10 Networking Papers: Recommended Reading

Jon Crowcroft
University of Cambridge
Computer Laboratory
William Gates Building
15 JJ Thomson Avenue
Cambridge CB3 0FD UK
http://www.cl.cam.ac.uk/~jac22/

In the last issue *ACM Computer Communication Review*, Christophe Diot, the Editor-in-Chief kicked off a series of contributions to *CCR* by members of our technical community on networking papers that they would recommend to others. Of course, we all have our lists of favourites (I have 4 book lists on Amazon!), but this is more than just stamp collecting.

Search engines and citation indexes have several problems, including: it is hard to balance recency of an article with popularity – a single dimension index doesn't really tell you whether a paper is seminal or just popular; some material is not available (too old, or not there any more; the average of everyone's opinion may not be as useful as a subjective view by someone you trust (or distrust); one day, these problems may be solved by contextualizing information that is retrieved and presenting the recommendation network that your retrieval was made by (Some search engines like http://beta.previewseek.com/ are starting to do this). Until that day, lists like this are a good substitute, and they are also fun starting points for discussion.

My list is explicitly not my "top ten" papers ever. Rather, it represents a sample made at a snapshot. These are papers that came up in recent discussions in PhD supervision, research project work, and in reviewing papers for conferences. For each paper, I've given some indication of the value I got from the paper. In some cases, I also give the context that I first saw the paper. Here they are, in random order:

- *"Experience with Grapevine: the growth of a distributed system,"* [Schroeder 1984]. This has so many ideas in how to actually do things right (compared say to DNS) and includes some things people have forgotten about 10 times (including later work that used both epidemic models and control theory applied to the update traffic). We used to work on Directory Systems at UCL in the 1980s – we also worked on comparing early DNS implementations (Berkeley BIND and Stanford's DRUID). The baseline for all of these, though, was GrapeVine. Paul Dourish (now at Irvine) visited us in UCL from Xerox's European PARC, round then and told us the stories of epidemic problems that showed up in this paper.
- *"The Design and Implementation of an Operating System to Support Distributed Multimedia Applications,"* [Leslie 2000]. It's a shame so few companies (perhaps only Be) actually read this. The parallels between OS and Nets have been well observed many times in the OS community, but only rarely (unfortunately) in the nets community. Of course, my colleagues now work on the hypervisor, Xen, but recently I also met folks at Microsoft Research in Redmond working on a cool new OS, Singularity. This is a good time to be in the OS space. But don't forget the past!
- *"The Synchronization of Periodic Routing Messages"* [Floyd 1994]. This is a very general concept - typically (for the 2nd author, Van is an ex-physicist) it uses an idea

from physics and an approach to observe, and then to come up with a system design to eliminate a problem. In the *ACM Sigcomm* conference presentation of the paper, Van explained that synchronization occurs to good effect in two commonplace situations - clockmakers used to display all their wares on a wooden board to make sure that all the clocks that were working reasonably well stayed in agreement. Fireflies at dusk in southeast asian river deltas start randomly, but synchronize. As with his 1988 ACM Sigcomm paper about congestion avoidance [Jacobson 1988], where the real reason he worked on it was that "he couldn't read rec.singles as the net was too clogged", this is another paper with a surprising source for its ideas.

- *"Analysis of the Parallel Packet Switch Architecture"* [Iyer 2003]. This is a lovely mix of fundamental algorithms and hardware structures. I used to teach Computer Architecture #101 from the books by Patterson and Hennessy, which lived and breathed on the software/hardware interface – I very much like the way Computer Science and Computer Engineering co-exist on that plane.
- *"Small Forwarding Tables for Fast Routing Lookups,"* [Brodnik1997]. This led to a successful startup. It contains a very good example of how to use a well-characterised workload to evaluate an elegant data structure, right on the limit of the processor/memory system performance. It spawned an industry of follow-up papers. People have criticised some aspects of the work - so what, it was first.
- *"How to Build a Gateway,"* [Strazisar 1979] This is amazingly clear, and prescient. It pre-dates all the post-hoc rationalisation of the IP world, but is quite beautifully discursive on how and why an IP router software system should be the way it was, and would remain, for 10-15 years after this was written. I implemented IP forwarding twice, and referred to this both times, although now of course, we might do things differently.
- *"Implementing remote procedure calls,"* [Birrell 1984]. This was pretty much the last word on RPC semantics, and implementation - most code after this got slower, did less until now we have all the glories of SOAP (XML "RPC"), with no more functionality at any level of portability or performance. Bruce Jay Nelson's thesis [Nelson 1981] is also a beautiful piece of writing. There used to be an IETF anti-ISO T-Shirt, that I believe was designed by Marshall Rose with the headline "Same Day Service in a Nanosecond World". The lessons behind that can be revisited here today.
- *"Models for a self-managed Internet,"* [Kelly 2000]. This really takes the biscuit as it brings together so many strands of modeling (whether control theory, game theory, optimisation), and then uses the results to give a prescription of where the Internet should head in both end-to-end controls, and within routers (small buffers, again!). It also links with much work on mechanism design and incentive alignment very naturally. The best mathematicians make things very clear, and very general, as does this.
- *"Approximate aggregation techniques for sensor databases,"* [Considine 2004]. This is a very good example of why research groups should not stay stovepiped - this paper came out of the database and networking groups at BU talking to each other as a result of mixing deliberately between subject disciples in the faculty. As a component of any good research and teaching department, this shows why we need to do both, and we need to cycle people around somehow (but not so as to overload them with interrupts and churn/context switches:-))
- *"Compressing TCP/IP Headers for Low-Speed Serial Links,"* [Jacobson 1990] This argues from rationale, to very simple, but efficient code with lots of hints that are useful to implementers along the way (and those hints pay off in other ways too) - the postscript

