Neural Adaptive Video Streaming with Pensieve

Hongzi Mao

Ravi Netravali Mohammad Alizadeh

Massachusetts Institute of Technology

CSAIL

Users start leaving if video doesn’t play in 2 seconds

https://gigaom.com/2012/11/09/online-viewers-start-leaving-if-video-doesnt-play-in-2-seconds-says-study/
Video: La Luna (Pixar 2011)

Dynamic Streaming over HTTP (DASH)

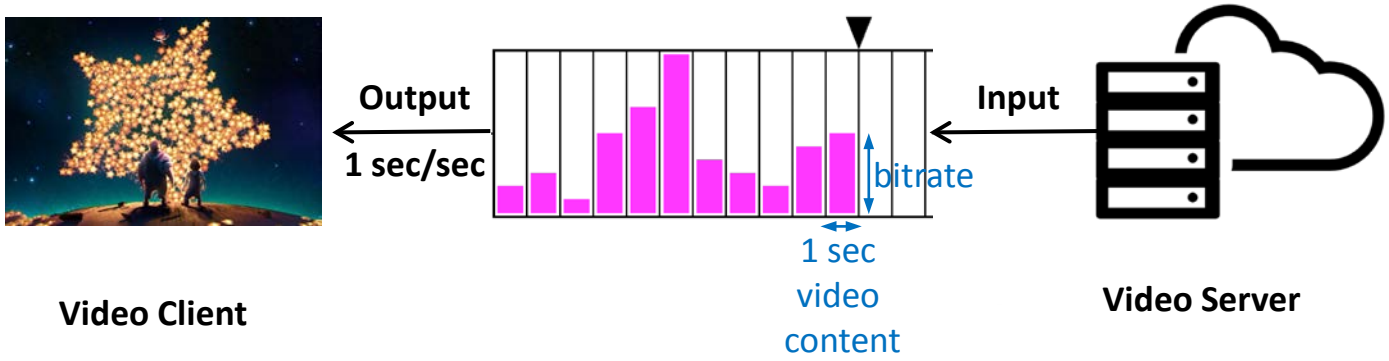
Video Client

Request:
next video chunk at bitrate *r*

Response:
video content

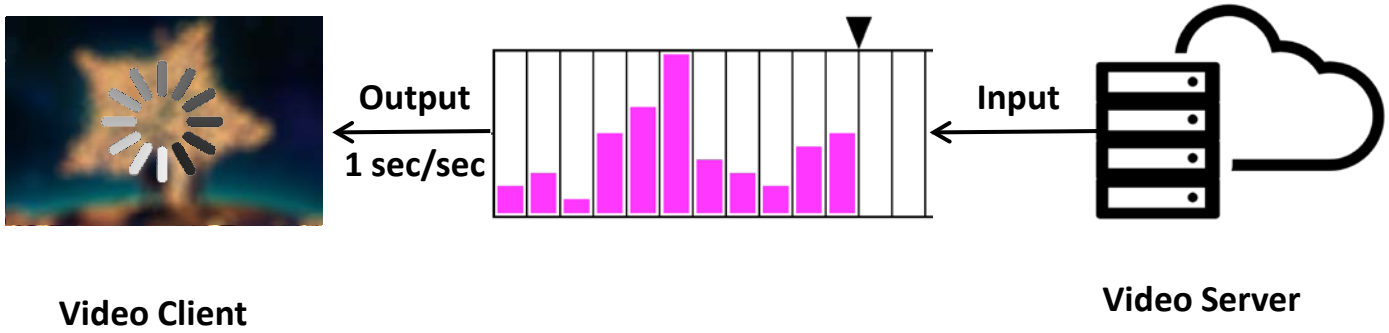
Video Server

Dynamic Streaming over HTTP (DASH)

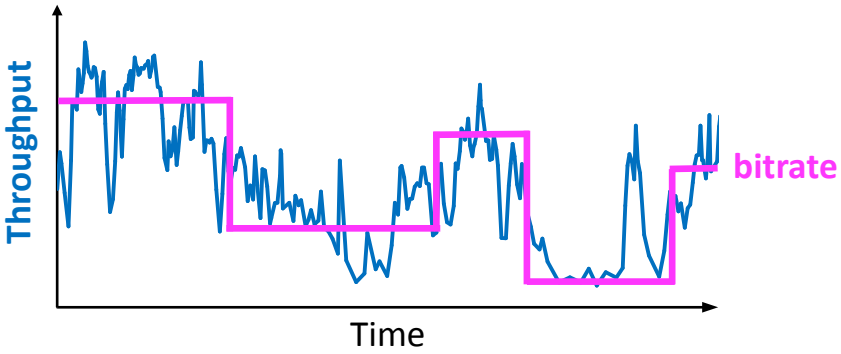


Animation borrowed from Te-Yuan Huang (SIGCOMM '14) http://conferences.sigcomm.org/sigcomm/2014/doc/slides/38.pdf

Dynamic Streaming over HTTP (DASH)

