Hydra
An Open Source Wireless Testbed

Hyrum K. Wright, Robert D. Grant

Wireless Communication and Networking Group
Department of Electrical and Computer Engineering
The University of Texas at Austin

July 23, 2008

Wired network != Wireless network

Traditional wired-network assumptions break down in the wireless world.

- The medium is half-duplex broadcast, not full-duplex point-to-point
- Channels vary in time and frequency domains
- Mobility results in rapidly changing link states

What we study

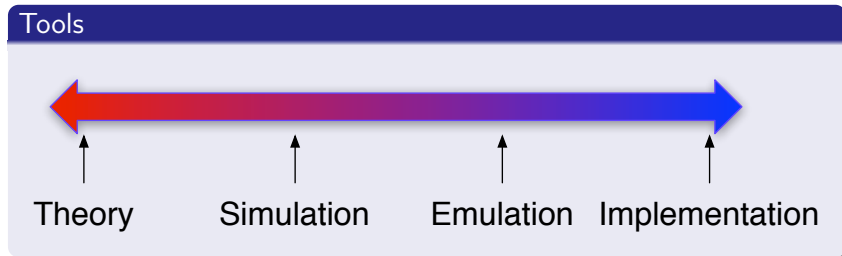Wireless communication at the PHY, MAC, and routing layers.
Particularly,

- Cross-layer algorithms (e.g., rate adaptation)
- MIMO (multi-antenna systems)
- Multi-hop networks
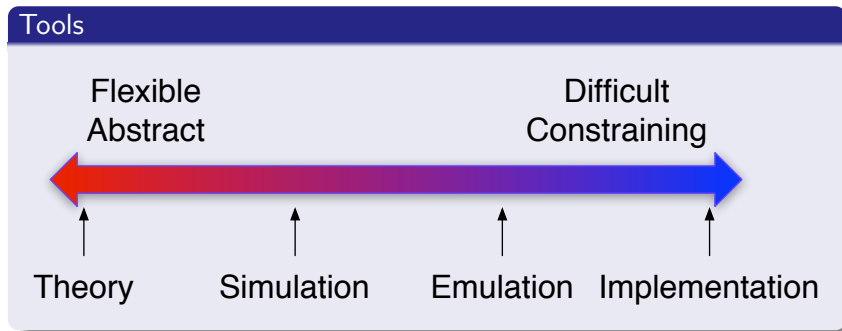
Why we study this stuff

Practical research for the next generation of wireless networking.

- Ad hoc networking can extend network range, help networks scale, and connect places previously unconnected by traditional networks
- MIMO and cross-layer algorithms can help improve network performance

How we study wireless networks

Tools

Theory — Simulation — Emulation — Implementation

How we study wireless networks

Tools

Flexible
Abstract

Difficult
Constraining

Theory   Simulation   Emulation   Implementation

Current Open Source Tools

- Theory - GNU Octave, R
- Simulation - NS2, OMNeT++
- Emulation - Emulab, NS Emulator
- Implementation - GNU Radio, IT++, Click

Weaknesses of Current Open Source Tools

- NS2, OMNeT++ - Difficult to get details of wireless right in simulation
- Emulab, NS Emulator - Built for wired networks, though people are trying to extend them to wireless
- GNU Radio, IT++ - Good tools, but low level. GNU Radio was built for stream processing, not packet processing.

Wireless Networking Research
Hydra
Demonstration

Implementation Options
High-level Overview
Nitty-gritty details

Hydra

Goal

Develop a flexible platform for wireless research and experimentation which uses real channels.

