

A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service

Te-Yuan (TY) Huang
Stanford / Netflix
Aug 19th, 2014

Joint work with Nick McKeown, Ramesh Johari,
Matt Trunnell, and Mark Watson

Video Streaming Services are Popular



8+ billion hours streamed per month globally

Video is 50+% of peak traffic in the US

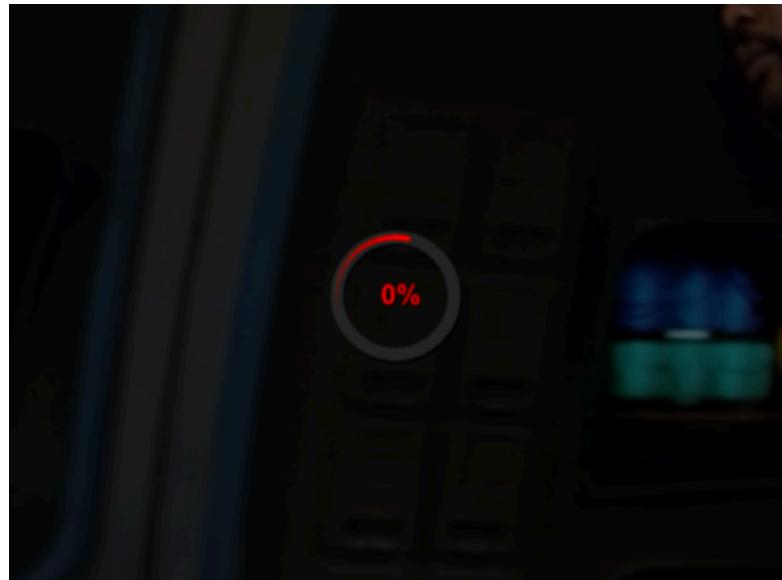
Streaming experience could be better

Problem #1: Too many rebuffers

Rebuffers means interruption

Definition: A rebuffer is **unnecessary** if:

- Picking a lower video rate would have avoided it
- 20 – 30% of rebuffers are unnecessary



