HotNets 2012 Highlights

P. Brighten Godfrey
University of Illinois at Urbana-Champaign
pbg@illinois.edu

ABSTRACT

This article captures some of the discussion and insights from this year's ACM Workshop on Hot Topics in Networks (HotNets-XI).

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General

Keywords

HotNets

1. INTRODUCTION

In his keynote at SIGCOMM 2012, Nick McKeown observed that the networking community is often too skeptical of new ideas and too committed to old assumptions. One reason I enjoy HotNets is that the program committee has a sense of adventure, resulting in an engaging set of papers that often push the boundaries of conventional wisdom. In fact, several attendees this year suggested that instead of SIGCOMM's Outrageous Opinion Session, HotNets should feature an Entirely Reasonable Opinion Session.

However unorthodox the ideas, quite a few of this year's papers presented real working implementations—including secure multiparty computation of interdomain routes, unexpectedly tiny compression of forwarding tables, creation of shared secrets based on physical location, measurements of price discrimination on shopping sites, and more.

In addition to differences in the flavor of the papers, HotNets is a smaller, more intimate forum than most conferences, leading to more conversational discussions. This document is an attempt to capture some of that discussion and insights from this year's papers.

At the outset, I should note that this is not an official or complete summary of the workshop. This article, presented here thanks to a suggestion of Dina Papagiannaki, was originally a live-blog of the workshop and certainly does not do justice to the ideas and results in all of the papers. I hope the brief summaries here will inspire you to read the original articles!

2. MONDAY

2.1 Architecture and Future Directions

- Software-Defined Internet Architecture
  Barath Raghavan, Teemu Koponen, Ali Ghodsi, Martin Casado, Sylvia Ratnasamy, Scott Shenker
- Towards Systematic Roadmaps for Networked Systems
  Bin Liu, Hsunwei Hsiung, Da Cheng, Ramesh Govindan, Sandeep Gupta
- FreeDOM: a new Baseline for the Web
  Raymond Cheng, Will Scott, Arvind Krishnamurthy, Tom Anderson

Teemu Koponen kicked off the workshop, arguing that combining the ideas of edge-core separation (from MPLS), separating control logic from the data plane (from SDN), and general-purpose computation on packets at network edges (from software routers) can lead to a more evolvable software defined Internet architecture. While there is some recent work on making core Internet protocols more modular and evolvable, the focus here is on decoupling the architecture from the physical infrastructure, so that new designs can be implemented entirely in software, vastly reducing the cost of deployment.

Sandeep Gupta discussed hardware trends, including increasing error rates in memory, and how this may affect networks (potentially increasing loss rates). This was probably the scariest talk of the workshop. The goal, however, is to prepare for the future: plan a *roadmap* predicting future capabilities of networked systems as a function of hardware roadmaps which are already used extensively in the semiconductor industry.

Raymond Cheng talked about how upcoming capabilities which will be widely deployed in web browsers will enable P2P applications among browsers, so free services can really be free. Imagine databases in browsers, or every browser acting as an onion router.

2.2 Security and Privacy

- A New Approach to Interdomain Routing Based on Secure Multi-Party Computation
  Debayan Gupta, Aaron Segal, Gil Segev, Aurojit Panda, Michael Schapira, Joan Feigenbaum, Jennifer Rexford, Scott Shenker
- Creating Shared Secrets out of Thin Air
  Iris Safaka, Christina Fragouli, Katerina Argyraki, Suhas Diggavi
- Act for Affordable Data Care
  Saikat Guha, Srikanth Kandula

Scott Shenker examined how to build inter-domain routing with secure multi-party computation (SMPC), which al-

lows a group of independent participants to compute a function of shared data, while providing a cryptographic guarantee that nothing more than the output of the function is revealed. The proposal is to compute routes not as a complex distributed algorithm across all routers (as BGP does today), but instead using an SMPC protocol on a relatively small number of compute clusters. SMPC might seem a little magical, but the paper demonstrates an implementation for a special case of BGP policies. The key benefits: autonomy, privacy, simple convergence behavior, and a policy model not tied to computational model. The last item should be emphasized: changing BGP today requires changing routers globally, but changing a route computation cluster could be much easier. For example, could this policy flexibility allow domains to cooperate to offer new services? Do other classes of policies have different or better oscillation properties?

