This assignment describes the course project. The goal of the course project is to provide depth in a particular area of wireless networking in a hands-on fashion.

Overview
Course projects are very open ended as one would expect in a graduate course. We list a number of possible projects at the end of this hand out but you can also propose your own project.

One of the challenges you face is to identify the platform you will use for your project. Here are some possibilities:

- The wireless network emulator has 15 nodes that can be used remotely. More information can be found on the emulator web page: www.cs.cmu.edu/~emulator
- For some projects you may be able to use your own laptops.
- A number of wireless testbeds are available on the web. These include: Emulab (www.emulab.net), Purdue (https://engineering.purdue.edu/MESH), Orbit (http://www.orbit-lab.org/), etc.

Here are some things to consider when defining your project:

- The project needs to result in a concrete outcome that will offer an interesting result or insight. Ultimately, the primary goal is that you learn something.
- You must have the necessary background to complete the project. For example, do not propose a project at the physical layer if your only background in this area is the four lectures in this course. Similarly, do not propose a project that involves kernel or driver hacking if you have limited programming experience.
- The project needs to be feasible. For example, the amount of effort must be reasonable given the time available and you must have access to the equipment needed to execute the project.

This project must be done in teams of 3 students. Please contact the instructors before the project proposal deadline if you want to propose a project with a different team size.

The specific deliverables for the project include a proposal, two check points and a final poster.

Project proposal
The project proposal should include the following information:

- Names of team members
- Project topic and title
- Description of activities that will be performed as part of the project.
- Expected results
• Equipment and resources needed
• Project plan: rough list of milestones, responsibilities of team members, qualifications

Project proposals will typically be between one and two pages in length. Project proposals must be emailed to both instructors.

**Project checkpoints**
There will be two project checkpoints. Checkpoints are short status reports that cover the following:
• Team information
• Progress so far: what was completed
• Any results so far
• Problems encountered
• Changes in the project plan

The first checkpoint will typically be about one to two pages while the second checkpoint may be about double the size. Please submit checkpoints through the blackboard Digital Drop box. Checkpoints will be graded.

**Poster presentation**
Each team must prepare a poster that summarizes the result of their project. There will be a poster session in the last week of the semester.

**Milestones**

<table>
<thead>
<tr>
<th>Milestone</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Friday, Feb 27</td>
<td>Submit team information</td>
</tr>
<tr>
<td>Friday, Mar 6</td>
<td>Project proposals due</td>
</tr>
<tr>
<td>Monday March 30</td>
<td>Check point 1</td>
</tr>
<tr>
<td>Wednesday April 15</td>
<td>Check point 2</td>
</tr>
<tr>
<td>Last week of classes</td>
<td>Poster session</td>
</tr>
</tbody>
</table>
Example Topics
We describe a number of examples of types of projects that we believe are feasible. However, you are welcome to define your project. It is very important that your project matches the background and interests of your team.

Which one can you trust?
There is some controversy in the wireless networking community about what are appropriate techniques for evaluating wireless technologies. Possibilities include simulation, various types of emulation, and testbeds. A number of projects can be defined that use different techniques to run similar experiments and compare the results. One option is to compare the following:

- Plain ns-2 simulation, probably the most widely used platform in academia
- ns-2 with extensions for improved physical layer accuracy, e.g. the package from the University of Karlsruhe (see: http://portal.acm.org/citation.cfm?id=1298155 and http://www.nabble.com/Announcement-of-NS-2-802.11Ext-td14664620.html)
- The CMU wireless network emulator or another testbed.

The domain for evaluation can be any topic covered in class. One option is to implement some of the techniques yourself. The other is to use open source implementations (e.g. for transmit rate control, ad hoc routing, TCP over wireless, etc.)

Directional antennas
Omni-directional antennas are easy to use and they are by design supported by the 802.11 standard. For example, carrier sense is used to reduce collision rates and RTS/CTS can be used to avoid collisions in hidden terminal situations. Directional antennas increase spatial reuse and, therefore, capacity. They can also improve the distance and quality of links. Directional antennas can be installed manually, or it is possible to install software steerable antennas, for example, phased-array antennas. Directional antennas unfortunately have some disadvantages, specifically hidden and exposed terminal scenarios are more common and more complex, especially when using software-steerable antennas. The goal of the project is to design a protocol that makes use of directional antennas to improve network capacity. You would need to focus on a particular scenario, e.g. physical context, mobility properties, timescale for adaptation, etc. Directional antennas can be emulated on the wireless network emulator testbed.

Handoff
Increasingly, people are using WiFi while being mobile, e.g. web browsing or VoIP from a PDA. As a result, smooth handoffs between APs become much more critical. Unfortunately, the handoff procedure in existing WiFi drives are often designed for nomadic users, i.e. users who use their wireless device while being stationary in different locations. A number of projects can be defined to improve the performance of handoff, e.g. in the MadWifi driver for Atheros or
possibly in simulation.

**Migrate for Community Wireless**

One of the problems in most mesh access networks is that because they use a NAT, flows are "pinned" to the gateway that they first go out of. If the network topology changes, or the gateway fails, the connection will die or remain bound to a sub-optimal path. This represents a case of unfortunate—and unnecessary—fate sharing between two communicating endpoints and a box in the middle. Design a scheme to circumvent this problem. Possibilities include using TCP migrate on an end-to-end basis (with its respective drawbacks for deployment) or having the gateway nodes perform Mobile IP-like services and pass the traffic to each other over the wired Internet.

**World models for the wireless emulator**

The control software for the wireless network emulator is described in:


One option for controlling experiments is through a “world model” in which the location and movement of wireless devices is emulated. The characteristics of the wireless channels in the experiment are then derived automatically from the properties of the devices in the world model. The current world model is unfortunately very primitive. A possible project would be to extend the world model (or add world models) to emulate more interesting physical environments. (At least) two approaches are possible:

- You could create a world with artificial objects (e.g. walls) and then have the characteristics of the channels (e.g. attenuation, multi path) be determined (or influenced by) the presence of these objects. Note that you would need to extend the control interface of the world model to account for the additional functionality.
- You could characterize the wireless channels in a specific environment (e.g. based on measurements, or results published in the literature) and then use that information to adjust the wireless channel characteristics of your world model.

This project requires a pretty good understanding of the physical layer.

**Software-defined Radios**

Software-defined radios implement most of the radio functionality in software, so they offer a lot of flexibility. As a result, they are often used as the based for cognitive radios, i.e. radios that can adapt their characteristics (modulation, coding, MAC, etc.) to the specific context (e.g. interference, node density, traffic load, etc.).

possible project is to add support for “signal flipping”, i.e. taking part of an incoming packet and inserting it in an outgoing packet. This may be useful, for example, to generate fast ACKs. Interested teams should contact George Nychis (gnychis@cmu.edu).

**Hybrid ARQ/FEC**

ARQ and FEC are two possible techniques to recover from lost or corrupted packets. A CMU research project on high quality video streaming over wireless networks has design an opportunistic ARQ mechanism. A possible project is to combine it with FEC at the application so the efficiency of error recovery can be tuned dynamically based on the runtime conditions. Interested teams should contact Amy Lu (meihsual@andrew.cmu.edu).