Problem #2: Video rate is too low

Video rate represents video quality

Video rate is **too low**

- When network could sustain a higher video rate



Picked the Wrong Video Rate

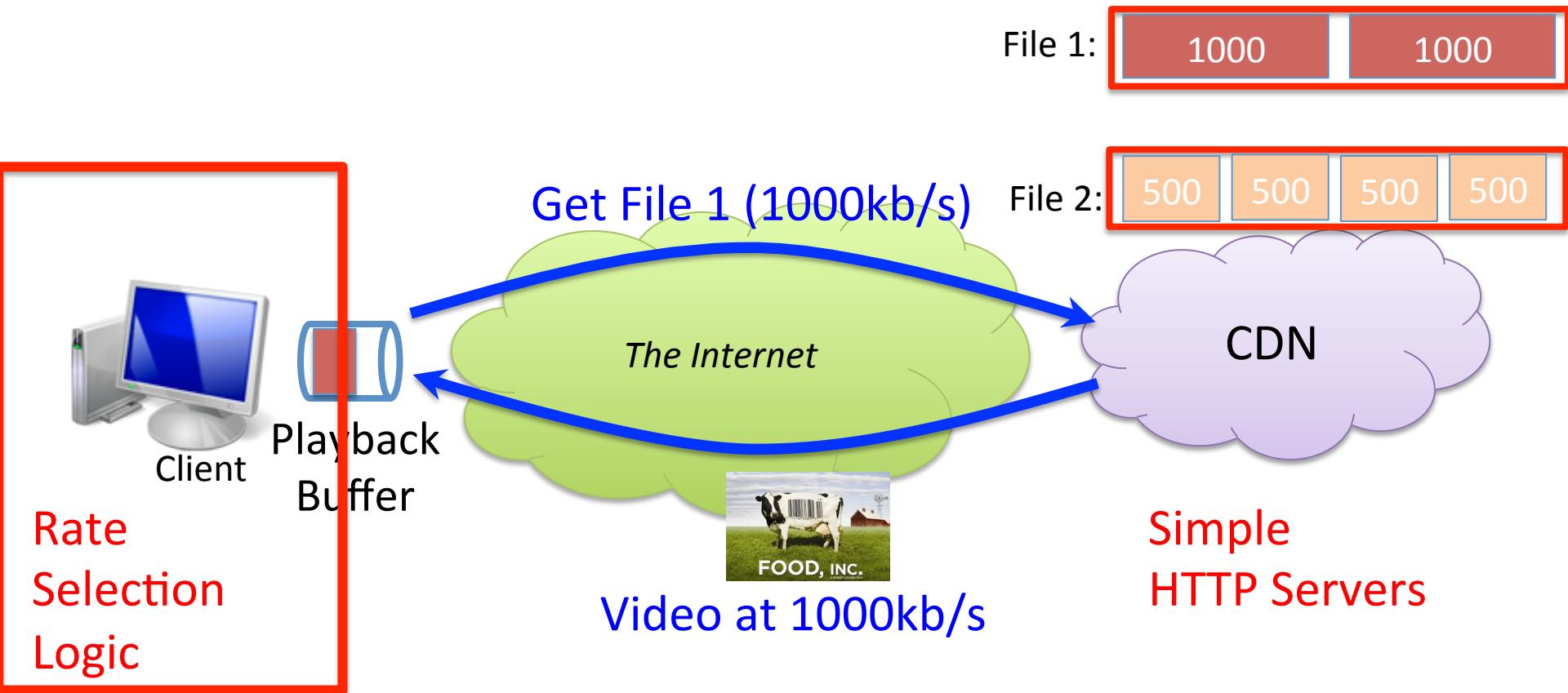
Problem #1: Unnecessary Rebuffers

- The chosen video rate is **too aggressive**
- A lower rate would have avoided the rebuffers

Problem #2: Low Video Rate

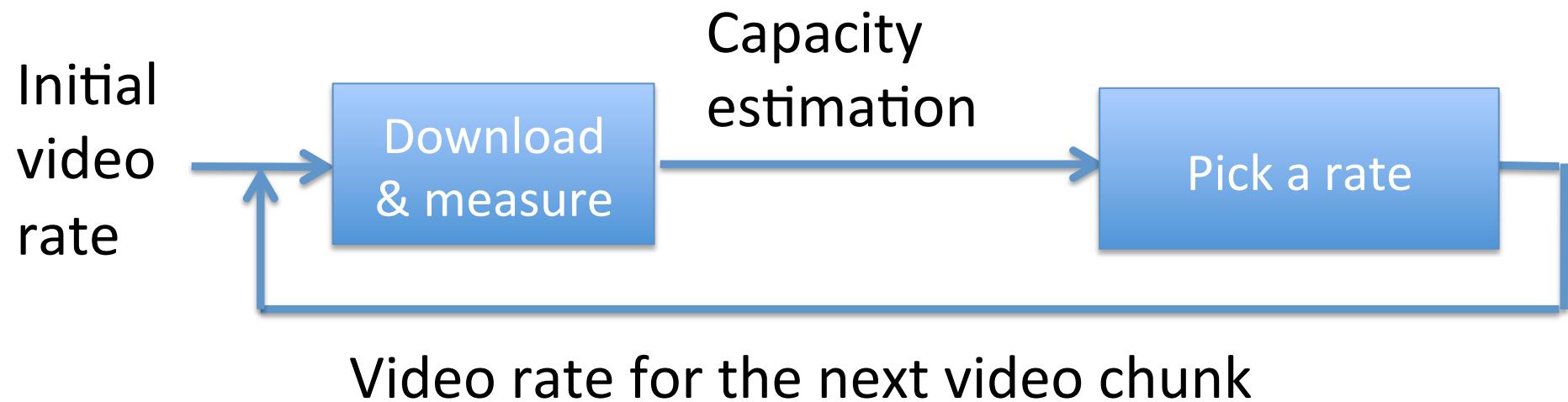
- The chosen video rate is **too conservative**
- The network could have sustained a higher rate

How is the Video Rate Picked?

