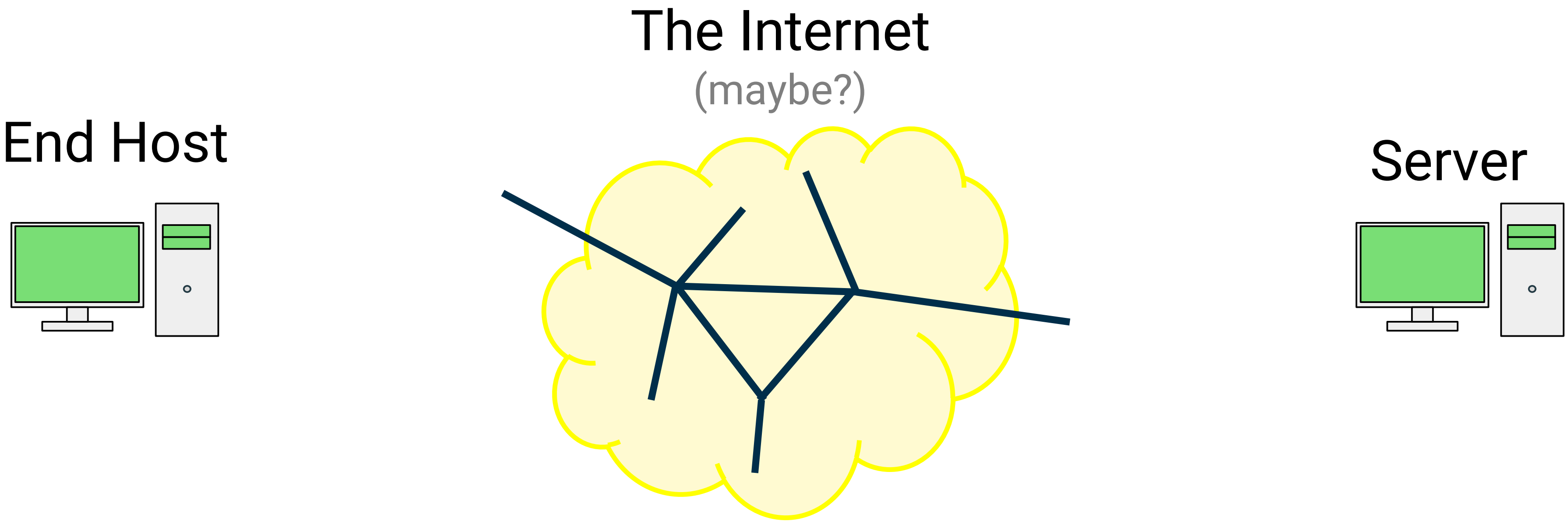
A Deep Reinforcement Learning Perspective on Internet Congestion Control

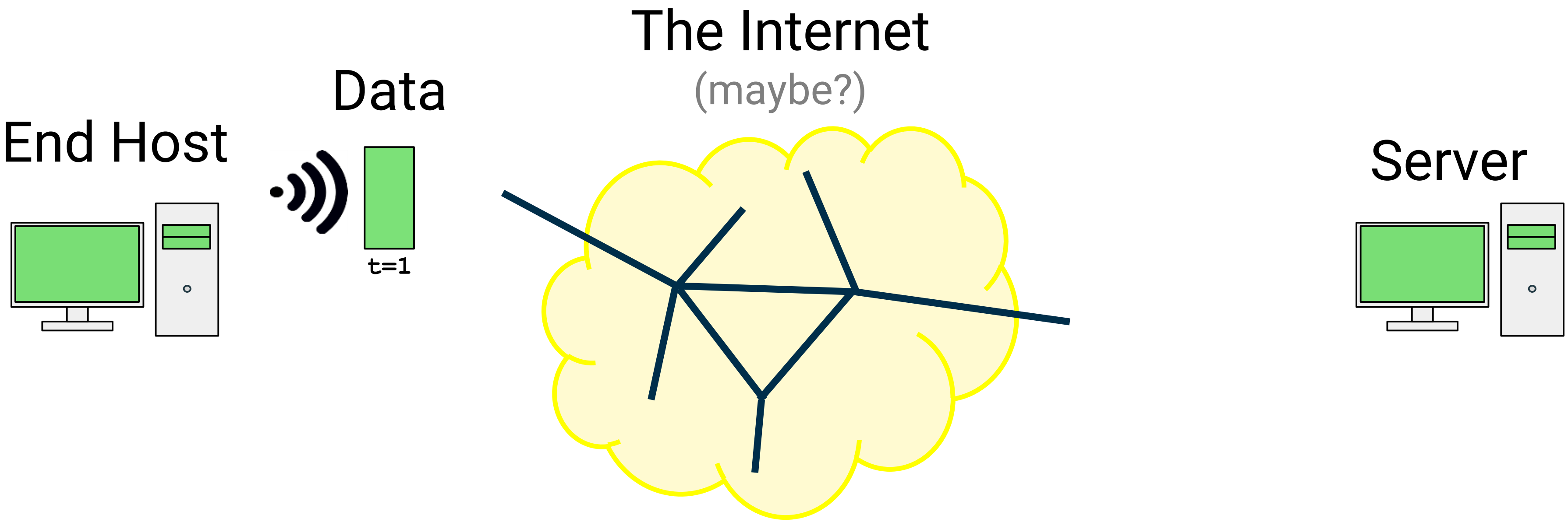
by **Nathan Jay***, Noga H. Rotman*, Brighten Godfrey, Michael Schapira, and Aviv Tamar

