Berkeley Lab — Bringing Science Solutions to the World

U.S. Department of Energy — Office of Science

Engineering Self-driving Networks using Deep Learning

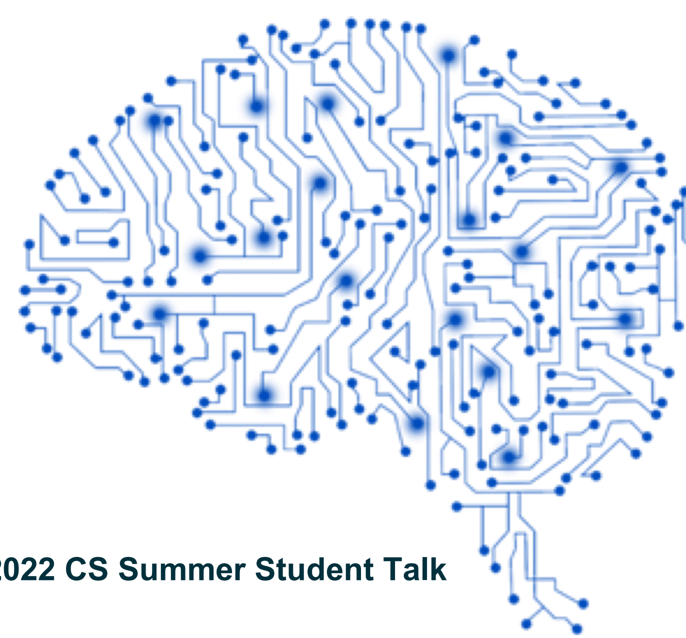
Mariam Kiran

2017 DOE Early Career

2021 N2Women Rising Star in Networking

Scientific Networking Division

ESnet, LBNL



ESnet Energy Sciences Network

SDM

July 2022 CS Summer Student Talk

Introductions

- 2011 PhD in Computer Science (University of Sheffield, UK)
  - Optimizing HPC jobs in agent-based simulations
- Postdocs in Cloud Computing (University of Oxford, Leeds)
- Royal Society Scientist at Westminster London
- 2016 Joined ESnet, LBNL
- Research Area : AI for Networking - **Self-driving networks**
  - Impacts distributed science workflows
  - Expanding AI from optic networks to wireless and quantum networks
  - Contributions to other science areas:
    - Self-driving **lasers**
    - Self-driving **batteries**
    - Self-driving **quantum transducers**

    and more..

Talk Agenda

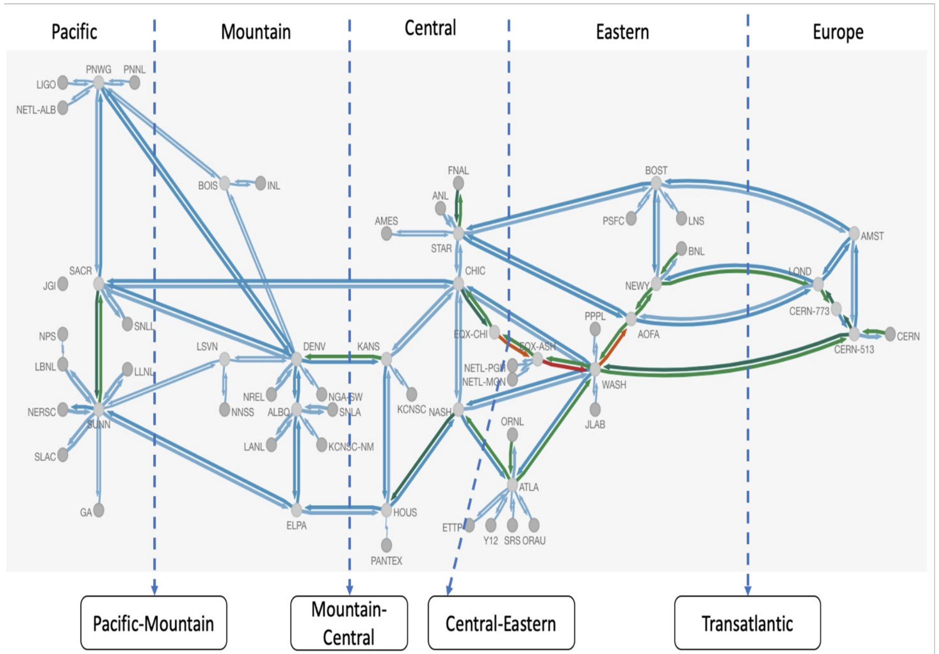
- What is ESnet (Energy Sciences Network)?
- Why Self-driving network?
- Application of LSTMs and GNN in traffic prediction
- Application of Deep Reinforcement Learning on traffic engineering
- Bringing deep learning into 5G networks
  - Classification for network slicing and multi-domain routing
- Acks and Conclusions

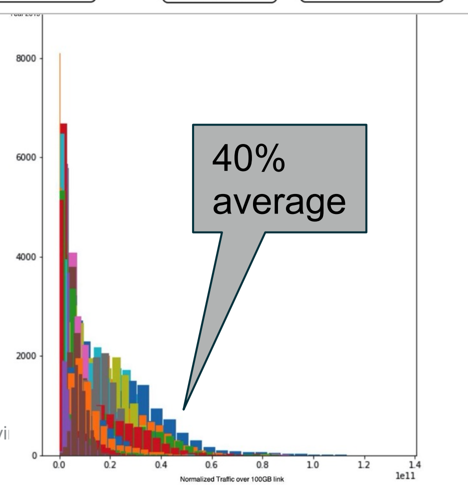
ESnet (www.es.net)

DOE’s Internet

- >100 engineers and scientists working together to improve data movement across DOE
- Write their own software to control networks e.g. OSCARS, NSI, SENSE to seamlessly ‘stitch’ multiple networks together
- Network monitoring tools e.g. Bro, perfSONAR, etc
- Innovative networking research e.g. SDN, intent-based networking, automation, etc
- Testbeds for experiments

ESnet: R&E Network the nervous system for Large-scale Science





ESnet Network and its utilization in 2019 over 100GB links

- Networks are built for resilience
- Science traffic is highly variable
- Resources are often underutilized and expensive
- Quality of Network Performance is crucial for Science

Need for **predictability** and infrastructure **adaptability**

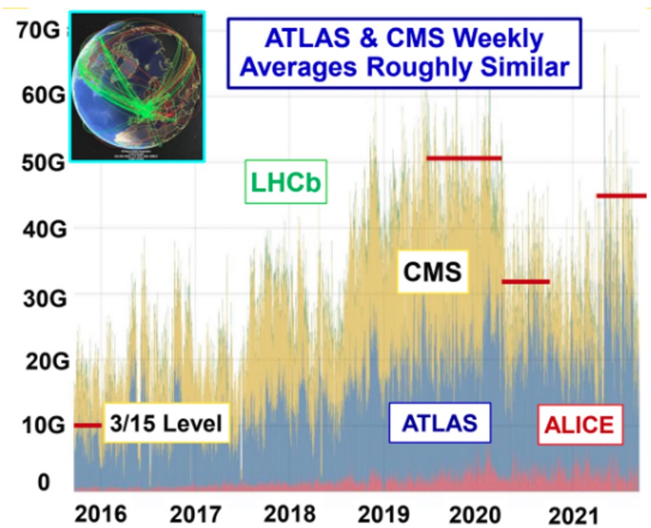
Challenge: how can we optimize and automate network resources (i.e. links) to adapt to variable workloads?

Why Self-Driving Networks?

Why Self-Driving Networks?

Networks are breaking boundaries

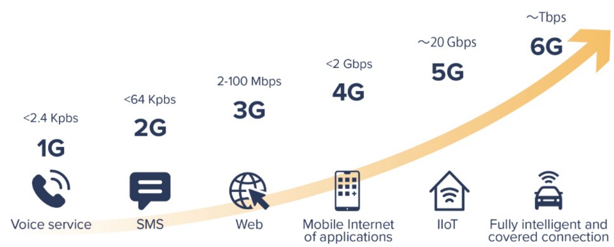
Data rate increases



Neuman, GNA-G 2021

- Upto 58 GBytes/s week avg
- 50% increase/year & 60X/decade
- Within ESnet backbone 62% inc/year

Network beyond walls

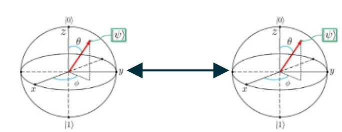


- 10x decrease in latency
- 10x increase throughput
- 100x network performance, traffic capacity
- VR/AR, ability to handle real-time emergency edge intelligence

Edge evolution in fabric

- Increased instruction/s
- Machine learning at edge
- Access to more data

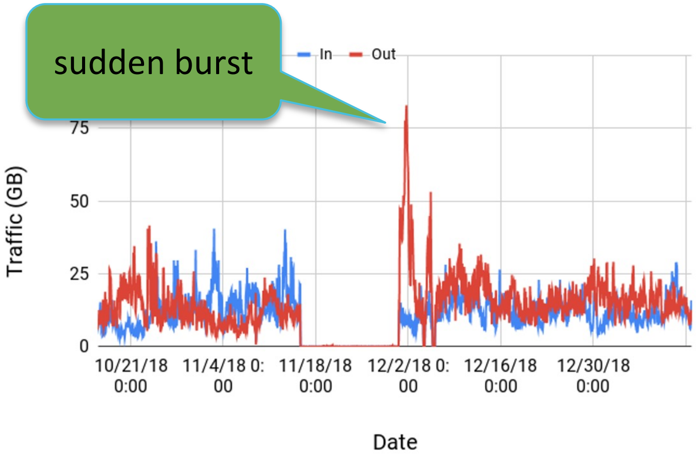
Still untapped…



Challenge 1: Resources are often underutilized

- Traffic is highly variable and ‘bursty’ (big versus small transfers)
  - Varies time of the day, month, year
- Congestion leads to packet loss
  - Risk to science data
- Networks are capped at 40% for resilient
  - Under-utilized resources

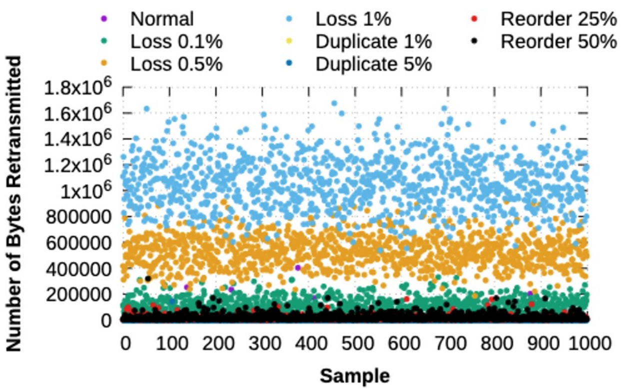
Congested Traffic Animation

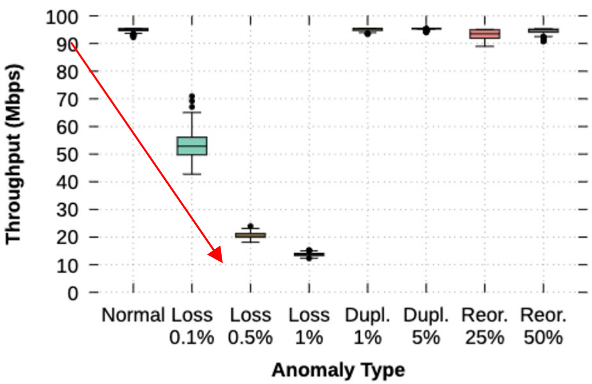


Challenge 2: TCP is Fragile to Congestion

TCP = Transmission Control Protocol

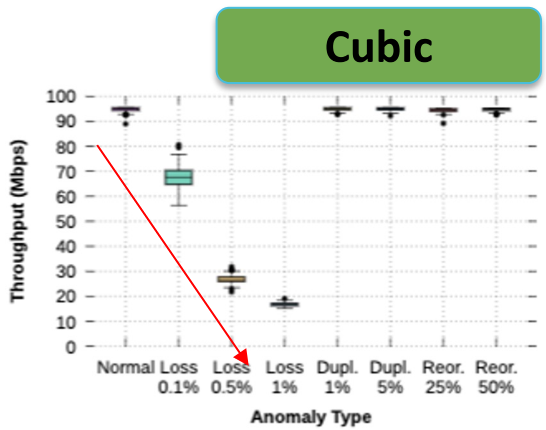
- 95% of Science Traffic uses TCP
- Transfer protocols (TCP) are sensitive to congestion
- Leads to congestion and packet loss

