mhv/2023

Towards Algorithmic Foundations for Customizable Video Delivery

Michael Schapira (Compira Labs)
Eden Zerbib-Itah (Compira Labs)

European Innovation Council

Compira Labs

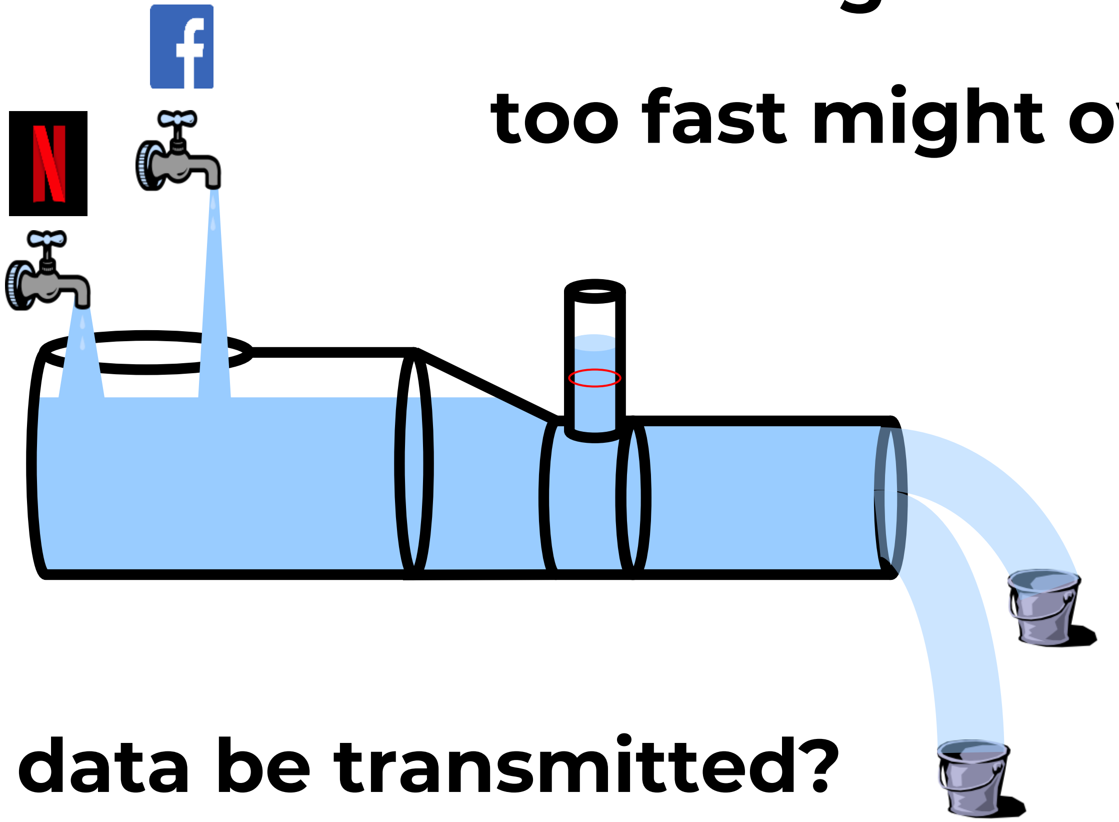
Bad content delivery impacts user experience

- Poor picture resolution
- High latency / delay ('time behind live')
- Re-buffering
- Long startup time

Internet Congestion Control (CC)

too slow might harm performance

too fast might overflow the pipe



How fast should data be transmitted?

The limitations of one-size-fits-all CC

- Congestion control **logic** is traditionally oblivious to both
  - the service-specific QoE requirements
  - the prevailing network conditions wrt different users

- Congestion control algorithms are expected to perform well across a daunting breadth of application domains and networks.

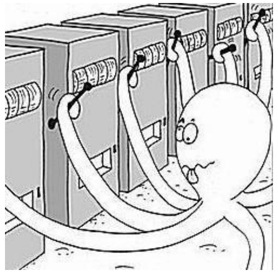
- No universal CC logic can optimize performance across all networks and performance metrics.

Warmup: a simple scenario

- A single traffic sender is **repeatedly** sending traffic across a single link
- The link has certain characteristics
  - BW, latency, non-congestion loss, buffer size
- We wish to learn which of a set of possible CC configurations is optimal for the sender.

mhv/2023

A multi-armed bandit problems!



