Iroko

A Data Center Emulator for Reinforcement Learning

Fabian Ruffy, Michael Przystupa, Ivan Beschastnikh

University of British Columbia

https://github.com/dcgym/iroko

THE UNIVERSITY OF BRITISH COLUMBIA

Reinforcement Learning and Networking

Reinforcement Learning and Networking

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

Li Chen, Justinas Lingys, Kai Chen, Feng Liu†
SING Lab, Hong Kong University of Science and Technology, †SAIC Motors
{lchenad,jlingys,kaichen}@cse.ust.hk,liufeng@saicmotor.com

Reinforcement Learning and Networking

AuTO: Scaling Deep Reinforcement Learning
Optimization

Resource Management with Deep Reinforcement Learning

Hongzi Mao⋆, Mohammad Alizadeh⋆, Ishai Menache†, Srikanth Kandula†
Massachusetts Institute of Technology⋆, Microsoft Research†
{hongzi, alizadeh}@mit.edu, {ishai, srikanth}@microsoft.com

Reinforcement Learning and Networking

AuTO: Scaling Deep Reinforcement Learning
Optimization

ı†
IC Motors
om

Resource Management with Deep Reinforcement Learning

Hongzi Mao⋆, Mohammad Alizadeh⋆, Ishai Menache†, Srikanth Kandula†
Massachusetts Institute of Technology⋆, Microsoft Research†
{hongzi, alizadeh}@mit.edu, {ishai. srikanth}@microsoft.com

Learning To Route

Asaf Valadarsky
Hebrew University of Jerusalem
asaf.valadarsky@mail.huji.ac.il

Michael Schapira
Hebrew University of Jerusalem
schapiram@huji.ac.il

Dafna Shahaf
Hebrew University of Jerusalem
dafna.shahaf@mail.huji.ac.il

Aviv Tamar
UC Berkeley
avivt@berkeley.edu

Reinforcement Learning and Networking

AuTO: Scaling Deep Reinforcement Learning
Optimization

1†
IC Motors
om

Resource Management with Deep Reinforcement Learning

Hongzi Mao⋆, Mohammad Alizadeh⋆, Ishai Menache†, Srikanth Kandula†
Massachusetts Institute of Technology⋆, Microsoft Research†
{hongzi, alizadeh}@mit.edu, {ishai. srikanth}@microsoft.com

Cellular Network Traffic Scheduling with Deep Reinforcement Learning

Sandeep Chinchali [1], Pan Hu [2], Tianshu Chu [3], Manu Sharma [3], Manu Bansal [3], Rakesh Misra [3]
Marco Pavone [4] and Sachin Katti [1,2]
[1] Department of Computer Science, Stanford University
[2] Department of Electrical Engineering, Stanford University
[3] Uhana, Inc.
[4] Department of Aeronautics and Astronautics, Stanford University
{csandeep, panhu, pavone, skatti}@stanford.edu, {tchu, manusharma, manub, rakesh}@uhana.io

Learning To Route

Michael Schapira
Hebrew University of Jerusalem
schapiram@huji.ac.il

Aviv Tamar
UC Berkeley
avivt@berkeley.edu

Reinforcement Learning and Networking

AuTO: Scaling Deep Reinforcement Learning
Optimization

Resource Management with Deep Reinforcement Learning

Hongzi Mao[⋆], Mohammad Alizadeh[⋆], Ishai Menache[†], Srikanth Kandula[†]
Massachusetts Institute of Technology[⋆], Microsoft Research[†]
{hongzi, alizadeh}@mit.edu, {ishai. srikanth}@microsoft.com

IC Motors

Learning To Route

Michael Schapira
Hebrew University of Jerusalem
schapiram@huji.ac.il

viv Tamar
JC Berkeley
@berkeley.edu

Cellular Network Traffic Scheduling with Deep Reinforcement Learning

Sandeep Chinchali [1], Pan Hu [2], Tianshu Chu [3], Manu Sharma [3], Manu Bansal [3], Rakesh Misra [3]
Marco Pavone [4] and Sachin Katti [1,2]
[1] Department of Computer Science, Stanford University
[2] Department of Electrical Engineering, Stanford University
[4] Department
{csandeep, panhu, pavone, sk