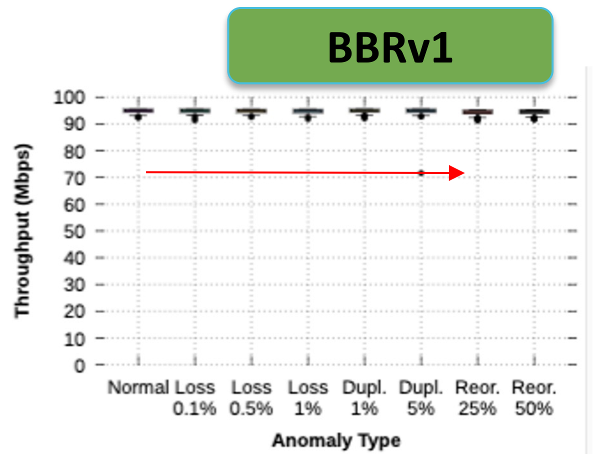




Challenge 3: TCP Congestion Control for Performance

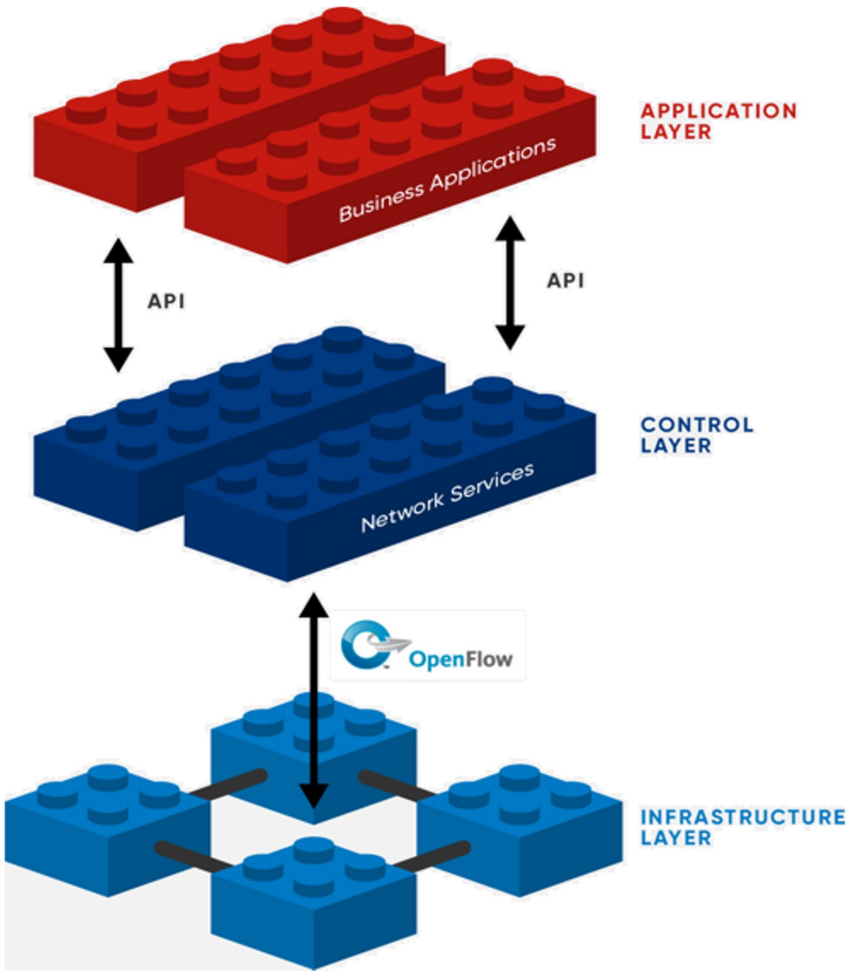
- Each TCP uses complex algorithms to optimize performance (slow down or quicken data sending)
- Examples include video buffering on Youtube, VOIP (zoom) or big file transfers in few hours (Astronomy/physics from LHC)





Software/Hardware Fusion Challenge

- Deep Learning trend
  - Image recognition e.g. Cats 2011
  - Self-playing game e.g. AlphaGo 2016
- Hardware acceleration
  - GPU advances
  - FPGAs
- Industry and Academic Efforts
  - Smart NICs e.g. Barefoot
  - AI @ Control Plane e.g. juniper, cisco
  - AI enabled TCP
  - Traffic patterns

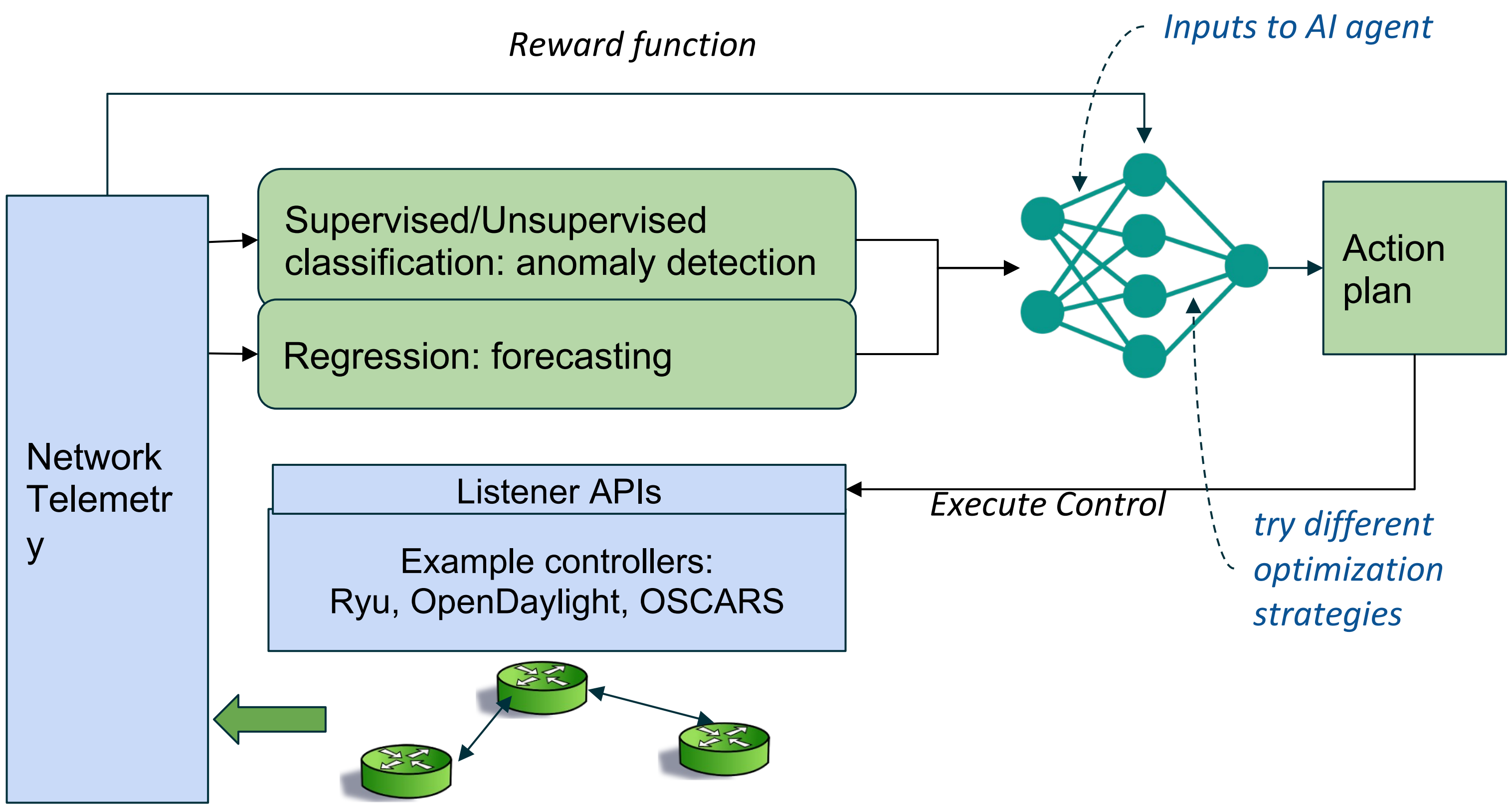


Our Vision: Self-Driving Network

Networks Today	Self-Driving Network
Static Routing Algorithms, manual setups (e.g., shortest route/least hops)	Adaptive Routing (e.g., real-time data for routing decisions)
Leads to congestion and packet loss	Learns to avoid congestion
Transfer protocols (TCP) are sensitive to congestion	congestion free -> loss free network
Inefficient utilization	100% utilization
Reactive fault tolerance	Proactive fault repair

"Networks should learn to drive themselves"*

Can do simple actions such as improving availability, attack resilience and dealing with scale. Our argument is AI is needed for mission critical actions.

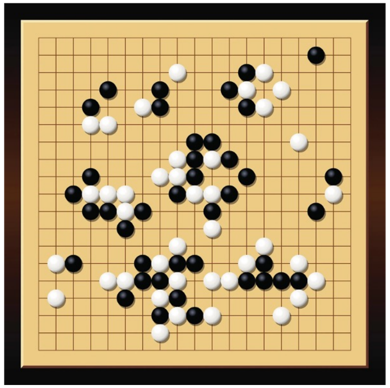


[*] Why (and How) Networks Should Run Themselves, Feamster, Rexford

Using Deep Learning for Self-Driving

Trained to recognize cats, "object identification"

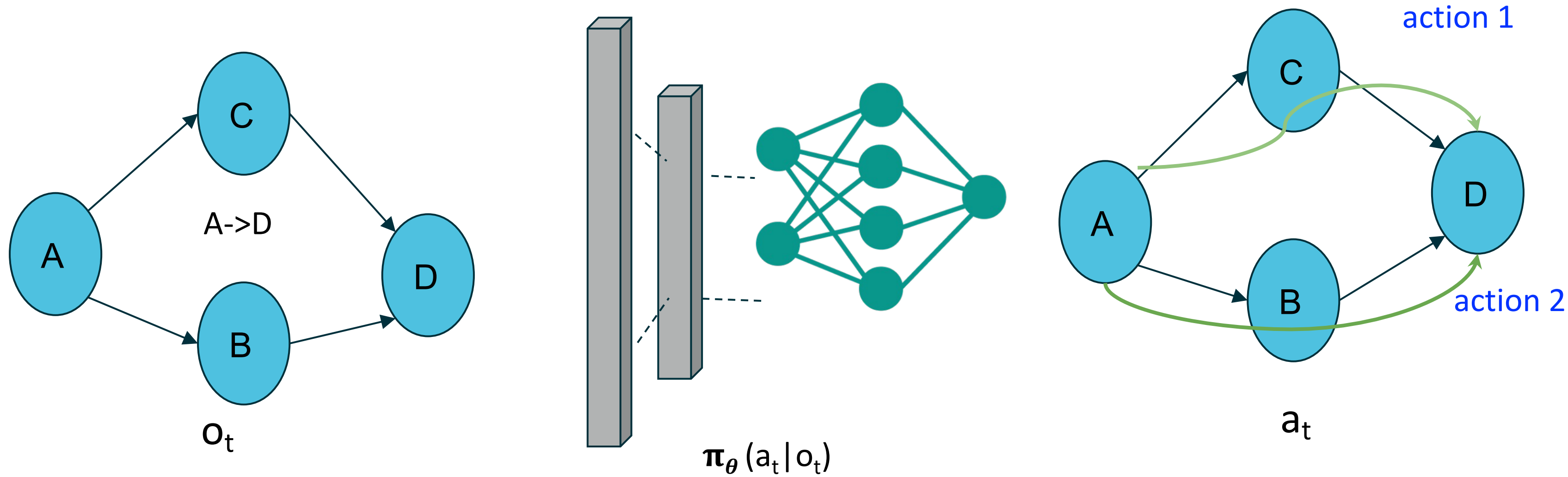
Play games, "Best strategies for winning game"



Deep learning (neural networks - NN) introduces 'data-driven learning' to build bespoke solutions