There are a couple connections here to other work. The topic of coordinating routing across domains to offer new services was explored in more depth by Kotronis et al. in a later session (§3.3). The proposal's convergence behavior appears related to Consensus Routing [2], which also uses a relatively small set of compute nodes to produce a single global routing outcome installed at routers. Jeff Mogul mentioned an interesting point: Adding a layer of cryptographically-secure privacy may make it very hard to figure out what's going on inside the algorithm and debug why it arrived at a particular result.

Katerina Argyraki spoke about how we can change the basic assumption of secure communication: creating a shared secret not based on computational difficulty, but on physical location. The idea is to leverage differences in wireless interference across location. The scheme's security is more robust than one might think, in that it relies only on a lower bound on how much information the eavesdropper Eve misses, rather than which pieces of message Eve missed. An implementation generated 38 secret Kbps between 8 nodes. However in a few corner cases Eve learned a substantial amount about the secret. There is some hope to improve this.

Saikat Guha linked the problem of data breaches to money and proposed data breach insurance—"Obamacare for data". In a survey, 77% of users said they would pay for such insurance, with a median amount of $20. (Saikat thought this may be optimistic.) They're working to develop a browser-based app to monitor user behavior, offer individuals incentives to change to more secure behavior, and see if people actually change. The shift here from attacking the problem via technology, to attacking it with economic incentives, is interesting.

2.3 Software-Defined Networking

- Toward Software-Defined Middlebox Networking
  Aaron Gember, Prathmesh Prabhu, Zainab Ghadiyali, Aditya Akella

- Rethinking End-to-End Congestion Control in Software-Defined Networks
  Monia Ghobadi, Soheil Hassas Yeganeh, Yashar Ganjali

- Live Migration of an Entire Network (and its Hosts)
  Eric Keller, Soudeh Ghorbani, Matthew Caesar, Jennifer Rexford

Aaron Gember spoke about designing an architecture for software defined middleboxes, taking the idea of SDN to more complex processing. Existing protocols for decoupling the control and data planes, such as OpenFlow, enable simple matching but fall short of common middlebox features like stateful processing of flows and deep packet inspection. The paper attacks the challenge of distributed management of middlebox state.

Monia Ghobadi has rethought end-to-end congestion control in software-defined networks. The work observes that TCP has numerous parameters that operators might want to tune—initial congestion window size, TCP variant, AIMD knobs, and more—and these can have a dramatic effect on performance. But the effects they have depend on current network conditions. The idea of the system they're building, OpenTCP, is to provide automatic and dynamic network-wide tuning of these parameters to achieve performance goals of the network. This is done in an SDN framework with a central controller that gathers information about the network and makes an educated decision about how end-hosts should react. The paper presents results from an early deployment of OpenTCP in a roughly 4,000-node HPC cluster, showing some very nice improvements in flow completion time. Questions discussed: Did you see cases when switching parameters dynamically offered an improvement? And in general, how often do you need to switch to approach the best performance? Some of that remains to be characterized in experiments.

Eric Keller, now at the University of Colorado, spoke about network migration: Moving your virtual enterprise network between cloud providers, or moving within a provider to be able to save power on underutilized servers, for example. Doing this while keeping the live network running reliably is not trivial. But if you can do it, it's pretty powerful: rather than being a disruptive event to be avoided whenever possible, migrating groups of services and whole networks can be a common operation. The solution here involves cloning switches, using tunnels from old to new, and then migrating VMs. But then, you need to update switch state in a consistent way to ensure reliable packet delivery. Some questions were discussed: How do you deal with SLAs? How do you deal with networks that span multiple controllers?