version also has a nice diagram illustrating the fields that change packet by packet, plus the invariants - this is also incidentally gives you information about what can be used in TCP/IP for covert signaling, and what can be used as a signature for a flow - it's the counter part to the work on header prediction, and some of the back-of-an-envelope performance analysis gives clues about how to do Type-of-service forwarding efficiently. The step of doing the combined analysis of TCP and IP layers together rather than separately, hints at other work such as Integrated Layer Processing and Application Layer Framing, discussed by Clark and Tennenhouse in a later paper [Clark 1990].

So – that's *my* list. Colleagues Steven Hand (Cambridge), and Umar Saif (MIT) commented on an initial version, causing me to switch a couple of papers for slightly different ones by the same authors, better to fit my selection. Jim Kurose (University of Massachusetts) encouraged me to submit it (and typed it up!). I also hope that others will contribute their own lists in future issues of *CCR*, so that I and others can discover yet other new gems.

References

[Birrell 1984] A. Birrell, B. Nelson, "Implementing remote procedure calls," *ACM Transactions on Computer Systems*, 2(1). pp. 39 – 59.

[Brodnik 1997] A. Brodnik, S. Carlsson, M. Degermark, S. Pink, "Small Forwarding Tables for Fast Routing Lookups," *Proc. ACM SIGCOMM 1997*, (Cannes, France 1997), pp. 3 – 14.

[Considine 2004] J. Considine, F. Li, G. Kollios, J. Byers "Approximate aggregation techniques for sensor databases," *Proceedings 20th International Conference on Data Engineering* (ICDE '04).

[Clark 1990] D. Clark, D, Tennenhouse, "Architectural considerations for a new generation of protocols," *Proc. ACM Sigcomm 1990*, (Philadelphia, PA), pp. 200 – 208.

[Floyd 1994] S. Floyd, V. Jacobson, "Synchronization of periodic routing messages," *IEEE/ACM Transactions on Networking*, Vol. 2, No. 2, pp. 122 – 136.

[Iyer 2003] S. Iyer, N. McKeown, "Analysis of the Parallel Packet Switch Architecture," *IEEE/ACM Transactions on Networking*, Vol. 11, No. 2 (April 2003), pp. 314-324.

[Jacobson 1988] V. Jacobson, "Congestion Avoidance and Control," *Proc. ACM Sigcomm 1988*, (Stanford CA, Aug. 1988), pp. 314-329.

[Jacobson 1990] V. Jacobson, "Compressing TCP/IP Headers for Low-Speed Serial Links," Request for Comments 1144, February 1990.

[Kelly 2000] F. P. Kelly, "Models for a self-managed Internet," Network Modeling in the 21st Century, Royal Society Discussion Meeting, London, December 1999. Published in *Philosophical Transactions of the Royal Society* A358 (2000) 2335-2348.

[Leslie 2000] I. Leslie, D. McAuley, R. Black, T. Roscoe, P. Barham, D. Evers, R. Fairbairns, E. Hyden, "The Design and Implementation of an Operating System to Support Distributed Multimedia Applications," http://www.cl.cam.ac.uk/Research/SRG/netos/old-projects/nemesis/documentation.html

[Nelson 1981] B.J. Nelson, "Remote procedure call," Tech. Rep. CSL-81-9, Xerox Palo Alto Research Center, Palo Alto, Calif. 1981.

[Schroeder 1984] M Schroeder, A. Birrell, R. Needham, "Experience with Grapevine: the growth of a distributed system," *ACM Transactions on Computer Systems* 2, 1 (February 1984), 3--23.

[Strazisar 1979] V. Strazisar, "How to Build a Gateway," Internet Engineering Note #109, August 1979. http://www.isi.edu/in-notes/ien/ien109.txt