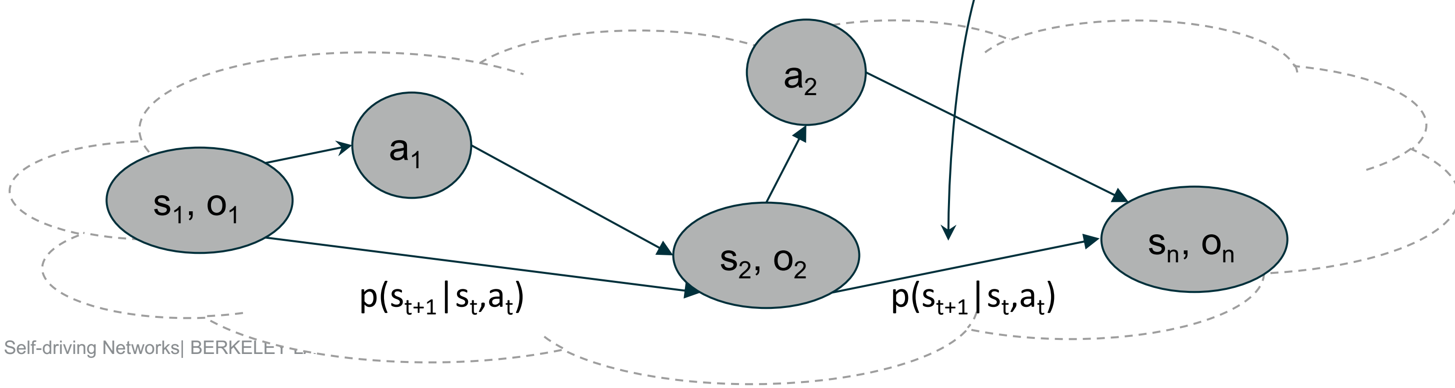
e.g. 4-node network

Data-driven learning: Network as a Markov Decision Process

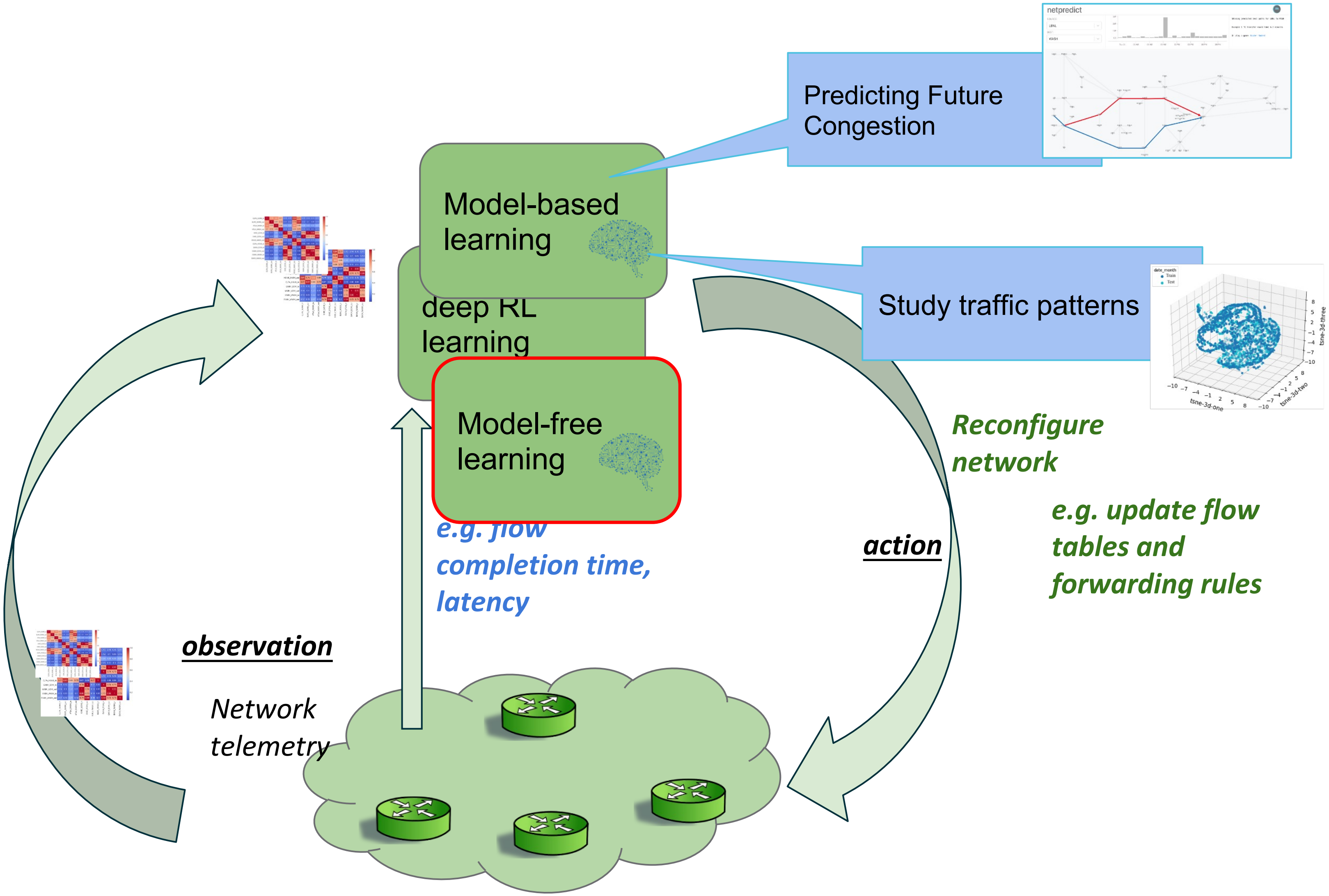


s_t - state
o_t - observation
a_t - action

Markov property helps represent system as state-action pairs



Representing Network as DL Problem

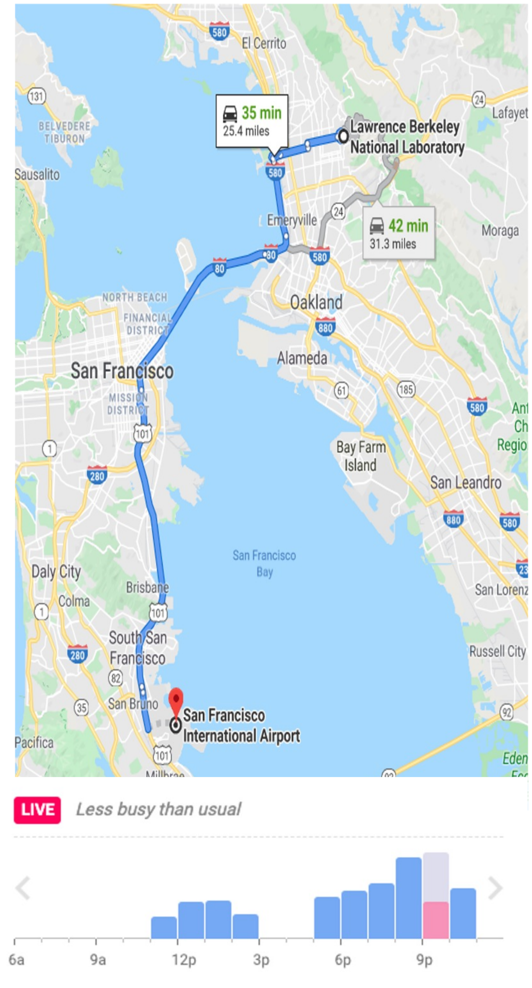


Dynamic Graph Neural Network for Traffic Forecasting in Wide Area Networks, IEEE BigData, 2020
Unsupervised Anomaly Detection in Daily WAN Traffic Patterns, SMC, 2020

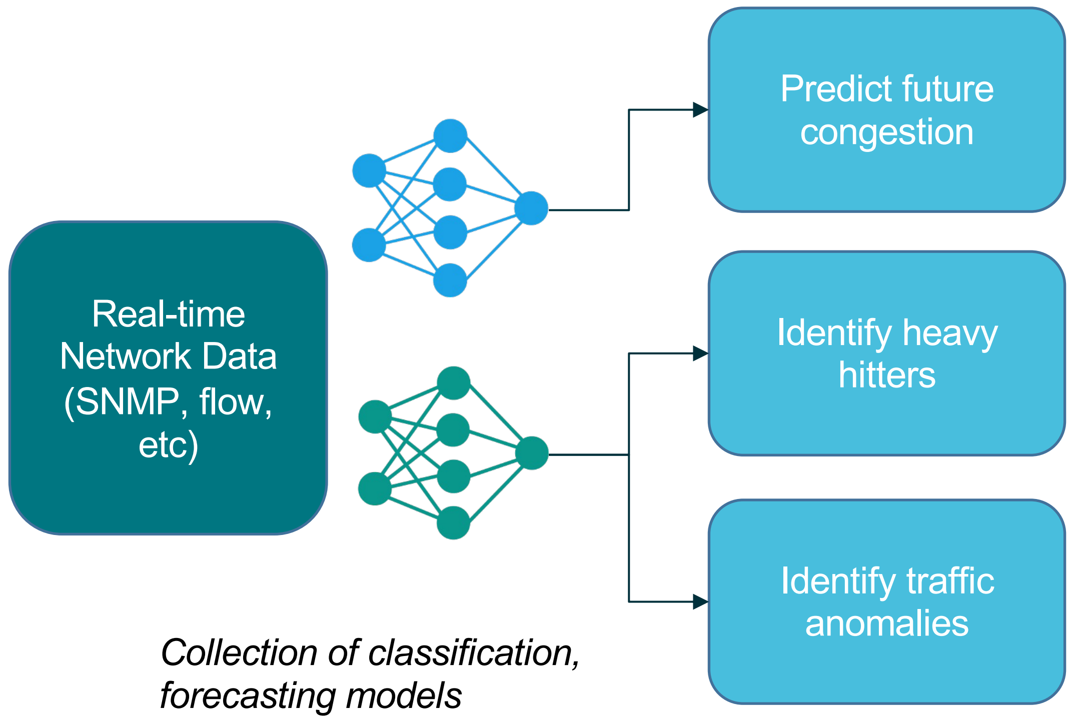
Predicting Network Congestion Before it Happens

Deep Learning Application 1

NetPredict: Predicts Congestion in next 1 week

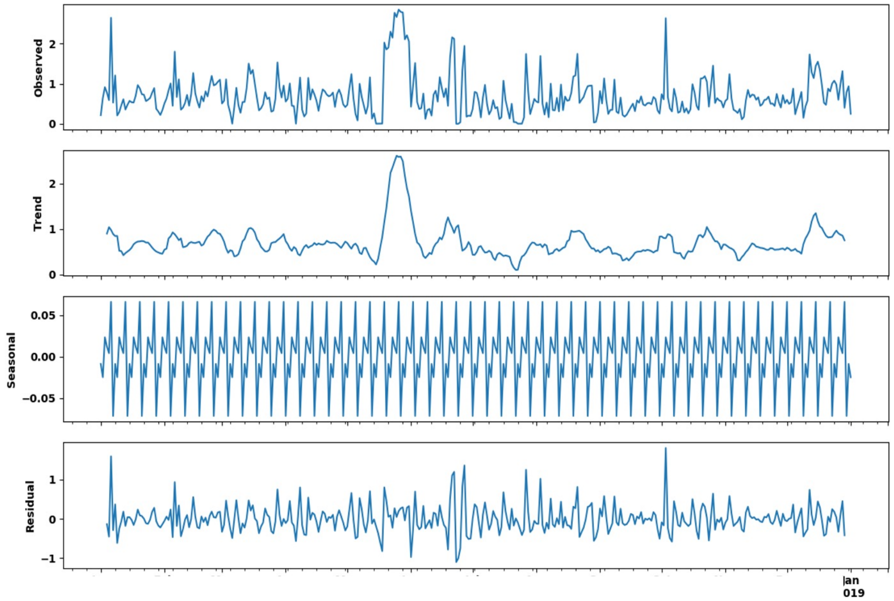


- Real-time data to inform flow directions with SDN tools
- Streaming machine learning backend to predict congestion
- Plan your data transfers to avoid congestion

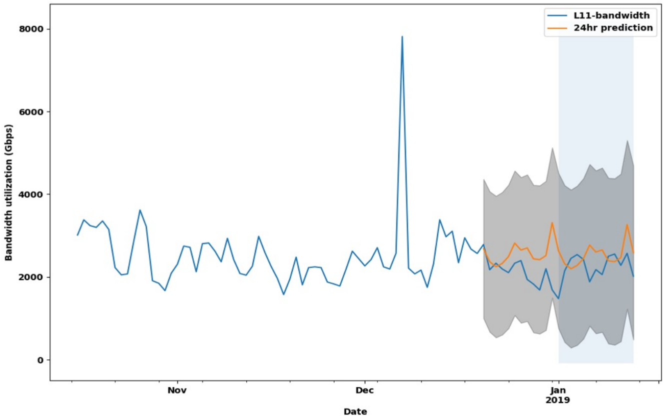


Learning Seasonality in the Traffic across Links

- ARIMA
- Holt-Winters
- SARIMA (Seasonal ARIMA)
- Fourier Transform
- LSTMs (per link)