*Equal contribution

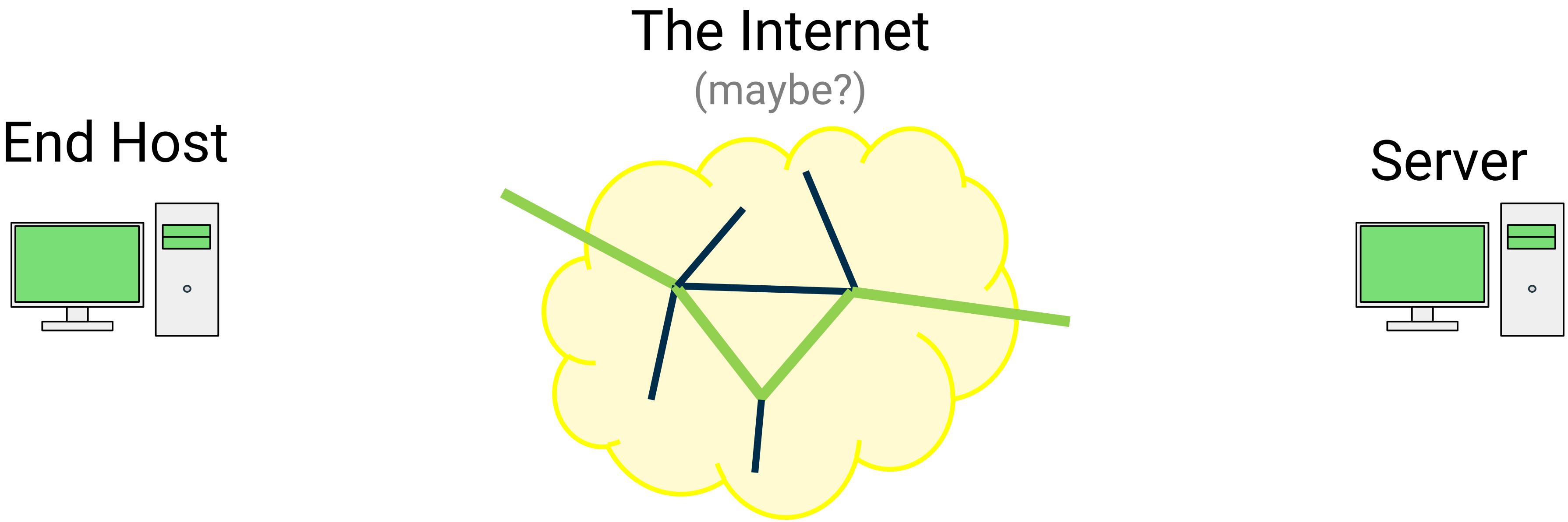
Internet Congestion Control



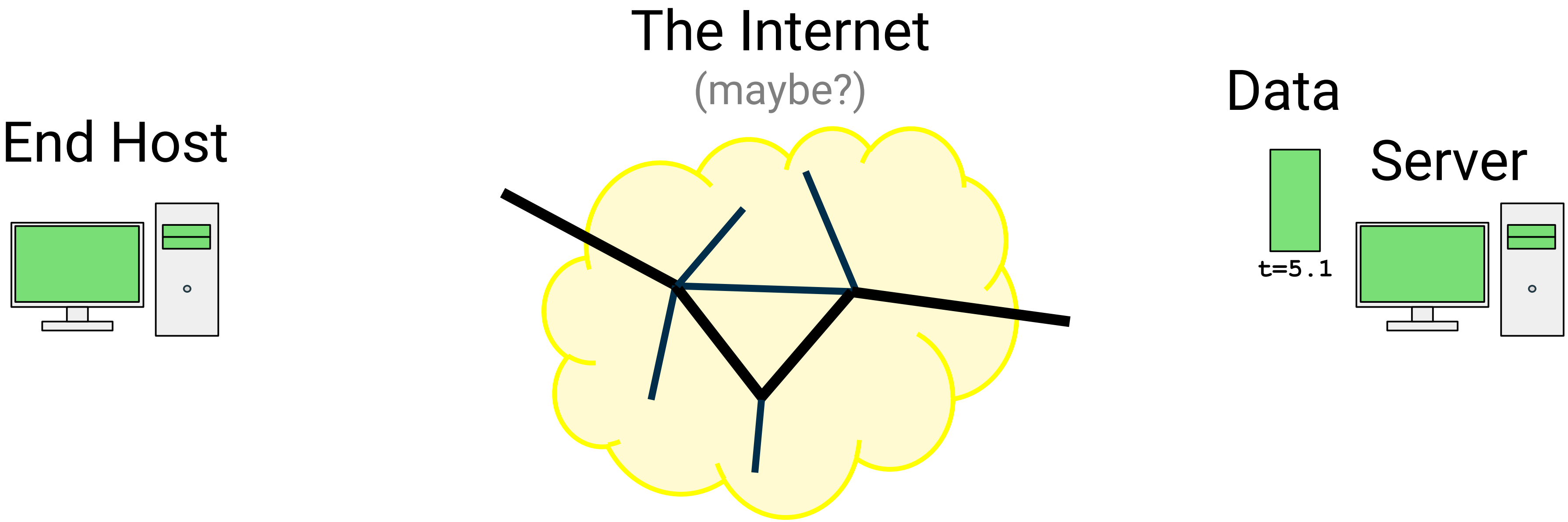
Internet Congestion Control



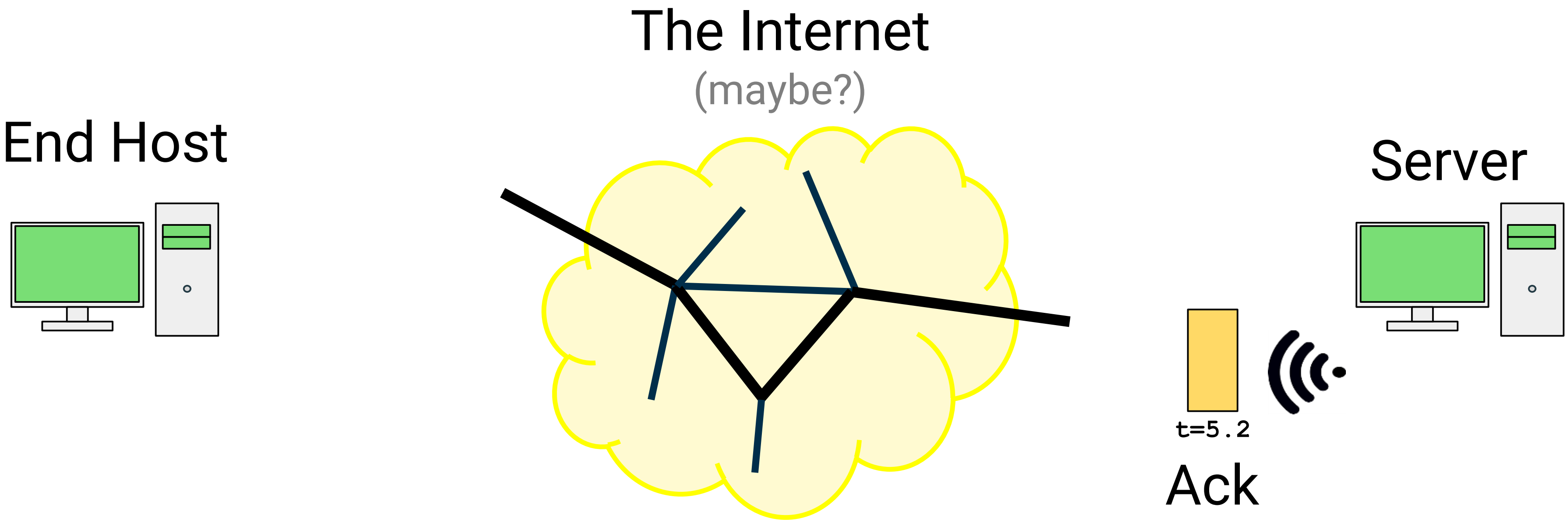
Internet Congestion Control



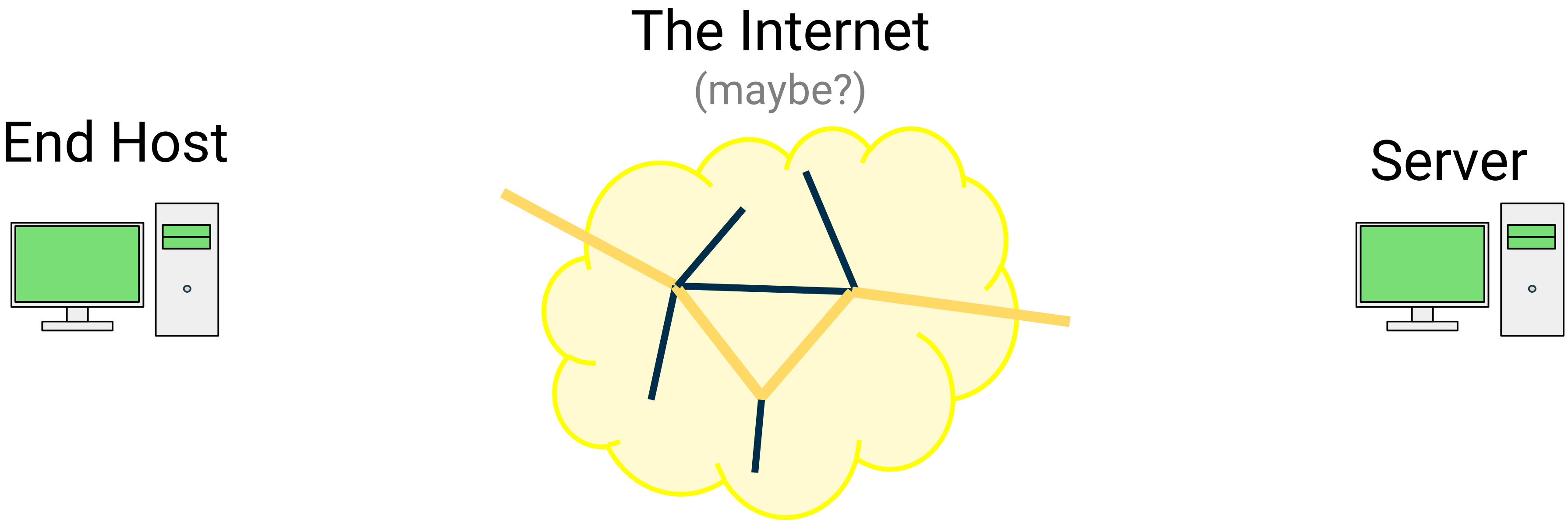
Internet Congestion Control



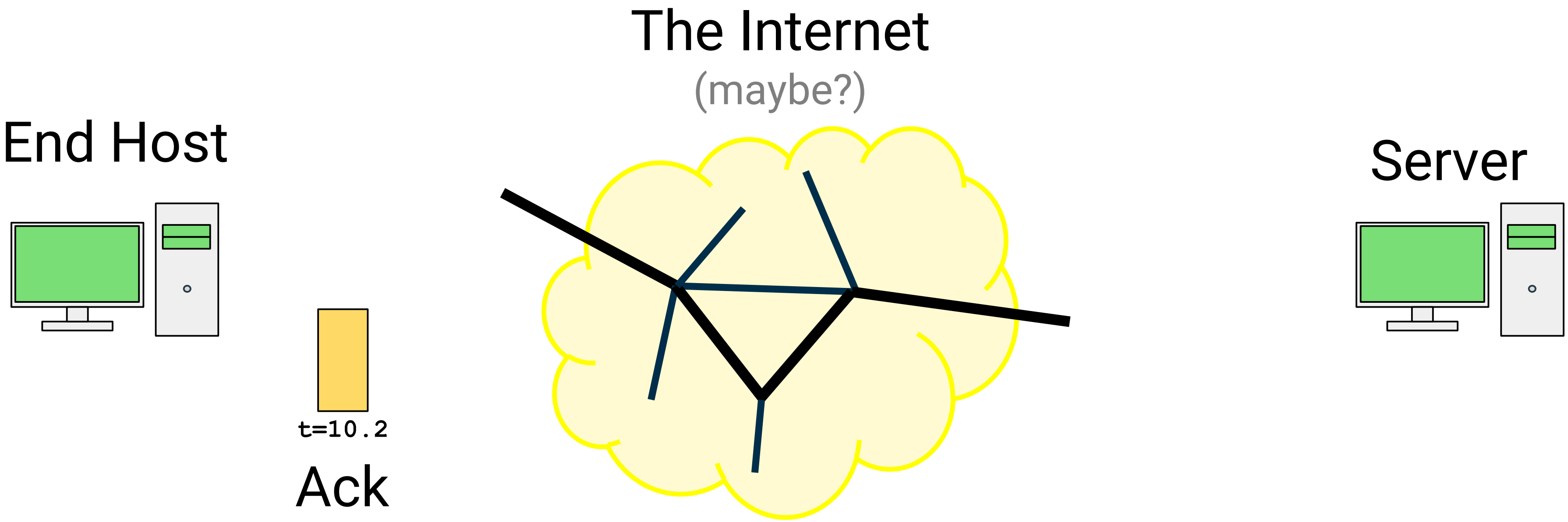
Internet Congestion Control



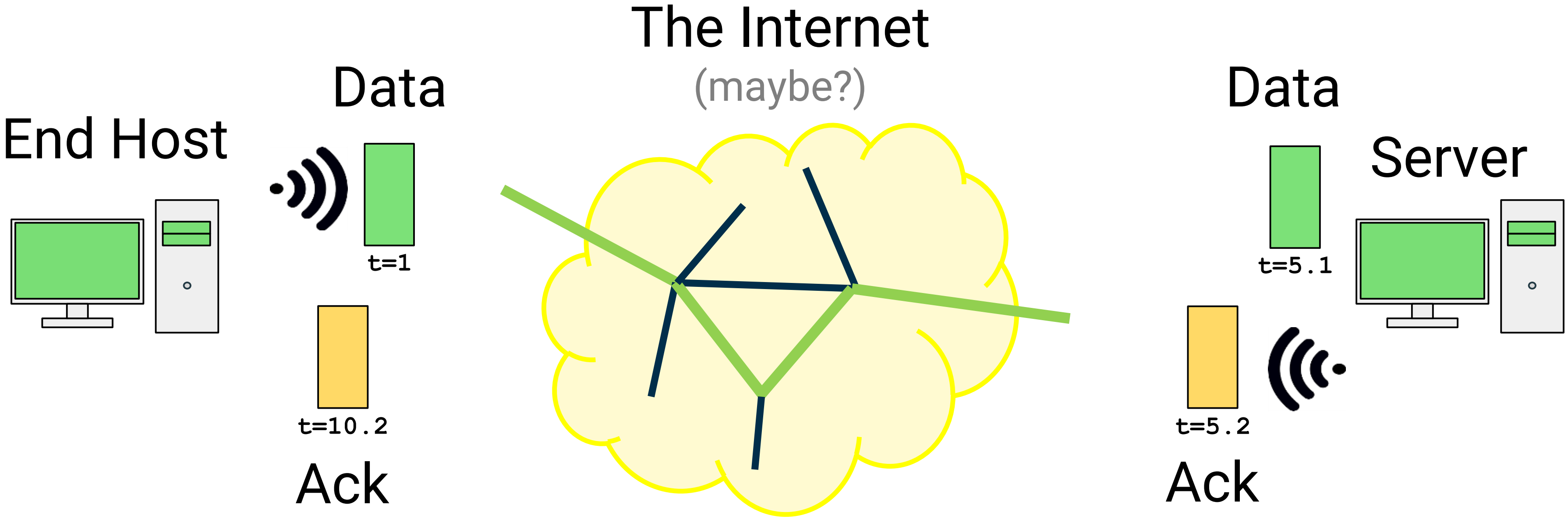
Internet Congestion Control