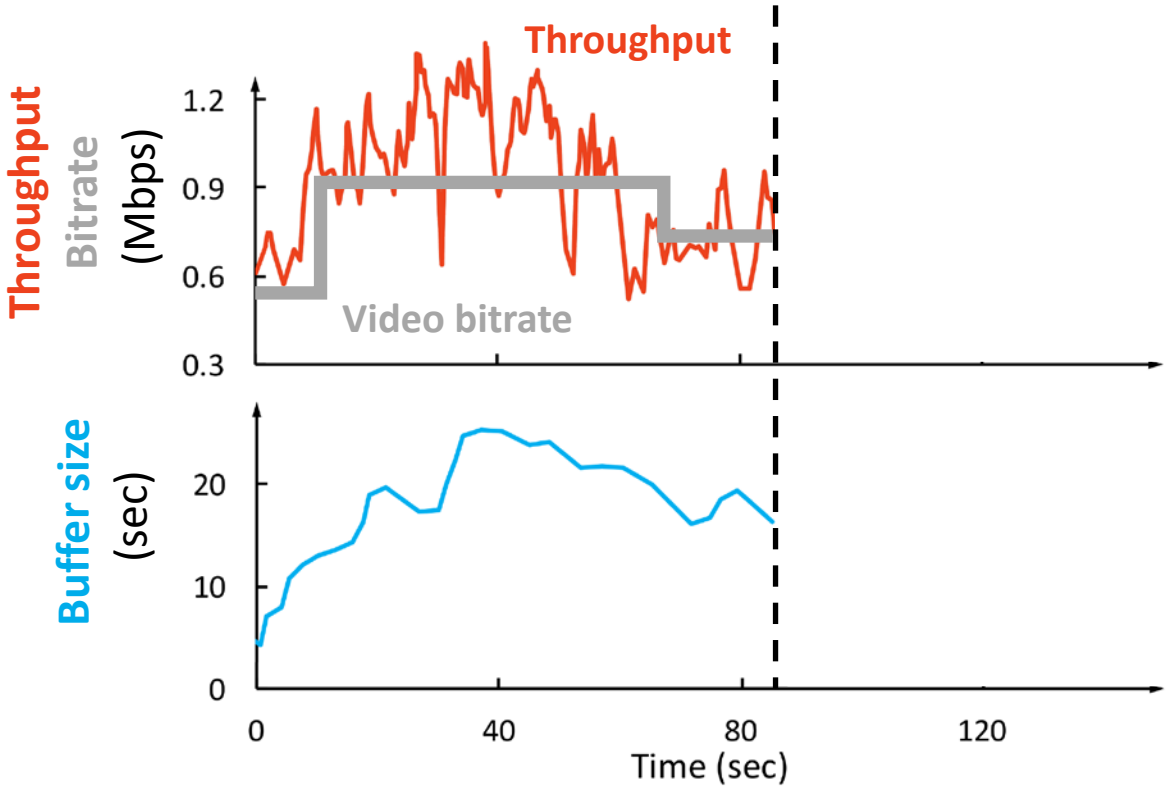


Adaptive Bitrate (ABR) Algorithms

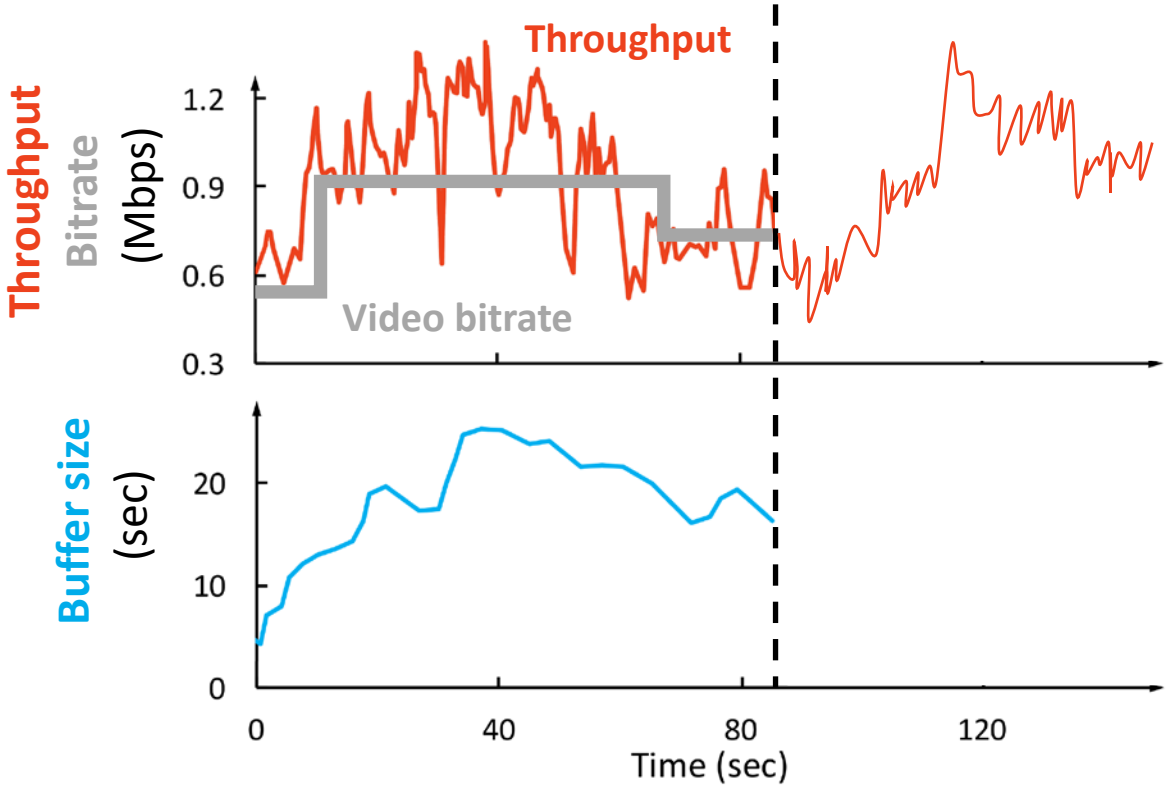


Animation borrowed from Te-Yuan Huang (SIGCOMM '14) http://conferences.sigcomm.org/sigcomm/2014/doc/slides/38.pdf

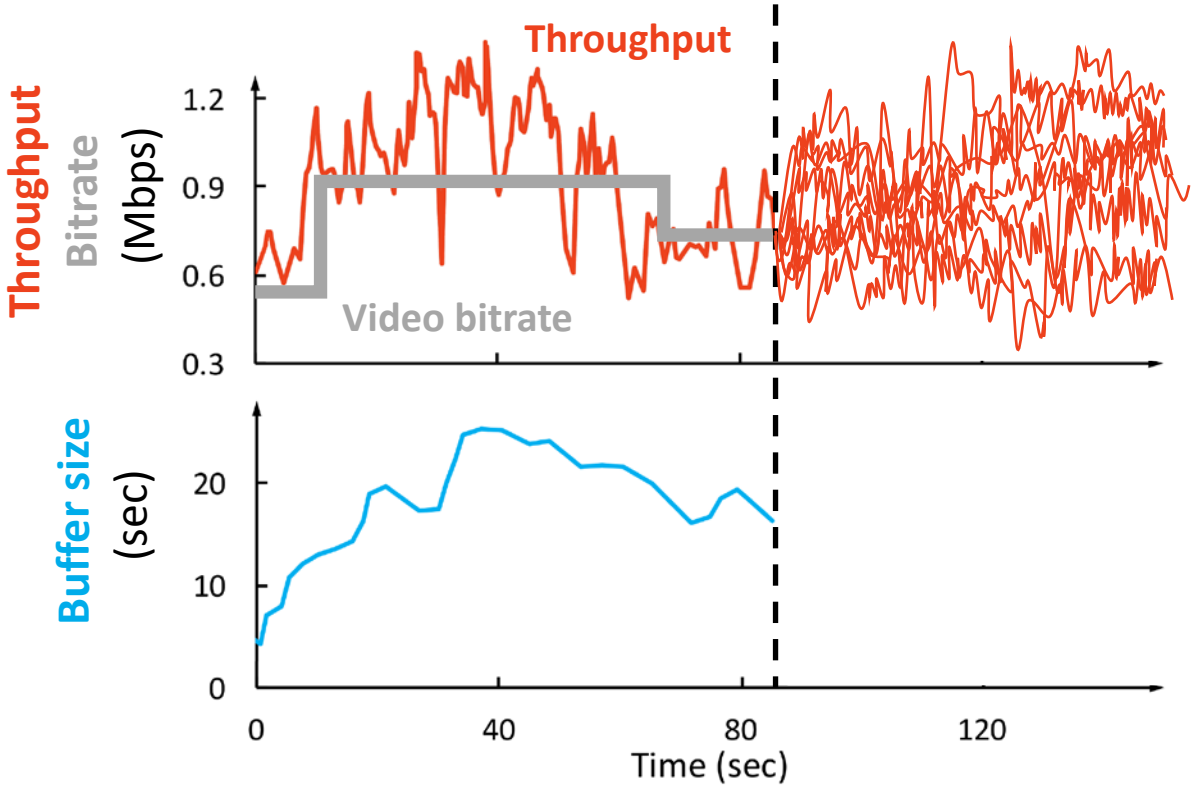
Why is ABR Challenging?



Why is ABR Challenging?

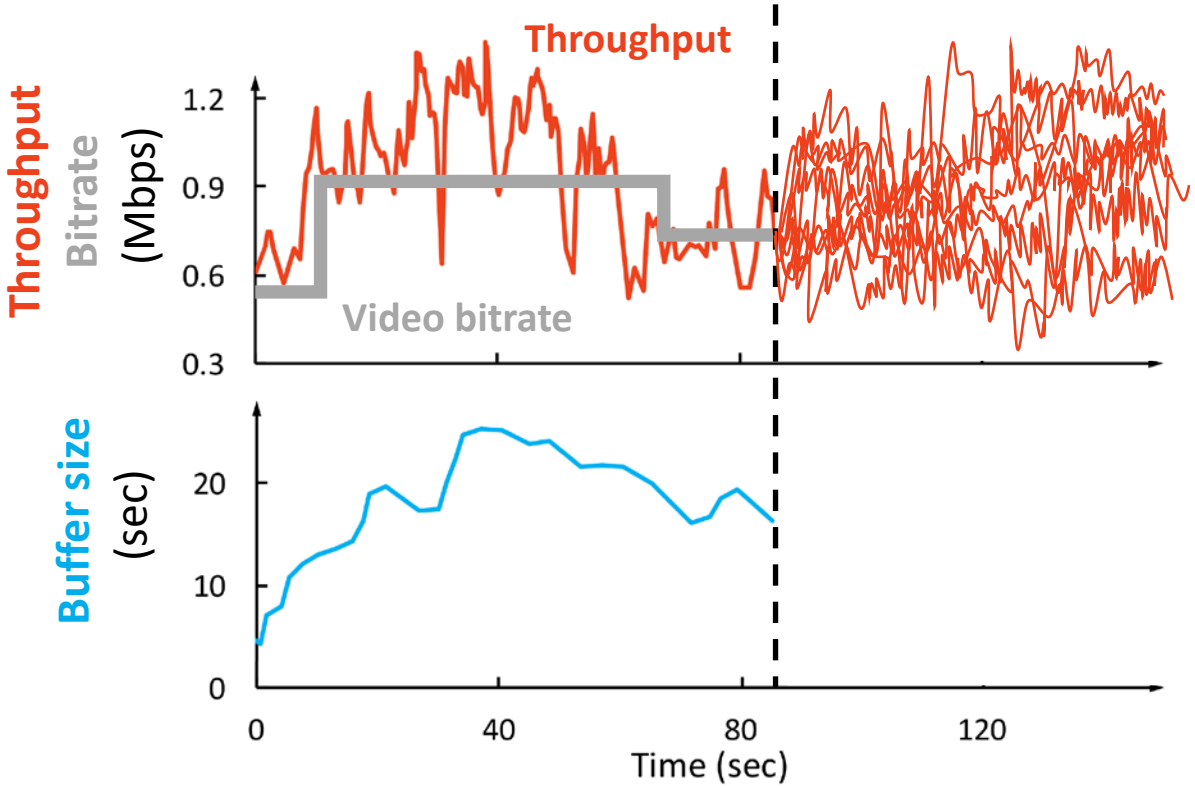


Why is ABR Challenging?