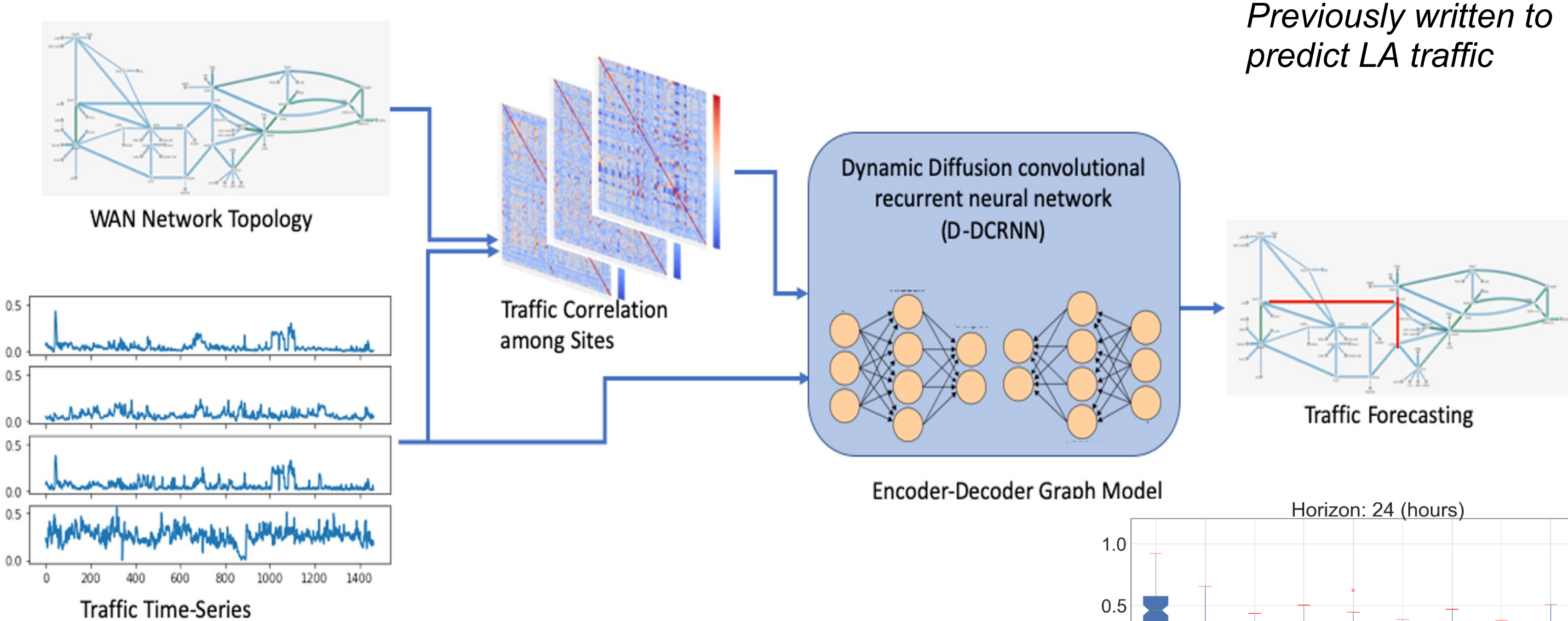
Predicts traffic in the future 3 months on

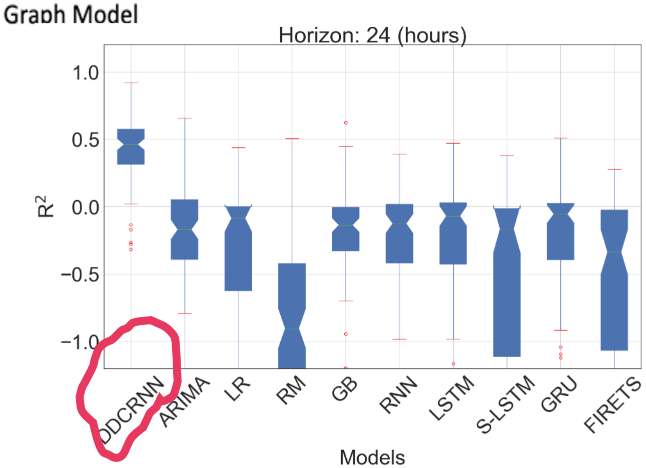


M Kiran et al. Predicting WAN Traffic Volumes using Fourier and Multivariate SARIMA Approaches, International Journal of Big Data Intelligence, 2020

Graph Neural Networks to represent Spatio-Temporal Data

Previously written to predict LA traffic



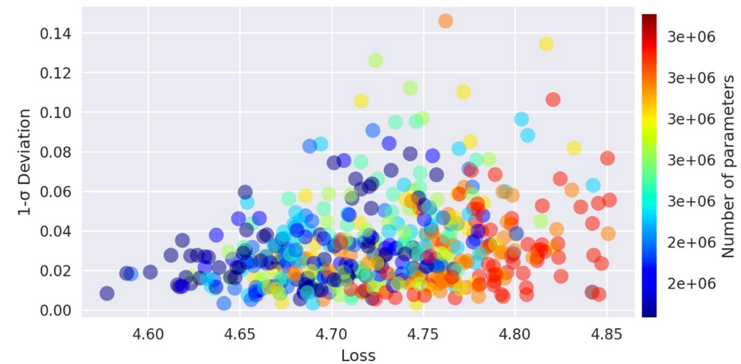


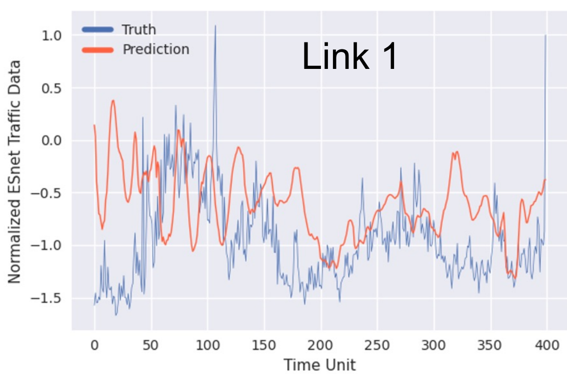
T. Mallick et al. "Dynamic Graph Neural Network for Traffic Forecasting in Wide Area Networks," in *2020 IEEE International Conference on Big Data (Big Data)*.

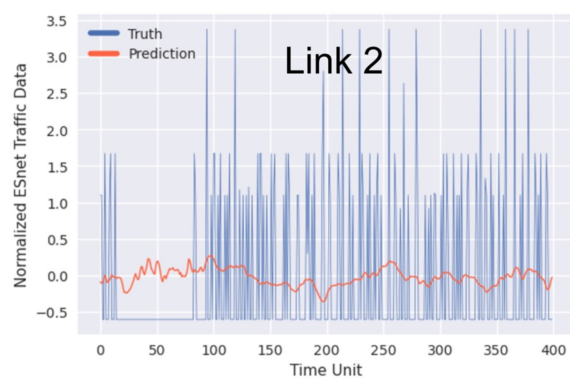
Hyperparameter tuning with GNN

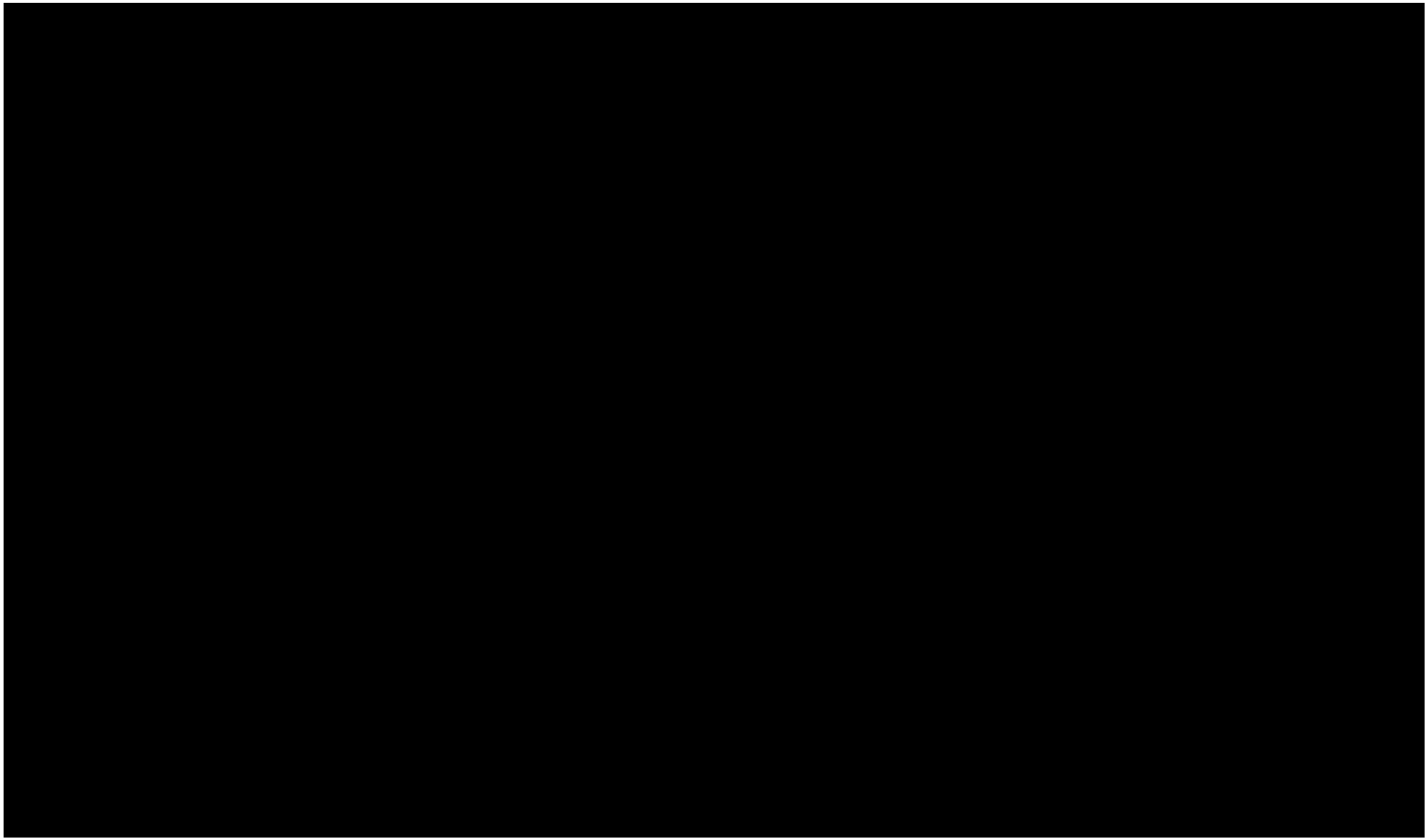
Credit: Juli Mueller, Vincent Dumont

- Large number of parameters to tune
  - Need innovative ways to find the best models
- Learning Window:
  - Tuned for optimum training data length:1 week of data to predict 2 days
- Still some links are better
  - Training data did not contain peaks