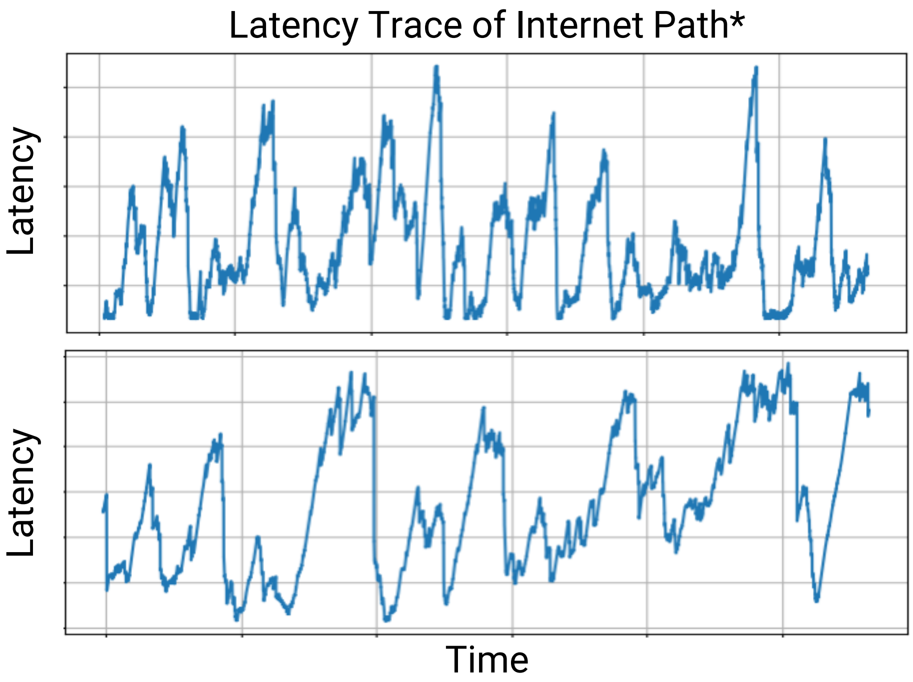
Internet Congestion Control



Internet Congestion Control



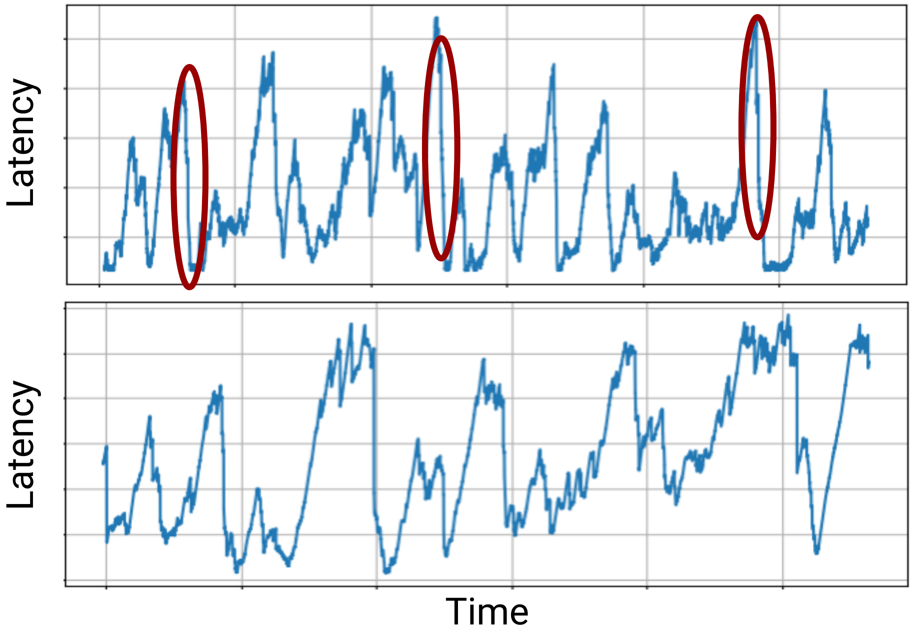
Internet Congestion Control



*from pantheon.stanford.edu

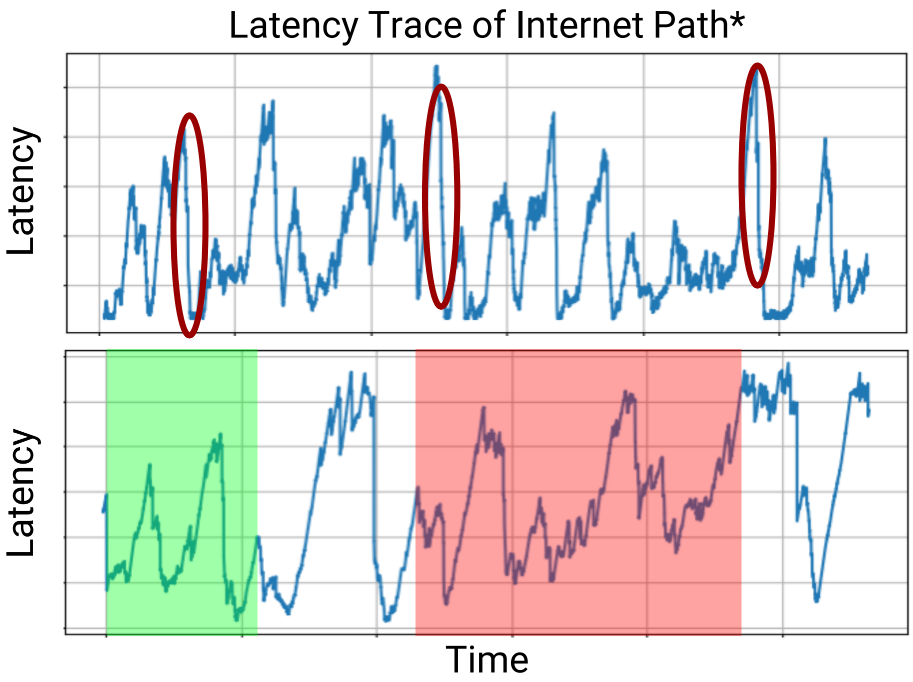
Internet Congestion Control

Latency Trace of Internet Path*



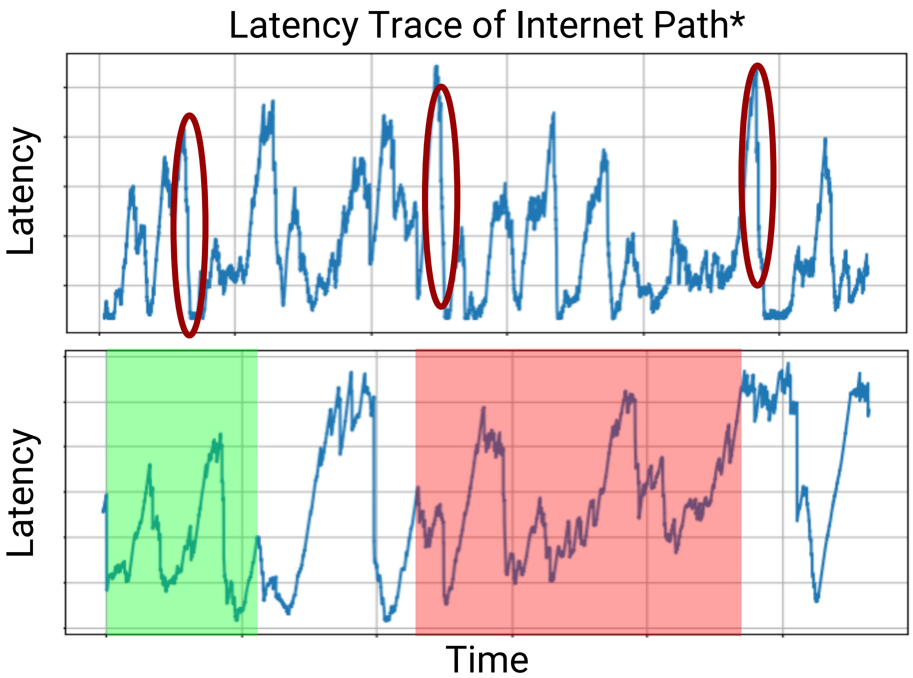
*from pantheon.stanford.edu

Internet Congestion Control



*from pantheon.stanford.edu

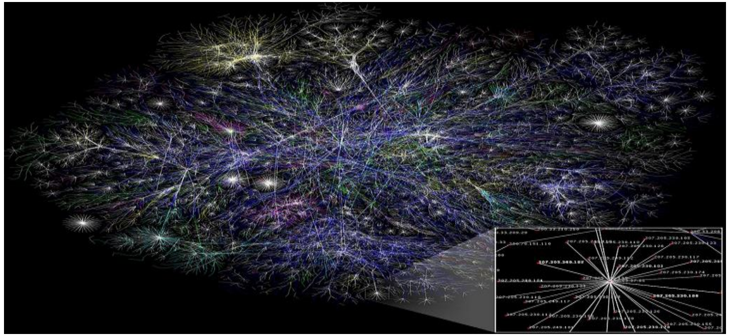