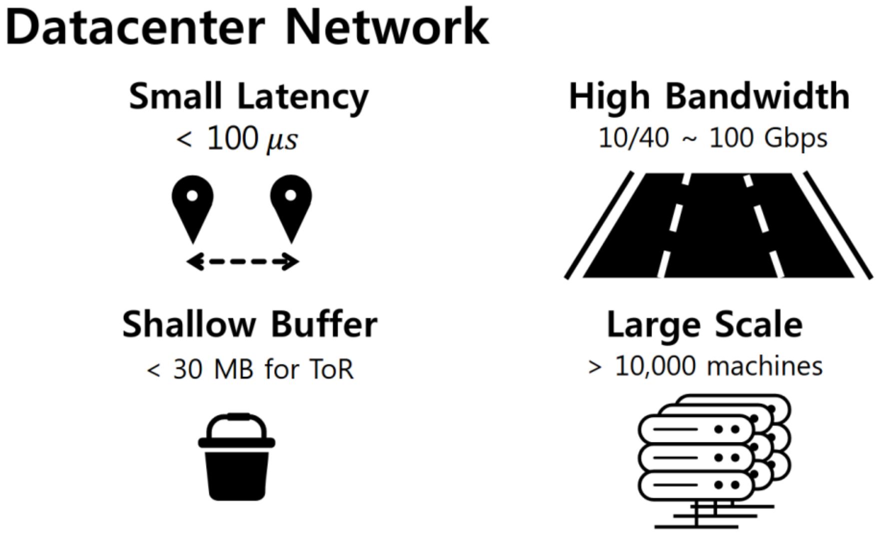
Knowledge-Defined Networking

Albert Mestres*, Alberto Rodriguez-Natal*, Josep Carner*, Pere Barlet-Ros*, Eduard Alarcón*,
Marc Solé[†], Victor Muntés-Mulero[†], David Meyer[‡], Sharon Barkai[§], Mike J Hibbett[¶], Giovani Estrada[¶],
Khaldun Ma'ruf[‖], Florin Coras**, Vina Ermagan**, Hugo Latapie**, Chris Cassar**, John Evans**, Fabio Maino**,
Jean Walrand[††] and Albert Cabellos*
* Universitat Politècnica de Catalunya [†] CA Technologies [‡] Brocade Communication [§] Hewlett Packard Enterprise
[¶] Intel R&D [‖] NTT Communications ** Cisco Systems [††] University of California, Berkeley

The Data Center: A perfect use case

- DC challenges are **optimization** problems
  - Traffic control
  - Resource management
  - Routing

- Operators have **complete** control
- Automation possible
- Lots of data can be collected



Cho, Inho, Keon Jang, and Dongsu Han.
"Credit-scheduled delay-bounded congestion control for datacenters.“
SIGCOMM 2017

Two problems…

- Typical reinforcement learning is not **viable** for data center operators!
  - Fragile stability
  - Questionable reproducibility
  - Unknown generalizability

- Prototyping RL is **complicated**
  - Cannot interfere with **live** production traffic
  - Offline traces are **limited** in expressivity
  - Deployment is tedious and **slow**

Our work: A platform for RL in Data Centers

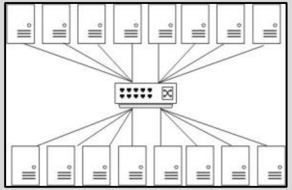
- **Iroko**: **open** reinforcement learning gym for data center scenarios
  - Inspired by the Pantheon* for WAN congestion control
- Deployable on a **local** Linux machine
  - Can scale to topologies with many hosts
- Approximates **real** data center conditions
- Allows **arbitrary** definition of
  - Reward
  - State
  - Actions

*Yan, Francis Y., et al. "Pantheon: the training ground for Internet congestion-control research." ATC 2018