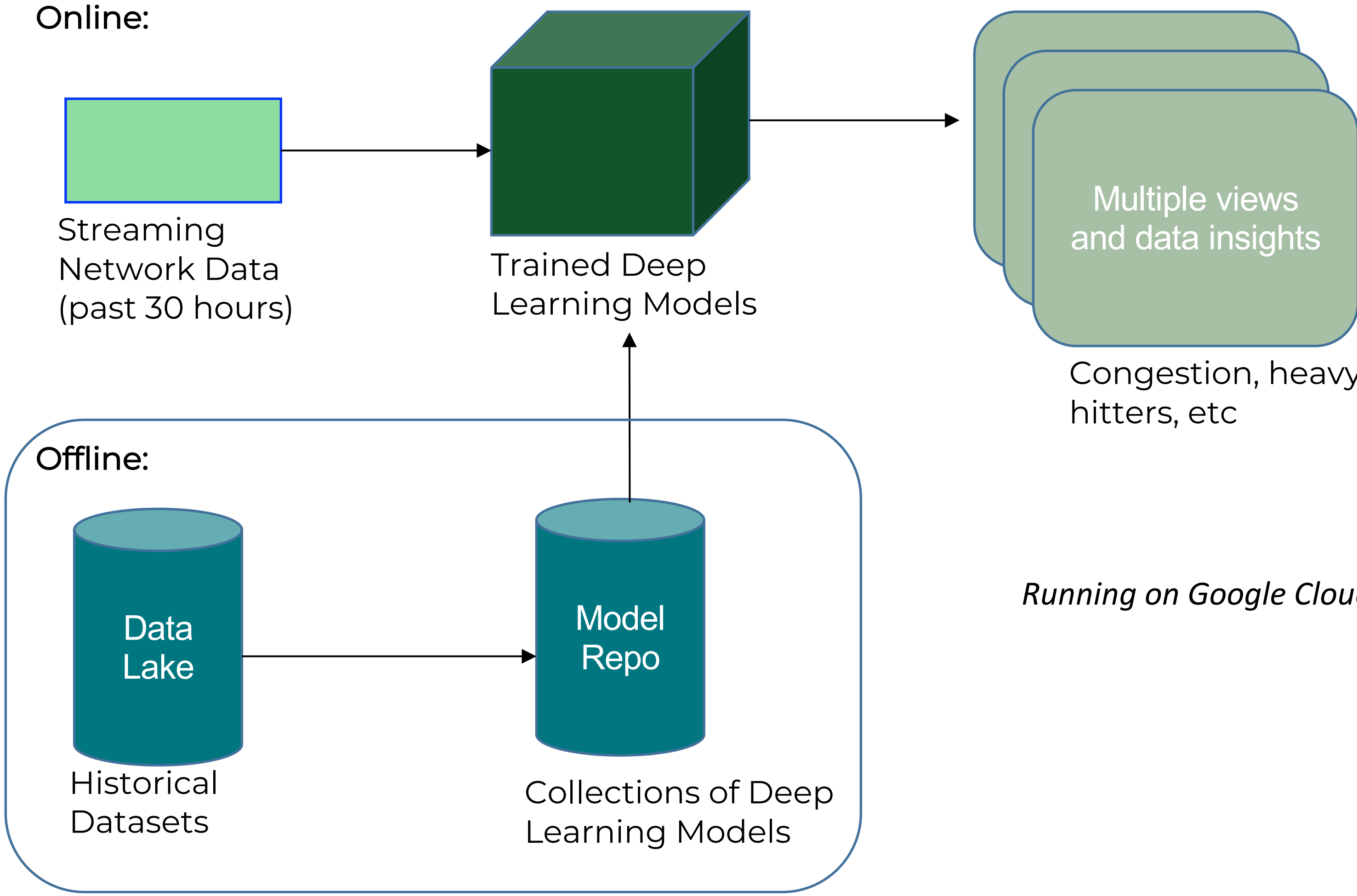






Summer SULI Student:
Nishat Tabassum

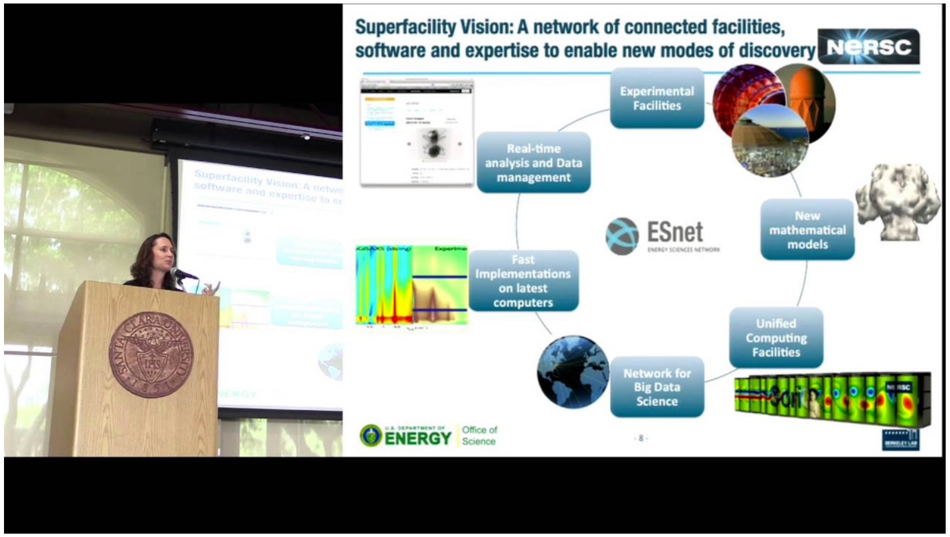
NetPredict as a Cloud Service



Running on Google Cloud

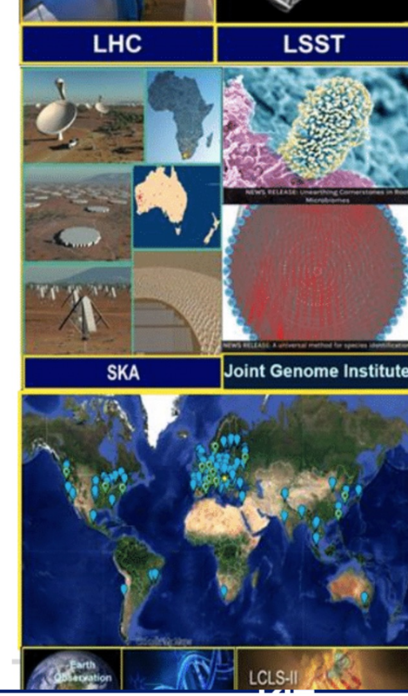
Part of the Superfacility Vision

- Schedule transfers for end-to-end congestion-free performance
- Design better transfer protocols to prevent packet loss
- Optimizing distributed science workflows

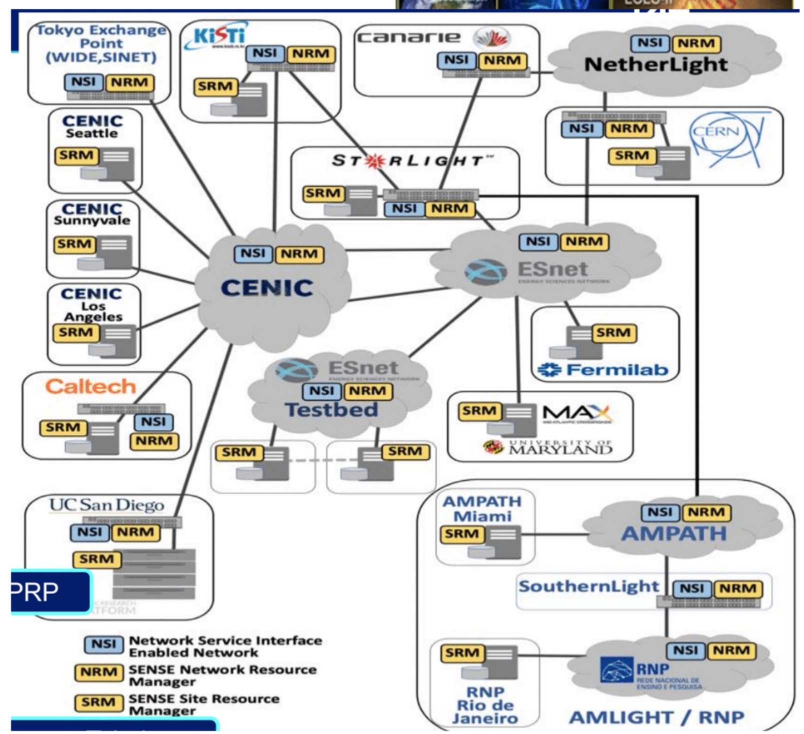


Credit: H. Neuman, CalTech (HEP)

Beyond Network Programmability



- Identifying network bottlenecks to inform better design (Collab. Reservoir Labs)
- Building future internet architectures for data intensive applications
- Thinking **Global Science** Challenges!

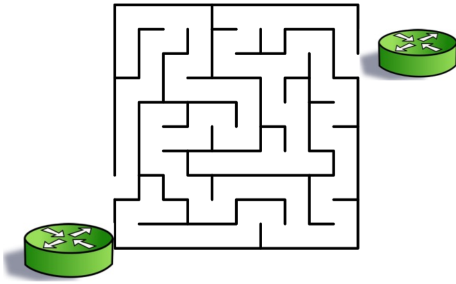


Intelligent Flow Control
(building Hecate)

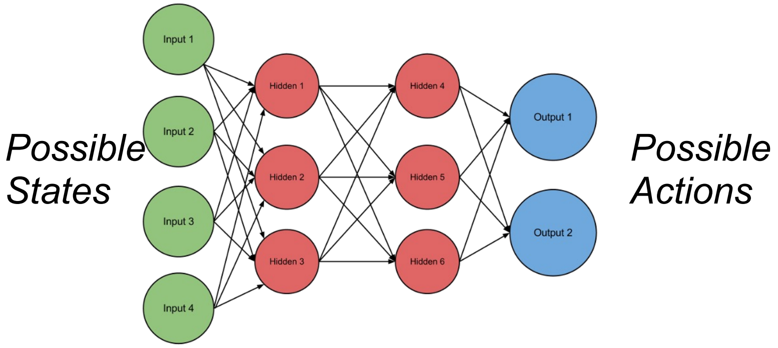
Deep Learning Application 2

Deep Reinforcement Learning for Traffic Engineering

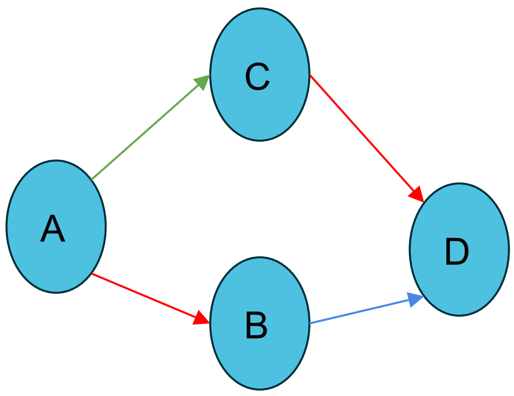
- Multiple ways between source and destination
- Learning the Q function (value function or Bellman equation)



state	action	Q value
state0	Move path1	10
state1	Move path2	2
…	…	..
…	..	..



What does the AI/DL learn?



State contains:

- current traffic allocated on all links
- current loss, latency, throughput on all links
- is the current traffic elephant or mice flow

Action:

- allocate incoming flows to all paths
- learn through trial-and error
- collecting reward value (at the end of each episode)

Once trained:

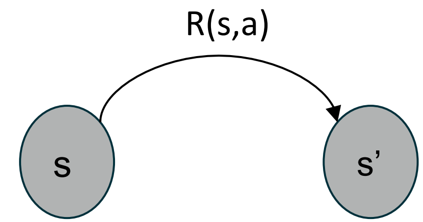
AI knows if “current next conditions are X”, I do action “a”, will get me the optimum result

Design of the Reward Function

$$Q_t(s,a) = Q_{t-1}(s,a) + \alpha\,(R(s,a) + \gamma \max_{a'} Q(s', a') - Q_{t-1}(s,a))$$

Bellman Optimality Equation

Learned Value (expected reward at next state)

