Tackling Deployability Challenges in ML-Powered Networks

Noga H. Rotman

AMC SIGMETRICS'23 Workshop on Measurements for Self-Driving Networks

האוניברסיטה העברית בירושלים
THE HEBREW UNIVERSITY OF JERUSALEM

Motivation

ML for the Win?

Learning *in situ*: a randomized experiment in video streaming

Francis Y. Yan, Hudson Ayers, Chenzhi Zhu[†], Sadjad Fouladi, James Hong, Keyi Zhang, Philip Levis, Keith Winstein

Stanford University, [†]Tsinghua University

Neural Adaptive Video Streaming with Pensieve

Hongzi Mao, Ravi Netravali, Mohammad Alizadeh
MIT Computer Science and Artificial Intelligence Laboratory
{hongzi,ravinet,alizadeh}@mit.edu

A Deep Reinforcement Learning Perspective on Internet Congestion Control

Nathan Jay[*1] **Noga H. Rotman**[*2] **P. Brighten Godfrey**[1] **Michael Schapira**[2] **Aviv Tamar**[3]

- These ML-based algorithms **do very well** when their **training** environment is **faithful** to the **deployed** environment
- **Real-world adaptation**, however, has been **stunted** due to **bad generalization**

Example: Adaptive Video Streaming (ABR)

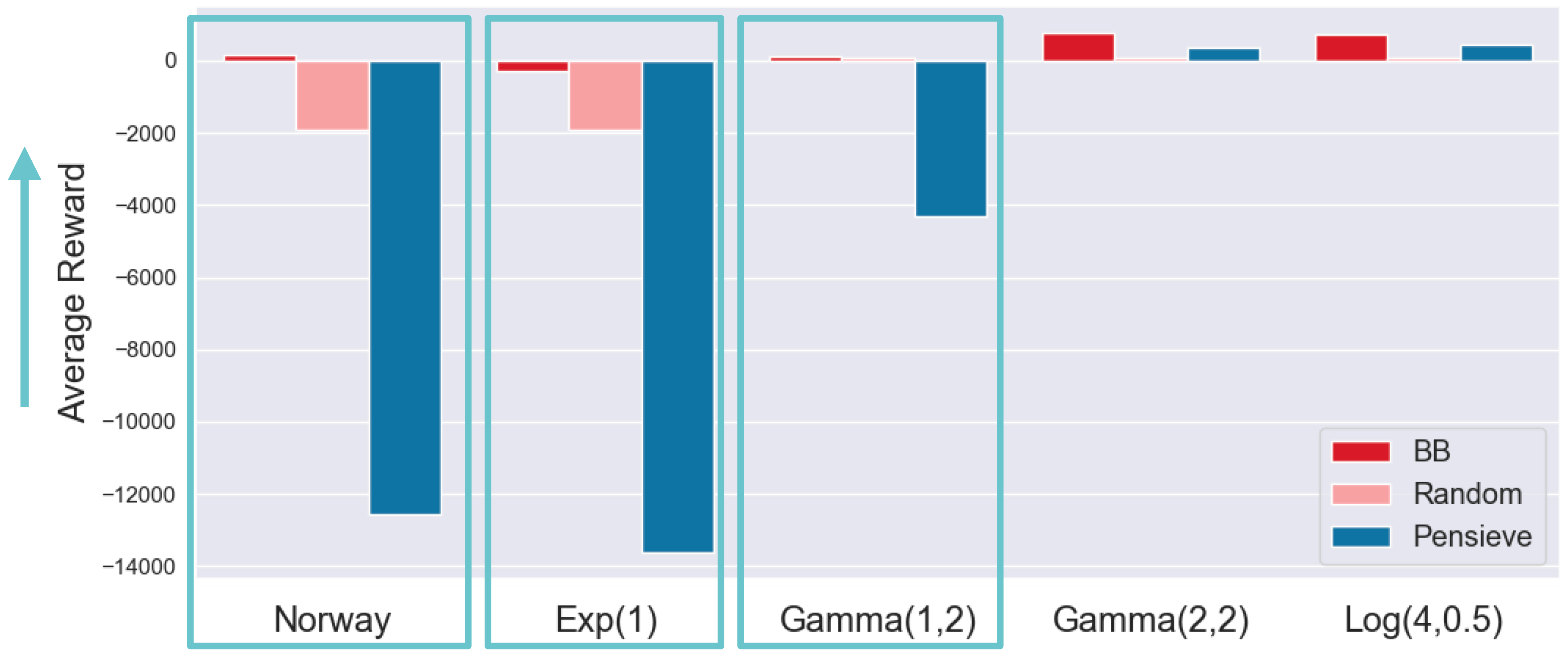
- Each video split into chunks
- Each stored in different discrete bitrates
  - e.g. 240P, 480P, 720P (HD)...