2.4 Performance

- More is Less: Reducing Latency via Redundancy
  Ashish Vulimiri, Oliver Michel, P. Brighten Godfrey, Scott Shenker

- CARE: Content Aware Redundancy Elimination for Disaster Communications on Damaged Networks
  Udi Weinsberg, Qingxi Li, Nina Taft, Athula Balachandran, Gianluca Iannaccone, Vyas Sekar, Srinivasan Seshan

Ashish Vulimiri presented our paper on making the Internet faster. Consistent low latency is extremely difficult to achieve, because it requires eliminating unpredictable exceptional conditions—congestion, an object not being cached, a delay due to virtualization, and so on. Redundancy is an effective technique to deal with this uncertainty: execute latency-sensitive operations twice, and use the first answer that finishes. Redundancy has been used to improve latency in various forms previously. This paper's argument

is that redundancy should be used much more pervasively than it is today. A very simple cost-benefit analysis indicates that conservatively, redundancy is useful when it saves at least 10 milliseconds per kilobyte of added traffic—and we can often do much better than this: sending a redundant DNS query, for example, empirically saves roughly 100 milliseconds per KB. Intuitively, redundancy converts the hard problem of achieving consistent low latency into an easier and often cheaper one of dealing with somewhat higher utilization. Several questions revolved around making the cost-benefit analysis more realistic, for example taking into account energy on mobile devices and server-side costs other than bandwidth.

Udi Weinsberg went in the other direction: redundancy elimination. This is an interesting scenario where a kind of content-centric networking may be a big help: in a disaster which cuts off high-throughput communication, a DTN can provide a way for emergency response personnel to learn what response is most effective, through delivery of photos taken by people in the disaster area. But in this scenario, as the authors have verified using real-world data sets, people tend to take many redundant photos. Since the throughput of the network is limited, smart content-aware redundancy elimination can more quickly get the most informative photos into the hands of emergency personnel.

3. TUESDAY

3.1 Mobile and Wireless

- Power-Aware Rateless Codes in Mobile Wireless Communication
  Calum Harrison, Kyle Jamieson

- One Strategy Does Not Serve All: Tailoring Wireless Transmission Strategies to User Profiles
  Shailendra Singh, Karthikeyan Sundaresan, Amir Khojastepour, Sampath Rangarajan, Srikanth Krishnamurthy

- When David helps Goliath: The Case for 3G OnLoading
  Narseo Vallina-Rodriguez, Vijay Erramilli, Yan Grunenberger, Laszlo Gyarmati, Nikolaos Laoutaris, Rade Stanejovic, Konstantina Papagiannaki

Calum Harrison presented work on making rateless codes more power-efficient. Although rateless codes do a great job of approaching the Shannon capacity of the wireless channel, they're computationally expensive, and this can be a problem on mobile devices. This paper tries to also optimize for cost of decoding measured in terms of CPU operations, and gets roughly 10-70% fewer operations with competitive rate.

Shailendra Singh showed that there isn't one single wireless transmission strategy that is always best. CSMA, diversity-based schemes, NetMIMO—for each there exists a profile of the user (are they moving, how much interference is there, etc.) for which that scheme is better than the others, which this paper experimentally verified. TRINITY is a system they're building to automatically get the best of each scheme in a heterogeneous world.

Narseo Vallina-Rodriguez argued for something that may be slightly radical: "onloading" traffic from a wired DSL network onto wireless networks. We often think of wireless bandwidth as a scarce resource, but if there is spare wireless capacity, why not use it? Wireless throughput could easily be twice that of DSL in some situations, and 40% of users use less than 10% of their allocated wireless data volume (based on data from a medium-sized European mobile virtual network operator). As shown in "onloading" experiments in a variety of locations at different times, the authors can get order-of-magnitude improvements in video streaming buffering. Apparently the reviewers suggested that wireless providers wouldn't be a big fan of this—but Narseo noted that his coauthors are all from Telefonica. Interesting question from Brad Karp: How did we get here? Telefonica owns the DSL and wireless; if you need additional capacity is it cheaper to build out wireless capacity or wired? One response is that wired is significantly cheaper per byte, but we need to have wireless anyway. Another commenter: Onloading is promising because measurements show congestion on wireless and DSL peaks at different times. Open question: Is this benefit going to be true long term or is it an artifact of current overprovisioning?