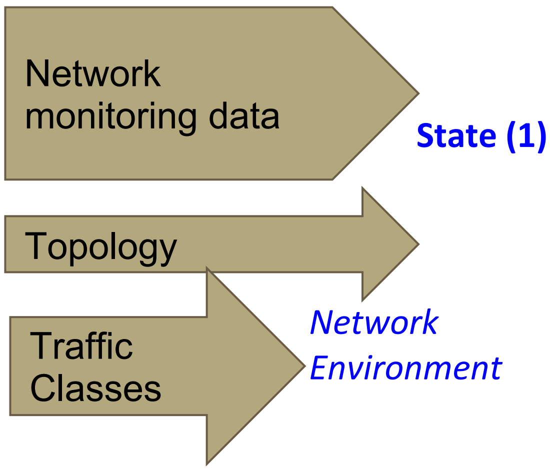


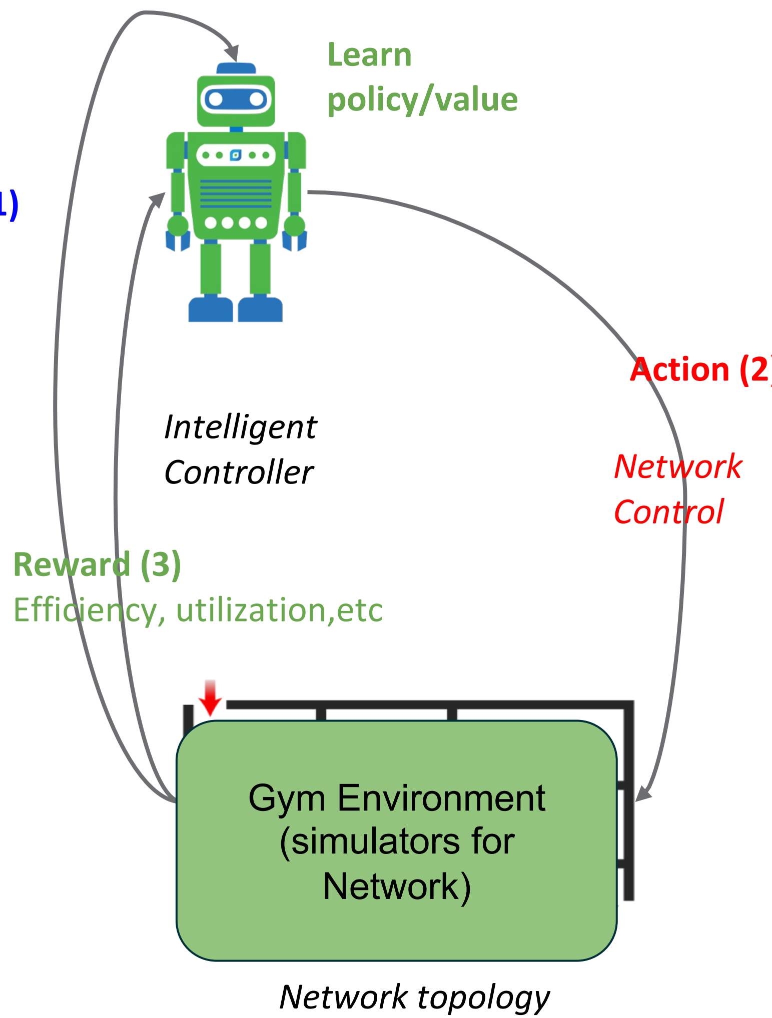
- Hyperparameters: discount factor (γ) - prioritize immediate rewards versus future; learning rate (α)- how quickly it learns and exploration factor (ϵ)

For Network problem:

- optimize for average network utilization
- optimize for latency OR bandwidth OR loss
- optimize for specific traffic class*

Intelligent Controller (HECATE)



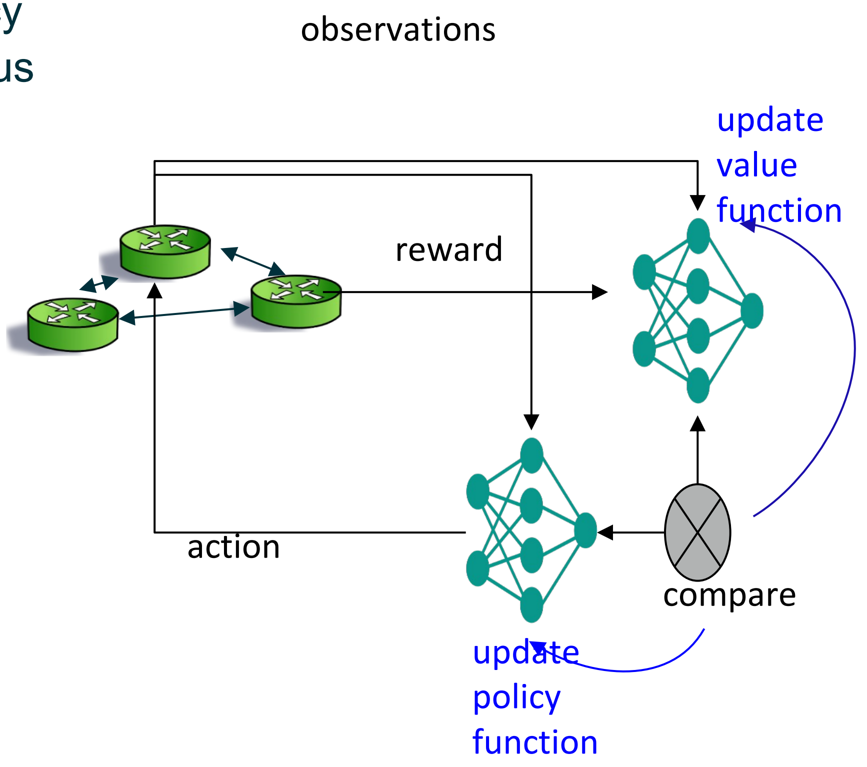


Gyms:

- Own custom gyms
- designed to learn either throughput, utilization or loss (different reward functions)
- Also connect to historical data to simulate real conditions

Training the HECATE algorithm

- Actor-critic model: Deep Deterministic Policy Gradient (DDPG) allows to model continuous state and action pairs
- 2 neural networks compare and improve in value and policy learning
- DDPG works well in complex environments (vs DQN)

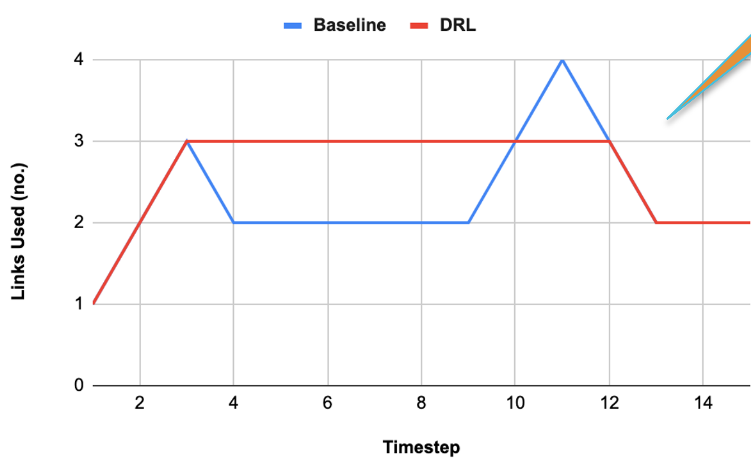


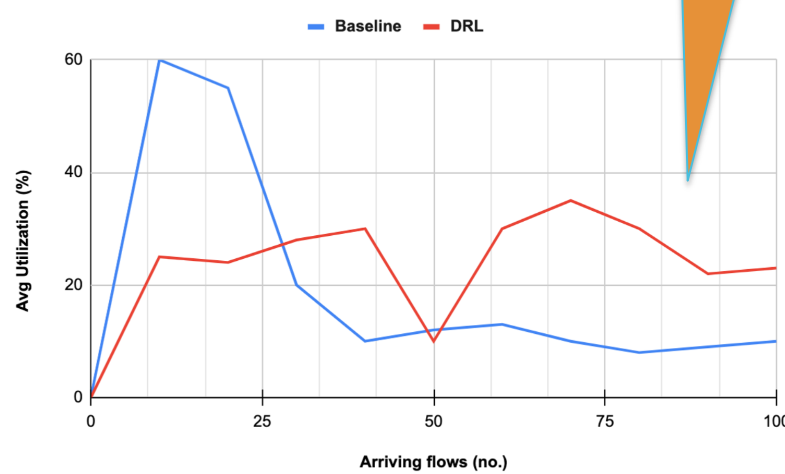
HECATE Overview

- Centralized Control
- Baseline - Dijkstra algorithm
- Flows arriving with Poisson Distribution

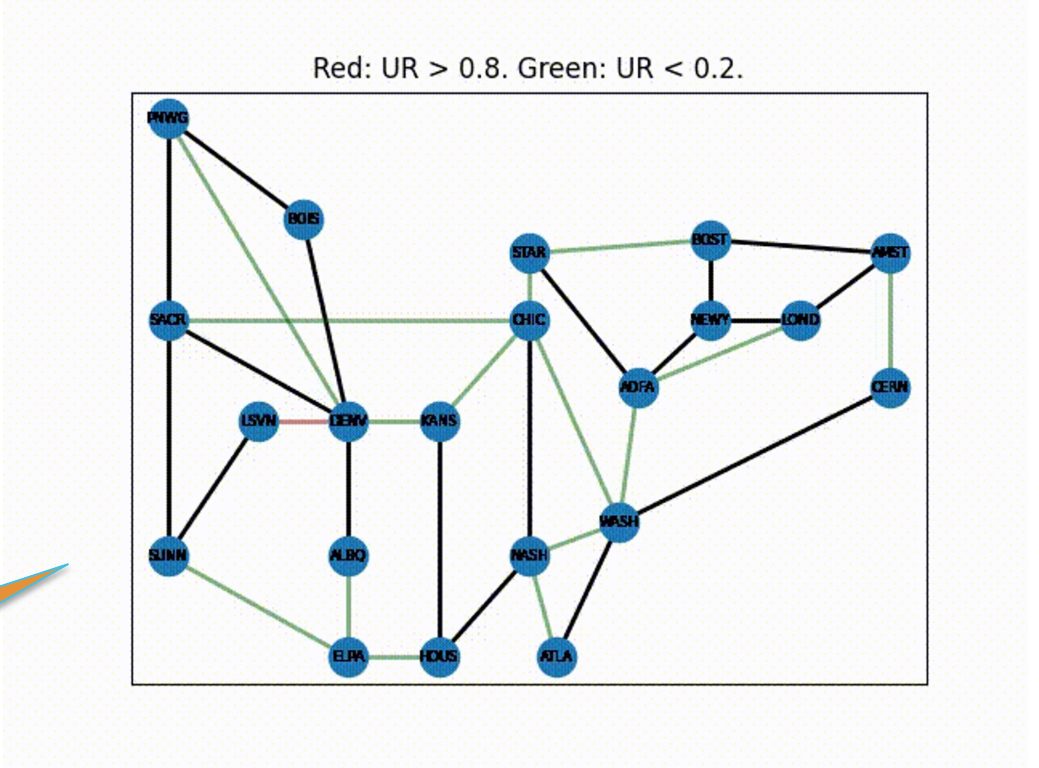
More links used

Better load balancing





HECATE Simulation

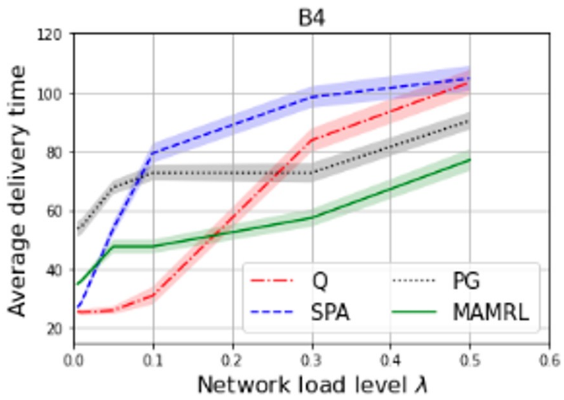


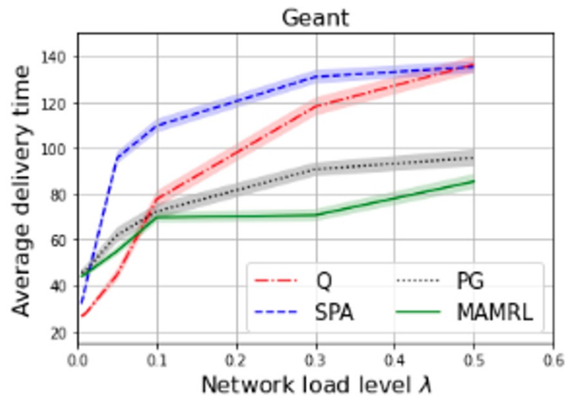
Moves incoming traffic to less used paths

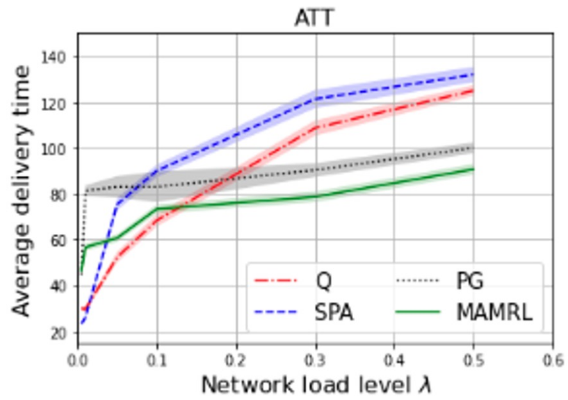
UR: link utilization rate

Improve packet delivery at high loads

- Improves network performance at high loads
- Leverage traffic patterns into learning to cater to different characteristics
- Can be extended to ESnet traffic engineering protocols





