

Experimental Networking

ECSE-4690 (UG)

ECSE-6966 (G)

I hear and I forget.

I see and I remember.

I do and I understand.

-- Chinese Proverb

Shivkumar (“Shiv”) Kalyanaraman

Rensselaer Polytechnic Institute

shivkuma@ecse.rpi.edu

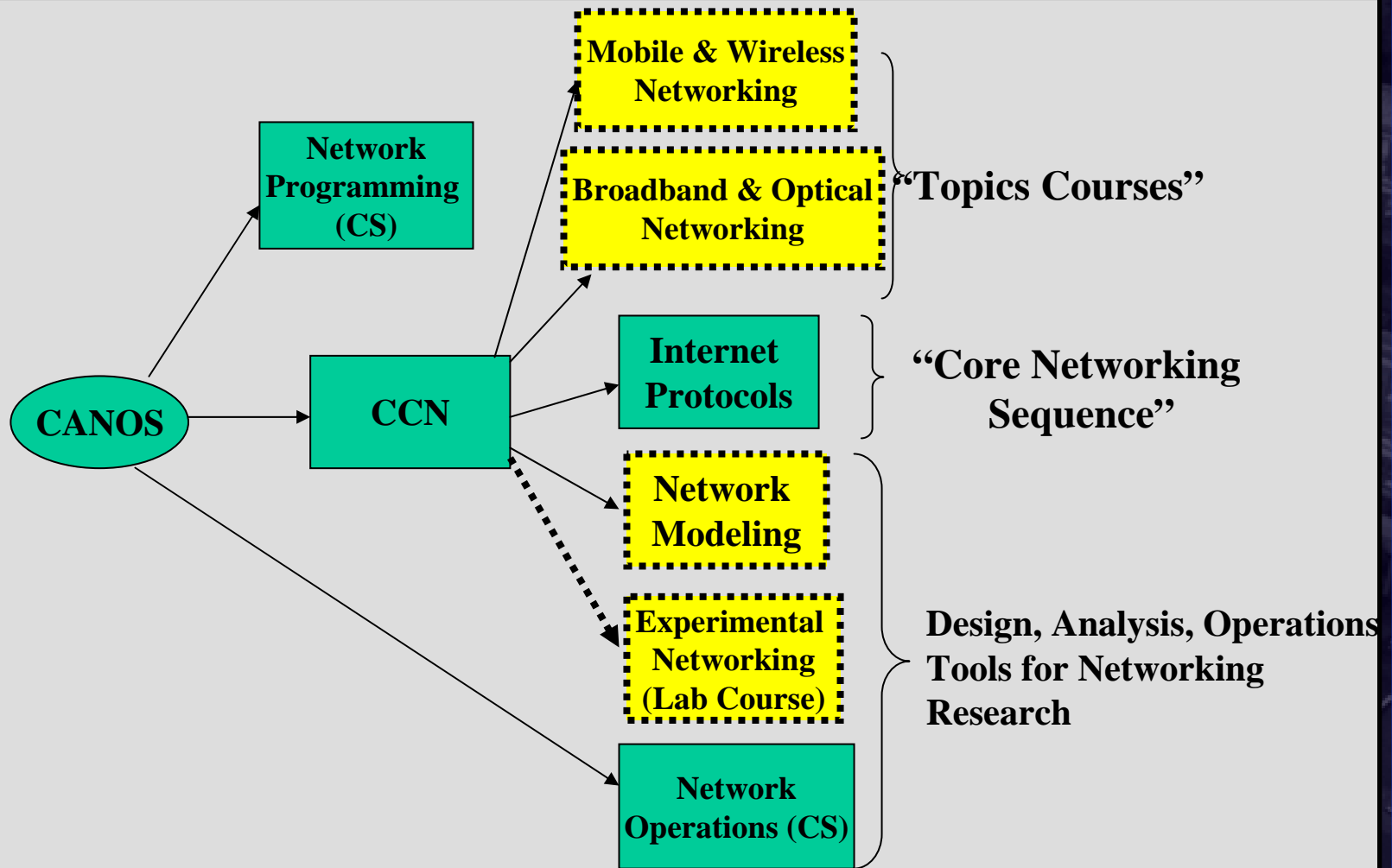
<http://www.ecse.rpi.edu/Homepages/shivkuma/>

GOOGLE: “Shiv RPI”

Who's Who

- ❑ **Instructor:** Shiv Kalyanaraman, kalyas@rpi.edu
 - ❑ Room: JEC 6042, Phone: x8979
- ❑ **TA:**
 - ❑ Neeraj Jaggi, jaggin@rpi.edu
 - ❑ Room: JEC 6213, Phone: x8289
 - ❑ Other research students (esp. Vijay Subramanian, Yufeng Shan) will help from time to time

Networking Courses @ RPI



Prerequisites

- ❑ Required (***no exceptions***):
 - ❑ VERY GOOD C programming knowledge
- ❑ **Co**-requisites:
 - ❑ ESCE-4670 Computer Communication Networks or equivalent
 - ❑ Probability Class (usually required for CCN)
- ❑ Desirable:
 - ❑ Operating Systems
 - ❑ Computer Architecture (ECSE-4730 or equivalent)
 - ❑ Basic ideas of statistics
- ❑ If you **do not have the required prerequisites**, you **must drop the course** and take it later.