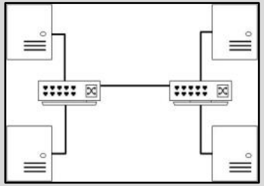
Iroko in one slide

Iroko in one slide

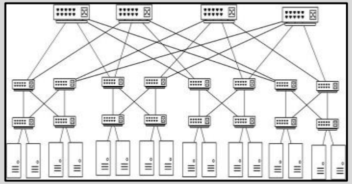
Topology



Rack



Dumbbell

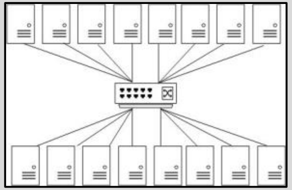


Fat-Tree

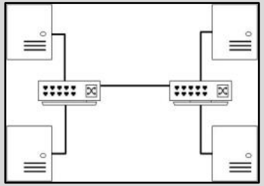
Iroko in one slide

Traffic Pattern	Action Model

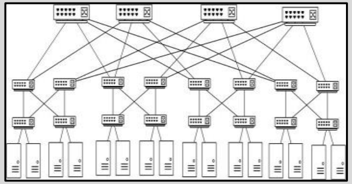
Topology



Rack

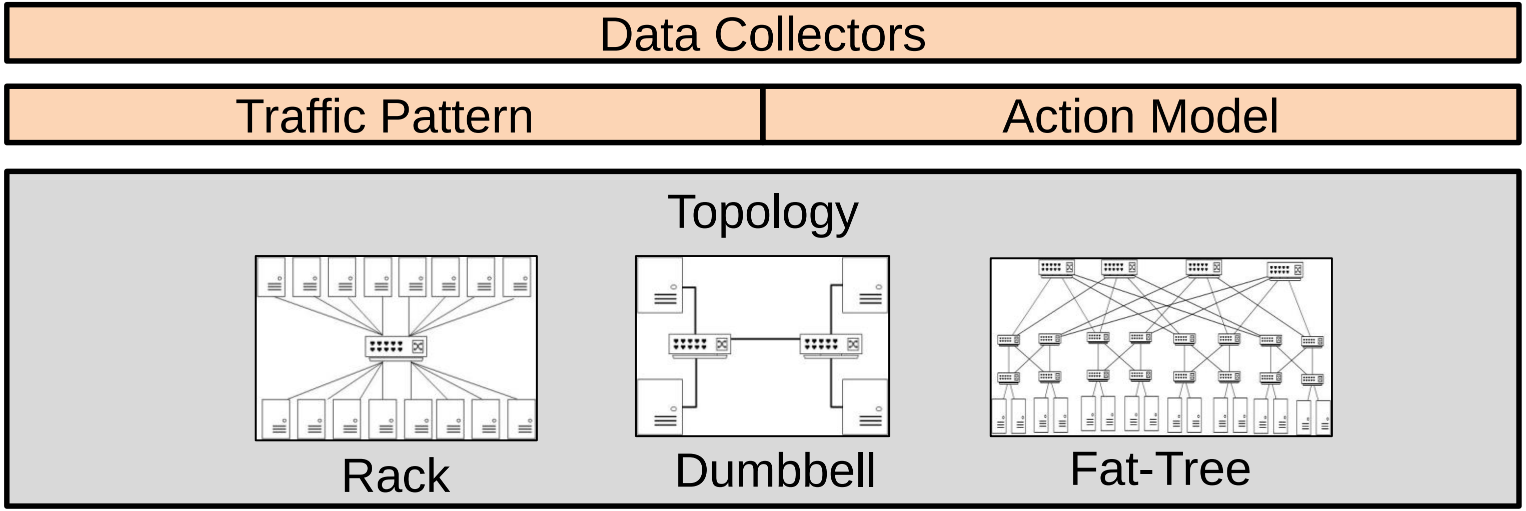


Dumbbell