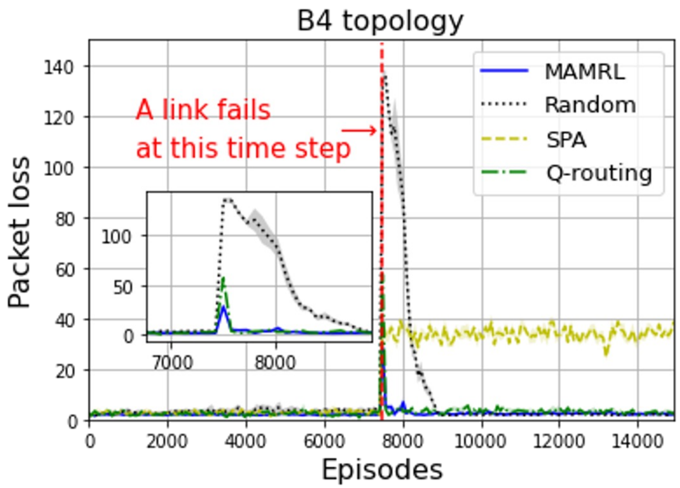


SPA: shortest possible
Q: Q-learning
PG: policy-gradient
MAMRL: multi-agent learning

MAMRL: Exploiting Multi-agent Meta Reinforcement Learning in WAN Traffic Engineering, arxiv, 2021

Quick Adaptation

- Decentralized learning shows quicker adaptation to disruptions
  - link failures or
  - network topology changes

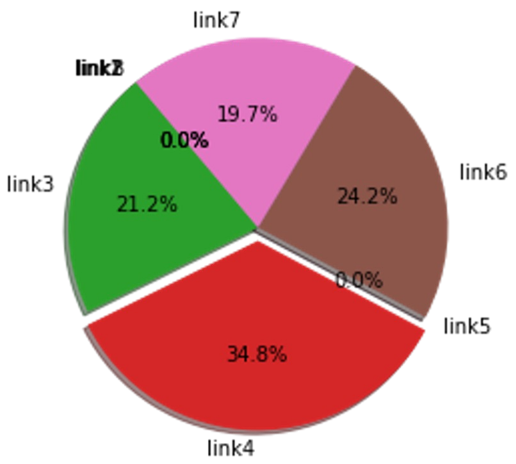


SPA: shortest possible
Q: Q-learning
PG: policy-gradient
MAMRL: multi-agent learning

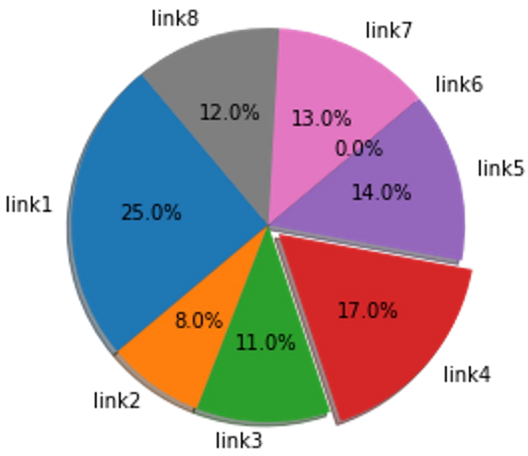
MAMRL: Exploiting Multi-agent Meta Reinforcement Learning in WAN Traffic Engineering, arxiv, 2021

Supercomputing Networking Research 2021 DEMO

Before

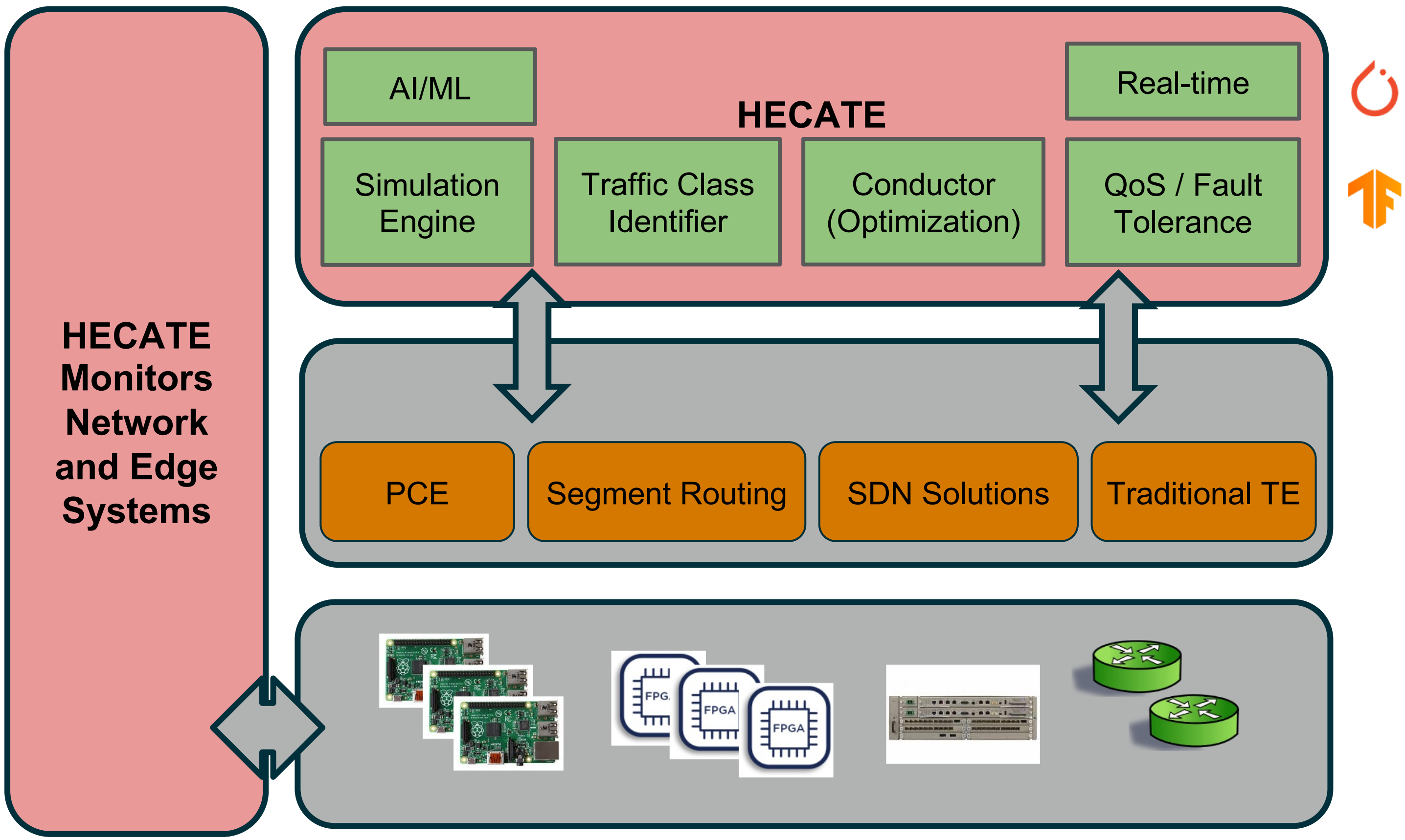


After



MAMRL: Exploiting Multi-agent Meta Reinforcement Learning in WAN Traffic Engineering, arxiv, 2021

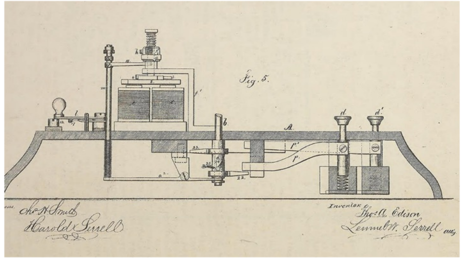
HECATE architectural diagram (high-level)

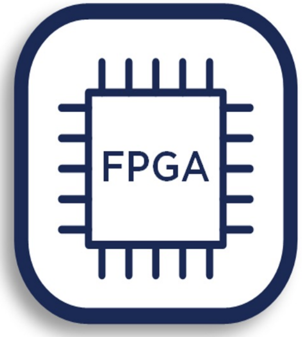


In-House development - HECATE

Patent filed 2021

- Operating in the data plane
- AI communicates with ESnet network tools
- Adding regression models with DRL





Deep Learning with 5G Networks

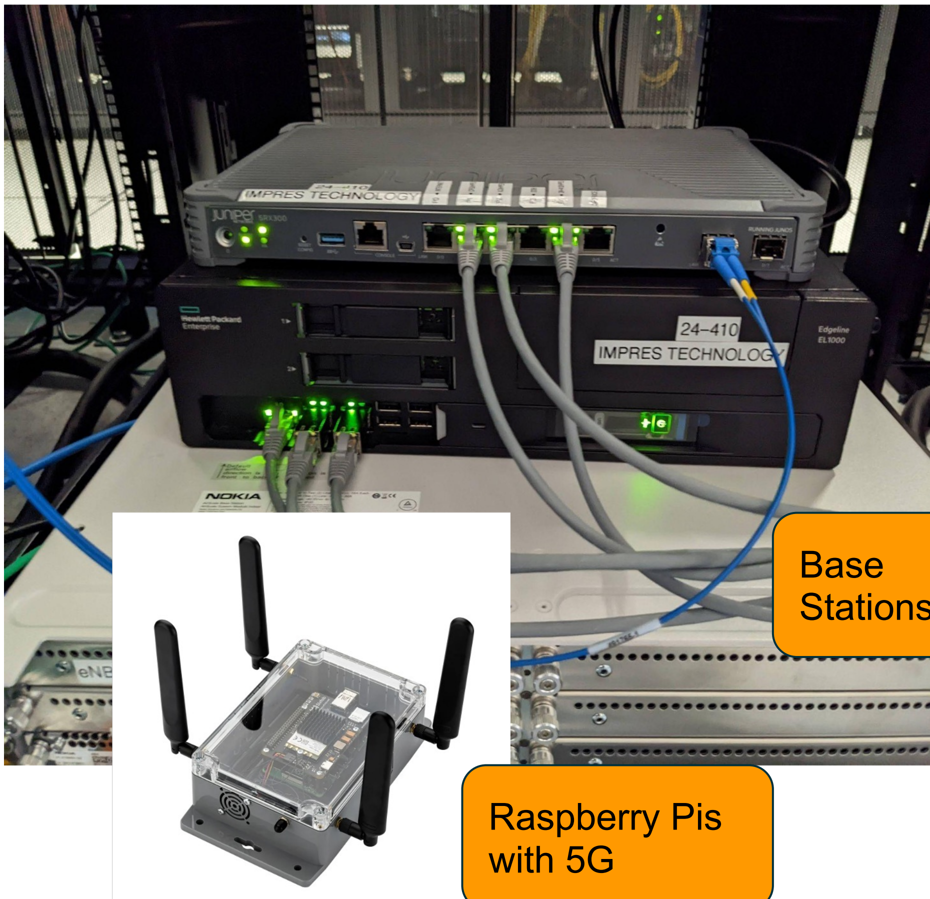
The Future!

