Principles of Communications - A Course for Final Year UK Computer Science Undergraduates

Jon Crowcroft
Computer Laboratory
University of Cambridge
United Kingdom
jon.crowcroft@cl.cam.ac.uk

ABSTRACT

In this paper, we present the outline of our final year course on the Principles of Communications, a course approximating what some people might call *Network Science*, although including some more traditional materials as well.

Categories and Subject Descriptors

C.2.3 [**Network Architecture and Design**]: Network Operation; I.0 [**Computing Methodologies**]: General

General Terms

Design, Models

Keywords

Protocols

1. INTRODUCTION

Last year, we re-wrote the systems strand of our undergraduate programme in Computer Science. In the process, the two courses that were taught on networks (Digital Communications 1&2) were completely revamped. The first course now loosely follows the Kurose/Ross book structure, which we found very good as a general introduction, and also as foundational material which people take who might go off in another direction (CS theory or NLP or what-have-you). The 2nd course, a 3rd year optional (but relatively popular) one, is now called "Principles of Communications".

This course is an attempt to structure the last 10 years of "network science" and related work into a set of material that underpins later courses for the student that want to make a career in systems in general and networking in particular.

Hence, the course is (in contrast to the prior introductory and generic course for the generalist CS students):

- A bottom up, principles to practice based course.
- Is theory based - there are many theoretical frameworks that apply in networks, some old, new, borrowed, none blue. We present the core set.
- We pick theories and methods for their generality and wide applicability for example to other networks than interconnects for computers, including potentially energy and transportation.
- We include recent complex systems and network science results because of their applicability to communications systems usage, as well as to other communicable diseases such as illness and opinion.

Some of the material may look like it is here just for fad and fashion reasons, however, the criteria above hopefully mitigate against that view.

1.1 Assumptions about prior knowledge

In our particular course, we can make relatively strong assumptions about the knowledge and ability of students in mathematics, because our degree programme shares intake with mathematics and with natural sciences. Hence it is reasonable to assume solid knowledge of statistical methods (e.g. central limit theorem) and of calculus (e.g. Laplace transform) amongst other areas.

1.2 Pre-requisites

The formal pre-requisites for this, aside from those implicit in being accepted for our degree programme, are the courses in Networks.

1.3 Course Structure

Systems Design Arguments& Layering Recapitulation This is just reminding students of the material they should be familiar with from the prior course (readers should revisit the Kurose/Ross book to see what is assumed here).

Information Theory This is a very fast re-cap of Bayes, Nyquist, Entropy & Shannon. There is another course that runs overlapping this which explores the whole area in depth as Information theory is also foundational for other CS areas (Security and Image Processing and Machine Learning just for starters) so it needs a full programme of its own).

Modulation and Coding There are some nice basic ideas in modulation and coding, but recently also some cunning examples of "Computational Thinking" especially with some of the recursive and random tricks used in code design.

Channel Capacity This is an opportunity to illustrate how we can go from a specific instantiation of a modulation and coding technique and with knowledge of a channel's properties, compute the information rate achievable

Graph Theory Most (but perhaps not all) networks are graph like. Graph theory has a lot of quite simple but very useful results. Some of the basic knowledge of the evolution of graphs (percolation over small world networks, growth through preferential attachment, assortative networks etc) is very useful when thinking about algorithms for networking (routing, content distribution, defense against DDoS) and evaluation of said algorithms.

Social Networks Obviously this will smack of being trendy. It is, which is always pedagogically good (this year's course overlapped with the release of the Movie of the Startup of the Subject, which was nice.). However, social networks exist not in a vacuum, but in the context of real world systems (e.g. human social networks are studied by epidemiologists and social scientists, but are now measured using artificial social networks through OSN and Smart Phone instrumentation). Some properties (path length in expander graphs, robustness of assortative networks, etc) follow naturally from the previous lecture.

Routing Properties of routing algorithms are interesting. Looking at Vectoring and Diffusing style algorithms, and their properties (global versus local consistency, cost, evaluation on real world graphs) is fun. It is, of course, eminently easy to test the knowledge of a student as well.

Errors Earlier, we have looked at coding, modulation and channel capacity. However, errors are more general a problem that the physical/link layer would have us believe. Protocol design for error detection and recovery has a very long history. It also doesn't really have a coherent foundational theory yet. Perhaps by placing this material here, a student of this course might successfully devise one in the future.