Fat-Tree

Iroko in one slide



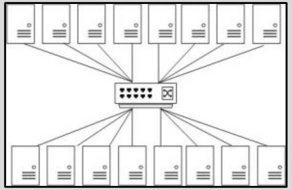
Iroko in one slide

Reward Model | State Model

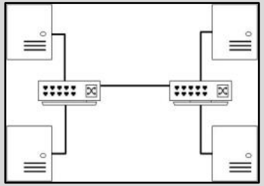
Data Collectors

Traffic Pattern | Action Model

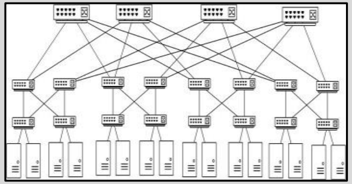
Topology



Rack

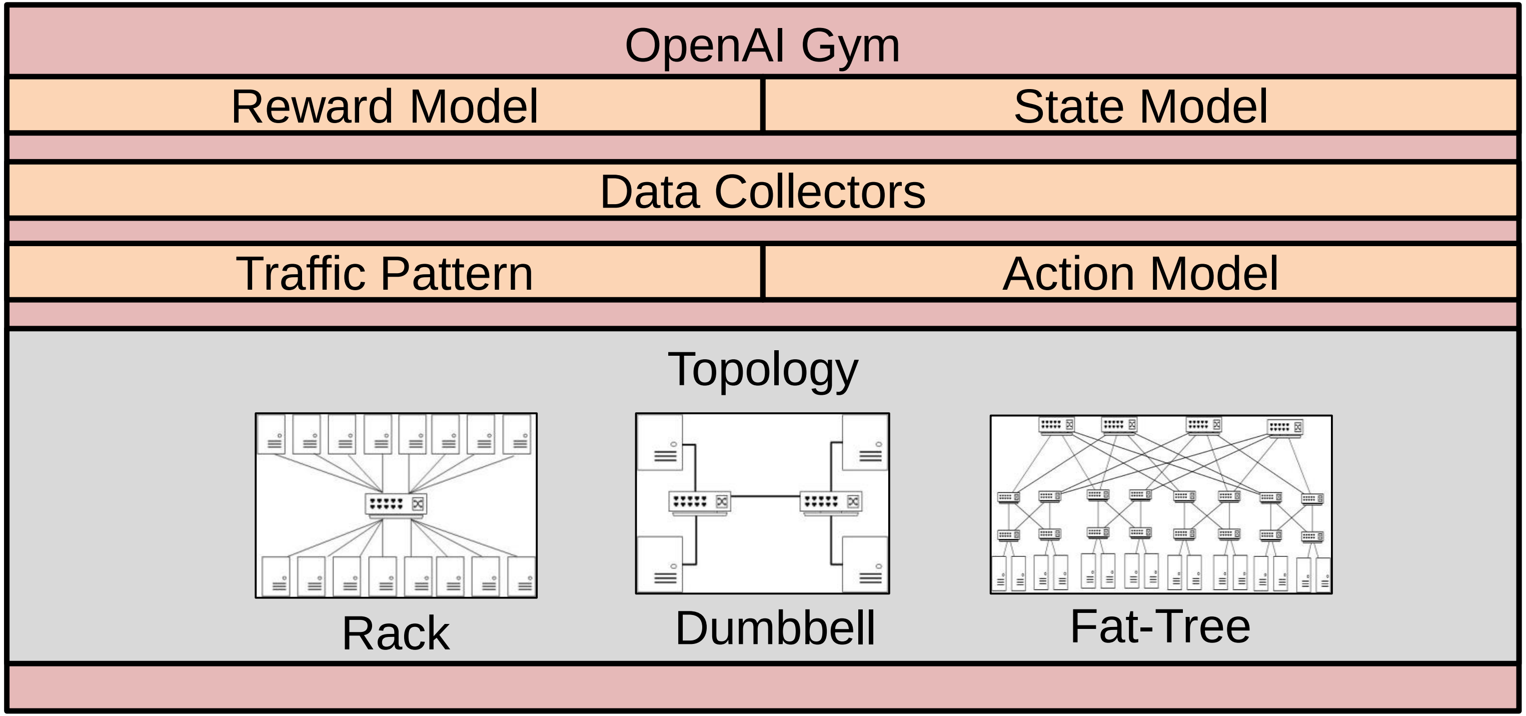


Dumbbell

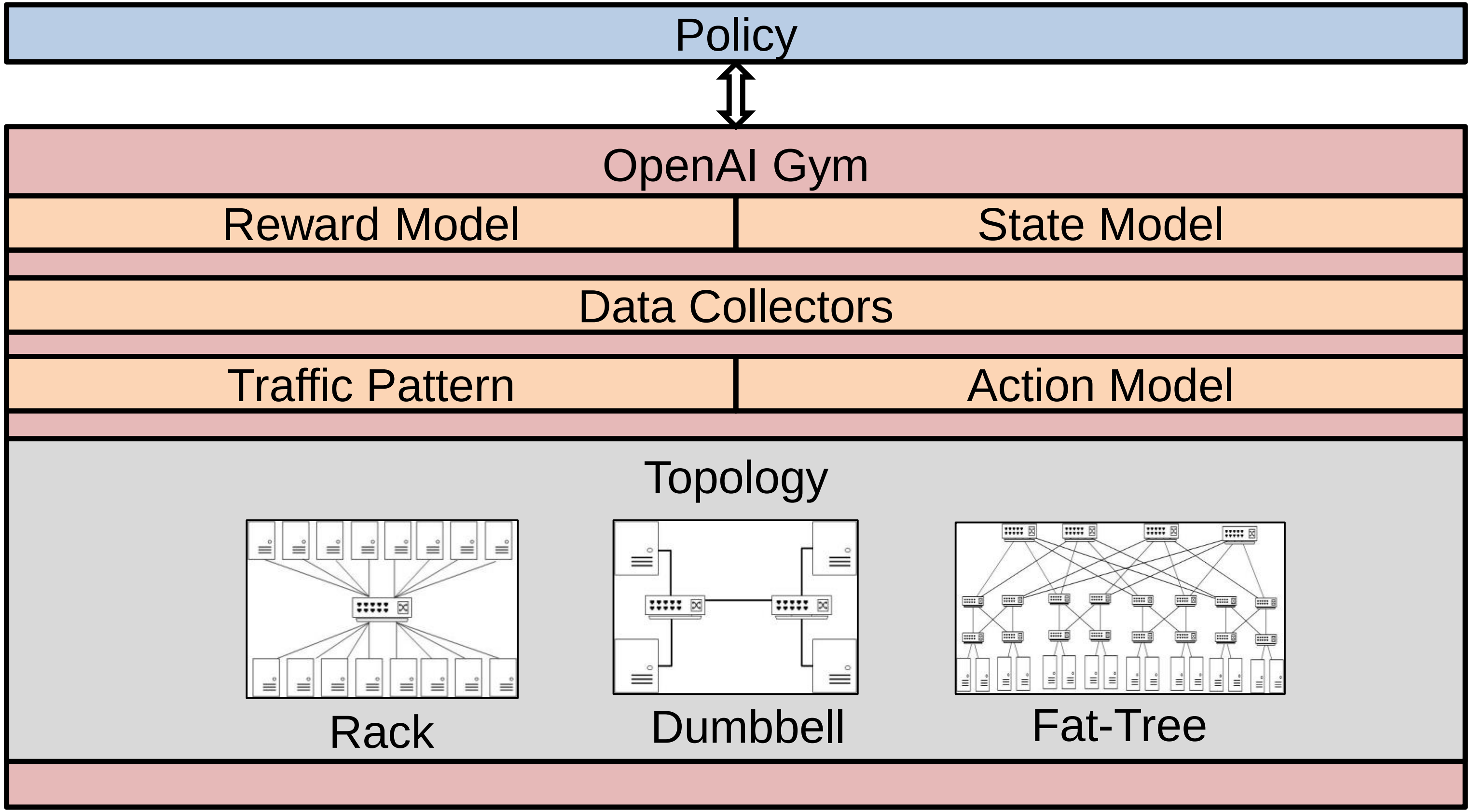


Fat-Tree

Iroko in one slide



Iroko in one slide