Internet Congestion Control



*from pantheon.stanford.edu

Underlying Complexity:

- Enormous, dynamic network



- Massive agent churn

  ~80,000 agents/second

  YouTube

- Very little information

Revisiting Congestion Control

Congestion Control Timeline

1988 | Flavors of TCP Congestion Control (Tahoe, Reno, Cubic, Illinois, Vegas, …) | 2016 | 2019

- Same network model
- Same action space
- Slightly different control algorithms

Revisiting Congestion Control

Congestion Control Timeline

1988 2016 2019

Flavors of TCP Congestion Control
(Tahoe, Reno, Cubic, Illinois, Vegas, ...)

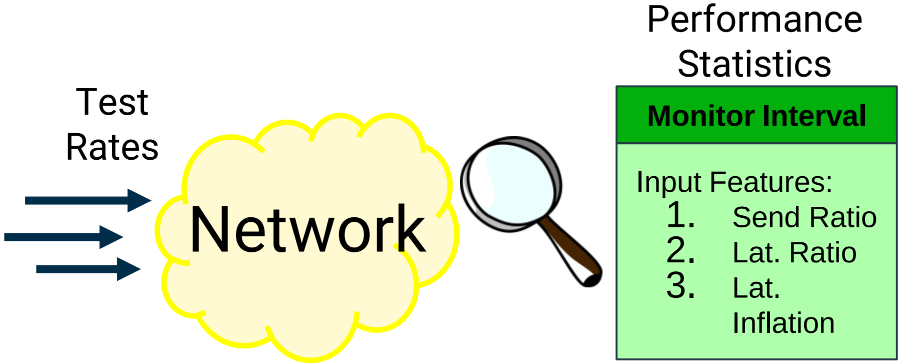
- Same network model
- Same action space
- Slightly different control algorithms

Introduction of **QUIC**, replaces significant amount of Google traffic.

- New models
- New action space (packet pacing added to Linux)
- Novel control algorithms and research (BBR, Copa, PCC)