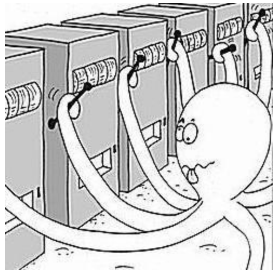
- A decision maker (agent) repeatedly chooses one of n actions
- After each choice a_t, a reward r_t is observed
- Let $E\langle r_t \mid a_t \rangle = Q^*(a_t)$
- The objective is to maximize the long-term reward

- CC configuration learning as multi-armed bandit task
  - agent = sender
  - action = CC configuration
  - reward = performance score

European Innovation Council

CompiraLabs

A multi-armed bandit problems!



- Had we known $E\langle r_t \mid a_t \rangle = Q^*(a_t)$ in advance, we would have simply chosen the best action (CC configuration).

- To solve the multi-armed bandit problem, you must explore a variety of actions and then exploit the best of them.
  - A delicate tradeoff…

- Different algorithmic approaches
  - ε-Greedy, softmax, UCB

Simple Lab Experiments

- Single sender on a single link, emulated using mininet.
- 4 candidate CC configurations.
- A CC configuration is learned over time through the sender's repeated interaction with the link.

- We vary
  - the link parameters
  - the performance metric
  - the CE configurations

Experiment 1

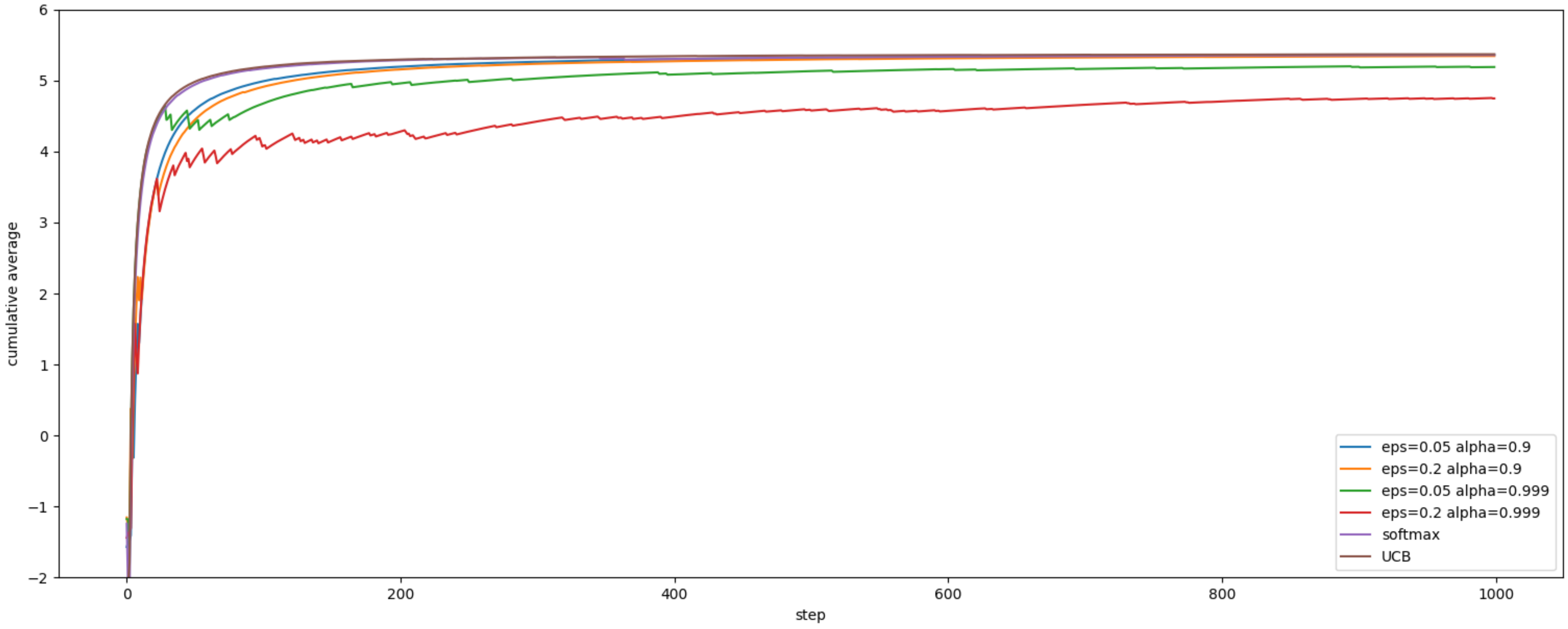
- **Link**: BW=60Mbps, latency=30ms, 1 BDP buffer, random loss = 0
- Performance metric:

$$\frac{sum-delivered}{duration} - \alpha \times \max\{loss-0.01, 0\} - \beta \frac{\max\_RTT}{\min\_RTT}$$