Network throughput is variable & uncertain

Why is ABR Challenging?

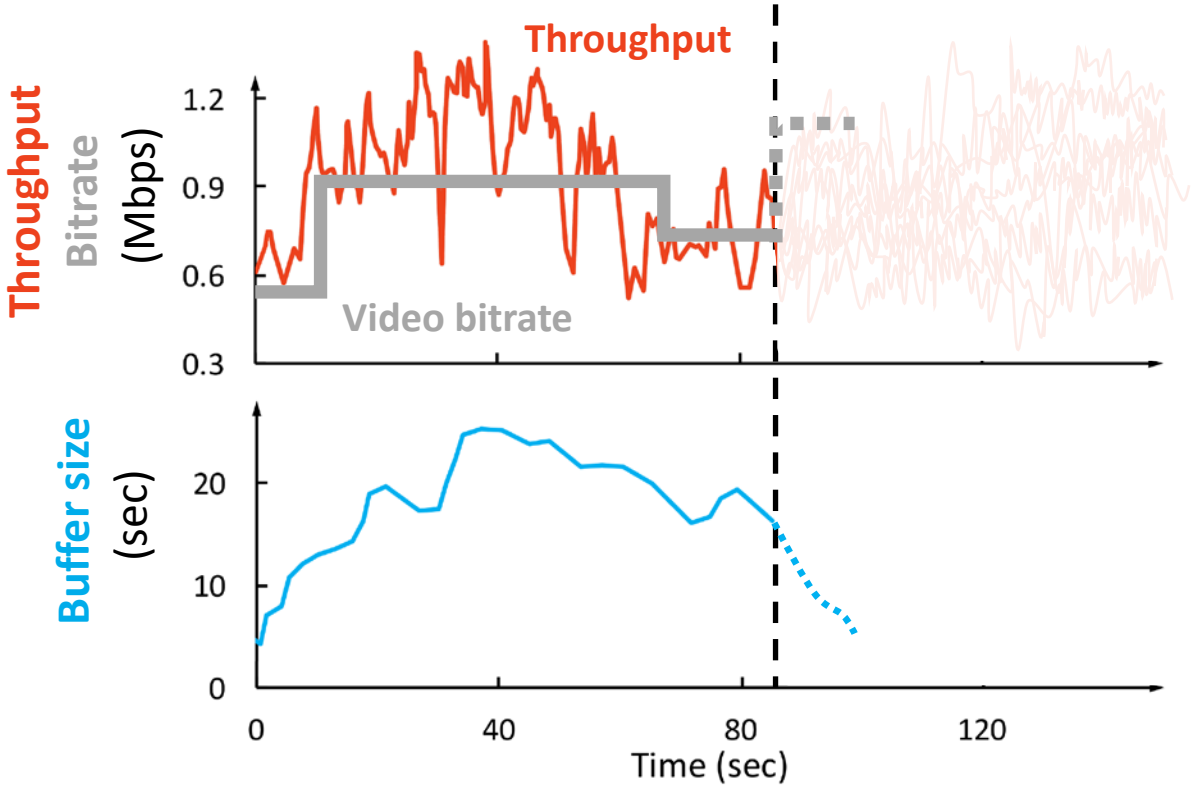


Network throughput is variable & uncertain

Conflicting QoE goals

- Bitrate
- Rebuffering time
- Smoothness

Why is ABR Challenging?



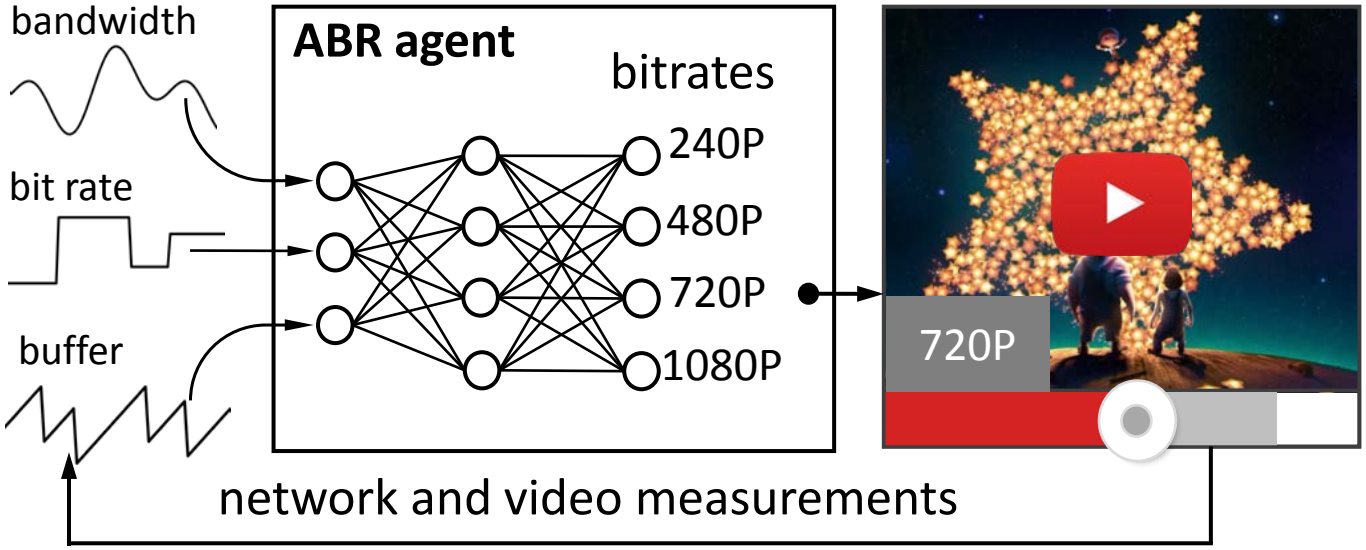
Network throughput is variable & uncertain