Course/Grading Format

- ❑ **Lab time:** 1 hr Lecture + 2.5 hr Lab Work
 - ❑ Lab Report for Each Class (groups of 2) submit via WebCT
 - ❑ Grad students (6000-level) will have more submissions & readings etc.
 - ❑ Grading will be simple: submit reasonable responses to each class assignment, get a full grade.
 - ❑ WebCT bulletin board: Post your questions!
- ❑ **1 term project** in the last month (complete design exercise)
- ❑ **2 exams:** mid-term and final, on concepts, theory, etc.
 - ❑ Term project and exams will be graded by TA/instructor
- ❑ **Grading:**
 - ❑ **Lab Work/Reports** {40 pts}
 - ❑ **Term Project:** {25 pts}
 - ❑ **Mid-Term and Final Exam:** {35 pts}

I do and I understand...

□ **What** to do?

- **Play** around with **real & messy stuff**: wires, routers, real networking code: builds character!
- **Simulate/animate** it: avoid the mess, focus on basic understanding of a subset of properties
- **Poke/Peek around** the network: peek at packets, measure n/w performance, collect/analyze traces/routing tables etc
- Structure a **set** of above activities to maximize information derived with minimum effort

□ **Why?**

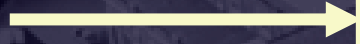
- Semantic behavior of protocols/networks: **how does it work?**
- Performance behavior of protocols/networks: **how good is it?**
- Use such techniques in the design process: **design your own new protocols/networks !**

Course Objectives

- ❑ **Hands-on networking:** “do networking” at the hardware, software, simulation, configuration (a.k.a. messy!) levels
- ❑ **Experimental Method:** How to correctly use a variety of abstract tools (measurement, simulation, animation, experiment) for design and analysis of computer/network systems
- ❑ **Tools:** specific tools/platforms useful for networking research and advanced development

System-Under-Test Model

Parameters



**System-
Under-
Test**



Metrics

Subject system to a set of tests (workloads/conditions)

Course Description Highlights

1. **Simulation and animation tools**: understand complex networking concepts by viewing the system as a **black box**
 - Vary external “knobs” (parameters)
 - Someone else has designed the system!
 - Why? Limited views of protocols allows a self-paced, visual understanding...
2. **Simulation development**: develop the networking protocol code, in a controlled environment, the simulator.
 - Run simulations and vary parameters to **incrementally refine** design.
 - You are designing & building the system!
 - Why? Understand what it takes to **embed your idea** in an existing event-driven system.

Course Description Highlights

- 3. **Systematic Tracing, Graphing, Profiling:**
 - Define *parameters* (input) and *metrics* (output)
 - **Parameter** criteria: all params that have performance impact (or a subset relevant to the performance “view”)
 - **Metric** criteria: must capture the relevant tradeoffs
 - Time series graphs vs point estimates
 - Examples of good, poor graphs;
 - **Workloads**: must stress test the system, capture relevant aspects of reality (in stages)
 - Issues with randomness: confidence intervals etc
 - **Profiling**: accounting for performance: contributions of components. Does it add up? Apply amdahl’s law to decide where to make changes
 - **Tracing**: at different degrees of resolution (low pass, high pass): helps in design debugging

Course Description Highlights

4. **Experiment design:** one simulation does not give you the answer (i.e. characterize system behavior)
- Systematic design of a *set of experiments* to maximize information extracted
 - Fit regression or other functional models to *correlate parameters to observed metrics*.
 - Why? A practical tool for incremental design and performance analysis. Understand the nature of protocols in-depth.
- [Some applied probability, statistics and simulation theory will be covered as necessary.]

5. **Linux-based protocol development:** Develop variants of protocols on a real OS platform (Linux)
- Set up experiments to instrument, measure and visualize system behavior.
 - Introduction to advanced community platforms: **Emulab, Planetlab**

Course Description Highlights

6. Measure, Model and Analyze the Internet:

- Understand tools to observe and measure network/protocol properties
- Develop and analyze measurement archives to understand protocol and network behavior

6. Experimentation with a combination of Linux and Cisco routers:

- Learn how to create experimental scenarios with a combination of customized/prototyped systems and off-the-shelf networking equipment.

Course Description Highlights

7. Development on modular platforms (Click router and Intel IXA):

- Recent developments include modular code development inside the OS kernel (Click), and network processor platforms (Intel IXA)
- Powerful, realistic prototypes can be created rapidly!
- Introduction to advanced community platforms: **Emulab**, **Planetlab**

8. Term project: Take a problem and use a mix of relevant tools to incrementally design, prototype, test and validate solutions

- Students are welcome to define a project of their own; and should get a written project definition approved by the instructor. Approval will require a critical mix of key ingredients to be present.
- Graduate students should choose a challenging research-driven project. Topics from their current research are welcome.
- Measurement projects or projects using Emulab/Planetlab/Intel IXA platforms are also welcome. The instructor will do his best to facilitate them.