PCC configuration (slow start, loss penalty, latency penalty)	Average performance score
(1.92,2,1)	1.77
(1.92,2,2)	1.47
(1.92,2,3)	1.68
(1.92,2,5)	**5.4**

mhv/2023

Experiment 1: Results



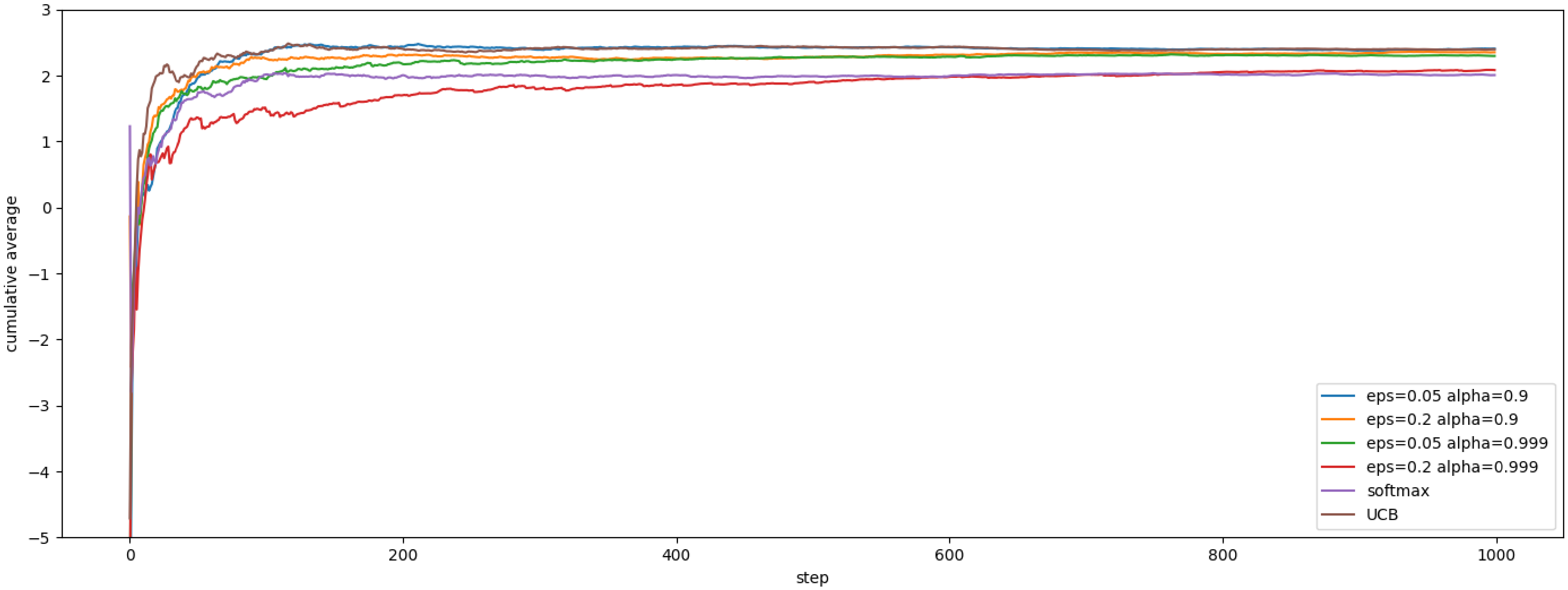
Experiment 2

- **Link**: BW=60Mbps, latency=30ms, 1 BDP buffer, **random loss = 3%**
- Performance metric:

$$\frac{sum-delivered}{duration} - \alpha \times \max\{loss-0.03, 0\} - \beta \times \max\{\frac{\max\_RTT}{\min\_RTT} - 5,\ 0\}$$

PCC configuration (slow start, loss penalty, latency penalty)	Average performance score
(1.92,0,0.5)	-5.8
(1.92,1,0.5)	0.17
(1.92,5,0.5)	1.24
(1.92,20,0.5)	**2.27**

Experiment 2: Results



Generalizing from the simple scenario

- **Temporal** patterns in traffic (and competition)!
- How can configurations be **safely** probed?
- How can we quantify performance in terms of what we actually care about (**video QoE**)?
- **Where** should customization be performed?
- At what **granularity**?

Conclusion and next steps

- Customizing CC to both the service and the network is key to improving over today's one-size-fits-all CC

- This can have significant implications for video QoE in the field (stay tuned)

- We are currently taking our first steps towards realizing this broad agenda.

Thanks!