Conflicting QoE goals

- Bitrate
- Rebuffering time
- Smoothness

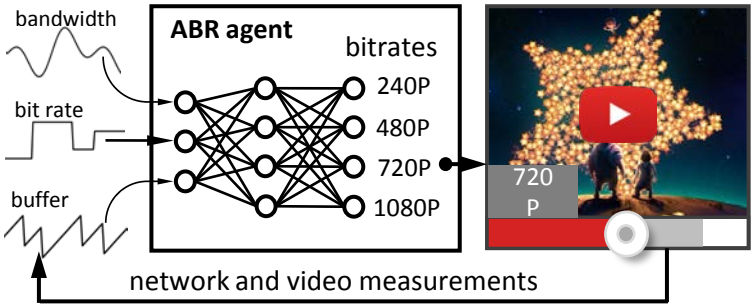
Cascading effects of decisions

Our Contribution: Pensieve



Pensieve learns ABR algorithm automatically through experience

Our Contribution: Pensieve



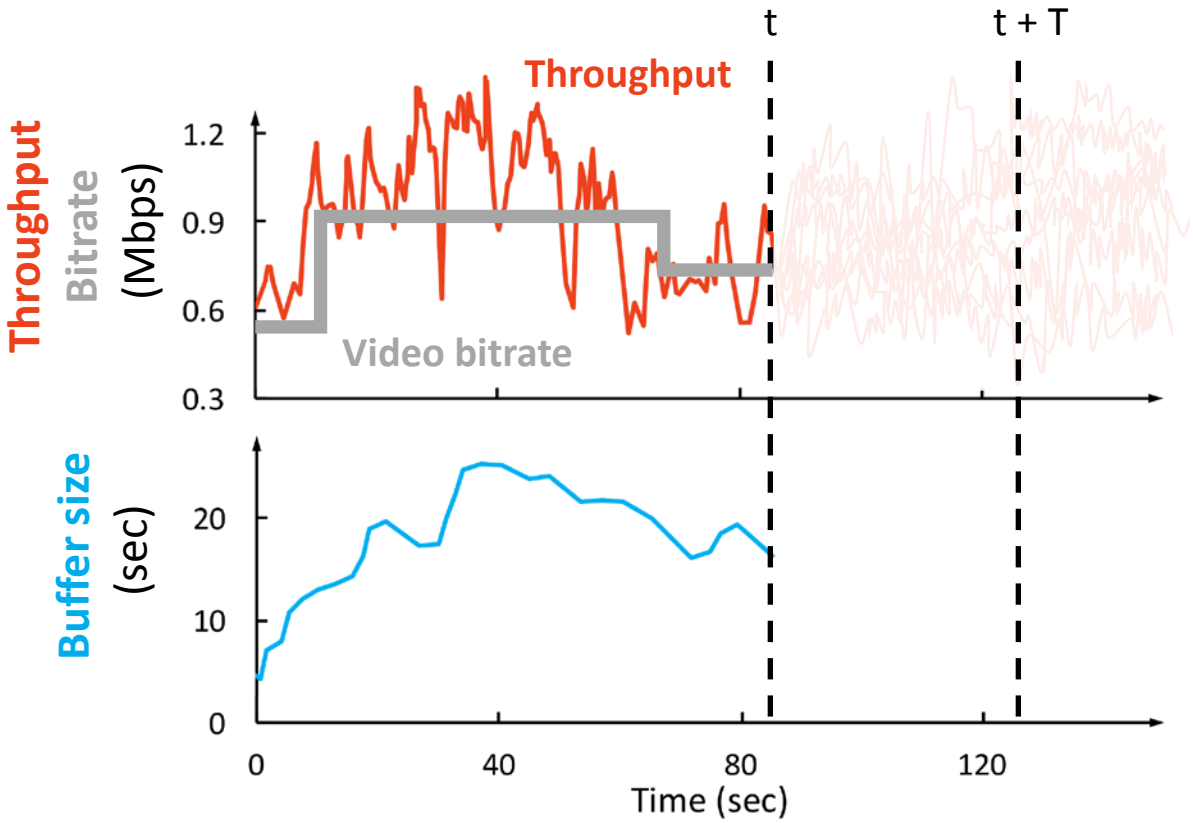
1. First network control system using modern “deep” reinforcement learning
2. Delivers 12-25% better QoE, with 10-30% less rebuffering than previous ABR algorithms
3. Tailors ABR decisions for different network conditions in a data-driven way

Previous Fixed ABR Algorithms

- Rate-based: pick bitrate based on predicted throughput
  - FESTIVE [CoNEXT'12], PANDA [JSAC'14], CS2P [SIGCOMM'16]
- Buffer-based: pick bitrate based on buffer occupancy
  - BBA [SIGCOMM'14], BOLA [INFOCOM'16]
- Hybrid: use both throughput prediction & buffer occupancy
  - PBA [HotMobile'15], MPC [SIGCOMM'15]