Wireless Networking Research
Hydra
Demonstration

Implementation Options
High-level Overview
Nitty-gritty details

Implementation Options

- Application Specific Integrated Circuit (ASIC)
    - Great performance
    - High cost
    - Difficult to change, revise
- FPGA
    - Easier than ASIC to modify
    - Commodity part = easy to obtain
    - Still requires hardware knowledge to develop
- Software Defined Radio (SDR) Platform
    - Acceptable (but worse) performance
    - Highly customizable
    - Does not require hardware knowledge to program

Hydra

An open-source, flexible testbed for wireless experimentation, almost entirely in software.
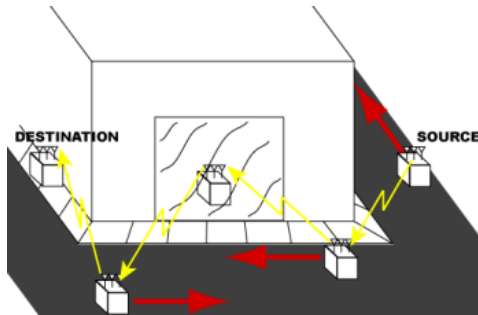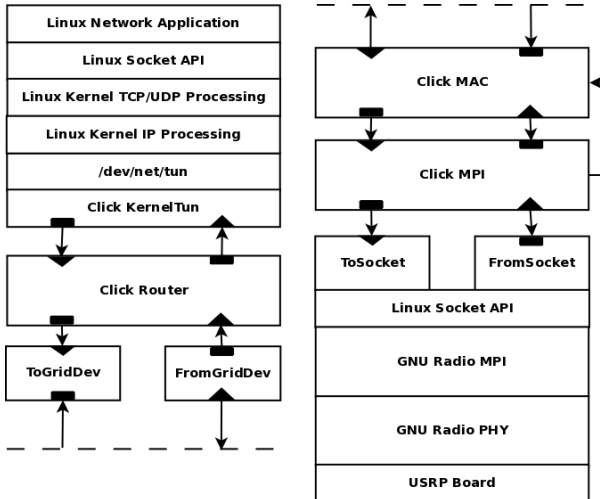
Hydra Node
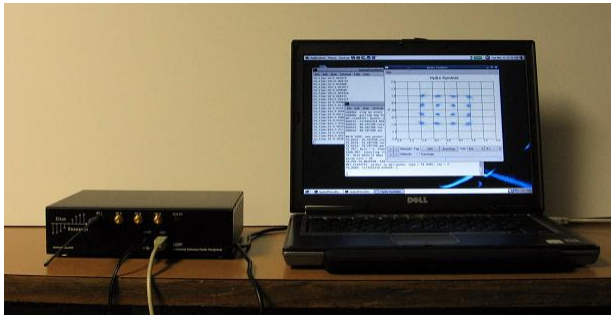
High Level

- Each node is a general purpose x86 machine
- Upper layers are GNU/Linux
- Link layer and MAC layer are implemented in Click Modular Router
- PHY layer is implemented in C++ and wrapped in GNU Radio
- Hardware is one or more Universal Software Radio Peripheral (USRP) boards, attached via USB

Hydra Block Diagram

Wireless Networking Research
Hydra
Demonstration

Implementation Options
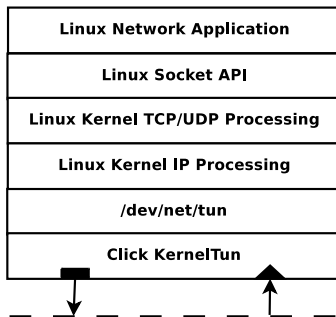**High-level Overview**
Nitty-gritty details

All Open Source Components

- The USRP - all FPGA code and designs are open!
- GNU/Linux operating system
- GNU Radio Software Radio Framework
- Click Modular Router
- IT++ (and its dependencies)
- C++, Python, and SWIG
- GNU Octave (in the works!)