3.2 Data Center Networks

- Coflow: An Application Layer Abstraction for Cluster Networking
  Mosharaf Chowdhury, Ion Stoica

- Hunting Mice with Microsecond Circuit Switches
  Nathan Farrington, George Porter, Yeshaiahu Fainman, George Papen, Amin Vahdat

- Deconstructing Datacenter Packet Transport
  Mohammad Alizadeh, Shuang Yang, Sachin Katti, Nick McKeown, Balaji Prabhakar, Scott Shenker

Mosharaf Chowdhury's work starts with the observation that the multiple recent projects improving data center flow scheduling are just that—*flow* schedulers—with each flow in isolation. Such a design ignores the fact that there are dependencies between flows due to applications: for example, a partition-aggregate workload may need all of its flows to finish, and if one finishes earlier, it's useless. The goal of Coflow is to expose that information to the network to improve scheduling. One question that was asked: What is the tradeoff with complexity of the API?

Nathan Farrington presented a new approach to building *hybrid* data center networks, that is, with both a traditional packet-switched network and a circuit-switched (e.g., optical) network. An optical switch provides much higher point-to-point bandwidth but switching is slow—far too slow for packet-level switching. Prior work on hybrid data center networks used *hotspot scheduling*, where the circuit switch is configured to connect input-output pairs that best help the elephant flows over a relatively long period of time. But performance of hot spot scheduling depends on the traffic matrix. Here, Nathan introduced Traffic Matrix Scheduling: the idea is to repeatedly iterate between a series of switch configurations (input-output assignments), such that the collection of all assignments fulfills the entire traffic matrix, including the mice. This algorithm requires much faster switching than hotspot schedulers, but by a remarkable stroke of good fortune, Farrington et al. [1] are also building an optical switch with microsecond-level switching times. Question: Once you reach 100% of traffic over optical, is there anything stopping you from eliminating the

packet switched network entirely? Yes, there is still latency on the order of 1 ms to complete one round of assignments, which is much higher than electrical data center network RTTs. Question: Where does the traffic matrix come from? Do you have to predict it, or wait until you've buffered some traffic? Either way, there's a tradeoff. (Presumably, however, other schedulers will run into a similar challenge.)

Mohammad Alizadeh took another look at finishing flows quickly in data centers. There are a number of recent protocols which are relatively complex, at end-hosts and especially in packet processing algorithms in routers. This new design is beautifully simple: each packet has a priority, and routers simply forward high priority packets first. They can have extremely small queues since the dropped packets are likely low priority anyway. End-hosts can set each packet's priority based on flow size, and perform only very simple rate control, to avoid congestion collapse. Performance is very good, though with some more work to do for elephant flows in high-utilization regimes.

3.3 Routing and Forwarding

- Compressing IP Forwarding Tables for Fun and Profit
  Gábor Rétvári, Zoltán Csernátony, Attila Kőrösi, János Tapolcai, András Császár, Gábor Enyedi, Gergely Pongrácz
- LOUP: Who's Afraid of the Big Bad Loop?
  Nikola Gvozdiev, Brad Karp, Mark Handley
- Outsourcing The Routing Control Logic: Better Internet Routing Based on SDN Principles
  Vasileios Kotronis, Bernhard Ager, Xenofontas Dimitropoulos

Gábor Rétvári tackled a compelling question: How much information is actually contained in a forwarding table? Turns out, there's less than you might think: with some new techniques, a realistic DFZ FIB compresses down to 60-400 KB, or 2-6 bits per prefix! A 4 million prefix FIB can fit in just 2.1 MB of memory. Now, the interesting thing is that this compression can support reasonably fast lookup directly on the compressed FIB, at least asymptotically speaking, based on an interesting new line of theory research on string self-indexing. The hope is that one could use this compressed representation in router hardware, making routers simpler and longer-lasting. In fact, Gábor demoed a prototype running as a Linux kernel module. One problem: They need more realistic FIBs to gain confidence in the conclusions. Widely-available looking glass servers are not good enough because they obscure the next-hops, which affect compression. Before the authors turn to a life of crime to obtain FIBs, why not send them yours?[1] Key question for the future: Can we use compressed forwarding tables at line speed?