Simplified inaccurate model leads to suboptimal performance

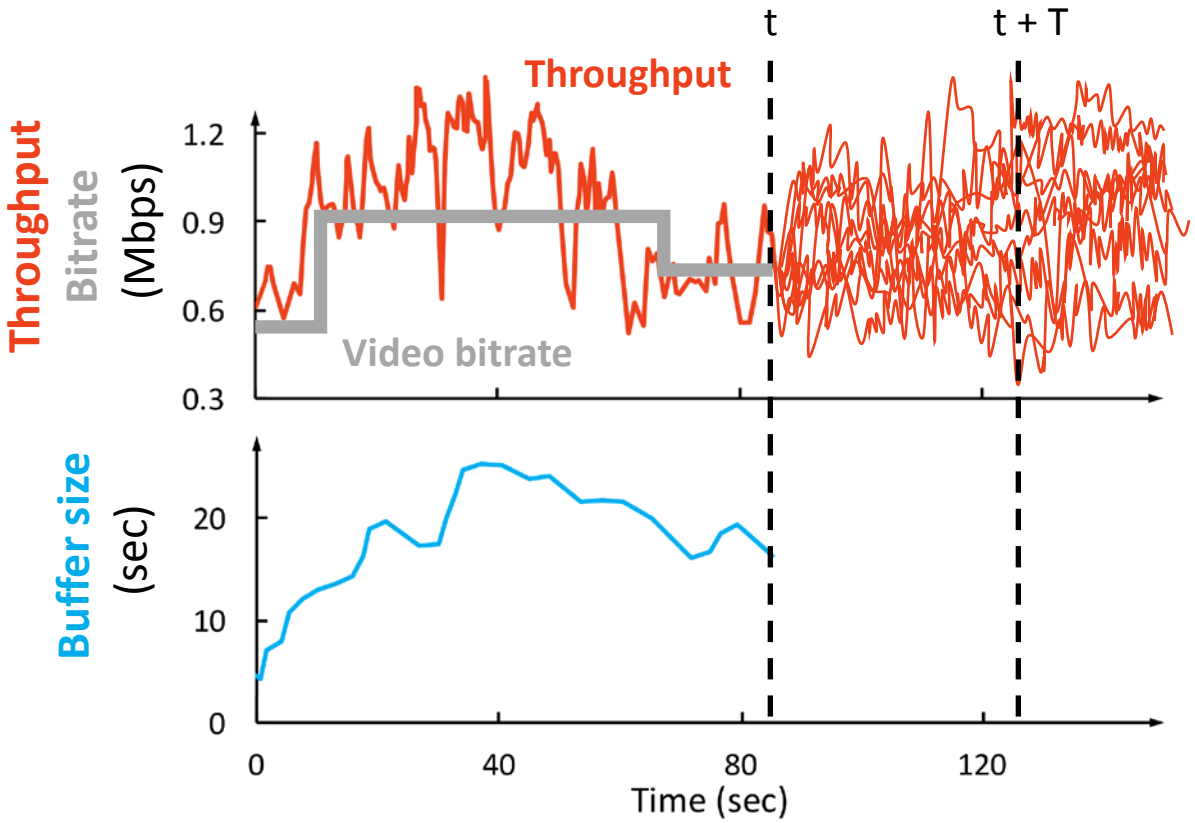
Example: Model Predictive Control



maximize QoE (t, t + T)
subject to system dynamics

Problem: Needs accurate throughput model

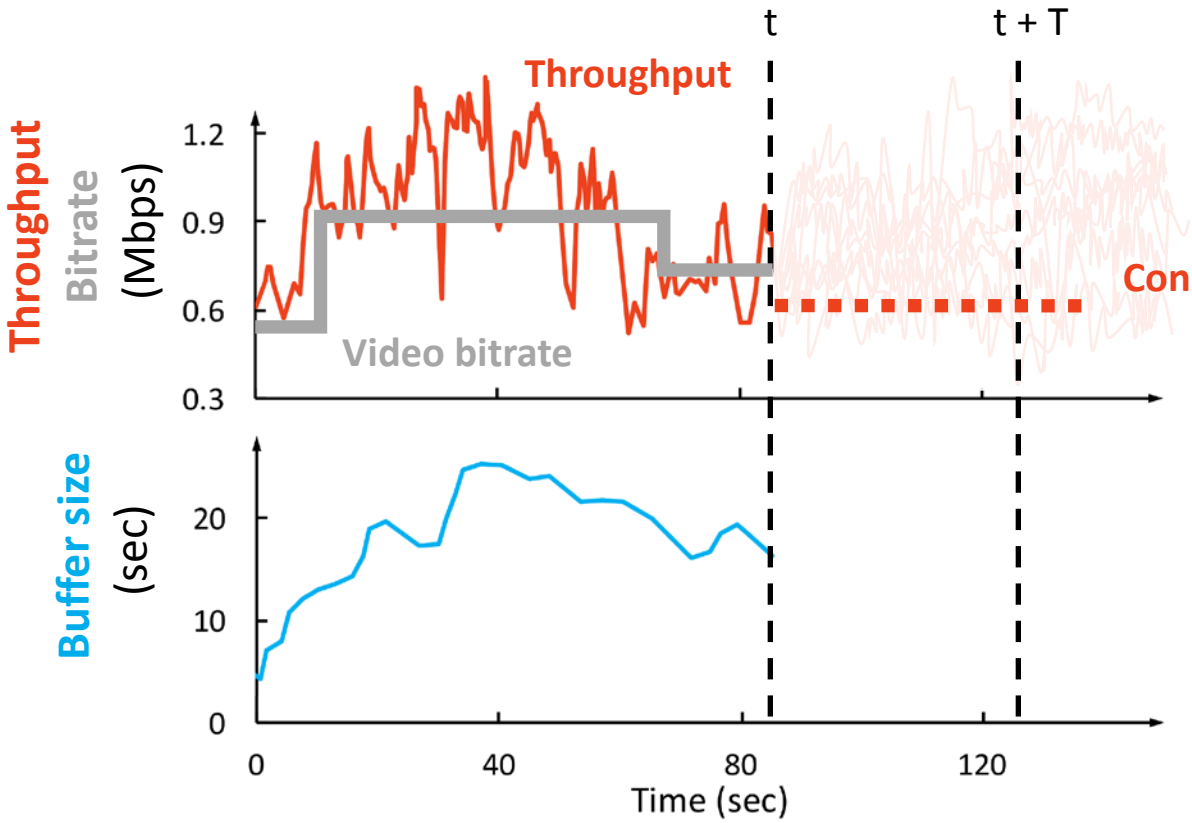
Example: Model Predictive Control



maximize QoE (t, t + T)
subject to system dynamics

Problem: Needs accurate throughput model

Example: Model Predictive Control



maximize QoE (t, t + T)
subject to system dynamics

Conservative Throughput Prediction

Problem: Needs accurate throughput model

Example: Model Predictive Control

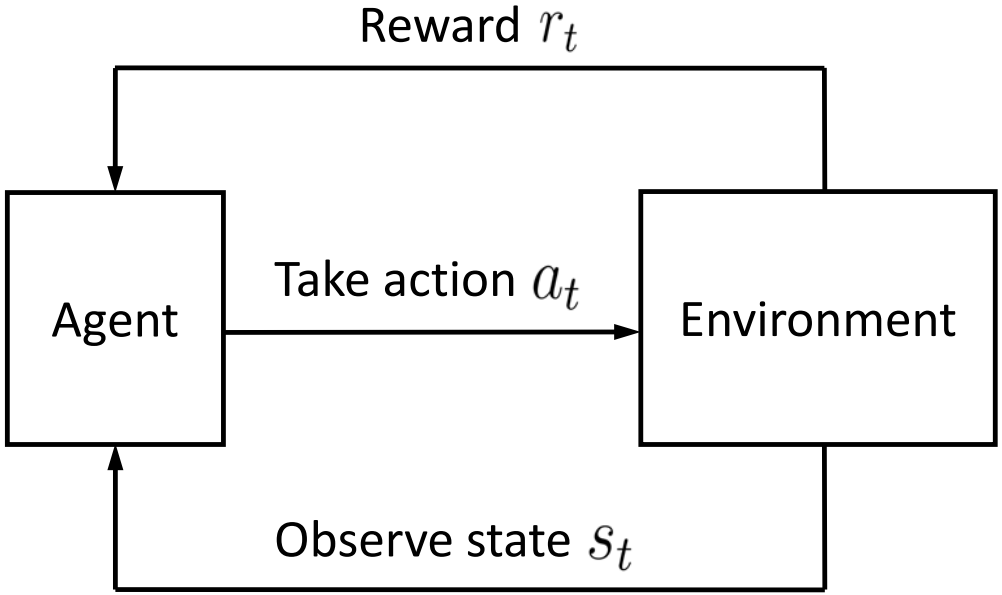
maximize QoE (t, t + T)
subject to system dynamics

Prediction

Problem: Needs accurate throughput model

Solution: learn from video streaming sessions in actual network conditions

Reinforcement Learning



Goal: maximize the cumulative reward $\sum_t r_t$

Pensieve Design

Reward r_t

$$+ q(b_t) - \mu T_t - \lambda |q(b_t) - q(b_{t-1})|$$

+ (bitrate) - (rebuffering) - (smoothness)

State s_t

Past chunk throughput

x_t x_{t-1} • • • x_{t-k+1}

Past chunk download time

τ_t τ_{t-1} • • • τ_{t-k+1}

Next chunk sizes

n_1 n_2 • • • n_m

Current buffer size s_t

Remaining chunks c_t

Past chunk bitrate b_t

Agent

240P

360P

720P

1080P

Action a_t

Pensieve Design

Reward r_t

$$+ q(b_t) - \mu T_t - \lambda |q(b_t) - q(b_{t-1})|$$

+ (bitrate) - (rebuffering) - (smoothness)

State s_t

Past chunk throughput

x_t x_{t-1} ••• x_{t-k+1}

Past chunk download time

τ_t τ_{t-1} ••• τ_{t-k+1}

Next chunk sizes

n_1 n_2 ••• n_m

Current buffer size s_t

Remaining chunks c_t

Past chunk bitrate b_t

Agent

1D-CNN

1D-CNN

1D-CNN

240P

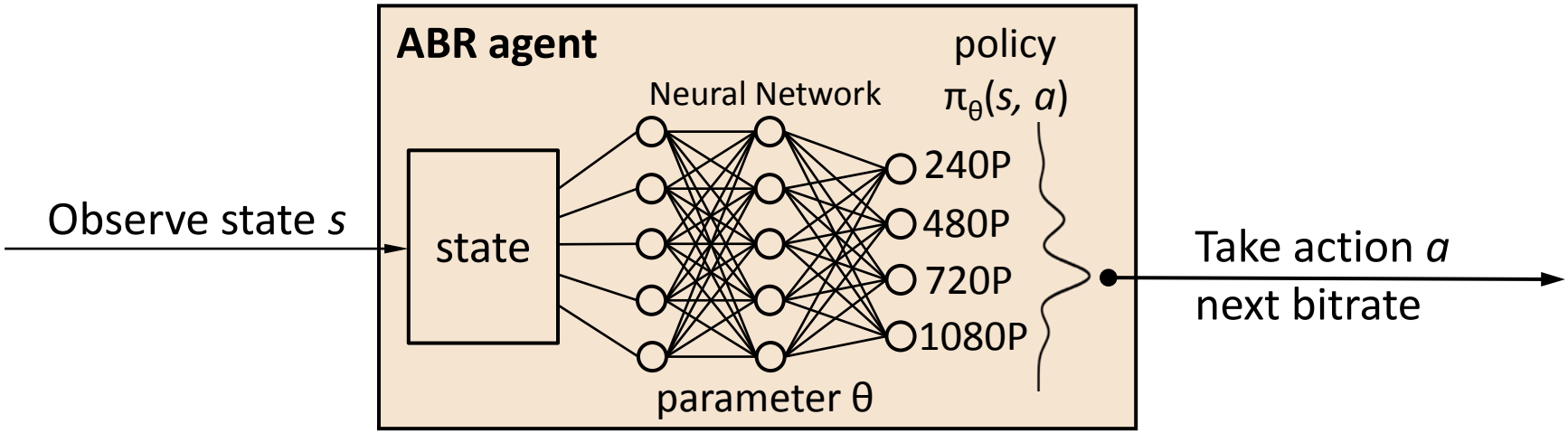
360P

720P

1080P

Action a_t

How to Train the ABR Agent



Collect experience data: trajectory of [state, action, reward]

Training: $\theta \leftarrow \theta + \alpha \nabla_\theta \mathbb{E}_{\pi_\theta} \left[\sum_t r_t \right]$

What Pensieve is good at

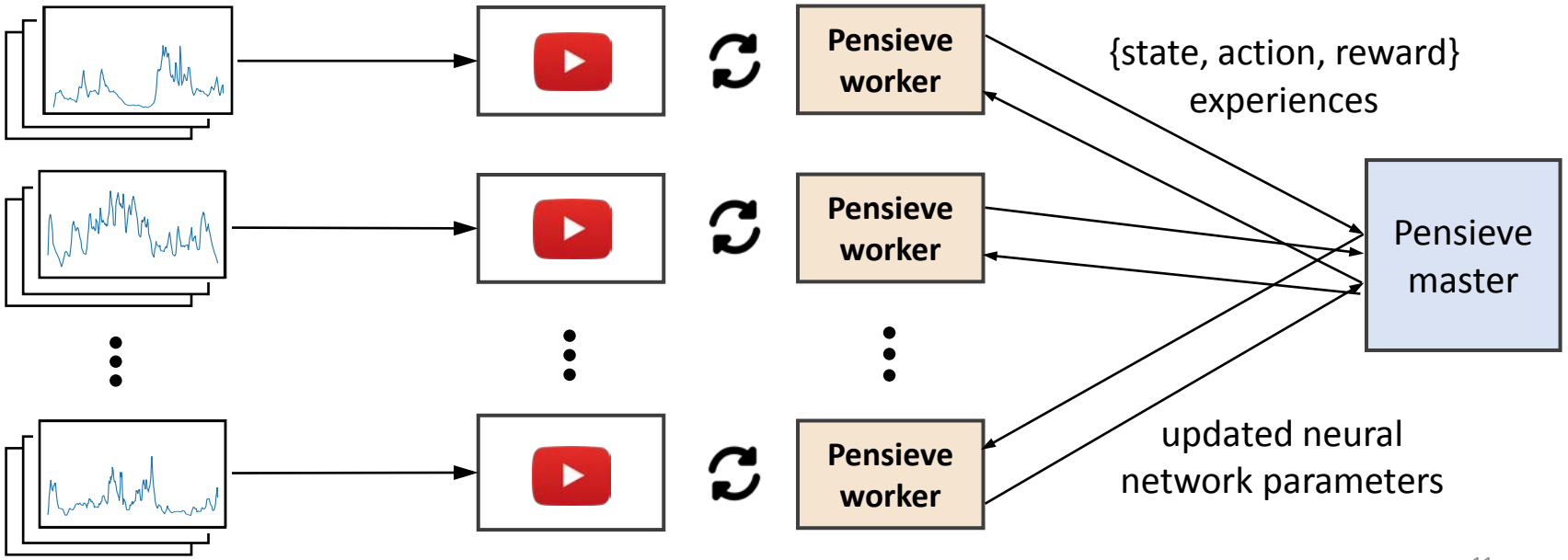
- Learn the dynamics **directly from experience**
- Optimize the high level QoE objective **end-to-end**
- Extract control rules from **raw high-dimensional** signals