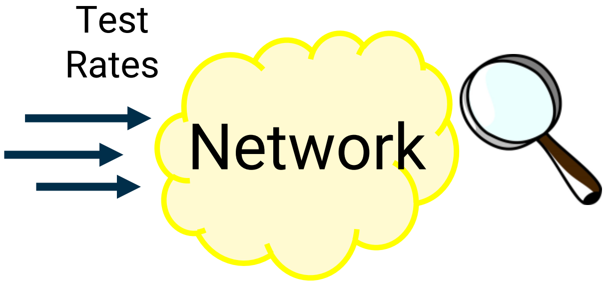
Reward-based architecture: PCC

Observations



Reward-based architecture: PCC

Observations



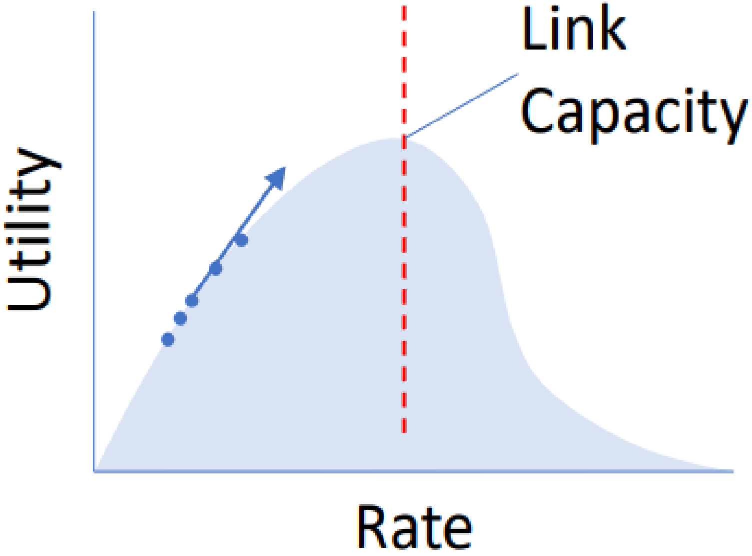
Performance Statistics

Monitor Interval

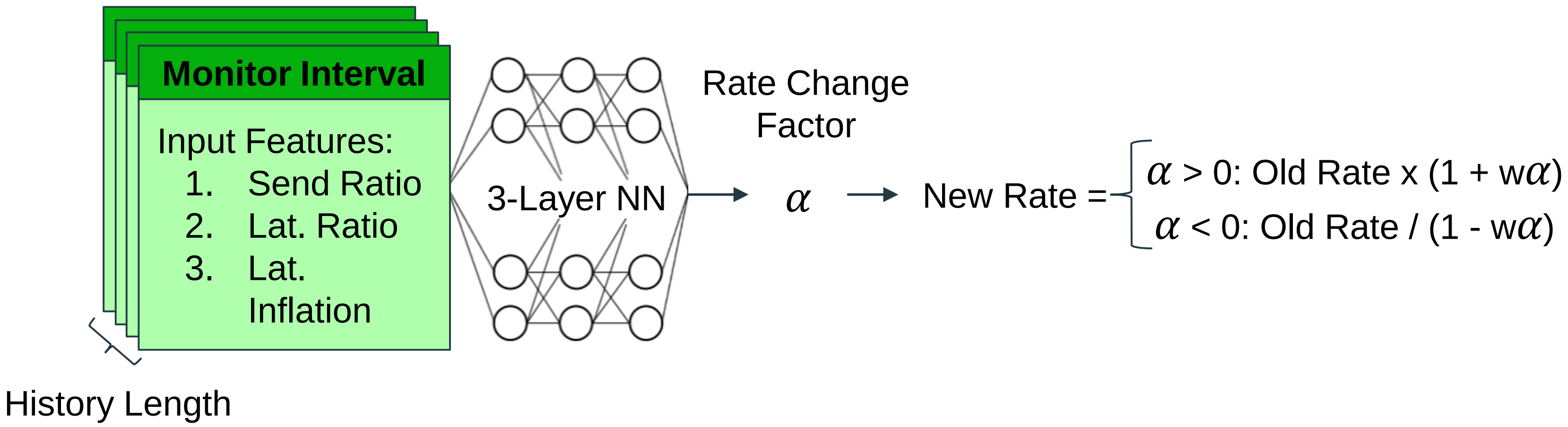
Input Features:

1. Send Ratio
2. Lat. Ratio
3. Lat. Inflation

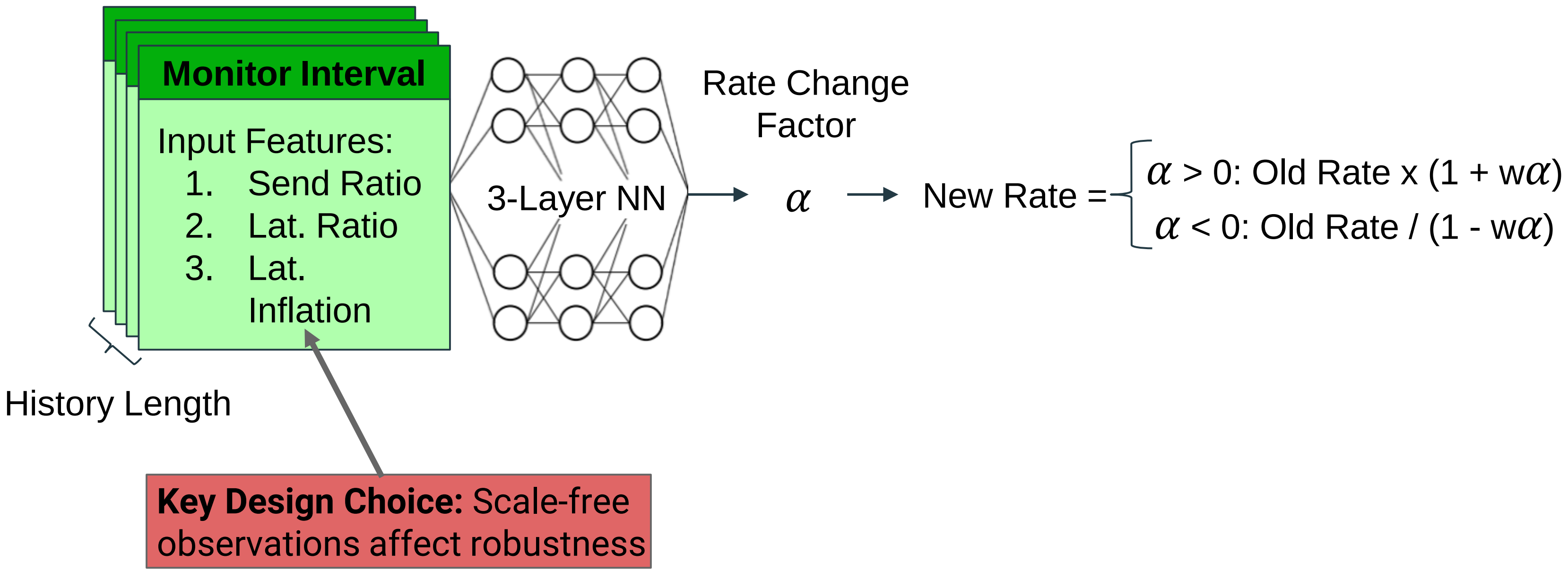
Actions



Agent Architecture



Agent Architecture



Training/Testing Environment

Training Environment:

- Simulated network
- Each episode chooses link parameters from a range:

Capacity	Latency	Loss	Queue
1 - 6mbps	50 - 500ms	0 - 5%	1 - ~3000pkt

- **Standard gym** at github.com/PCCProject/PCC-RL

Training/Testing Environment

Training Environment:

