Seminar in Communication Networks

Learning, Reasoning and Control

Prof. Laurent Vanbever

nsg.ee.ethz.ch

ETH Zürich (D-ITET)

18 September 2019

Let's start by introducing ourselves! ☺

What…

is your name?

are you studying?

are your previous experiences in

- communication networks
- control theory
- machine learning?

are your expectations?

In this class, we'll look at how we can solve some fundamental problems in computer networks

Networks are hard to

- manage
- optimize
- secure

and not necessarily improving

Networks are **hard** to

- manage
- optimize
- secure

and not necessarily improving

Networks consist of thousands of different devices, and managing them all is difficult

Networks consist of thousands of different devices, and managing them all is difficult

Both in the network…

- Different switches
- Firewalls
- Traffic Analysers
- Optimizers
- NATs
- …

Datacenter Switch (cisco.com)

Networks consist of thousands of different devices, and managing them all is difficult

Both in the network…

Datacenter Switch (cisco.com)

…and connected to it!

- Home Routers
- Personal Computers
- Servers
- Mobile Phones
- IoT Devices
- …

IoT Camera (logitec.com)

Google accidentally broke the internet throughout Japan

A mistake led to internet outages for about half of the country.

Mallory Locklear, @mallorylocklear
08.28.17 in Internet

Forbes / Personal Finance

The Little Black Book of Billionaire Secrets

JUL 8, 2015 @ 03:36 PM 11,261 VIEWS

United Airlines Blames Router for Grounded Flights

'Configuration Error' Blamed for AWS Outage

By David Ramel ■ 08/12/2015

The summer of network misconfigurations

Amazon's massive AWS outage was caused by human error

One incorrect command and the whole internet suffers.

By Jason Del Rey | @DelRey | Mar 2, 2017, 2:20pm EST

CenturyLink: 750 calls to 911 missed during Aug. 1 outage caused by human error in Minnesota, North Dakota

By Barry Amundson on Aug 15, 2018 at 4:43 p.m.

Data Centre ▸ Networks

Level3 switch config blunder for US-wide VoIP blackout

affected Comcast, Spectrum, Verizon and AT&T customers

BY: CNN

Data Centre ▸ Networks

CloudFlare apologizes for Telia screwing you over

Unhappy abou

By Kieren McCarthy in

Facebook struggles to deal with epic outage

By Donie O'Sullivan and Heather Kelly, CNN Business

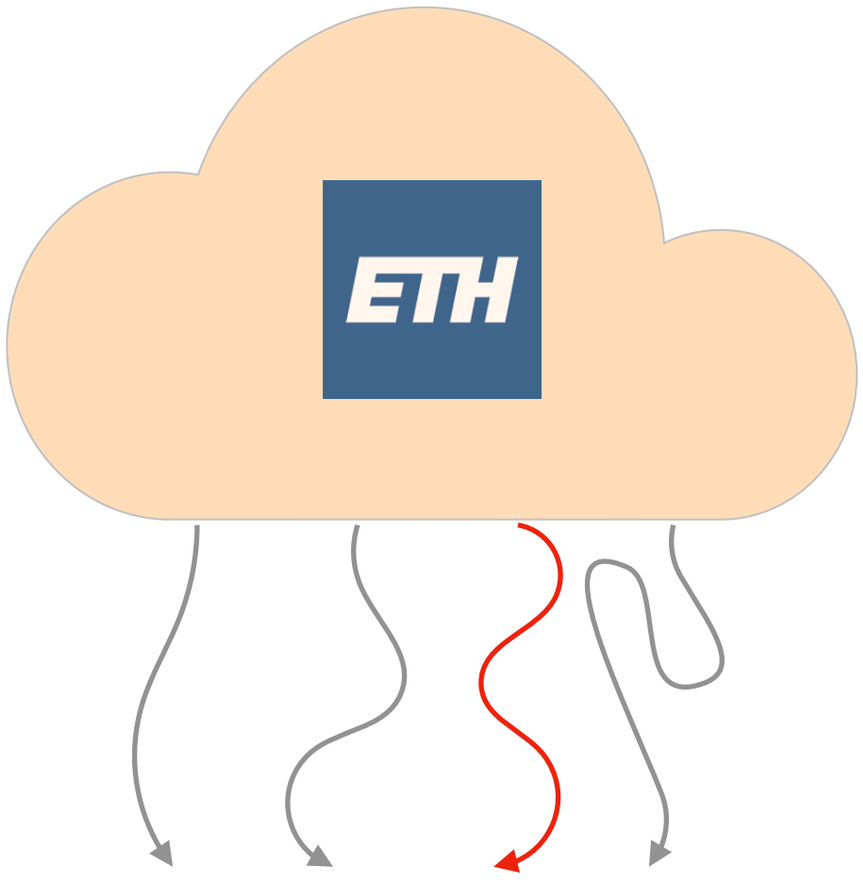
Updated 0654 GMT (1454 HKT) March 14, 2019

Networks are **hard** to

- manage
- optimize
- secure

and not necessarily improving

With most factors out of the networks control,
optimization is difficult.



Which path is the best?

- How can a network know how well a path works without using it?
- How will other networks react to sudden traffic shifts?
- How quickly does the environment change?

Networks are **hard** to

- manage
- optimize
- secure

and not necessarily improving

Network security and privacy are in conflict,

can learning solve this problem?

Encrypt traffic for privacy?

Or inspect everything
to discover attacks?

Network security and privacy are in conflict,
can learning solve this problem?

Encrypt traffic for privacy?

Or inspect everything
to discover attacks?

Can we learn to detect attacks without compromising privacy?

February 2018

1.35 Tbps DDoS largest to date

memcached-based

September 2016

>1 Tbps DDoS

botnet-based

Understanding the Mirai Botnet

Manos Antonakakis[◇] Tim April[‡] Michael Bailey[†] Matthew Bernhard[◁] Elie Bursztein[○]
Jaime Cochran[▷] Zakir Durumeric[◁] J. Alex Halderman[◁] Luca Invernizzi[○]
Michalis Kallitsis[§] Deepak Kumar[†] Chaz Lever[◇] Zane Ma[†*] Joshua Mason[†]
Damian Menscher[○] Chad Seaman[‡] Nick Sullivan[▷] Kurt Thomas[○] Yi Zhou[†]

[‡]*Akamai Technologies* [▷]*Cloudflare* [◇]*Georgia Institute of Technology* [○]*Google*
[§]*Merit Network* [†]*University of Illinois Urbana-Champaign* [◁]*University of Michigan*

Abstract

The Mirai botnet, composed primarily of embedded and IoT devices, took the Internet by storm in late 2016 when it overwhelmed several high-profile targets with massive distributed denial-of-service (DDoS) attacks. In this paper, we provide a seven-month retrospective analysis of Mirai's growth to a peak of 600k infections and a history of its DDoS victims. By combining a variety of measurement perspectives, we analyze how the botnet emerged, what classes of devices were affected, and how Mirai variants evolved and competed for vulnerable hosts. Our measurements serve as a lens into the fragile ecosystem of IoT devices. We argue that Mirai may represent a sea change in the evolutionary development of botnets—the simplicity through which devices were infected and its precipitous growth, demonstrate that novice malicious techniques can compromise enough low-end devices to threaten even some of the best-defended targets. To address this risk, we recommend technical and non-technical interventions, as well as propose future research directions.

1 Introduction

Starting in September 2016, a spree of massive distributed denial-of-service (DDoS) attacks temporarily crippled Krebs on Security [46], OVH [43], and Dyn [36]. The initial attack on Krebs exceeded 600 Gbps in volume [46]—among the largest on record. Remarkably, this overwhelming traffic was sourced from hundreds of thousands of some of the Internet's least powerful hosts—Internet of Things (IoT) devices—under the control of a new botnet named Mirai.

While other IoT botnets such as BASHLITE [86] and Carna [38] preceded Mirai, the latter was the first to emerge as a high-profile DDoS threat. What explains Mirai's sudden rise and massive scale? A combination of factors—efficient spreading based on Internet-wide scanning, rampant use of insecure default passwords in IoT products, and the insight that keeping the botnet's behavior simple would allow it to infect many heterogeneous devices—all played a role. Indeed, Mirai has spawned many variants that follow the same infection strategy, leading to speculation that "IoT botnets are the new normal of DDoS attacks" [64].

In this paper, we investigate the precipitous rise of Mirai and the fragile IoT ecosystem it has subverted. We present longitudinal measurements of the botnet's growth, composition, evolution, and DDoS activities from August 1, 2016 to February 28, 2017. We draw from a diverse set of vantage points including network telescope probes, Internet-wide banner scans, IoT honeypots, C2 milkers, DNS traces, and logs provided by attack victims. These unique datasets enable us to conduct the first comprehensive analysis of Mirai and posit technical and non-technical defenses that may stymie future attacks.

We track the outbreak of Mirai and find the botnet infected nearly 65,000 IoT devices in its first 20 hours before reaching a steady state population of 200,000–300,000 infections. These bots fell into a narrow band of geographic regions and autonomous systems, with Brazil, Columbia, and Vietnam disproportionately accounting for 41.5% of infections. We confirm that Mirai targeted a variety of IoT and embedded devices ranging from DVRs, IP cameras, routers, and printers, but find Mirai's ultimate device composition was strongly influenced by the market shares and design decisions of a handful of consumer electronics manufacturers.