Pensieve Training System

Large corpus of network traces
cellular, broadband, synthetic

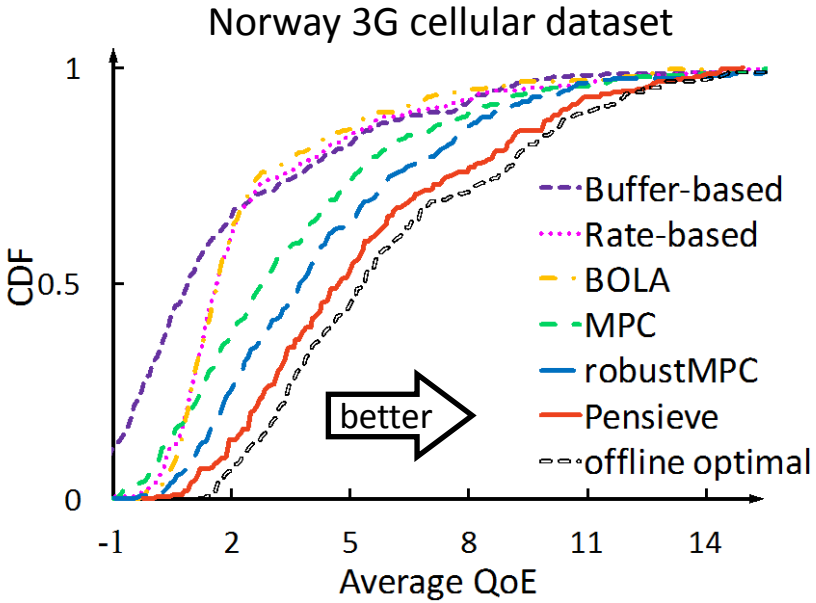
Video playback
Fast chunk-level simulator

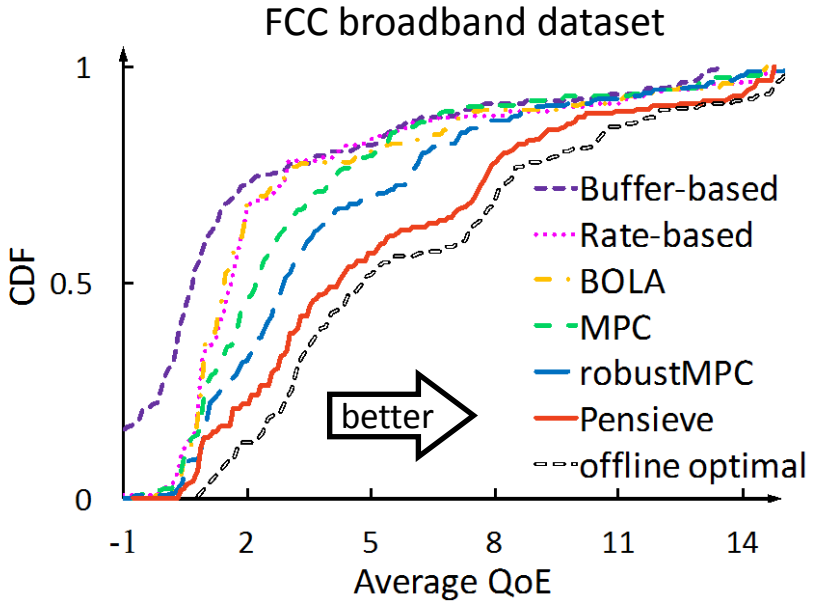
Model update
TensorFlow



Trace-driven Evaluation

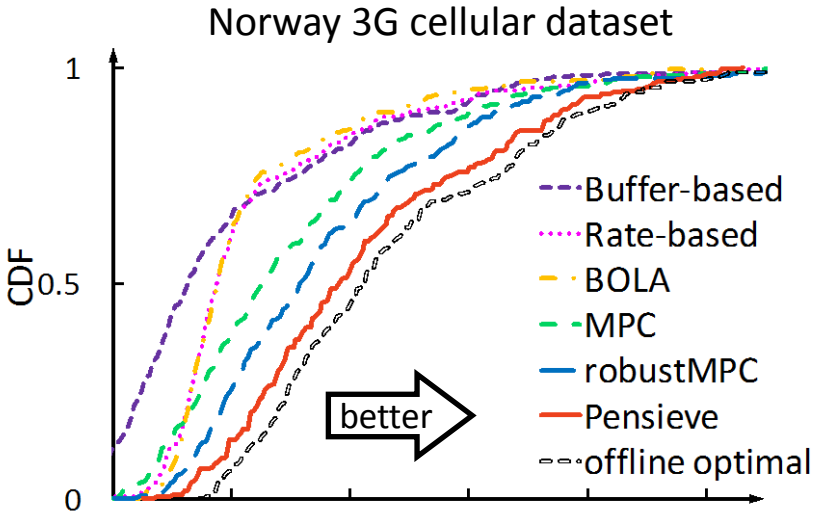
- **Dataset:** Two datasets, each dataset consists of 1000 traces, each trace 320 seconds.
- **Video:** 193 seconds. encoded at bitrates: {300, 750, 1200, 1850, 2850, 4300} kbps.
- **Video player:** Google Chrome browser **Video server:** Apache server