- The ABR algorithm needs to **choose** the **next bitrate** *r*
  - Undershoot - bad resolution
  - Overshoot - suffer rebuffering

Example: Adaptive Video Streaming

- BB - a heuristic solution
- Random - an algorithm which chooses the bitrate randomly
- Pensieve - an ML-based algorithm

ML agent trained on a dataset from Belgium



The ML agent does not generalize well, and may lead to performance worse than random!

* Taken from: Noga H. Rotman, Michael Schapira, and Aviv Tamar. "Online safety assurance for learning-augmented systems." In Proceedings of the 19th ACM Workshop on Hot Topics in Networks, pp. 88-95. 2020.

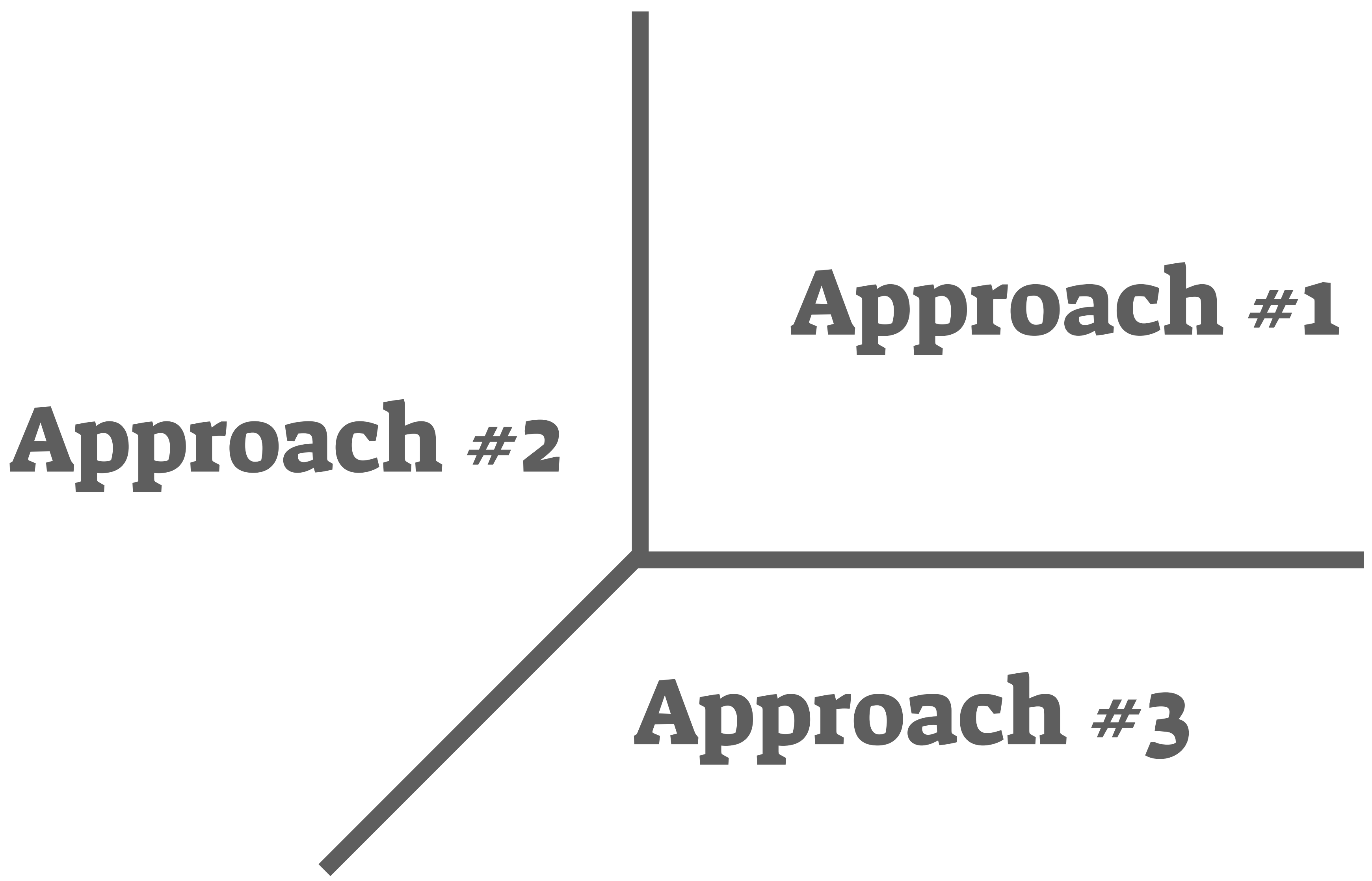
Deployment Challenges

Communication networks can be very dynamic
Machine crashes, routing changes…

Training data cannot encompass all possible scenarios
Environments are too diverse