By statically analyzing over 1,000 malware samples, we document the evolution of Mirai into dozens of variants propagated by multiple, competing botnet operators. These variants attempted to improve Mirai's detection avoidance techniques, add new IoT device targets, and introduce additional DNS resilience. We find that Mirai harnessed its evolving capabilities to launch over 15,000 attacks against not only high-profile targets (e.g., Krebs

*Denotes primary, lead, or "first" author

https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-antonakakis.pdf

Your turn! ☺

How can control theory help managing networks?

machine learning

optimizing

securing

Your teaching assistants for the semester

Albert Gran Alcoz

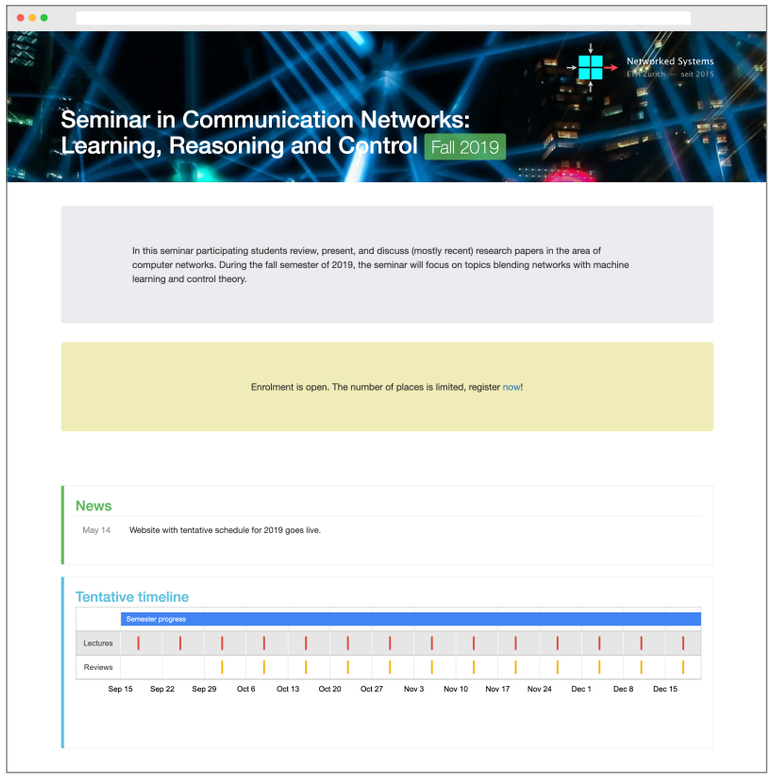
galberto@ethz.ch

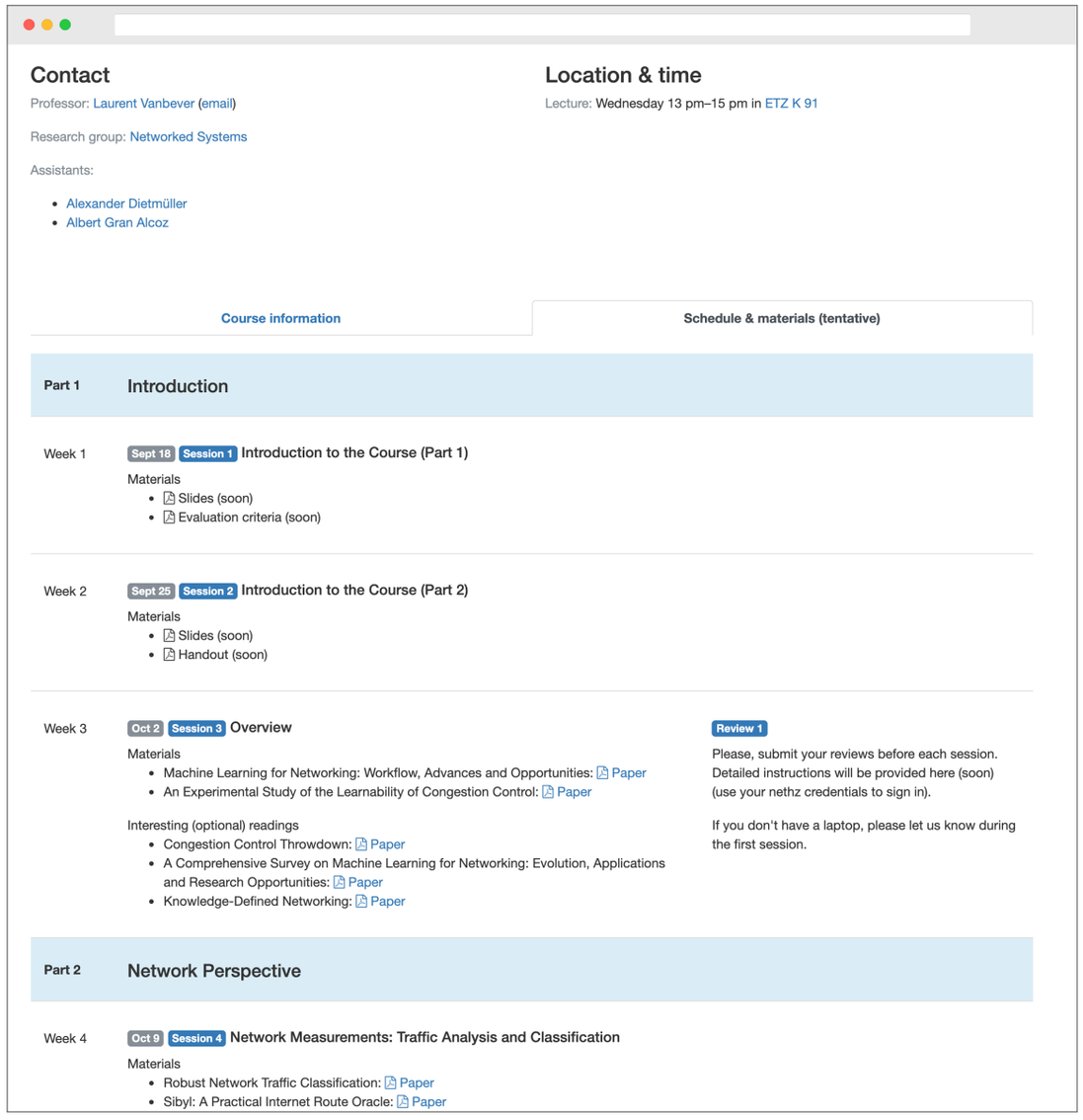
Alexander Dietmüller

adietmueller@ethz.ch

Regularly check out the course website

https://seminar-net.ethz.ch





Your final grade will be based on your

Presentation	Reviews	Participation
45% of the grade	40% of the grade	15% of the grade

(there is no final exam)

Your final grade will be based on your

Presentation

Reviews

Participation

45% of the grade

Each student will present and animate the discussion for one research paper during the semester

length	20 min (not including questions)
	you *cannot* reuse slides found online
format	summarize the problem
	highlight key ideas (skip the details)
	describe main results
discussion	discuss strengths/weaknesses
	propose follow-up work (if any)

Each student will present and animate the discussion for one research paper during the semester

submission	email your slides to nsg-seminar@ethz.ch by 23.59pm the day before your presentation
optional feedback	organize a meeting with the TAs the week before your presentation

Hint

#1 Prepare your presentation early

#2 Practice, *a lot*

#3 Watch "*Creating effective slides: Design, Construction, and Use in Science*"

https://www.youtube.com/watch?v=meBXuTIPJQk&t

Your final grade will be based on your

Presentation

Reviews

Participation

40% of the grade

Each week you'll write a short review
for one out of two selected paper

format

Summarize the problem. Is it real?

Describe the key insights

How is it different from previous solutions?

Highlight strengths/weaknesses

submission

Submit by Tuesday evening, 11.59pm

you can miss one without penalty

Check out these useful references

How to read a research paper?

by Srinivasan Keshav [PDF]

by Mitzenmacher and Ramsey [PDF]

How to write good reviews

by Timothy Roscoe [PDF]

Your final grade will be based on your

Presentation

Reviews

Participation

15% of the grade

A part of your grade will be based on

in-class participation

How do you moderate the discussion after your talk

How much do you participate in the discussions throughout the semester

Ask questions. Share your perspective. Be curious!

This is *not* a competition about who says the most

We'll communicate your expected participation grade half-way through the class

A glimpse at the papers

we're gonna discuss together

Course information | Schedule & materials (tentative)

Part 1 Introduction

Week 1

Sept 18 Session 1 **Introduction to the Course (Part 1)**

Materials
- Slides (soon)
- Evaluation criteria (soon)

Week 2

Sept 25 Session 2 **Introduction to the Course (Part 2)**

Materials
- Slides (soon)
- Handout (soon)

Exercise

Before this session, you should read the paper: Machine Learning for Networking: Workflow, Advances and Opportunities.

Week 3

Oct 2 Session 3 **Overview**

Materials
- An Experimental Study of the Learnability of Congestion Control: Paper

Interesting (optional) readings
- Congestion Control Throwdown: Paper
- Knowledge-Defined Networking: Paper
- A Comprehensive Survey on Machine Learning for Networking: Evolution, Applications and Research Opportunities: Paper

Review 1

Please, submit your reviews before each session. Detailed instructions will be provided here (soon) (use your nethz credentials to sign in).

If you don't have a laptop, please let us know during the first session.