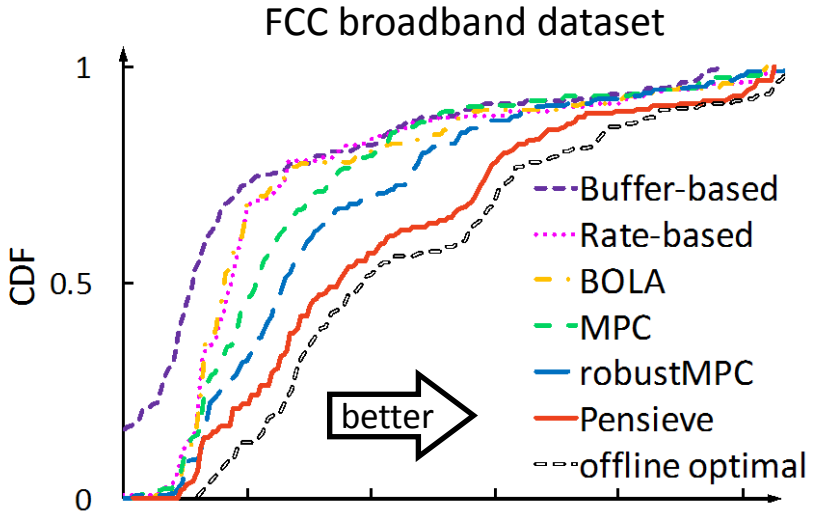




Trace-driven Evaluation

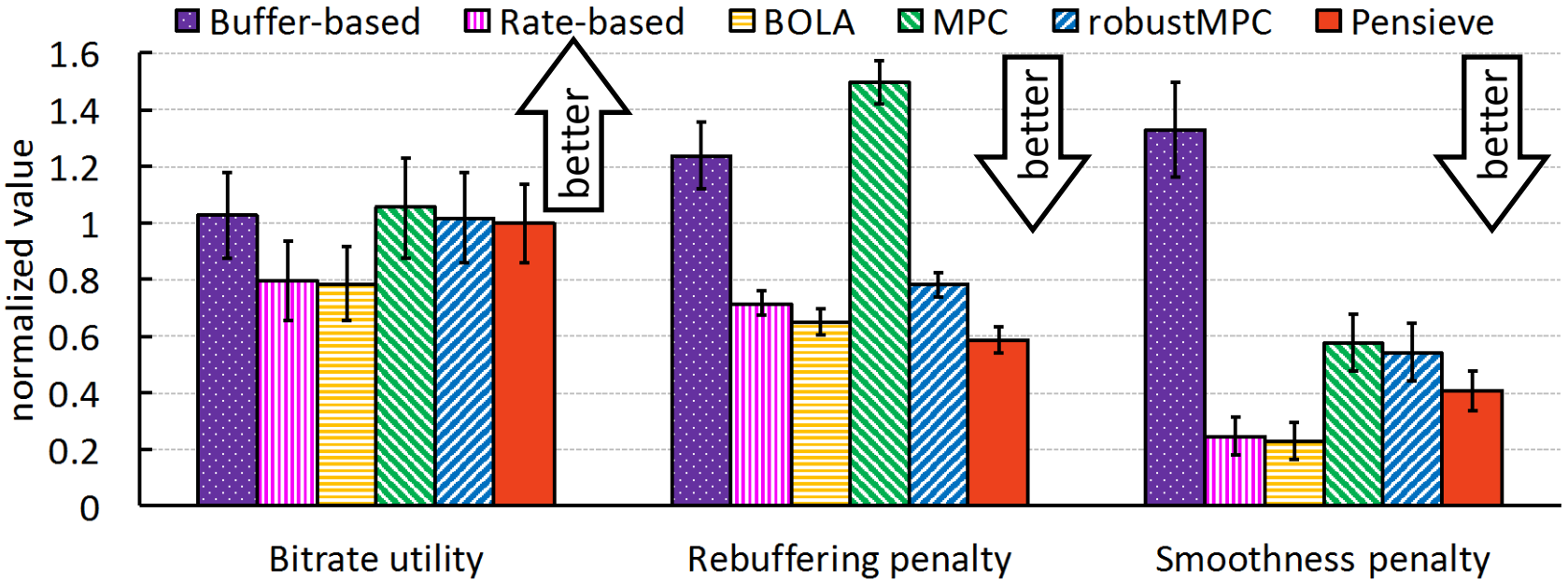
- **Dataset:** Two datasets, each dataset consists of 1000 traces, each trace 320 seconds.
- **Video:** 193 seconds. encoded at bitrates: {300, 750, 1200, 1850, 2850, 4300} kbps.
- **Video player:** Google Chrome browser **Video server:** Apache server





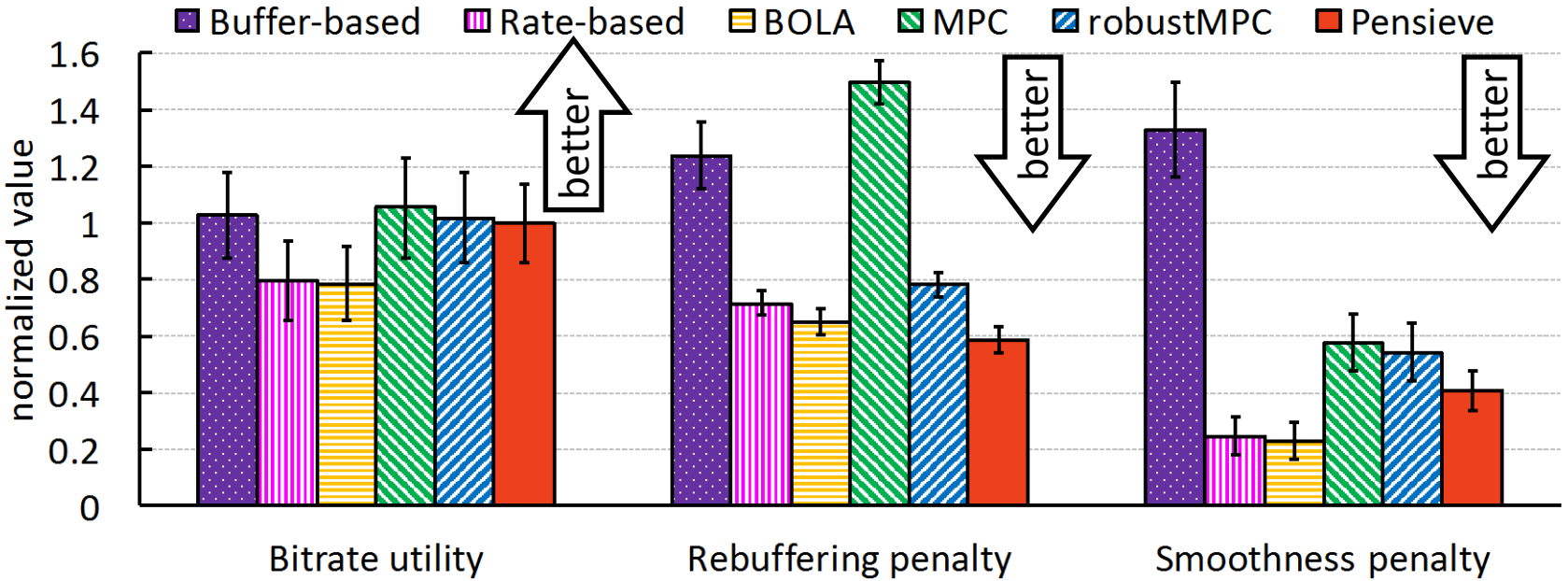
Pensieve improves the best previous scheme by 12-25%
and is within 9-14% of the offline optimal

QoE Breakdown



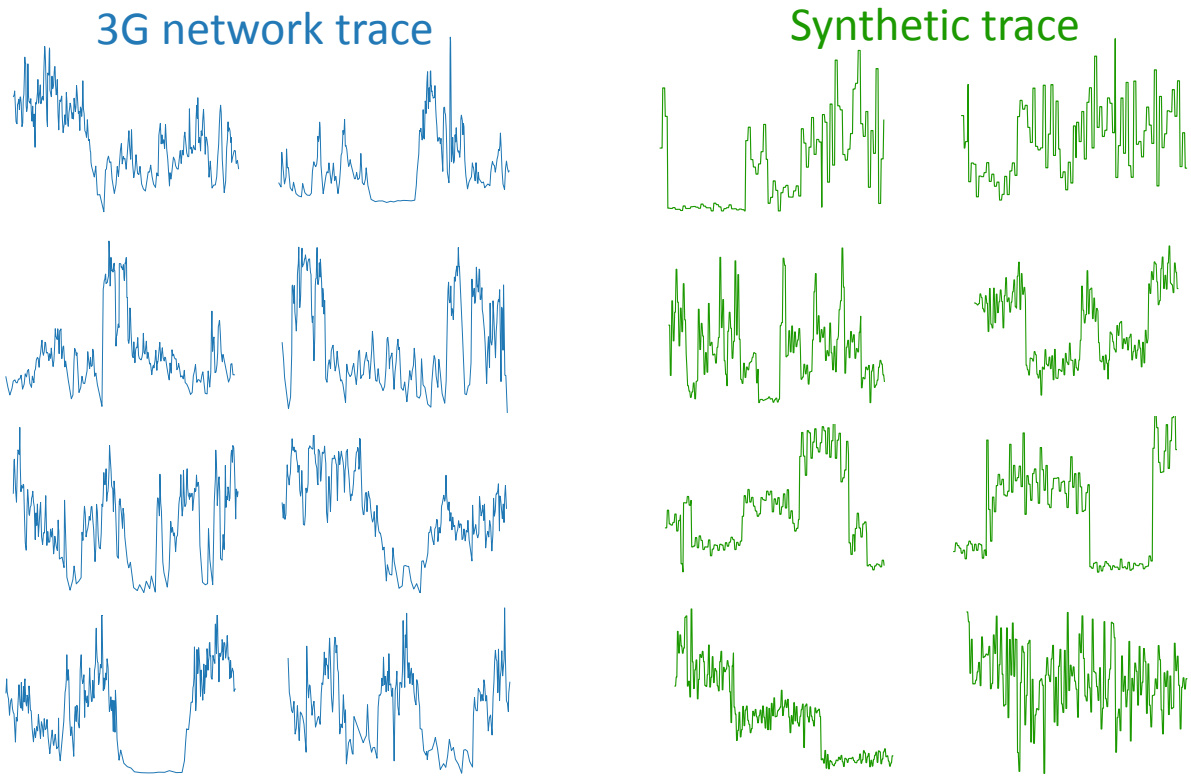
Reward/QoE $\sim$ + Bitrate utility – rebuffering penalty – smooth penalty