ML agents are ~~impossible~~ very hard for humans to understand
Essentially a “black box”

Complementary Approaches



Key insight:

The following approaches are **complimentary**, or **orthogonal**, and therefore can be used **in tandem** to help advance real-world deployment

Complementary Approaches

Approach #1: Improvement via Training

Leveraging existing data, improving the training data

Learning *in situ*: a randomized experiment in video streaming

Francis Y. Yan, Hudson Ayers, Chenzhi Zhu[†], Sadjad Fouladi
James Hong, Keyi Zhang, Philip Levis, Keith Winstein

Stanford University, [†]Tsinghua University

- Continuous re-training using real-world deployment data
- Winner of the NSDI Community Award

Genet: Automatic Curriculum Generation for Learning Adaptation in Networking

Zhengxu Xia[1*], Yajie Zhou[2*], Francis Y. Yan[3], Junchen Jiang[1]
[1]University of Chicago, [2]Boston University, [3]Microsoft Research

- Focused training on the most challenging environments

Complementary Approaches

Approach #2: Pre-Deployment Analysis

Explainability of models, formally verifying the neural network

Interpreting Deep Learning-Based Networking Systems

Zili Meng[1], Minhu Wang[1], Jiasong Bai[1,2], Mingwei Xu[1,2], Hongzi Mao[3], Hongxin Hu[4]
[1]Institute for Network Science and Cyberspace & BNRist, Tsinghua University
[2]Department of Computer Science and Technology, Tsinghua University
[3]Massachusetts Institute of Technology, [4]Clemson University
zilim@ieee.org, {{wangmh19, bjs17}@mails., xumw@}tsinghua.edu.cn, hongzi@mit.edu, hongxih@clemson.edu

- Translating the neural network into a decision tree or hypergraphs
- Could use insights for retraining

Verifying Learning-Augmented Systems

Tomer Eliyahu
tomer.eliyahu2@mail.huji.ac.il
The Hebrew University of Jerusalem
Jerusalem, Israel

Yafim Kazak
Yafim.Kazak@gmail.com
The Hebrew University of Jerusalem
Jerusalem, Israel

Guy Katz
guykatz@cs.huji.ac.il
The Hebrew University of Jerusalem
Jerusalem, Israel

Michael Schapira
schapiram@cs.huji.ac.il
The Hebrew University of Jerusalem
Jerusalem, Israel

- A mathematical approach for reasoning about a neural network's behavior
- Provides provably guarantee of specified requirements

Complementary Approaches

Approach #3: Online Assurances

What can we do while the agent is running?

Online Safety Assurance for Learning-Augmented Systems

Noga H. Rotman
nogar02@cs.huji.ac.il
Hebrew University of Jerusalem

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

Aviv Tamar
avivt@technion.ac.il
Technion

ABSTRACT

Recently, deep learning has been successfully applied to a variety of networking problems. A fundamental challenge is that when the operational environment for a learning-augmented system differs from its training environment, such systems often make badly informed decisions, leading to bad performance. We argue that safely deploying learning-driven systems requires being able to determine, in real-time, whether system behavior is coherent, for the purpose of defaulting to a reasonable heuristic when this is not so. We term

for such a mismatch is the standard practice of training on simulated/emulated environments [20, 27], which fail to capture the intricacies of real-world networks [61]. However, even if training occurs *in situ* on real data, as advocated in [61], the operational environment encountered after training might still greatly differ from the training environment due to variability in network conditions not adequately covered by the finite training data, or the introduction of new factors such as routing changes, network failures, the addition/removal of traffic sources, etc.

- Building into the system the means to **detect**, in real-time, when the agent encounters **scenarios** it was **not trained for**
- Allows **switching** to a reasonable, default **non-learning policy** while the agent adapts to the new environment

Conclusion

- Making self-driving networks deployable is a **complex**, **iterative process**
- **We need a lot of hands on deck!**

Thank You!