- Simulated network
- Each episode chooses link parameters from a range:

Capacity	Latency	Loss	Queue
1 - 6mbps	50 - 500ms	0 - 5%	1 - ~3000pkt

- **Standard gym** at github.com/PCCProject/PCC-RL

Testing Environment:

- Real packets in Linux kernel network emulation
- Much wider testing range:

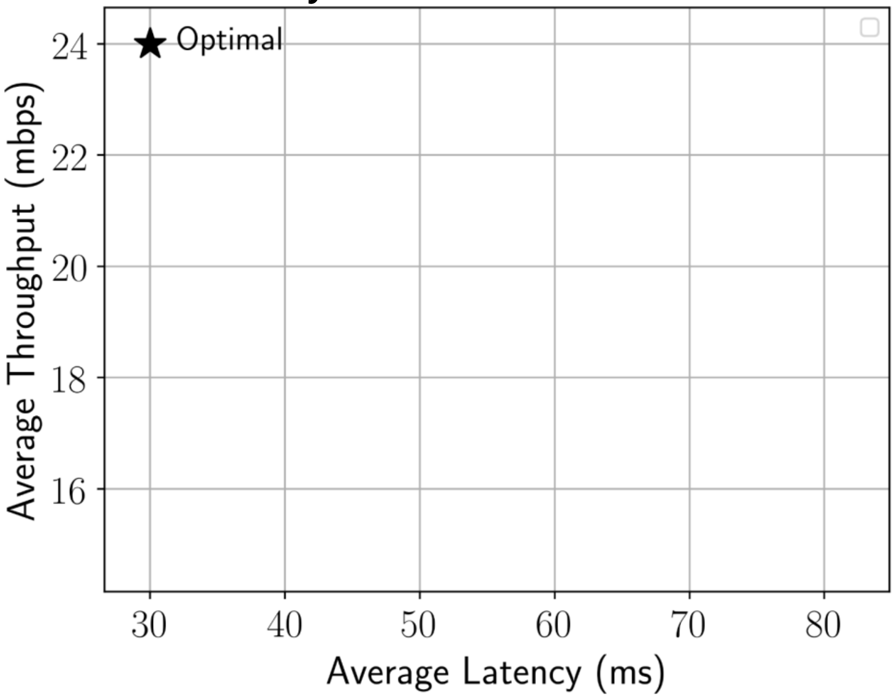
Capacity	Latency	Loss	Queue
1 - 128mbps	1 - 512ms	0 - 20%	1 - 10000pkt

State-of-the-art Results

Test Description:

- Emulated network, with real Linux kernel noise
- Time-varying link

Emulated Dynamic Link Performance

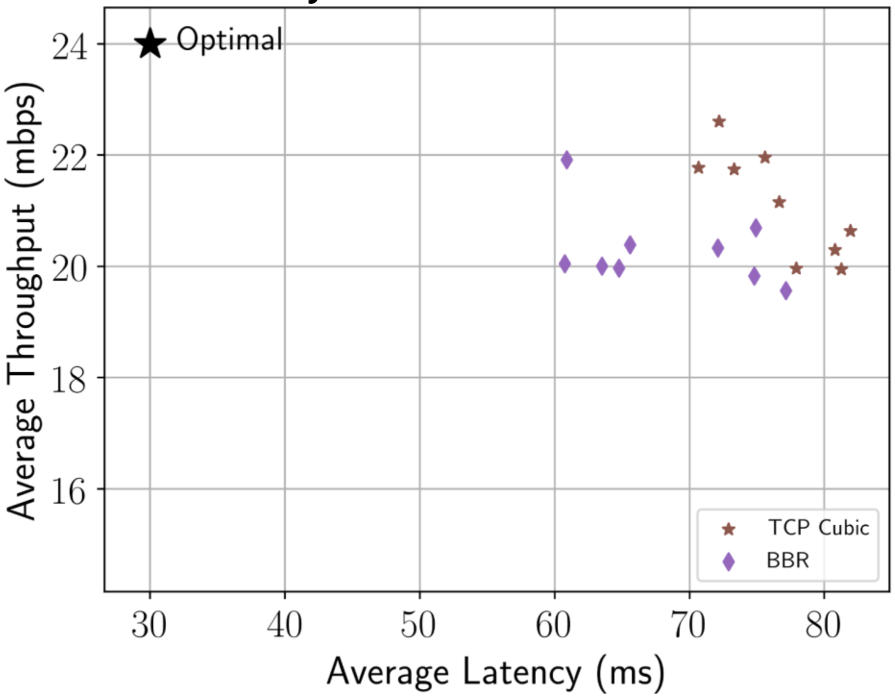


State-of-the-art Results

Test Description:

- Emulated network, with real Linux kernel noise
- Time-varying link

Emulated Dynamic Link Performance

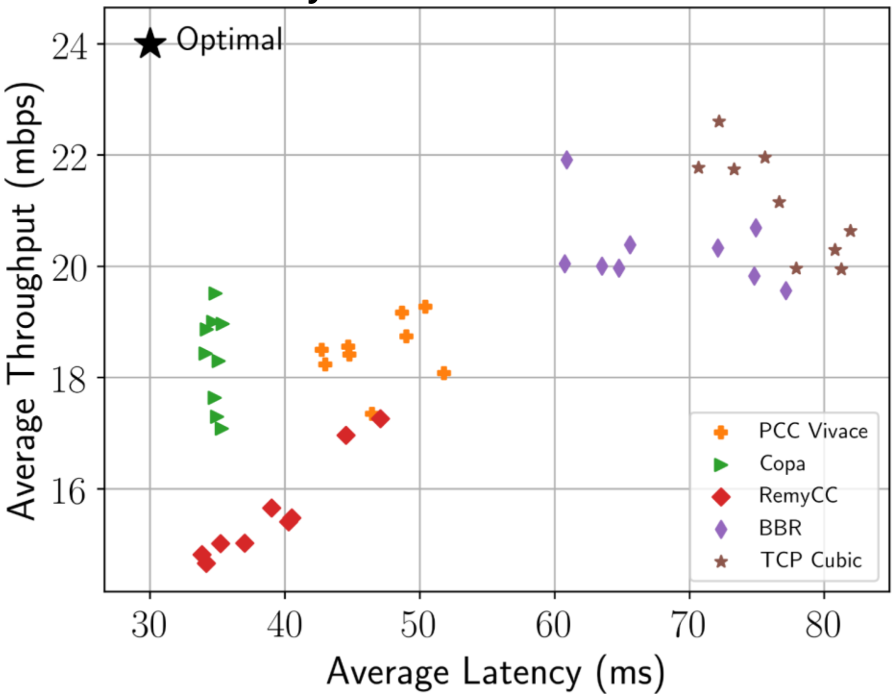


State-of-the-art Results

Test Description:

