or receivers**
  - Write LP to solve multi-commodity flow problem

• **Non-greedy sender**
  - Create a virtual sender
  - Include a “virtual link” of limited capacity from the virtual sender to the real sender in the connectivity graph
  - This link does not conflict with any other links
  - LP maximizes flow out of virtual sender

• **Single path routing**
  - Integer linear programming

• **Multiple radios on orthogonal channels**
  - Represent with multiple, non-interfering links between nodes

• **Directional antennas**
  - Include appropriate links in the connectivity graph
  - Conflict graph can accommodate any interference pattern
Other Generalizations (Cont.)

• **Multirate radios**
  
  • Create multiple virtual links corresponding to a physical link, one for each data rate
  
  • Only one of the virtual links corresponding to a physical link can be active at a time
  
  • The edge weights (under physical interference model) reflect the specific noise tolerance for each rate

• **Other objectives**
  
  • Any linear function (e.g., fairness or revenue) can be used
Limitations

• Linear programs can take a long time to solve
  - Especially when single path routing is used

• There is no guarantee that optimal solution will be found in less than exponential time

• Upper bound might not converge to optimal even if we find all cliques
  - Graphs with odd-holes and anti-holes

• It gives bounds on the optimal capacity
  - Not the real capacity under a given protocol
Summary

- A flexible framework for deriving capacity of specific topologies with specific traffic patterns
  - Computes upper and lower bounds on optimal throughput
  - Accommodate various models of network connectivity and interference, routing constraints, traffic demands