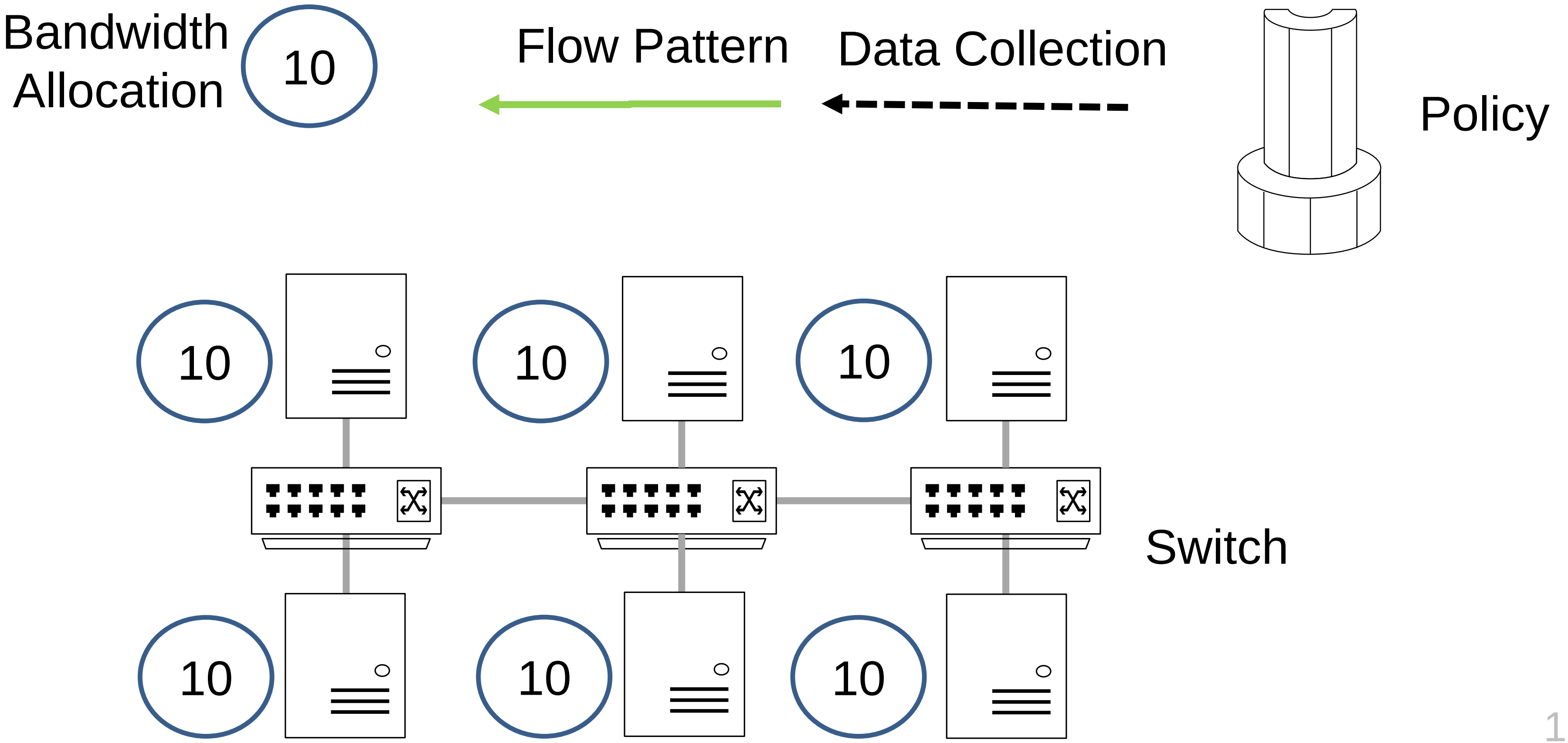


Use Case: Congestion Control

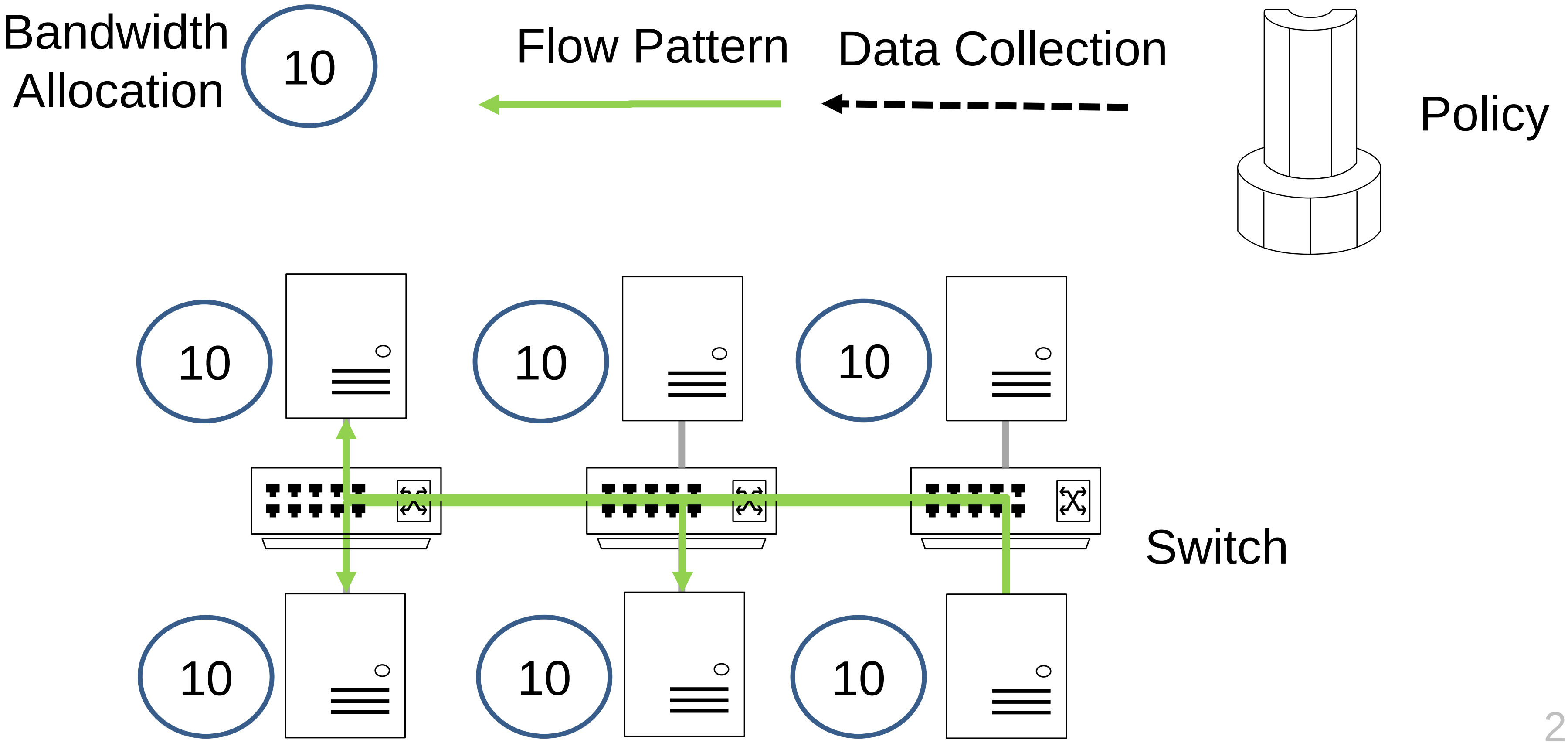
- Ideal data center should have:
  - Low latency, high utilization
  - No packet loss or queuing delay
  - Fairness

- CC variations draw from the reactive TCP
  - **Queueing latency** dominates
  - Frequent retransmits reduce **goodput**
  - Data center performance may be **unstable**