Queueing The Grandfather of network theories. Equally useful in operating systems, and so gets revisited here - some of this is covered in greater detail in a later undergraduate course delivered by Richard Gibbens, on Computer Systems Modeling, so this is a recapitulation, reminder of basics and link to that material

Flow Control (aka congestion control) One of the most published areas in networking in the last twenty years, and so worth visiting, but not just for historical reasons. One area to link with is in distributed resource management in other domains (economics, for example).

Control Theory James Clerk Maxwell wrote "On Governors" and presented it at the Royal Society in 1867. The material is still relevant for feedback control systems. This brief section covers models, controllers, z-transforms and stability conditions and efficiency considerations.

Scheduling Underpinning scheduling are ideas of fairness and efficiency (again). We rehearse the notion of max-min fairness and of work-conservation, and statistical multiplexing efficiency gains. We look at the Generalized Processor Sharing and compare practical fair queuing algorithms with idealised GPS.

Switching Switch fabrics adorn computing in many places, whether early parallel computing, modern data centers, or the back-plane of a modern router. The basic principles of switched design (and goals, e.g. non blocking) are covered in overview.

Contention Most networking today is accessed via shared media, typically wireless. So resource sharing and particularly, contention resolution techniques are needed. These reveal another side of Computational Thinking, as cunning inventors draw from various other areas to devise schemes that work with distributed cooperation or centralised enforcement, or somewhere in between.

Mesh Radio Capacity The capacity of multihop radio networks is another nice topic. The various results due to Gupta&Kumar, Grossglasuer and Tse, for various spatial organisation of nodes with and without motion, with cooperative relaying or not, are covered. The key idea to get across is that including geometry and movement makes life even more interesting than the already tricky problem of radio propagation.

Decentralised Traffic Optimization This is all about Kelly and Low and others models of the Internet traffic problem, and how TCP (or more recently, multipath TCP and multipath routing) provide a neat solution with a theoretical basis (in Lagrangian Optimisation methods).

Traffic Engineering and Linear Programming The longer term traffic engineering problem is usually solved as an LP formulation, so here we just recap that technique as it is generally quite useful.

It turns out that John Daugman's Information Theory, Markus Kuhn's Digital Signal Processing, and Richard Gibben's Computer Systems Modeling courses run overlapping or just after this course, hence the material in the start, and at the very end of the programme outlined above can be relatively introductory.

1.4 Subsequent courses

Our undergraduate degree programme is incredibly compressed compared even with other UK CS degrees. It is relatively hard to compare with a US degree too, since UK students start with more advanced, but narrow qualifications.

We have recently introduced a 4th year (aka Part III), for which see `http://www.cl.cam.ac.uk/teaching/1011/acs.html`, and that has a number of courses that are also suitable for the US style 'qualifier" type learning, and are intended to bridge from the undergraduate through a type of Masters material, to possible PhD or other full graduate research studies.

Some specific courses running in that programme, targeting communications interests include:

Network Architecture This course assumes a broad knowledge of the material above, and then looks at the Future Internet Design space (both evolutionary and revolutionary).

How to build an internet router This is a very intensive practical course, in which student teams construct software and hardware systems leading to an 8Gbps router based on NetFPGA.

Programming for Mobiles This is a new course about smart devices (Android etc) and what capabilities they have.

Data Centric Networking This is a specialised course about the quite wide range of new ideas for a particular new internet, based around content delivery rather than human-to-human communications.

Flows in Networks This is all about performance analysis in communications networks.

Social and Technical Network Analysis This goes in to great detail about network characteristics (graph models, mobility models, and information flows over those graphs).

2. DISCUSSION AND PLEA

It is too early to say how well the new structure is working, however, feedback on the new material in the course was largely positive, and in particular, asked for "more of these theories and examples of their use", which was highly encouraging. Suggestions for more material, examples and anything else are most welcome.

A book that covers this territory at the right level of detail, with suggestions for coursework, and associated slide-ware would be marvellous.

3. FURTHER READING

As stated above, the assumption in this course is that students have learned all the material in basic computer networking, as perhaps presented in the Kurose/Ross text, for which see `http://www.aw-bc.com/kurose_ross/`.

Underlying much of our teaching approach is algorithmic thinking, in line with the CMU work on Computational Thinking, for which see `http://www.cs.cmu.edu/~CompThink/`. More specifically, on networking, we believe that by the end of an undergraduate programme, or start of a Masters (or qualifying US PhD) year, students should understand and be able to apply all the ideas present in the work by George Varghese (see, for example `http://cseweb.ucsd.edu/users/varghese/TEACH/cs228/introslides.pdf`).

However, a number of new topics (notably in graph theory and in optimisation) need thorough coverage too - these are present in specialist texts, but we would like to see introductory material made available at a similar level to that we describe above.