Part 2 **Network Perspective**

Table of contents

Table of contents


ACCEPTED FROM OPEN CALL

Machine Learning for Networking: Workflow, Advances and Opportunities

Overview

Machine Learning for Networking: Workflow, Advances and Opportunities

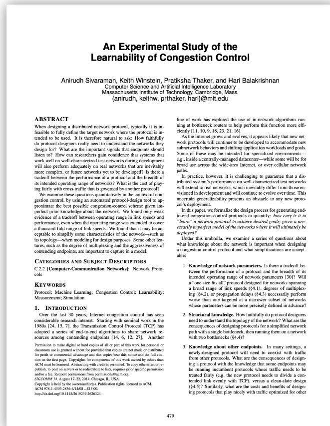
[IEEE Network '17]

…presents an overview of the workflow for ML in networking.

applications

performance

opportunities


An Experimental Study of the
Learnability of Congestion Control
Anirudh Sivaraman, Keith Winstein, Pratiksha Thaker, and Hari Balakrishnan
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, Cambridge, Mass.
{anirudh, keithw, prthaker, hari}@mit.edu

Overview

An Experimental Study of the Learnability of Congestion Control

[ACM SIGCOMM ’14]

How can we **generate a congestion control protocol?**

Can we **learn, or** must we **know**:
- Network Parameters
- Topology
- Cross Traffic
- Network Signals

These questions are **explored through experiments.**

Table of contents

Network Measurements: Traffic Classification and Analysis

Network Measurements: Traffic Classification


Neural Packet Classification

Neural Packet Classification

[SIGCOMM ’19]

- Fundamental problem

  firewalls, ACLs, traffic engineering, measurements

- Matching a packet to a rule

  TCAMs or hand-made decision trees

Class	Src IP	Dst IP	Protocol	Priority
1	10.0.0.0	10.0.0.0/16	*	2
2	*	*	TCP	1

- Deep RL to build decision trees

  given a set of rules and the objective (memory, time)

Network Measurements: Traffic Analysis

Sibyl: A Practical Internet Route Oracle

[NSDI ’16]

Sibyl: A Practical Internet Route Oracle

- Operators troubleshooting problems

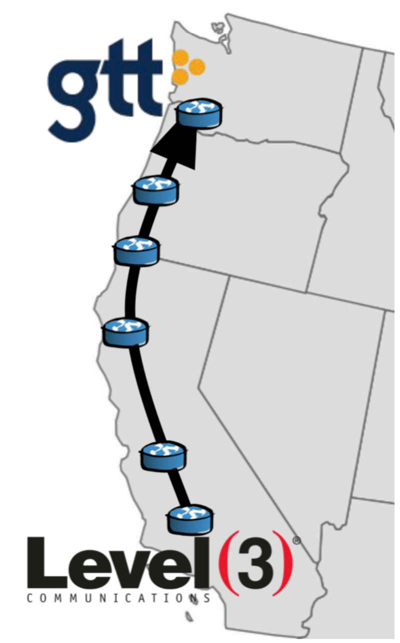
  are other routes through gtt in Seattle experiencing problems?

- Today, mailing lists are used

  .. or (limited) traceroutes from VPs

- Sibyl: High-level queries over internet routes

  optimizing measurements across platforms



Network Measurements: Traffic Analysis

Sibyl: A Practical Internet Route Oracle

[NSDI '16]

- Regular-expression syntax to express queries


Sibyl: A Practical Internet Route Oracle

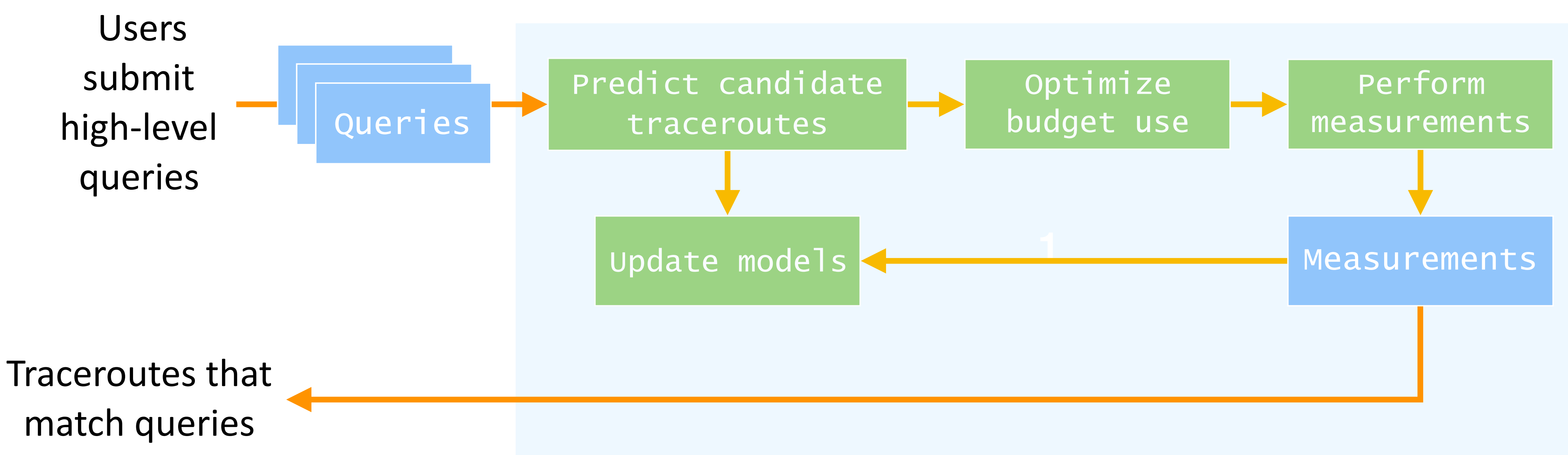
Network Measurements: Traffic Analysis

Sibyl: A Practical Internet Route Oracle

[NSDI ’16]

- Predict traceroutes that would likely match the query

  ML from the record of queries

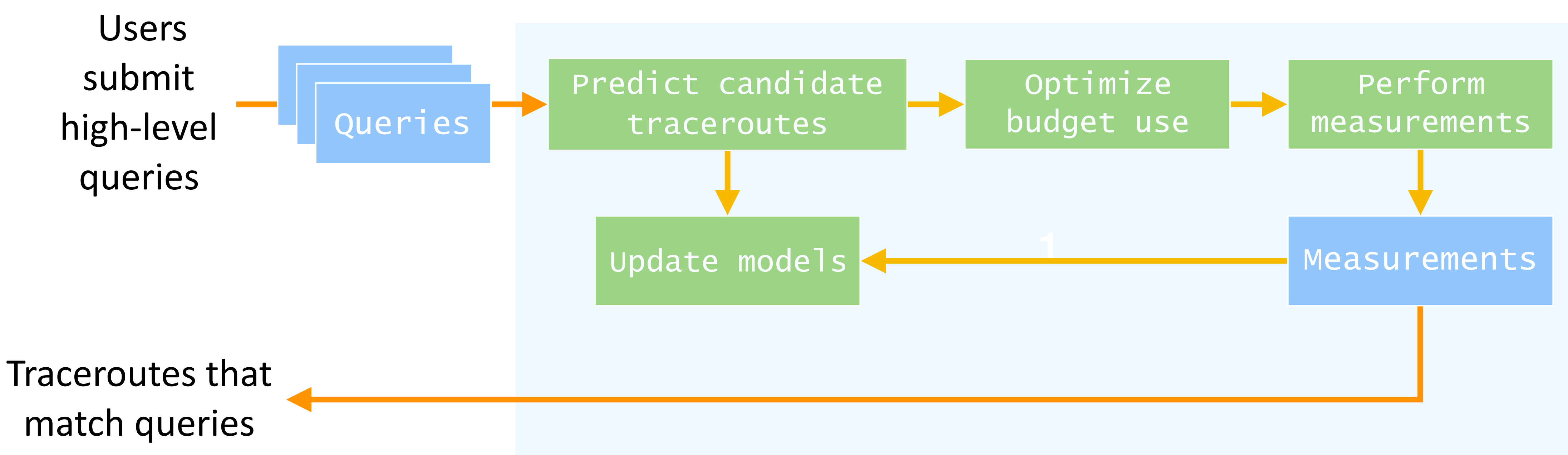


Network Measurements: Traffic Analysis

Sibyl: A Practical Internet Route Oracle

[NSDI ’16]

- An optimization framework that selects measurements to maximize query satisfaction within the budget



Network Measurements: Anomaly Detection


2010 IEEE Symposium on Security and Privacy

Outside the Closed World:
On Using Machine Learning For Network Intrusion Detection

Robin Sommer

Vern Paxson

Network Measurements: Anomaly Detection

Outside the Closed World: On Using Machine Learning for Network Intrusion Detection

[IEEE S&P '10]

Why is ML not being successful for anomaly detection in networking?

- Not suitable for *novel* attacks

  strength in detecting previously-seen activity

- High cost of errors

- Data variability and lack of data

It is not inappropriate,
but requires care…


2010 IEEE Symposium on Security and Privacy

Outside the Closed World:
On Using Machine Learning For Network Intrusion Detection

Robin Sommer

Vern Paxson

Network Measurements: Anomaly Detection

Outside the Closed World: On Using Machine Learning for Network Intrusion Detection

[IEEE S&P ’10]

Bonus track:
Demystifying DL in Networking

Why is ML not being successful for anomaly detection in networking?