QoE Breakdown



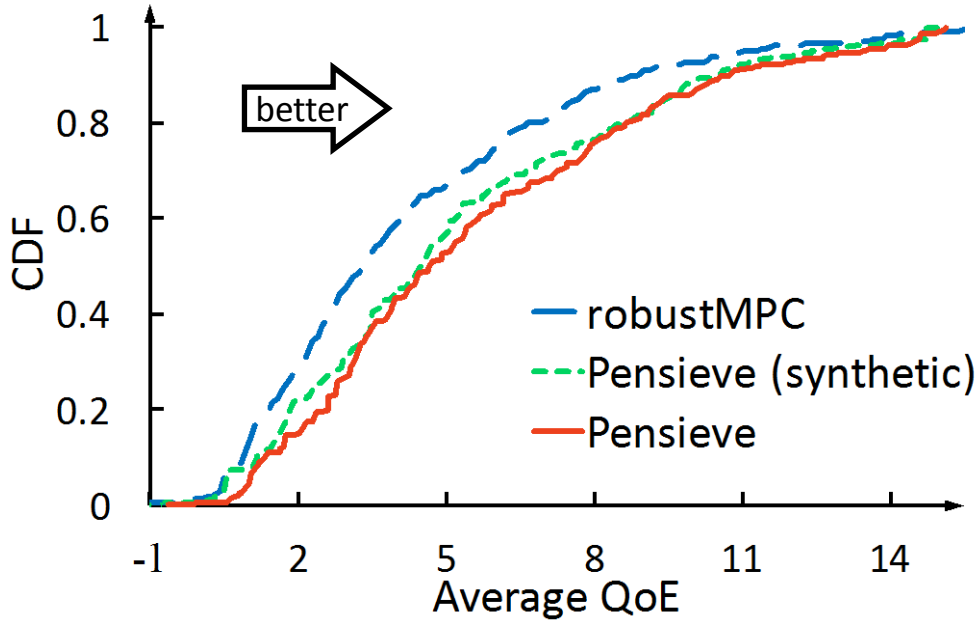
Pensieve reduces rebuffering by 10-32% over second best algorithm

Does Pensieve Generalize?



- Synthetic trace
- Covers a wide range of average throughput and network variation

Does Pensieve Generalize?



Train on synthetic traces then test on real 3G network trace

Only 5% degradation compared with Pensieve trained on real network trace

Summary

- Pensieve uses Reinforcement Learning to generate ABR algorithms
- Pensieve optimizes different network conditions through experience
- Pensieve outperforms existing approaches across a wide range of network environments and QoE preferences
- Policies generated by Pensieve have strong ability to generalize

http://web.mit.edu/pensieve/

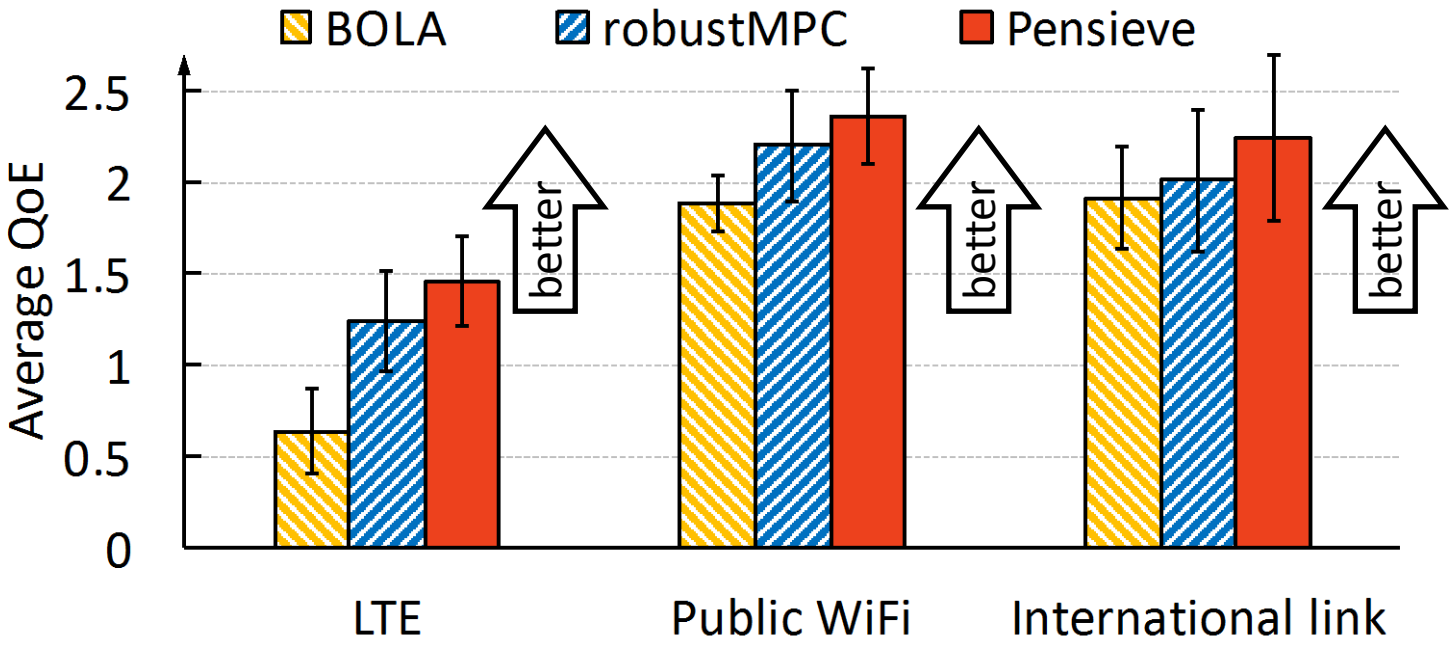
My thoughts

- In addition to heuristics, reinforcement learning based on neural networks provides a new way to solve network problems.

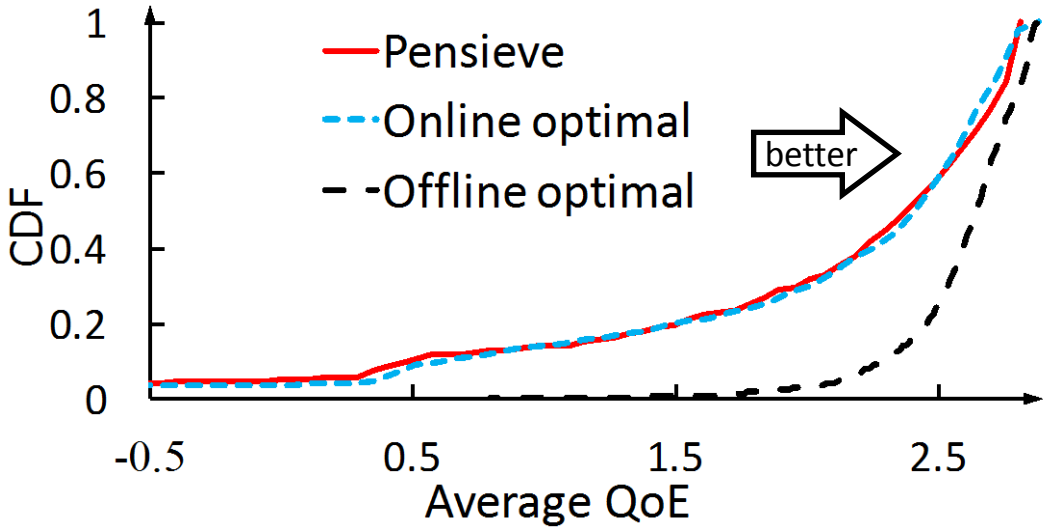
- Neural networks trained with synthetic data in a simulation environment may also have good performance, as long as the data amount and diversity is enough.

Thank you.

Experiments in the Wild



How Close is Pensieve to Optimal?



- Simulate chunk download time $T_n = T_{n-1} (R_n / R_{n-1}) + \varepsilon$
- Use dynamic programming to obtain online/offline optimal
- Train a Pensieve agent with exactly the same setting as before

Multi-video extension