Wireless Networking Research
Hydra
Demonstration

Implementation Options
High-level Overview
Nitty-gritty details

The Upper Layers

Hydra runs standard applications (ping, Firefox, Apache, FTP). Works through the Linux TUN device

- A virtual network device (/dev/net/tun)
- Allows Click to send IP packets from userspace to the Linux kernel's IP processing code
- Allows the kernel to pass IP packets to Click in userspace

| Linux Network Application |
| Linux Socket API |
| Linux Kernel TCP/UDP Processing |
| Linux Kernel IP Processing |
| /dev/net/tun |
| Click KernelTun |

Wireless Networking Research
Hydra
Demonstration

Implementation Options
High-level Overview
Nitty-gritty details

Link Layer

Media Access Control: Distributed algorithms for determining transmission priority. i.e., "Who gets to talk when"

Click Modular Router

- A modular packet router which integrates with Linux.
- Can run in either user or kernel mode.
- Creates an additional network interface which applications can use seamlessly.
- Current MAC implementations: 802.11, plus experimental protocols such as RBAR, OAR and others.
- Using C++, we define "blocks," and then connect those blocks in a flow graph.
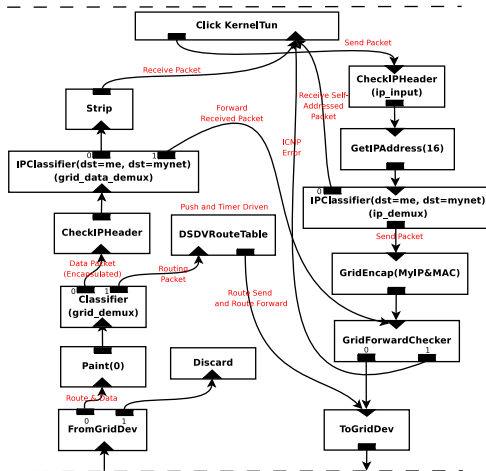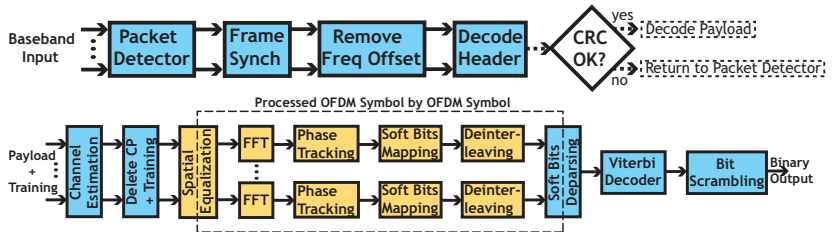
Link Layer

Figure: Click DSDV implementation

Physical Layer

- 802.11a and 802.11n style physical layers implemented
- Implemented in C++ using the IT++ communications library
- Entire physical layer wrapped in a single GNU Radio Block
- Visualizations written using Matplotlib, wxpython

Physical Layer

Side Project: Octave Integration

- Our current physical layer is written in C++
- Most physical layer researchers are used to MATLAB
- Solution: GNU Octave, an open source version of MATLAB

Octave Integration Structure

Client/Server architecture

- TCP server interface to USRP
- TCP client in GNU Octave
- Physical layer in GNU Octave
  - Functional structure. A series of transformations on data to transmit or receive.
  - Runs in Octave and MATLAB
  - Custom unit-testing framework

Research

Mostly in cross-layer MAC/PHY algorithms

- Physical layer rate control
- MAC frame aggregation
- MIMO
  - Beamforming
  - Limited feedback

Wireless Networking Research
Hydra
Demonstration

Implementation Options
High-level Overview
Nitty-gritty details

Future

- More experiments – Hydra is starting to stabilize as an experimental platform
- More nodes – scale up our test network
- Make easier to use. . .
    - More documentation
    - More visualization
    - More experimental tools
    - Easier network administration
- Release code – backend cleanup!
- Expand Octave integration
- Use in the classroom

Demo Time

BAM

Questions?

Resources

- Hydra: `http://hydra.ece.utexas.edu/`
- GNU Radio: `http://www.gnu.org/software/gnuradio/`

Obligatory Sponsor Info

- National Science Foundation, under grant CNS-626797
- Office of Naval Research (ONR), under grant number N00014-05-1-0169
- DARPA IT-MANET program, under grant W911NF-07-1-0028