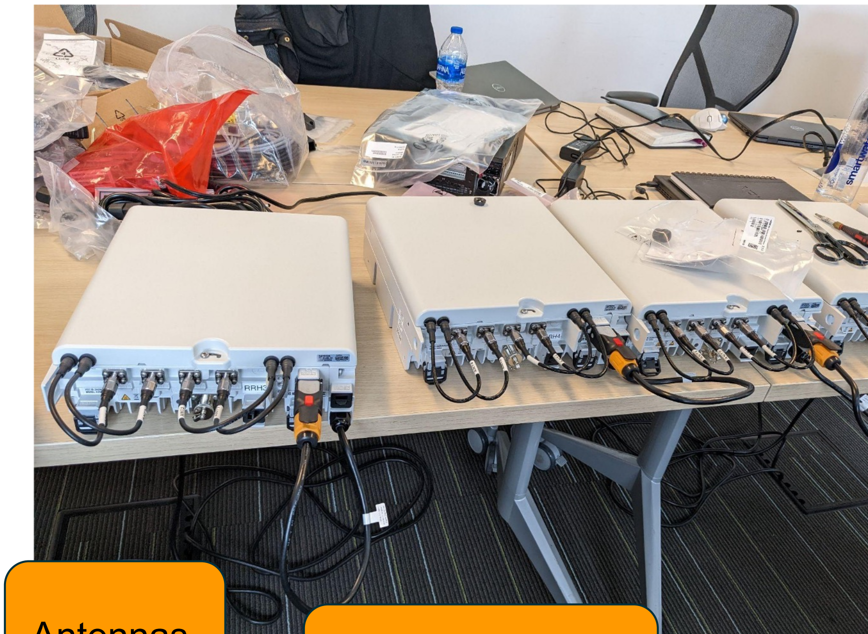
Science Networks are Expanding to Wireless, 5G, 6G and Beyond

- Network statically deployed, unattended sensors need full-time monitoring
- Sensors self-manage and optimize data collection
- Networks with mobile sensors including unmanned aerial systems
- We need continual monitoring of the sites, data is processed in real-time and transmitted to central location for further processing with 5G speed
- Develop novel mesh network topology for flexible inclusion of new sources and use 5G latency targeting off between edge and cloud processing - much like new cells in the body

θ-AWR: New Advanced Wireless 5G Testbed (Building 59)

Including CBRS
Collab: A.Wiedlea





Base Stations

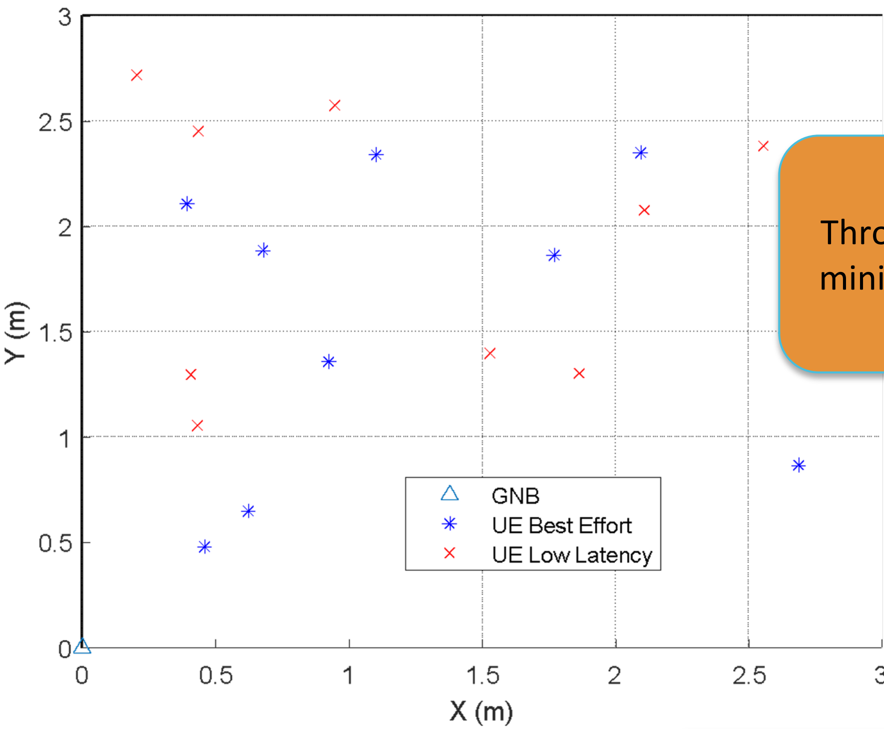
Antennas

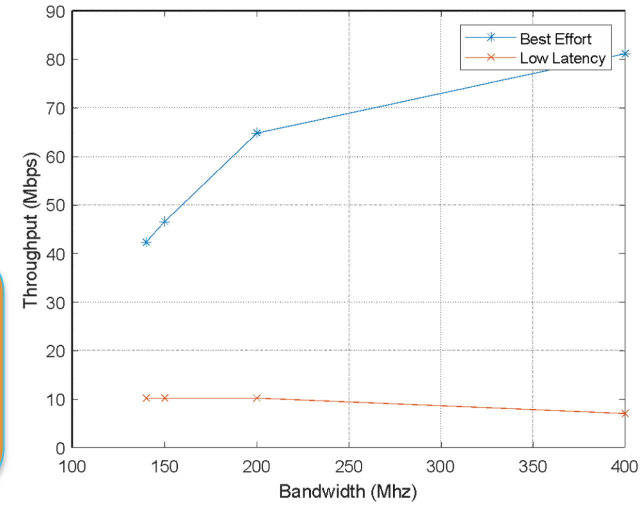
O-RAN developments

Raspberry Pis with 5G

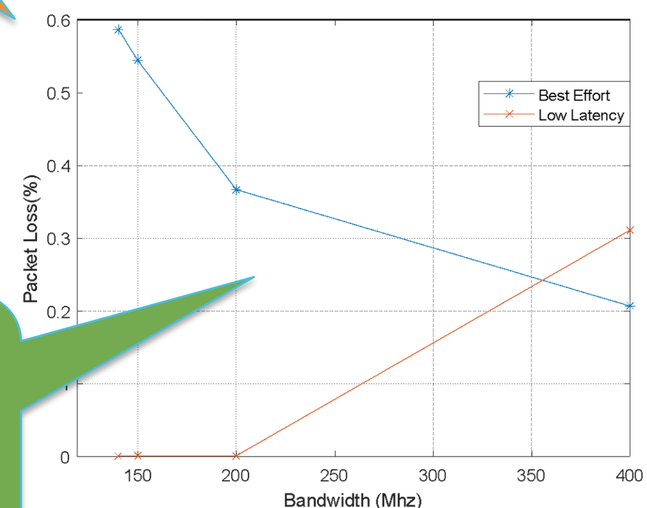
Credit: Qun Wang (Claud)

5G Deployments Research





Better Throughput and minimize packet loss



Tx Packets: 6000
Tx Bytes: 7680000
Tx Throughput Offered : 102.4

Simulations help inform our real world deployments for optimum networking

5G Network Slicing

Credit: Summer Student Piyush Nayak

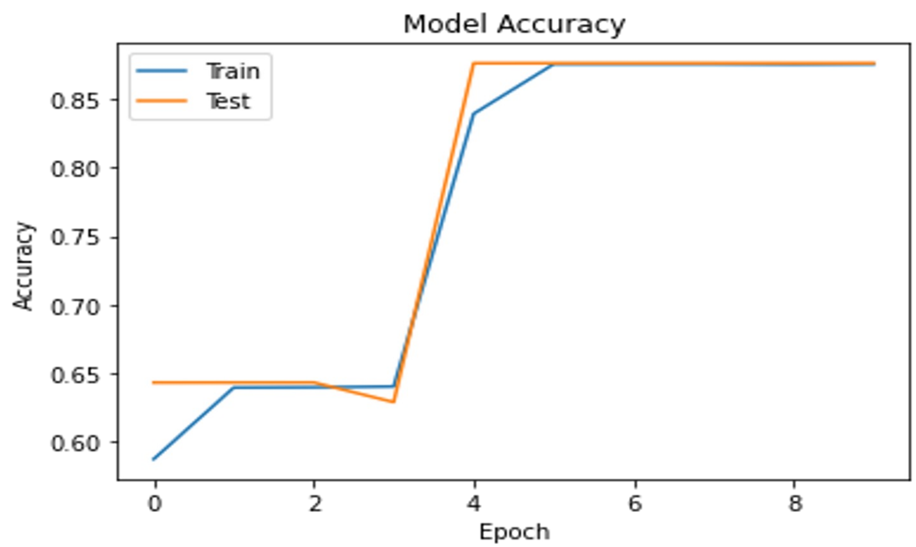
- Network slicing refers to the sharing of network resources to help meet the system's requirements
- Categories of 5G network slices - URLLC, eMBB and mMTC

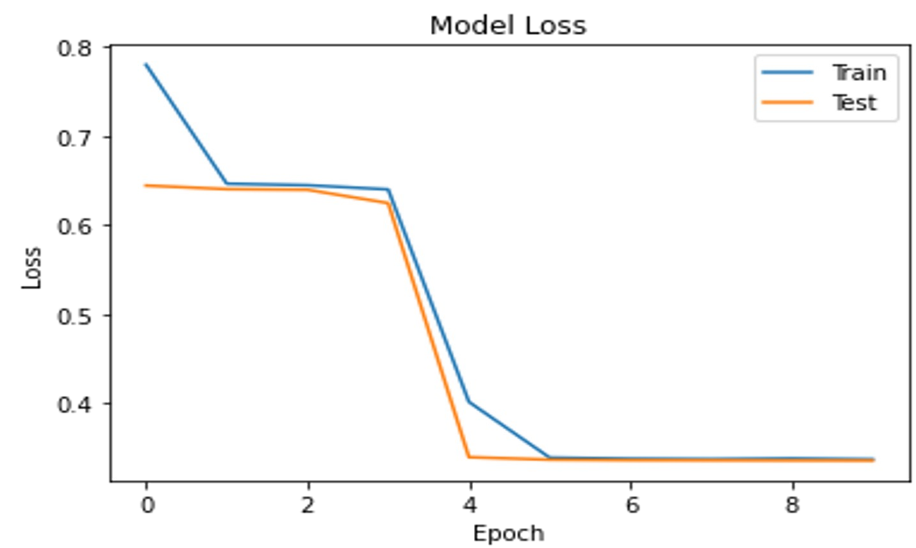
- Neural Network
  - Model maintains a funnel approach consisting of 5 stacks of CNN layers with activation functions in between
  - Softmax is used as the output layer of choice, providing us the probabilities of the three classes as the output variables
  - Adam was used as the optimizer of choice with a variable learning rate scheduler.

Feature Types
• Use Case type
• LTE/5G UE Category
• Technology Supported
• Day
• Time
• QCI
• Packet Loss Rate (Reliability)
• Packet Delay Budget (Latency)
• Slice Type (Output)

Experimental Results

- The model took 54 secs for each epoch to train on the train data.
- It was able to attain a validation accuracy of 90% during the training phase.
- After training, the model was tested on the test dataset, which resulted in an **accuracy of 89%.**
- The model was able to correctly predict the best network slice using the device's features.





Conclusions

Future is HyperConnected with many Self-driving Infrastructure!

Conclusions

Future is HyperConnected with many Self-driving Infrastructure!

- New networking challenges in 5G and Beyond will truly transform DOE science
- Rise to experimentation in virtual worlds - DOE Metaverse
- Intelligent control with optimal data movement
- New research needed in
  - AI data validation - uncertainty, adversarial attacks and faulty data
  - Correctness and testing AI
  - encourage wireless adoption across Science areas
  - new Science with new data
- New moving data sources seamlessly integrate into network fabric like human cells

Our work with the new testbed, self-driving systems, simulations via agent-based models and real DOE use cases will help push this new frontier for networking science!

Call for Papers: SC INDIS 2022

https://scinet.supercomputing.org/community/indis/

- INDIS 2022: 9th Workshop on Innovating the Network for Data-Intensive Science

  Held in conjunction with SC22 at Dallas, Texas. In cooperation with: IEEE Computing Society and Association for Computing Machinery (ACM)

- (Submissions due Monday, August 15th, 2022 AoE)

- Topics of interest include, but are not limited to:

  [Innovations in Networking Space - SDN, AI, monitoring, etc]

Acknowledgements

Teams working across multiple projects

DAPHNE

(Deep Learning for High Performance Networks)

NetPredict, Hecate

- Nishat Tabussam
- Piyush Nayak
- Scott Campbell
- Nick Buraglio

Self-driving 5G network

- Anastasiia Butko
- Ren Cooper
- Claud Wang

Intelligent Wireless Automation

- Andrew Wiedlea
- Xi Yang

Poseidon

- Ewa Deelman (USC)
- Anirban Mandal (RENCI)
- Cong Wang (RENCI)
- George Papadopoulos (USC)
- Prasanna Balaprakash (ANL)

Hyperparameter optimization and uncertainty quantification (HYPPO)

- Julianne Mueller
- Vincent Dumont

<mkiran@es.net>

"Self-Driving"

Systems manage themselves according to an administrator's goals. New components integrate as effortlessly as a new cell establishes itself in the human body. *These ideas are not science fiction, but elements of the grand challenge to create self-managing computing systems.*

- Kephart, Chess (IBM) 2003

Inspired by human nervous system

Thank You