Schedule

Every Wednesday 3pm – 7pm in Fall'06, Aug 30 ~ Dec 6

Basic Labs: Tools and Techniques

Week 1 Aug 30

Lab 1, Networking commands and socket programming

Week 2-5 Sept 6,13,20,27

Lab 2, Network simulator NS2 (and NAM)

Lab 3, TCP Tahoe, Reno, and SACK comparisons in simulation

Lab 4, Perf. Analysis methods: Metrics, Tracing, Experiment design

Lab 5, Active queue management (AQM): RED scheme

Week 6 Oct 4

Lab 6, TCP traffic experiment: how to encode/setup/measure real TCP dynamics

[MID-TERM EXAM: Oct 11th]

Week 7-8 Oct 11, 18

Lab 8, BGP routing table analysis, Internet Mapping, Measurement projects

Lab 9, MIT Click modular router, Linux kernel programming, Intel IXA Network Processor Platform, Introduction to the Utah Emulab facility, Planetlab

Term Project

Week 11-14 Oct 25, Nov 1, 8, 15, 29th:

1-month term project.

Project proposals due on Oct 18th, approved by weekend.

Mid-project updates + slides due on Nov 8th
(email ppts prev night to Neeraj).

{22: thxgiving}

Week 14: Nov 29th

Presentations [7 min/group]: Sharing Ideas, Experiences and Frustrations ☺

PPTs due before class (emailed to TA).

Submission of Written Project Reports on WebCT (DEC 3rd)

[FINAL EXAM: DECEMBER 6th]

Special Issues for Fall 2006

- ❑ Travel dates:
 - ❑ Sept 13, 27th
 - ❑ Oct 18.
 - ❑ But the class will go on. Professor Biplab Sikdar, and/or other PhD research students will be present.
 - ❑ I will cover lecture material in previous classes – so that on those days, you can focus on the lab portion.

Open Shop Hours

- ❑ MON-THU Evenings: 6pm-8pm
- ❑ Sun: 4-9 pm
- ❑ Note: TA will not be available then
- ❑ Open Shop Hour Qns: (or if lab is not open)
 - ❑ Amit Mukherjee, mukhea@rpi.edu

Term Project Ideas

- ❑ **1.** Design and Comparison of Active queue management (AQM) approaches: e.g.: ARED, BLUE, AVQ, REM...
 - ❑ Architectural twists: virtual AQM, edge-based policing
- ❑ **2.** TCP+AQM traffic dynamics: Various Flavors of TCP/Binomial/Uncooperative End-system schemes w/ AQM: (RED, ARED, REM, AVQ)
- ❑ **3.** TCP enhancements for high-bandwidth delay product and wireless networks (VCP, LT-TCP): evaluate, implement
- ❑ **4.** Performance and implementation of mesh wireless routing and name-resolution algorithms (eg: GHTs, coordinate space routing, location systems)
- ❑ **5.** Measurement study of 802.11 on campus (understanding performance problems)
- ❑ **6.** Measurement and inference problems using overlays on the Internet (using Planetlab)
- ❑ **7.** Implementation and experimentation of routing/AQM and new hybrid FEC techniques etc on Intel IXA platform

Term Project Ideas

- ❑ **8.** Using SSFNet to test a mix of new OSPF/BGP related concepts and perform larger scale simulation experiments
- ❑ **9.** Multimedia Streaming: understanding effect of various system components, joint source-network coding
- ❑ **10.** Delay tolerant networks: evaluation of various routing mechanisms
- ❑ **11.** Mixed Wireless (802.11) and Internet Experiments/Simulations
- ❑ **12.** Large-scale Internet Measurement Studies: TCP Latency/Bandwidth, Internet Mapping, Feeding online measurement and models into design
- ❑ **13.** Data mining and data streaming techniques applied to high-speed internet measurement, deep-packet-inspection, intrusion detection problems etc.
- ❑ **14.** Wimax (4G wireless) simulation modeling
- ❑ Other ideas welcome! Bring your favorite research topics or talk to me in advance to brainstorm

Shivkumar Kalyanaraman

Caveat!

- ❑ A lot of lab development has been done and labs have been tested
- ❑ Things could go wrong, and you may have to discover things on your own at times.
 - ❑ We will try to provide maximum help
 - ❑ Grading will be sensitive to such issues
- ❑ If you cannot live with some uncertainty; trying out some realistic hacking; and/or do not have the time to spend a larger-than-average effort on the class...
 - ❑ Please drop the class! 😊

Logistics

- ❑ Restart machine to Linux when you come
 - ❑ Username: net
 - ❑ Password: netnet
- ❑ Restart machine to Windows before you leave
- ❑ Openshop hours: Everyday (except Wednesday) after 4 pm
 - ❑ TA will NOT be available then. Please ask for help during regular class hours!
 - ❑
- ❑ Every week's Assignment due (WebCT): Sunday 11:55pm