Iroko
A Data Center Emulator for Reinforcement Learning
Fabian Ruffy, Michael Przystupa, Ivan Beschastnikh
University of British Columbia
https://github.com/dcgym/iroko
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 Resource Management with Deep Reinforcement Learning Optimization

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 Resource Management with Deep Reinforcement Learning Optimization

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

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

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
Resource Management with Deep Reinforcement Learning Optimization

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

Learning To Route

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

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 Resource Management with Deep Reinforcement Learning Optimization

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

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 of Electrical Engineering, Stanford University

Knowledge-Defined Networking

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

* viv Tamar
† IC Berkeley
‡@berkeley.edu

Marc Solé†, Victor Muntés-Mulero†, David Meyer†, Sharon Barkai‡, Mike J Hibbett‡, Giovanni Estrada‡,
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

---

**Datacenter Network**

- **Small Latency**
  - $< 100 \mu s$

- **High Bandwidth**
  - $10/40 \sim 100$ Gbps

- **Shallow Buffer**
  - $< 30$ MB for ToR

- **Large Scale**
  - $> 10,000$ machines

---

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

Topologies:
- Rack
- Dumbbell
- Fat-Tree
Iroko in one slide

<table>
<thead>
<tr>
<th>Traffic Pattern</th>
<th>Action Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology</td>
<td></td>
</tr>
<tr>
<td>Rack</td>
<td>Dumbbell</td>
</tr>
<tr>
<td>Fat-Tree</td>
<td></td>
</tr>
</tbody>
</table>
Iroko in one slide

<table>
<thead>
<tr>
<th>Data Collectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic Pattern</td>
</tr>
</tbody>
</table>

- Topology
- Rack
- Dumbbell
- Fat-Tree
Iroko in one slide

<table>
<thead>
<tr>
<th>Reward Model</th>
<th>State Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Collectors</td>
<td></td>
</tr>
<tr>
<td>Traffic Pattern</td>
<td>Action Model</td>
</tr>
</tbody>
</table>

- Topology
  - Rack
  - Dumbbell
  - Fat-Tree
Iroko in one slide

<table>
<thead>
<tr>
<th>OpenAI Gym</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reward Model</td>
<td></td>
<td>State Model</td>
</tr>
<tr>
<td>Data Collectors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Traffic Pattern</td>
<td></td>
<td>Action Model</td>
</tr>
</tbody>
</table>

**Topology**

- Rack
- Dumbbell
- Fat-Tree
Iroko in one slide

Policy

OpenAI Gym
Reward Model | State Model

Data Collectors

Traffic Pattern | Action Model

Topology
Rack | Dumbbell | Fat-Tree
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

Bandwidth Allocation

Flow Pattern

Data Collection

Policy

Switch

10

10

10

10

10

10

10

10
Predicting Networking Traffic

Bandwidth Allocation

Flow Pattern

Data Collection

Policy

Switch
Predicting Networking Traffic

Bandwidth Allocation

Flow Pattern

Data Collection

Switch

Policy
Predicting Networking Traffic

Bandwidth Allocation

Flow Pattern

Data Collection

Policy

Switch

10

10

10

10

10

10

10

10

10

3.3

3.3

3.4
Predicting Networking Traffic

Bandwidth Allocation

Flow Pattern

Data Collection

Policy

Switch
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

\[
R \leftarrow \sum_{i \in \text{hosts}} \left( \frac{bw_i}{bw_{max}} - \text{ifaces} \cdot \frac{q_i}{q_{max}} \right)^2 - \text{std} \]

\[
\text{bandwidth reward} \quad \text{weight} \quad \text{queue penalty} \quad \text{devpenalty}
\]
Experiment Setup

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

- DCTCP
- DDPG
- PPO
- REINFORCE
- TCP_NV

- Rewards
- Queue length
- Bandwidth

Timestep: 0 to 50000
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