- Not suitable for *novel* attacks

  strength in detecting previously-seen activity

- High cost of errors

- Data variability and lack of data

It is not inappropriate,
but requires care…

Detecting Credential Spearphishing Attacks in Enterprise Settings

Grant Ho[†] Aashish Sharma[◇] Mobin Javed[†] Vern Paxson[†*] David Wagner[†]

[†]UC Berkeley [◇]Lawrence Berkeley National Laboratory [*]International Computer Science Institute

Network Measurements: Anomaly Detection

Detecting Credential Spearphishing Attacks in Enterprise Settings

[USENIX Security '17]

Distinguished Paper Award

- Credential spear-phishing

  customized attack on a specific employee in a company

- Standard ML not useful

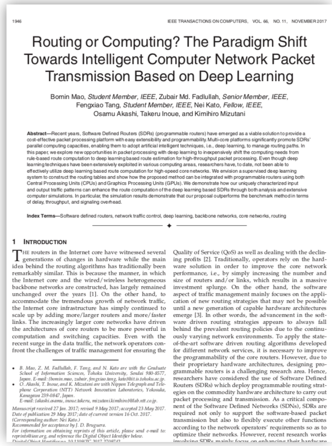
  extreme class imbalance

  10 attacks in 370 million emails

- Domain knowledge to reduce false-positive rate

  1 month of alerts, 15 min

Network Configuration


Routing or Computing? The Paradigm Shift Towards Intelligent Computer Network Packet Transmission Based on Deep Learning

Bomin Mao, Student Member, IEEE, Zubair Md. Fadlullah, Senior Member, IEEE, Fengxiao Tang, Student Member, IEEE, Nei Kato, Fellow, IEEE, Osamu Akashi, Takeru Inoue, and Kimihiro Mizutani

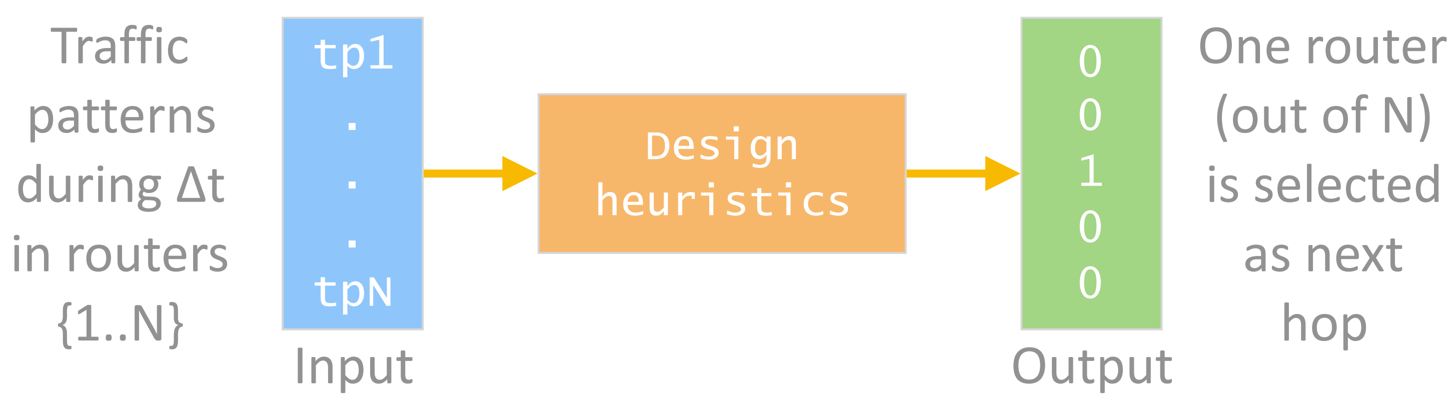
Network Configuration

Routing or Computing? The Paradigm Shift Towards Intelligent Computer Network Packet Transmission

[IEEE ToC '17]

- Route computation has remained the same over years
- Shortest-path algorithm has some caveats

  slow convergence and multi-metric cost
- Can we compute routing tables using DL?



CherryPick: Adaptively Unearthing the Best Cloud Configurations for Big Data Analytics

Omid Alipourfard[1], Hongqiang Harry Liu[2], Jianshu Chen[3], Shivaram Venkataraman[3], Minlan Yu[1], Ming Zhang[2]

[1]Yale University, [2]Microsoft Research, [3]University of California, Berkeley, [4]Alibaba Group

Abstract — Picking the right cloud configuration for recurring big data analytics jobs running in clouds is hard, because there can be tens of possible VM instance types and even more cluster sizes to pick from. Choosing poorly can significantly degrade performance and increase the cost to run a job by 2-3x on average, and as much as 12x in the worst-case. However, it is challenging to automatically identify the best configuration for a broad spectrum of applications and cloud configurations with low search cost. CherryPick is a system that leverages Bayesian Optimization to build performance models for various applications, and the models are just accurate enough to distinguish near-optimal configurations from the rest with only a few test runs. Our experiments on five analytic applications in AWS EC2 show that CherryPick has a 45-90% chance to find optimal configurations, otherwise near-optimal, saving up to 75% search cost compared to existing solutions.

1 Introduction

Big data analytics running on clouds are growing rapidly and have become critical for almost every industry. To support a wide variety of use-cases, a number of evolving techniques are used for data processing, such as Map Reduce, SQL-like languages, Deep Learning, and in-memory analytics. The execution environments of such big data analytic applications are structurally similar: a cluster of virtual machines (VMs). However, since different analytic jobs have diverse behaviors and resource requirements (e.g., CPU, memory, network), their cloud configurations – the types of VM instances and the number of VMs – cannot simply be satisfied.

Choosing the right cloud configuration for an application is essential to service quality and commercial competitiveness. For instance, a bad cloud configuration can result in up to 12 times more cost for the same performance target. The saving from a proper cloud configuration is even more significant for *recurring* jobs [10, 17] in which similar workloads are executed repeatedly. Nonetheless, selecting the best cloud configuration, e.g., the cheapest or the fastest, is difficult due to the complexity of simultaneously achieving high accuracy, low overhead, and adaptivity for different applications and workloads.

Accuracy The running time and cost of an application have complex relations to the resources of the cloud instances, the input workload, internal workflows, and configuration of the application. It is difficult to use straightforward methods to model such relations. Moreover, cloud dynamics such as network congestions and stragglers introduce substantial noise [23, 38].

Overhead Brute-force search for the best cloud configuration is expensive. Developers for analytic applications often face a wide range of cloud configuration choices. For example, Amazon EC2 and Microsoft Azure offer over 40 VM instance types with a variety of CPU, memory, disk, and network options. Google provides 18 types and also allows customizing VMs' memory and the number of CPU cores [2]. Additionally, developers also need to choose the right cluster size.

Adaptivity Big data applications have diverse internal architectures and dependencies within their data processing stages. Manually learning to build the internal structures of individual applications' performance model is not scalable.

Existing solutions do not fully address all of the preceding challenges. For example, Ernest [37] trains a performance model for machine learning applications with a small number of samples but since its performance model is tightly bound to the particular structure of machine learning jobs, it does not work well for applications such as SQL queries (poor adaptivity). Further, Ernest can only select VM sizes within a given instance family, and performance models need to be retrained for each instance family.

To address the above, we present CherryPick—a system that unearths the optimal or near-optimal cloud configurations that minimize cloud usage cost, guarantee application performance and limit the search overhead for executing big data analytic jobs. Each configuration is represented as the number of VMs, CPU count, CPU speed per core, RAM per core, disk count, disk speed, and network capacity of the VM.

The key idea of CherryPick is to build a performance model that is *just accurate enough* to allow us to distinguish near-optimal configurations from the rest. Tolerating the inaccuracy of the model enables us to achieve both low overhead and adaptivity: only a few samples

Network Configuration

CherryPick: Adaptively Unearthing the Best Cloud Configurations for Big Data Analytics

[NSDI '17]

- Big data analytics are very common today

  data bases, ML, stream processing…

- Many options available

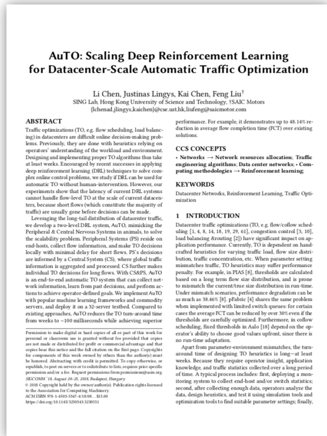
  cloud provider, machine type, cluster size

- How to find the best cloud configuration?

  minimizes the cost given a performance

  for a recurrent job, given its representative workload

Network Adaptation



AuTO: Scaling Deep Reinforcement Learning for Datacenter-Scale Automatic Traffic Optimization

Network Adaptation

AuTO: Scaling Deep Reinforcement Learning for Datacenter-Scale Automatic Traffic Optimization

[SIGCOMM ’18]

- Traffic optimization strongly impacts performance

  flow scheduling, congestion control, load balancing…

- Today, based on hand-crafted heuristics

  long design process, performance penalty if model mismatch

- Flow-level DRL traffic-optimization agent

  adapting to uncertain and volatile traffic

  2-step decision process to prevent short-flows

Network Adaptation

Learning Scheduling Algorithms for Data Processing Clusters

[SIGCOMM '19]

- Scheduling has a great impact in computation time

  which job should go next?

- Best scheduling algorithm depend on the specific workload and system

  cloud provider, machine type, cluster size

- Current scheduling algorithms rely on heuristics

  FIFO, SJF, Fair Queuing, far from optimal

Can ML tame this complexity?