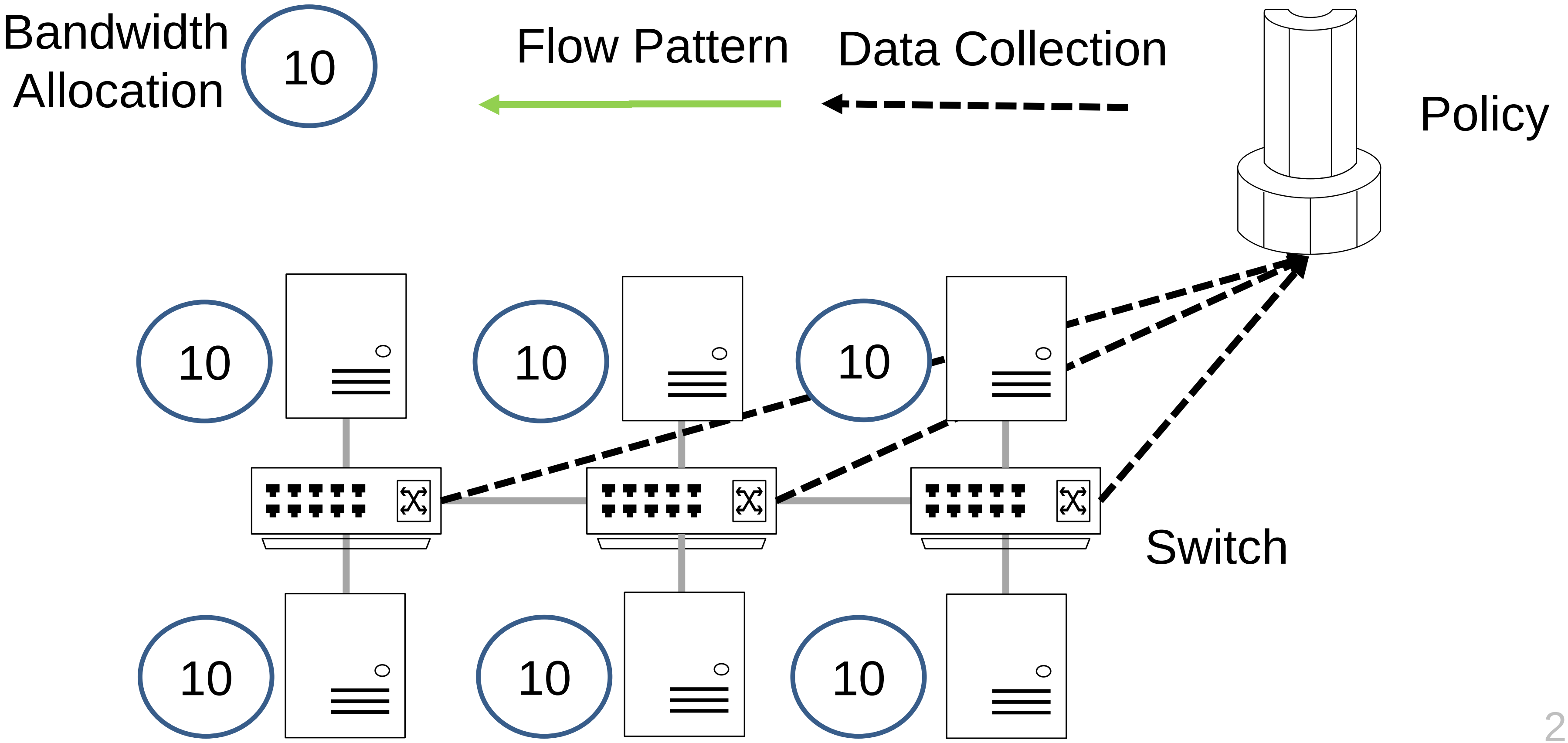
Predicting Networking Traffic



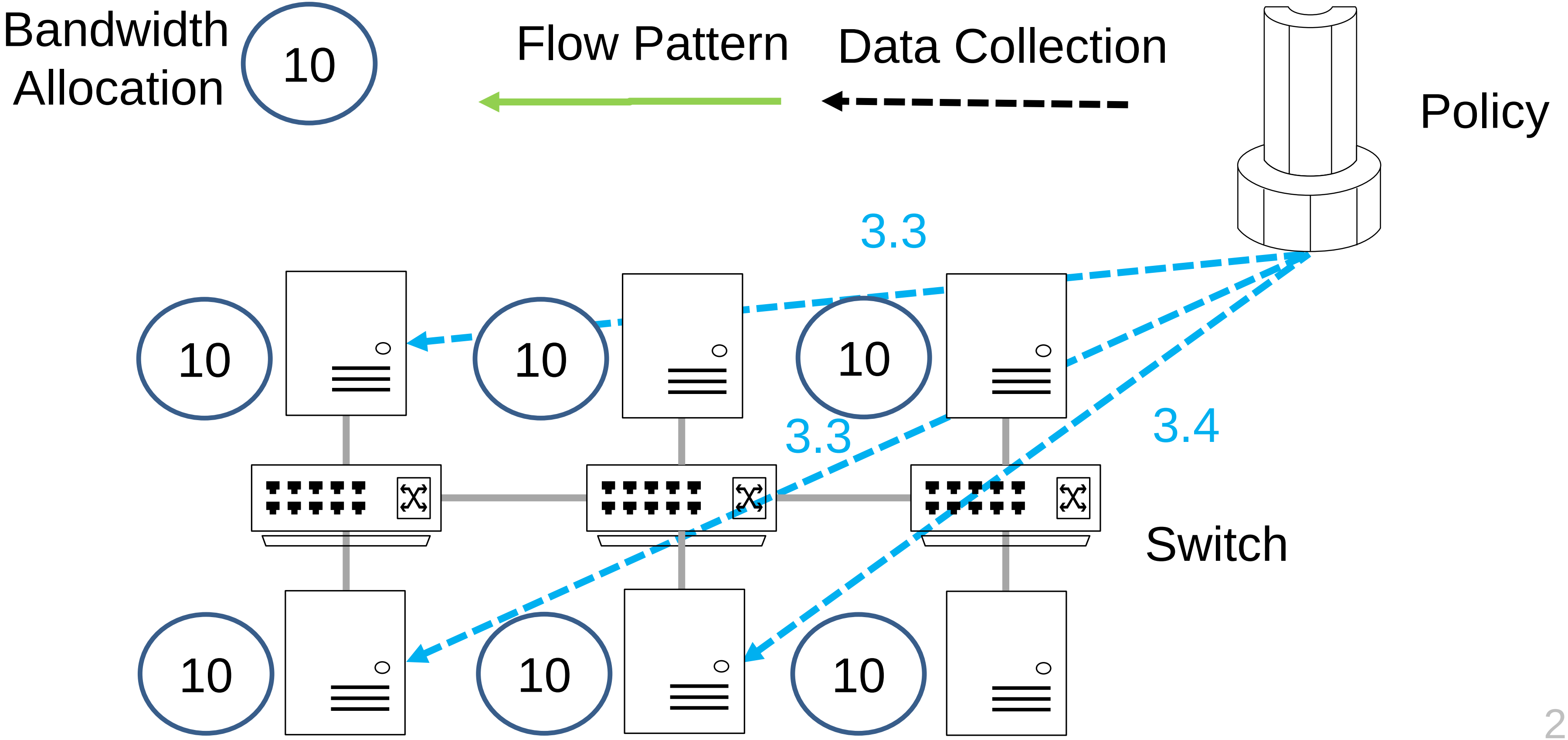
Predicting Networking Traffic



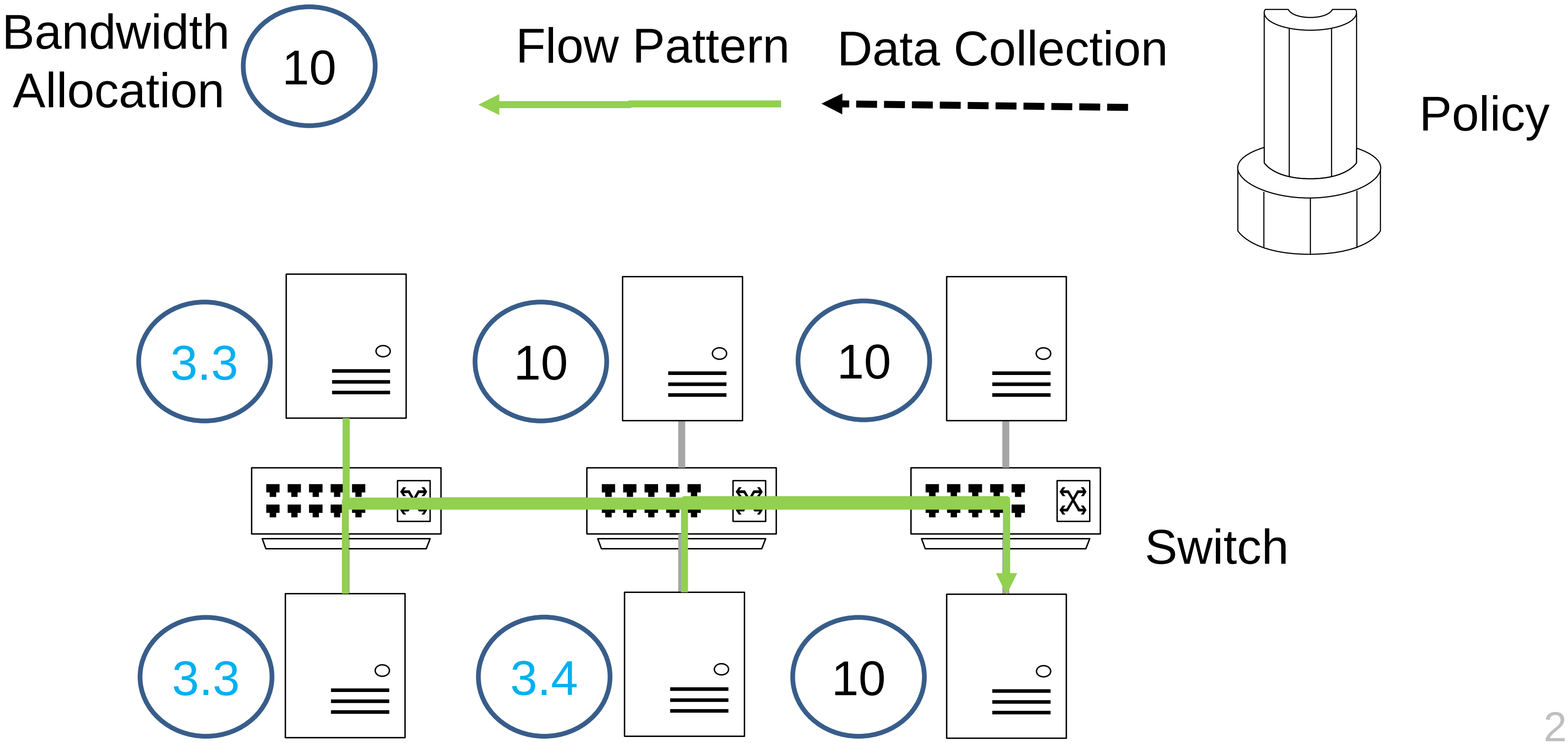
Predicting Networking Traffic



Predicting Networking Traffic



Predicting Networking Traffic



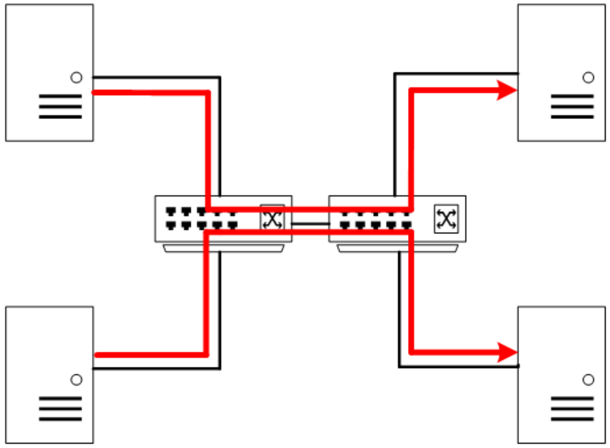
Can we learn to allocate traffic fairly?