Nicola Gvozdiev should win an award for best visualizations, with some nice animation of update propagation among iBGP routers. Their work is developing the algorithms and systems necessary to propagate state changes in iBGP, without causing any transient black holes or forwarding loops. Although we sometimes think of these problems as unavoidable consequences of distributed route convergence, they are not. The Link-Ordered Update Protocol (LOUP) presented here uses ordering constraints to avoid black holes and loops.

Vasileios Kotronis's work takes SDN-based routing a step further: Don't just centralize within a domain, outsource your routing control to a contractor! One tempting advantage here, besides reduced management costs, is that you can go beyond what an individual domain can otherwise do—for example, the contractor has interdomain visibility and can perform cross-domain optimization, debug policy conflicts, etc. which is typically difficult for a single domain.

3.4 User Behavior and Experience

- Understanding Rationality: Cognitive Bias in Network Services
  Rade Stanojevic, Vijay Erramilli, Konstantina Papagiannaki
- A Quest for an Internet Video Quality-of-Experience Metric
  Athula Balachandran, Vyas Sekar, Aditya Akella, Srinivasan Seshan, Ion Stoica, Hui Zhang
- Detecting Price and Search Discrimination in the Internet
  Jakub Mikians, László Gyarmati, Vijay Erramilli, Nikolaos Laoutaris

Rade Stanojevic presented results from a large data set of mobile service plans (roughly a billion each of calls, SMS/MMS messages, and data sessions). The question: Are economic models of how users select bandwidth and service plans realistic? What choices do real people make? Customers can choose from multiple service plans, which may be more or less advantageous for them. In fact, only 20% of customers choose the optimal tariff, resulting in a 37% mean and 26% median overpayment. Another interesting result: a customer's service utilization peaks immediately after purchase, and then decays steadily over at least a month, even with unlimited service (so it's not just because people are conservative as they near their service limits). Several questions were raised: Do these results really demonstrate irrationality? Users may buy more service than they need, so they don't need to worry about (and pay) comparatively pricey overage fees. Comment from an audience member: One has to imagine the marketing department of Telefonica has that exact same CDF of "irrationality" as their metric of success!

Athula Balachandran presented a study working towards a quantitative metric to score user experience of video delivery (in particular, how long users end up watching a video). The problem here is that predicting user experience based on quantitative observables is hard: it's a complex function of initial startup delay, how often the player buffers, buffering time, bit rate, the type of video, and more. The paper analyzes how well user experience can be predicted using several techniques, based on data from Conviva.

Vijay Erramilli presented a measurement study of how web sites act on information that they know about you. In particular, do sites use price discrimination based on information they collect about your browsing behavior? Starting with clean machines and having them visit sites based on certain high- or low-value browsing profiles, the authors measured how a set of search engines and shopping sites present results and prices to those different user profiles.

[1] retvari@tmit.bme.hu

They uncovered evidence of differences in search results, and some price differences on aggregators such as a mean 15% difference in hotel prices on Cheaptickets. Interestingly, there were also significant price differences based on the client's physical location. Saikat Guha asked a good question: How can you differentiate the vendor's intentional discrimination from unintentional? For example, in ad listings, having browsed a certain site can cause a Rolex ad to display, which bumps off an ad for a lower priced product, indirectly raising the mean price of displayed ads.

That's it! Congratulations to the organizing committee, chaired by Jitendra Padhye, Srikanth Kandula, Ramesh Govindan, and Emin Gün Sirer, for a very enjoyable event. I look forward to next year!

4. REFERENCES

[1] Nathan Farrington, George Porter, Pang-Chen Sun, Alex Forencich, Joseph Ford, Yeshaiahu Fainman, George Papen, and Amin Vahdat. A demonstration of ultra-low-latency datacenter optical circuit switching. In *ACM SIGCOMM Poster and Demo Session*, 2012.

[2] J.P. John, E. Katz-Bassett, A. Krishnamurthy, T. Anderson, and A. Venkataramani. Consensus routing: The Internet as a distributed system. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2008.