Network Adaptation

Learning Scheduling Algorithms for Data Processing Clusters

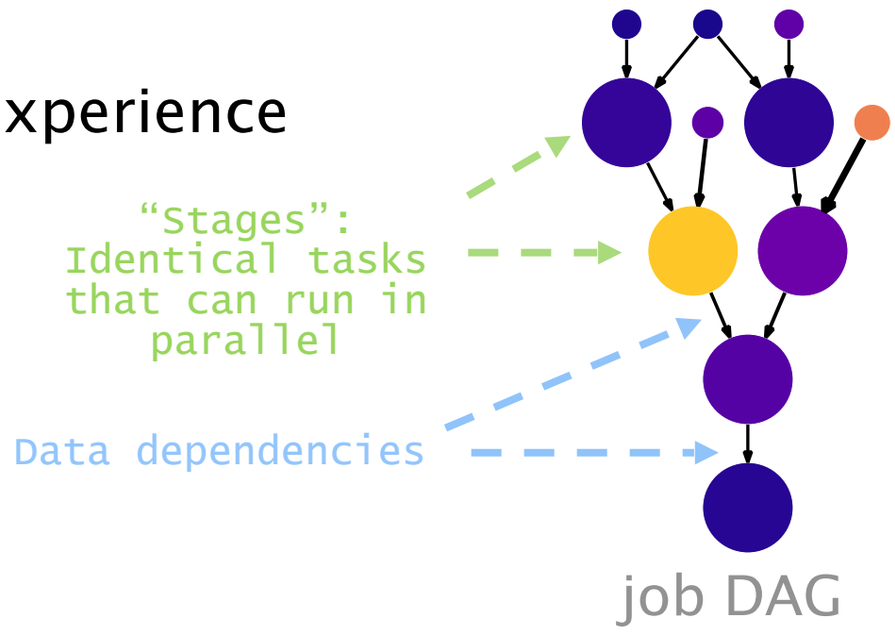
[SIGCOMM '19]

- Decima learns workload-specific scheduling algorithms

  jobs which have dependencies

  represented as directed acyclic graphs (DAG)

- Automatically through experience



Network Adaptation

Learning Scheduling Algorithms for Data Processing Clusters

[SIGCOMM '19]

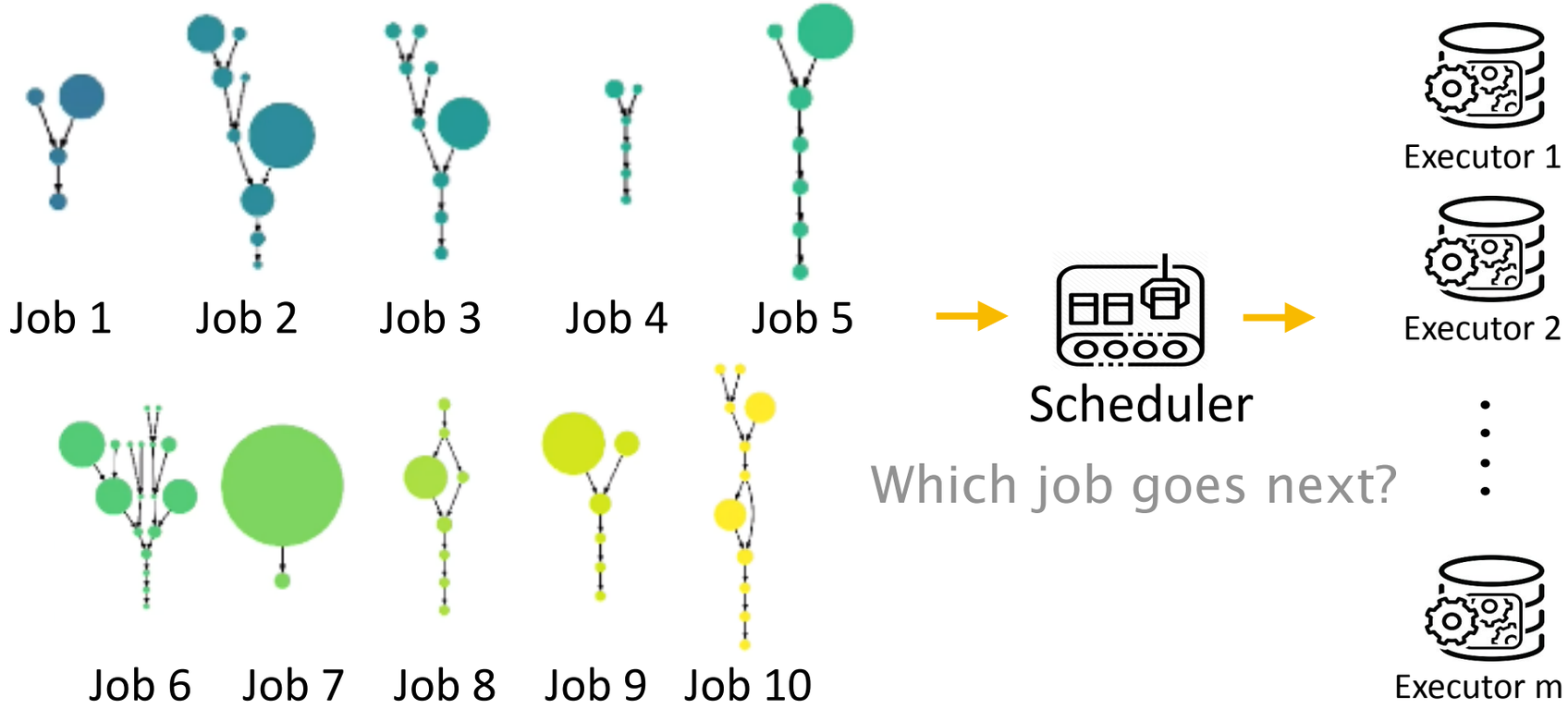
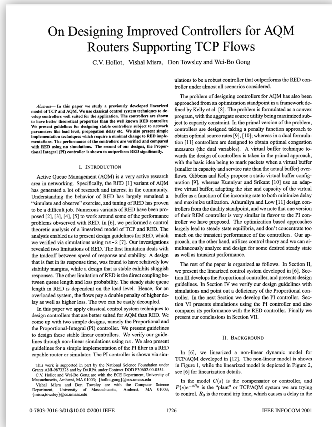


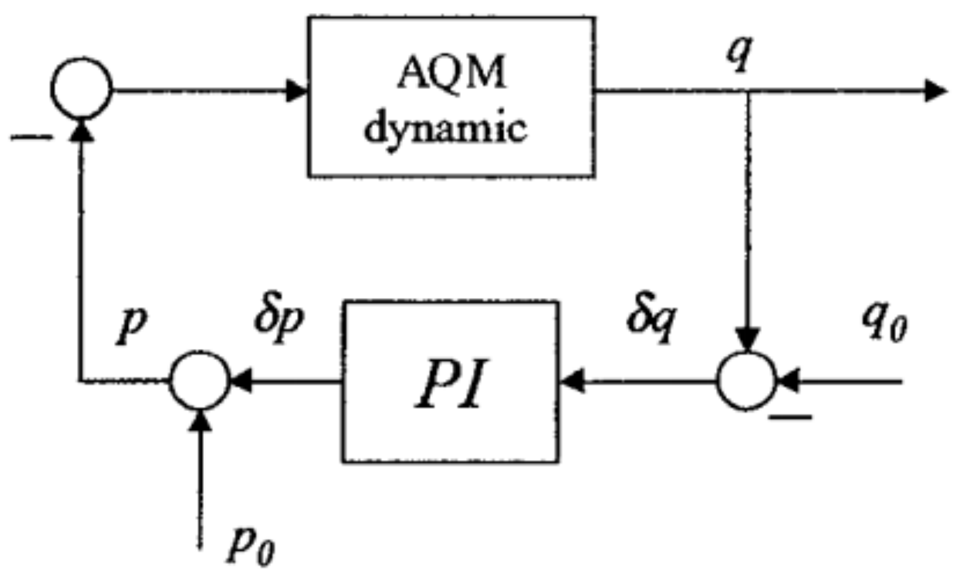
Table of contents



End-host Perspective

On designing improved Controllers for AQM routers supporting TCP flows

[IEEE INFOCOM '01]



The authors
use control theory and
model TCP dynamics to
design a PI controller
for Active Queue Management.

Rate control for communication networks: shadow prices, proportional fairness and stability

FP Kelly, AK Maulloo and DKH Tan
University of Cambridge, UK

This paper analyses the stability and fairness of two classes of rate control algorithm for communication networks. The algorithms provide natural generalisations to large-scale networks of simple additive increase/multiplicative decrease schemes, and are shown to be stable about a system optimum characterised by a proportional fairness criterion. Stability is established by showing that, with an appropriate formulation of the overall optimisation problem, the network's implicit objective function provides a Lyapunov function for the dynamical system defined by the rate control algorithm. The network's optimisation problem may be cast in primal or dual form: this leads naturally to two classes of algorithm, which may be interpreted in terms of either congestion indication feedback signals or explicit rates based on shadow prices. Both classes of algorithm may be generalised to include routing control, and provide natural implementations of proportionally fair pricing.

Keywords: ATM network; congestion indication; elastic traffic; Internet; Lyapunov function; proportionally fair pricing; queues; routing

Introduction

The design and control of modern communication networks raises several issues well suited to study using techniques of operational research such as optimisation, network programming and stochastic modelling. In this paper we illustrate this theme, through the presentation and analysis of a mathematical model that arises in connection with the development and deployment of large-scale broadband networks.