- Two environments:
  - **env_iroko**: centralized rate limiting arbiter
    - Agent can set the sending rate of hosts
    - PPO, DDPG, REINFORCE
  - **env_tcp**: raw TCP
    - Contains implementations of TCP algorithms
    - TCP Cubic, TCP New Vegas, DCTCP
- Goal: Avoid congestion

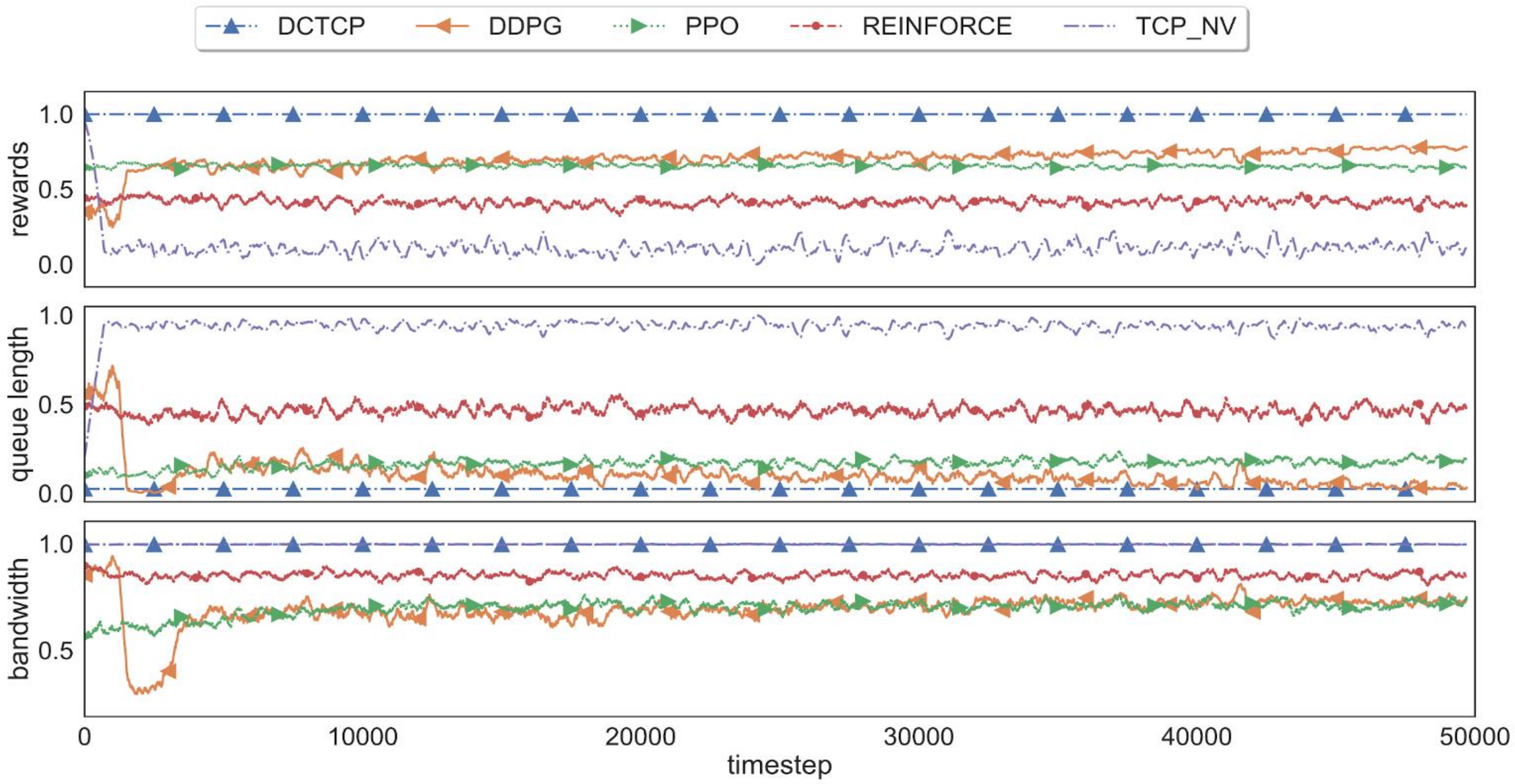
$$R \leftarrow \sum_{i \in hosts} \underbrace{bw_i/bw_{max}}_{\text{bandwidth reward}} - \underbrace{\text{ifaces}}_{weight} \cdot \underbrace{(q_i/q_{max})^2}_{\text{queue penalty}} - \underbrace{\text{std}}_{devpenalty}$$

Experiment Setup

- 50000 timesteps
- Linux default UDP as base transport
- 5 runs (~7 hours per run)
- Bottleneck at central link



Results – Dumbbell UDP



Results - Takeaways

- **Challenging** real-time environment
  - Noisy observation
  - Exhibits strong credit assignment problem

- RL algorithms show **expected** behavior for our gym
  - Achieve better performance than TCP New Vegas
  - More robust algorithms required to learn good policy
    - DDPG and PPO achieve near optimum
    - REINFORCE fails to learn good policy

Contributions

- Data center reinforcement learning is gaining traction

  …but it is difficult to prototype and evaluate
- **Iroko** is
  - a **platform** to experiment with RL for data centers
  - intended to train on live traffic
  - **early** stage work
    - but experiments are promising
  - available on Github:

    https://github.com/dcgym/iroko