- Emulated network, with real Linux kernel noise
- Time-varying link

Emulated Dynamic Link Performance

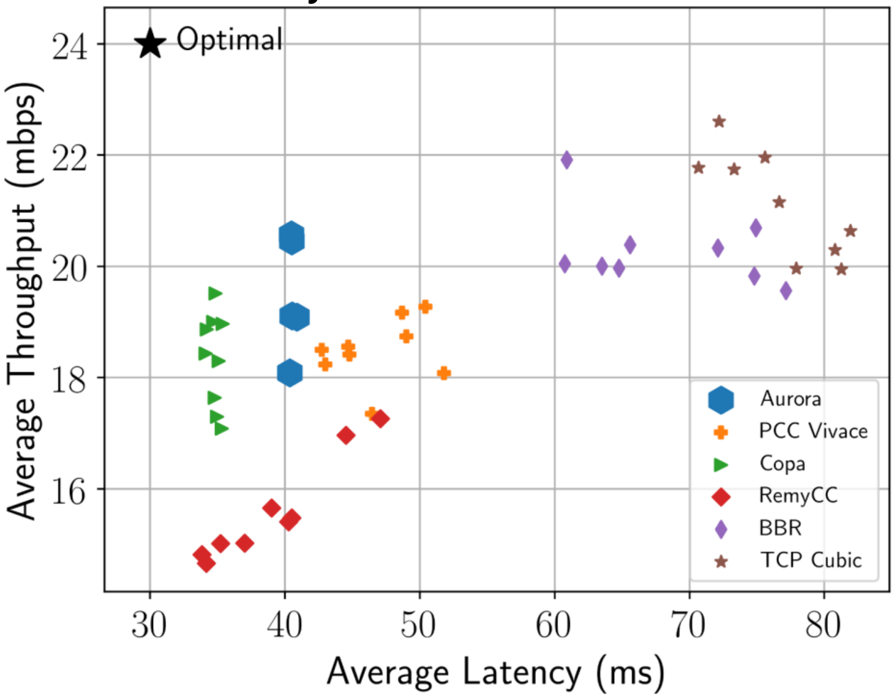


State-of-the-art Results

Test Description:

- Emulated network, with real Linux kernel noise
- Time-varying link

Emulated Dynamic Link Performance

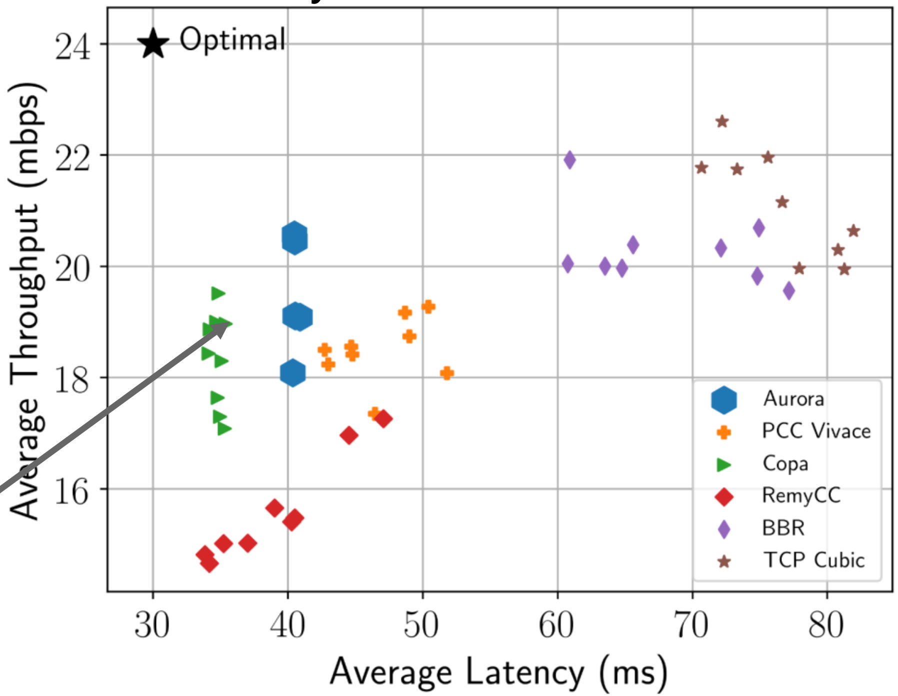


State-of-the-art Results

Test Description:

- Emulated network, with real Linux kernel noise
- Time-varying link

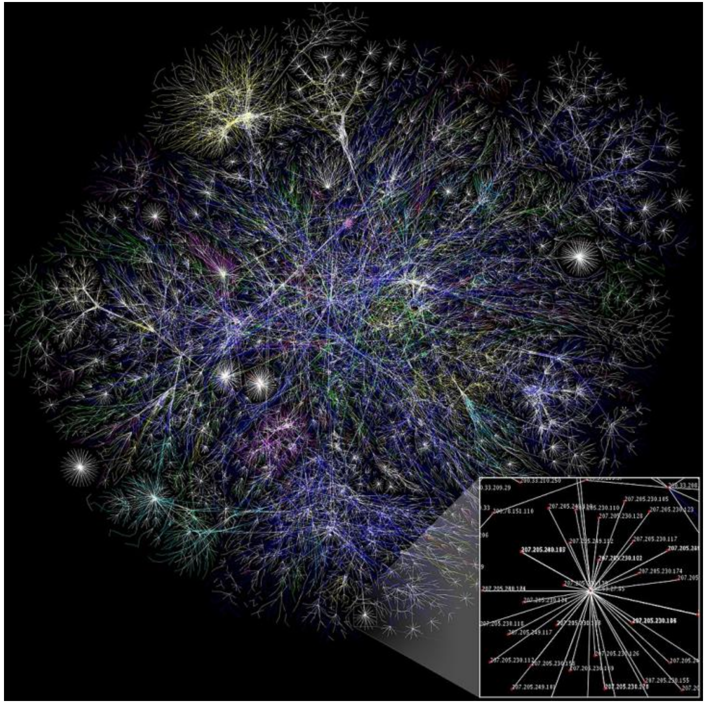
Emulated Dynamic Link Performance



Aurora is on the Pareto front of state-of-the-art algorithms

Exciting Directions

- Multi-agent scenarios:
  - Cooperative
  - Selfish
- Online training:
  - Few-shot training
  - Meta-learning
- Multi-objective Learning:
  - File transfer
  - Live video



See us at:

Poster #45
6:30pm - 9:00pm
Pacific Ballroom

Code available at
github.com/PCCProject/PCC-RL