In future communication networks there are expected to be applications that are able to modify their data transfer rates according to the available bandwidth within the network. Traffic from such applications is termed *elastic*;[1] a typical current example is TCP traffic over the Internet,[2] and future examples may include the controlled-load service of the Internet Engineering Task Force[3] and the Available Bit Rate transfer capability of ATM (asynchronous transfer mode) networks.[4]

The key issue we address in this paper concerns how the available bandwidth within the network should be shared between competing streams of elastic traffic; in particular, we present a tractable mathematical model and use it to analyse the stability and fairness of a class of rate control algorithms. Traditionally stability has been considered an engineering issue, requiring an analysis of randomness and feedback operating on fast time-scales, while fairness has been considered an economic issue, involving static comparisons of utility. In future networks the intelligence embedded in end-systems, acting on behalf of human users, is likely to lessen the distinction between engineering and economic issues and increase the importance of an inter-disciplinary view. (This general theme was the subject of the 1996 Blackett Memorial Lecture; further aspects are developed elsewhere, see Reference 5).

There is a substantial literature on rate control algorithms, recently reviewed by Hernandez-Valencia *et al*.[6] Key early papers of Jacobson[2] and Chiu and Jain[7] identified the advantages of adaptive schemes that either increase flows linearly or decrease flows multiplicatively, depending on the absence or presence of congestion. Important recent papers of Bolot and Shankar,[8] Fendick *et al*[9] and Bonomi *et al*[10] have analysed the stability of networks with a single bottleneck resource, where congestion is signalled by the build-up of a queue at the bottleneck's buffer, and where propagation delays are significant. (In wide-area networks propagation times may be significant in comparison with queueing times: for a transatlantic link of 600 Megabits per second, ten million bits may be in flight between queues.) The framework we adopt in this paper is simpler than that analysed by these authors in that we directly model only rates and not queue lengths, but more complex in that we model a network with an arbitrary number of bottleneck resources. Theoretical work[11,12] on queues serving the superposition of a large number of streams indicates circumstances when the busy period preceding a buffer overflow may be relatively short, and several authors have argued the advantages of preventing queue build-up through the bounding of rates (see Charny *et al*).[13]

Correspondence: Frank Kelly, Statistical Laboratory, University of Cambridge, 16 Mill Lane, Cambridge CB2 1SB, UK.
E-mail: f.p.kelly@statslab.cam.ac.uk

End-host Perspective

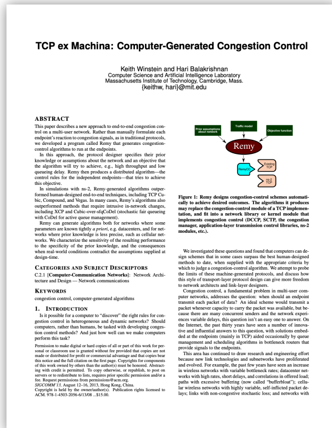
Rate control for communication networks: shadow prices, proportional fairness and stability

[Journal of the Operational Research Society '98]

What game are you playing?

Model the…
optimisation problems behind rate-control

Analyze the…
stability & fairness of the corresponding solution


TCP ex Machina: Computer-Generated Congestion Control

Keith Winstein and Hari Balakrishnan
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, Cambridge, Mass.
{keithw, hari}@mit.edu

End-host Perspective

TCP ex Machina: Computer-Generated Congestion Control

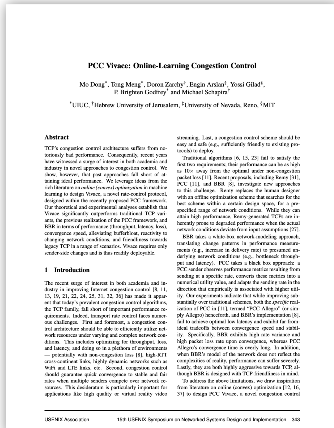
[ACM SIGCOMM '13]

Using a network model & global utility function,
optimize this mapping to generate a CC algorithm:

$$\langle \texttt{ack\_ewma}, \texttt{send\_ewma}, \texttt{rtt\_ratio} \rangle \to \langle m, b, r \rangle.$$

congestion signals measured by senders

parameters that define how senders react to the signals



End-host Perspective

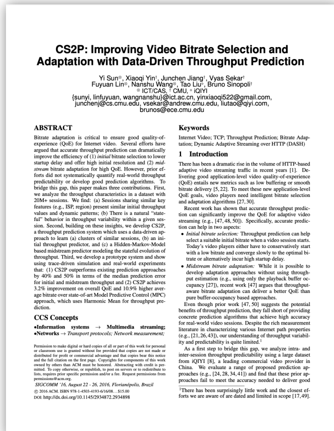
PCC Vivace:
Online-Learning Congestion Control

[USENIX NSDI ’18]

$$u\left(x_i, \frac{d(RTT_i)}{dT}, L_i\right) = x_i^t - b x_i \frac{d(RTT_i)}{dT} - c x_i \times L_i$$

t, b, and c are careful chosen to ensure fairness, convergence, and robustness.

senders compute their own utility, and
probe different sending rates to learn the rate for optimal utility

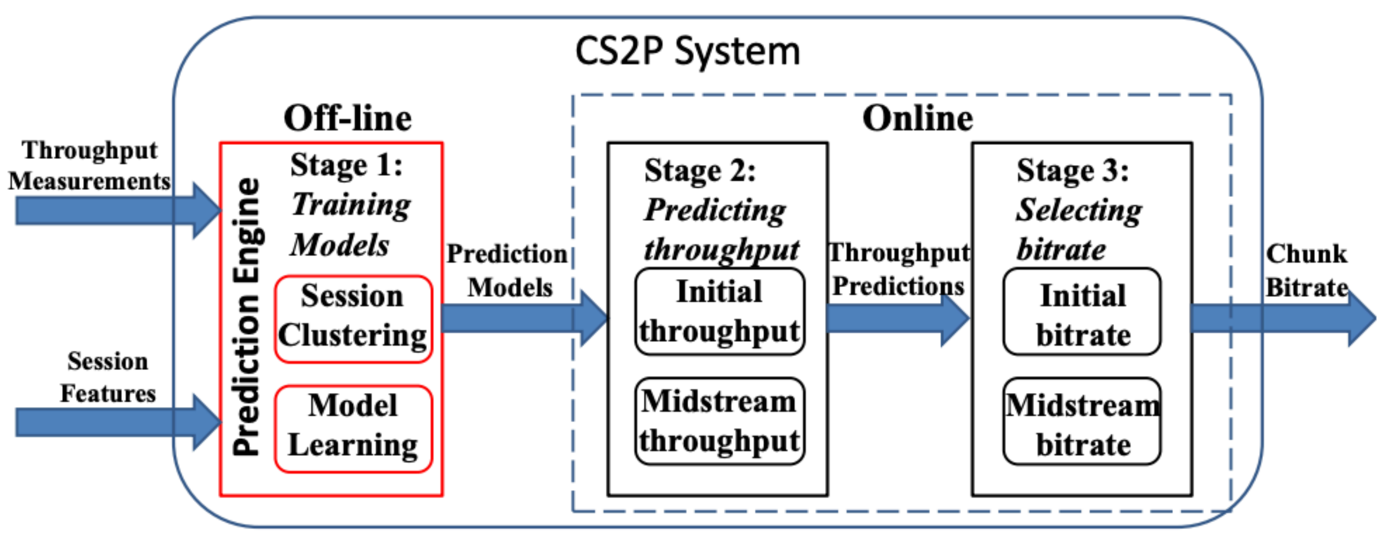

CS2P: Improving Video Bitrate Selection and Adaptation with Data-Driven Throughput Prediction

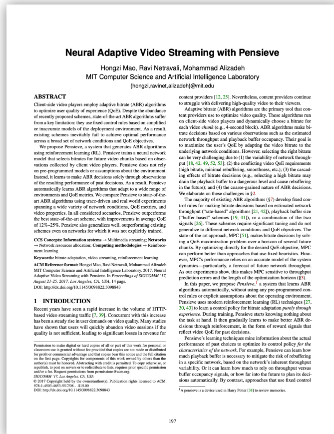
End-host Perspective

C2SP: Improving Video Bitrate Selection and Adaptation with Data-Driven Throughput Prediction

[ACM SIGCOMM ’16]

CS2P learns the dynamics of video streams from data
using Hidden Markov Models to adapt video bitrates.



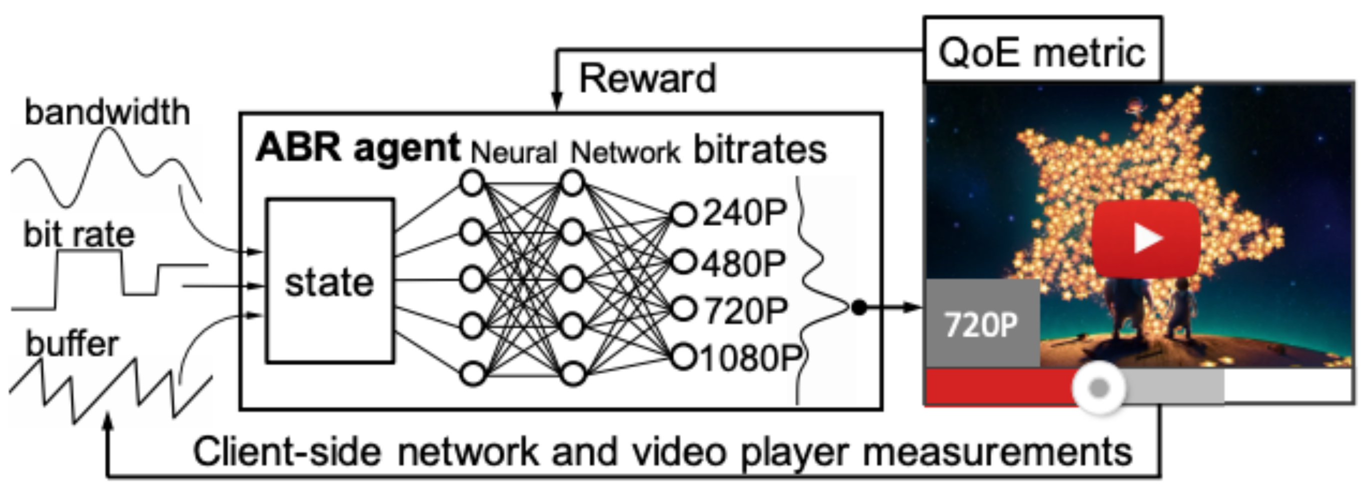

Neural Adaptive Video Streaming with Pensieve
Hongzi Mao, Ravi Netravali, Mohammad Alizadeh
MIT Computer Science and Artificial Intelligence Laboratory
{hongzi,ravinet,alizadeh}@mit.edu

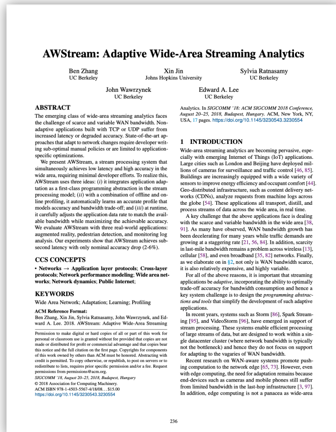
End-host Perspective

Neural Adaptive Video Streaming with Pensieve

[ACM SIGCOMM '17]

Pensieve uses Reinforcement Learning with Neural Networks
based on an Actor-Critic model to adapt video bitrates.




AWStream: Adaptive Wide-Area Streaming Analytics

Ben Zhang, UC Berkeley; Xin Jin, Johns Hopkins University; Sylvia Ratnasamy, UC Berkeley; John Wawrzynek, UC Berkeley; Edward A. Lee, UC Berkeley

ABSTRACT

1 INTRODUCTION

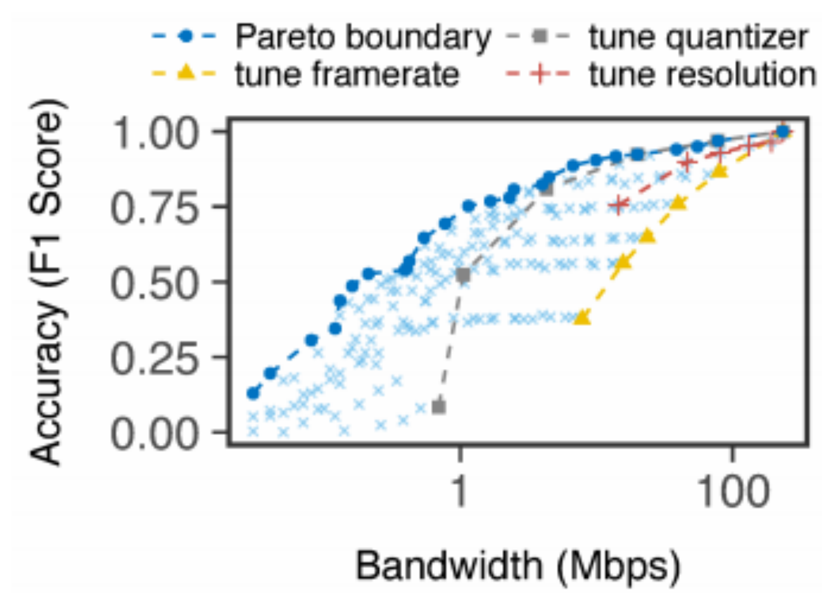
End-host Perspective

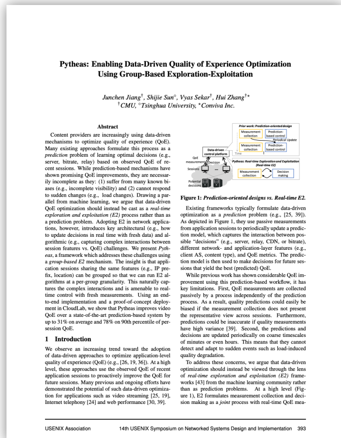
AWStream: Adaptive Wide-Area Streaming Analytics

[ACM SIGCOMM '18]

To optimize bandwidth adaptation, **AWStream** combines **custom degradation operations** with **off- and online profiling.**

```
let app = Camera::new((1920, 1080), 30)
    .maybe_downsample(vec![(1600, 900), (1280, 720)])
    .maybe_skip(vec![2, 5])
    .map(|frame| frame.show())
    .compose();
```





Pytheas: Enabling Data-Driven Quality of Experience Optimization Using Group-Based Exploration-Exploitation

Junchen Jiang[†], *Shijie Sun*[°], *Vyas Sekar*[†], *Hui Zhang*[†*]
[†]*CMU,* [°]*Tsinghua University,* [*]*Conviva Inc.*

Abstract

Content providers are increasingly using data-driven mechanisms to optimize quality of experience (QoE). Many existing approaches formulate this process as a *prediction* problem of learning optimal decisions (e.g., server, bitrate, relay) based on observed QoE of recent sessions. While prediction-based mechanisms have shown promising QoE improvements, they are necessarily incomplete as they: (1) suffer from many known biases (e.g., incomplete visibility) and (2) cannot respond to sudden changes (e.g., load changes). Drawing a parallel from machine learning, we argue that data-driven QoE optimization should instead be cast as a *real-time exploration and exploitation (E2)* process rather than as a prediction problem. Adopting E2 in network applications, however, introduces key architectural (e.g., how to update decisions in real time with fresh data) and algorithmic (e.g., capturing complex interactions between session features vs. QoE) challenges. We present *Pytheas*, a framework which addresses these challenges using a *group-based E2* mechanism. The insight is that application sessions sharing the same features (e.g., IP prefix, location) can be grouped so that we can run E2 algorithms at a per-group granularity. This naturally captures the complex interactions and is amenable to real-time control with fresh measurements. Using an end-to-end implementation and a proof-of-concept deployment in CloudLab, we show that Pytheas improves video QoE over a state-of-the-art prediction-based system by up to 31% on average and 78% on 90th percentile of per-session QoE.

1 Introduction

We observe an increasing trend toward the adoption of data-driven approaches to optimize application-level quality of experience (QoE) (e.g., [26, 19, 36]). At a high level, these approaches use the observed QoE of recent application sessions to proactively improve the QoE for future sessions. Many previous and ongoing efforts have demonstrated the potential of such data-driven optimization for applications such as video streaming [25, 19], Internet telephony [24] and web performance [30, 39].

Figure 1: ***Prediction-oriented designs vs. Real-time E2.***

Existing frameworks typically formulate data-driven optimization as a *prediction* problem (e.g., [25, 39]). As depicted in Figure 1, they use passive measurements from application sessions to periodically update a prediction model, which captures the interaction between possible "decisions" (e.g., server, relay, CDN, or bitrate), different network- and application-layer features (e.g., client AS, content type), and QoE metrics. The prediction model is then used to make decisions for future sessions that yield the best (predicted) QoE.

While previous work has shown considerable QoE improvement using this prediction-based workflow, it has key limitations. First, QoE measurements are collected passively by a process independently of the prediction process. As a result, quality predictions could easily be biased if the measurement collection does not present the representative view across sessions. Furthermore, predictions could be inaccurate if quality measurements have high variance [39]. Second, the predictions and decisions are updated periodically on coarse timescales of minutes or even hours. This means that they cannot detect and adapt to sudden events such as load-induced quality degradation.

To address these concerns, we argue that data-driven optimization should instead be viewed through the lens of *real-time exploration and exploitation (E2)* frameworks [43] from the machine learning community rather than as prediction problems. At a high level (Figure 1), E2 formulates measurement collection and decision making as a *joint* process with real-time QoE mea-

USENIX Association 14th USENIX Symposium on Networked Systems Design and Implementation 393

End-host Perspective

Pytheas: Enabling Data-Driven Quality of Experience Optimization Using Group-Based Exploration-Exploitation

[USENIX NSDI '17]

Pytheas is based on…

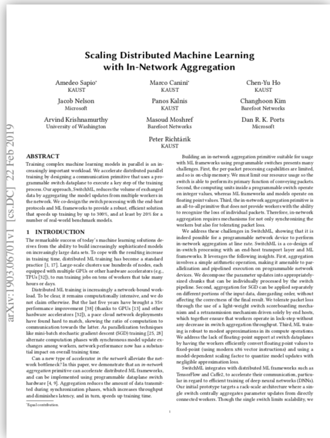
…learning the decision space in real time with **exploration/exploitation**

and

…grouping apps by **critical features** for scalability

Table of contents

In-Network Machine Learning



Scaling Distributed Machine Learning with In-Network Aggregation

In-Network Machine Learning

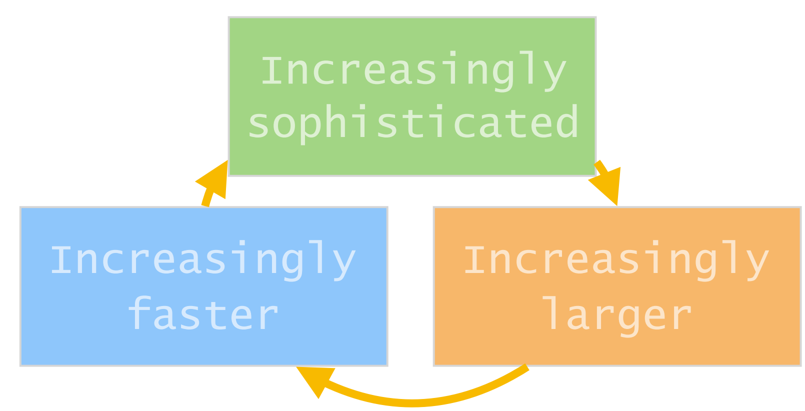
Scaling Distributed Machine Learning With In-Network Aggregation

[P4 Workshop'19]

- ML success is (also) thanks to hardware

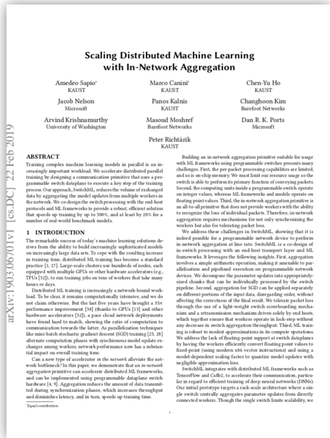
  clusters with hundreds of machines,

  each with many hardware accelerators (GPUs)



- Distributed training speeds up

  the network becomes the bottleneck

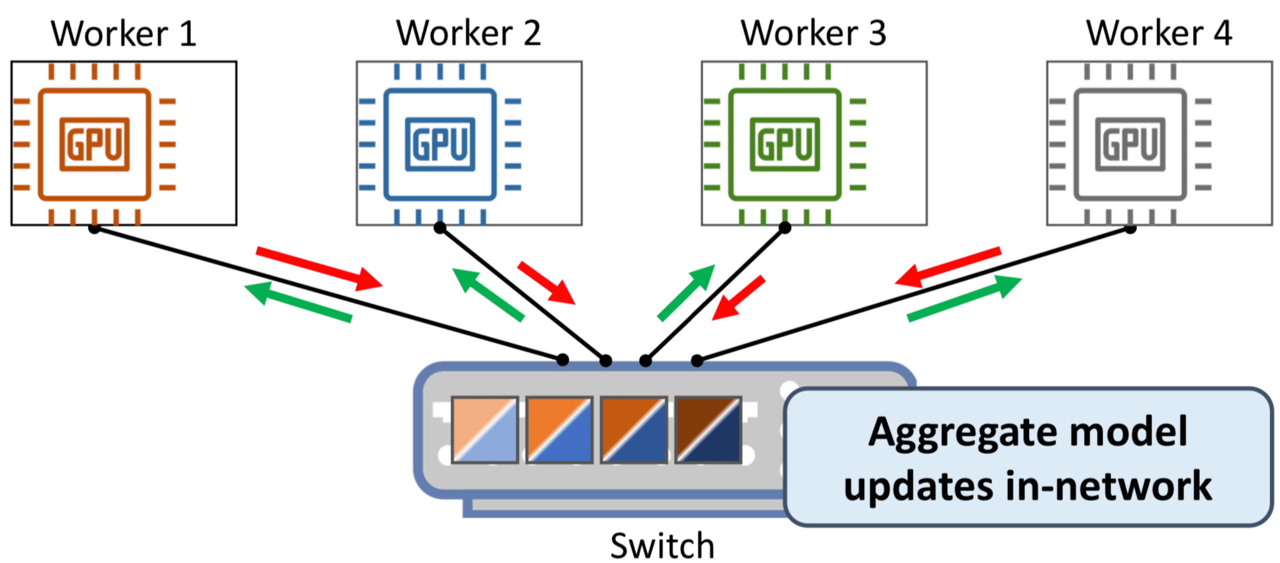

Scaling Distributed Machine Learning with In-Network Aggregation

In-Network Machine Learning

Scaling Distributed Machine Learning With In-Network Aggregation

[P4 Workshop'19]

- Can the network speed up as well?



- Leveraging data-plane programmability

This is just a subset of research in the area…

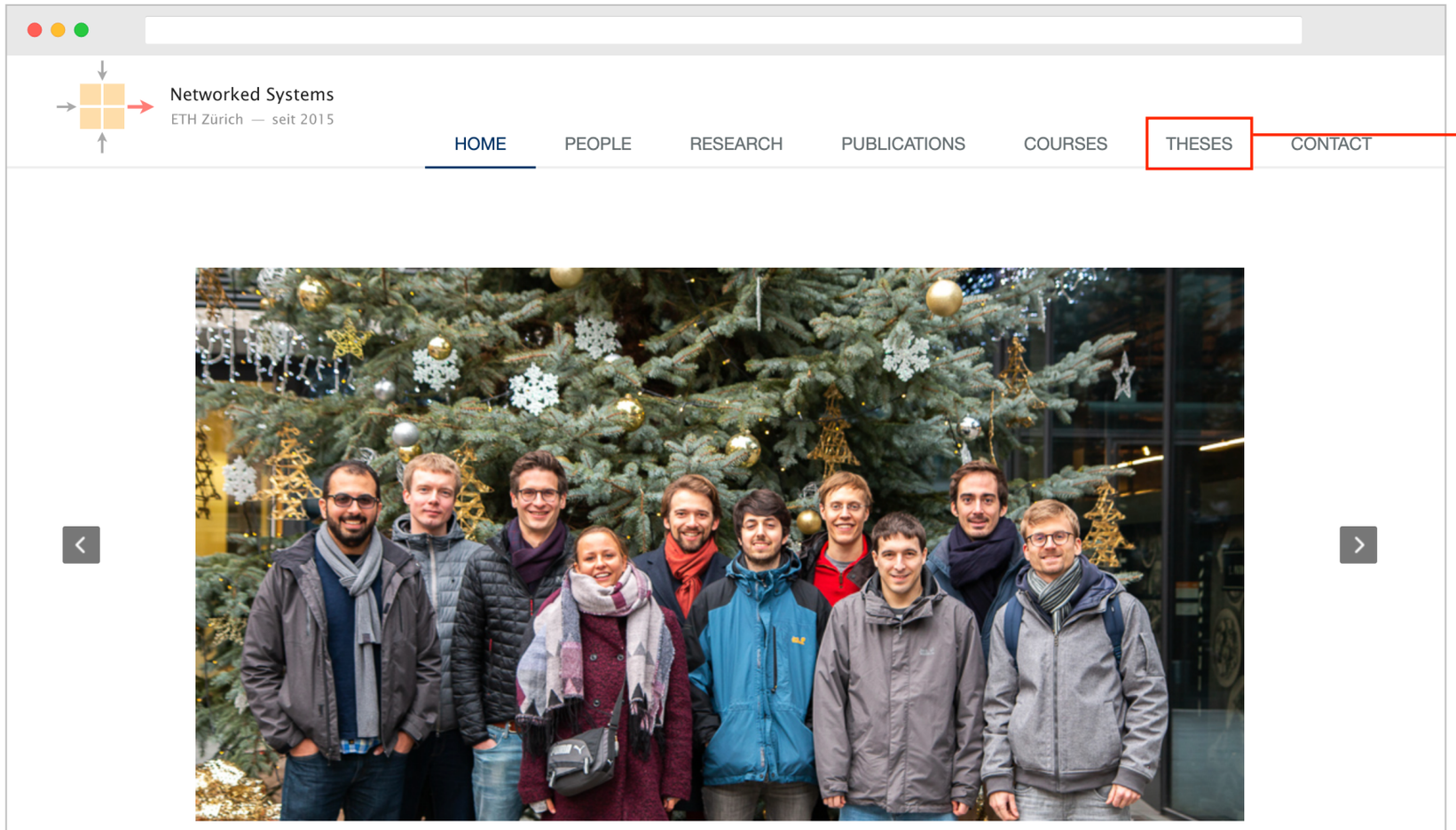
If you…

- find that something is missing
- want to discuss a particular topic
- want to share a paper you like
- want to learn a specific ML technique

Let us know! ☺

Our group is doing research on all these topics

Check out nsg.ee.ethz.ch for more info!



The Networked Systems Group (NSG) is a research group in the Department of Information Technology and Electrical Engineering (D-ITET) at ETH Zürich led by Prof. Laurent Vanbever. We are also part of the ETH ICE center.

Our research interests are centered around complex network management problems, with the larger goal of making current and future networks (especially the Internet) easier to design, understand and operate. We are currently active in multiple areas including network programmability, data-driven networking, verification, routing, and security. Most of our projects are inherently multidisciplinary and tend to involve recent advances in programming languages, algorithmics, and machine learning.

A few recent examples of practical systems we have built include: Blink, Config2Spec, Bayonet, Fibbing, iTAP, Net2Text, NetComplete, NetHide, SDX, SyNET, SDNRacer, SP-PIFO, Stroboscope, and SWIFT. We are also currently looking at the impact of routing attacks on systems overlays such as cryptocurrencies and anonymity networks. To learn about our work, please check out our research and publications pages.

Your TODOs for next week

Read "Machine Learning for Networking:
Workflow, Advances and Opportunities"

Select the paper you want to present
using the form on https://seminar-net.ethz.ch

Register for the lecture
(if not done already)

Seminar in Communication Networks

Learning, Reasoning and Control

Prof. Laurent Vanbever

nsg.ee.ethz.ch

ETH Zürich (D-ITET)

18 September 2019