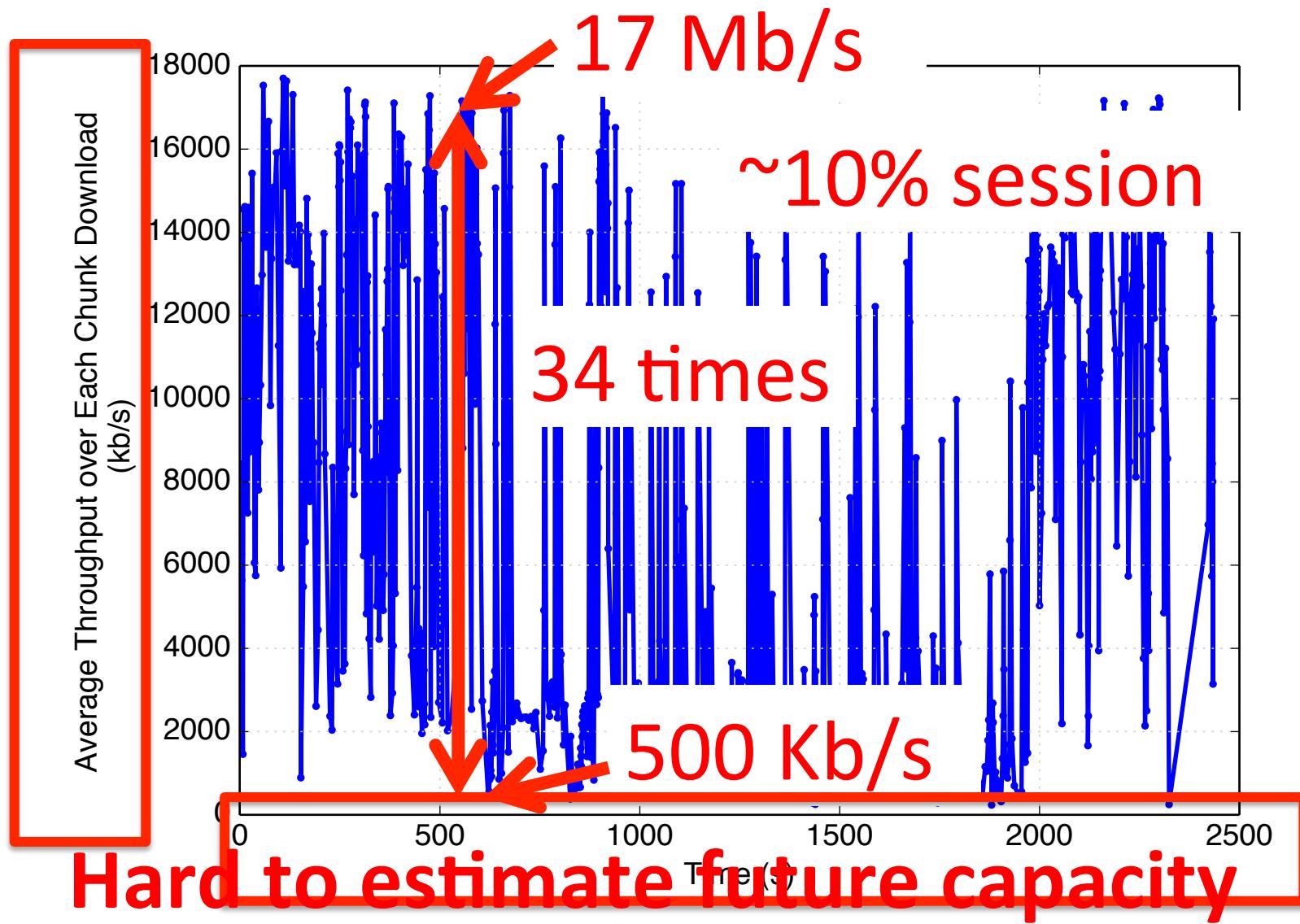


Adaptive Bit Rate (ABR)

How is the Video Rate Picked?



Highly Varying Capacity



Capacity Estimation is Unreliable

Confused, Timid, and Unstable: Picking a Video Streaming Rate is Hard TY Huang et al. [ACM IMC'12]

Downton Abbey without the Hiccups: Buffer-Based Rate Adaptation for HTTP Video Streaming TY Huang et al. [ACM FhMN'12]

Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE J. Jiang et al. [ACM CoNext'12]

An Experimental Evaluation of Rate Adaptation Algorithms in Adaptive Streaming over HTTP? S. Akhshabi et al. [ACM MMSys'11]

An Experimental Investigation of the Akamai Adaptive Video Streaming L.D. Ciccio et al. [USAB'10]

What can we do?

Can we avoid estimating capacity?

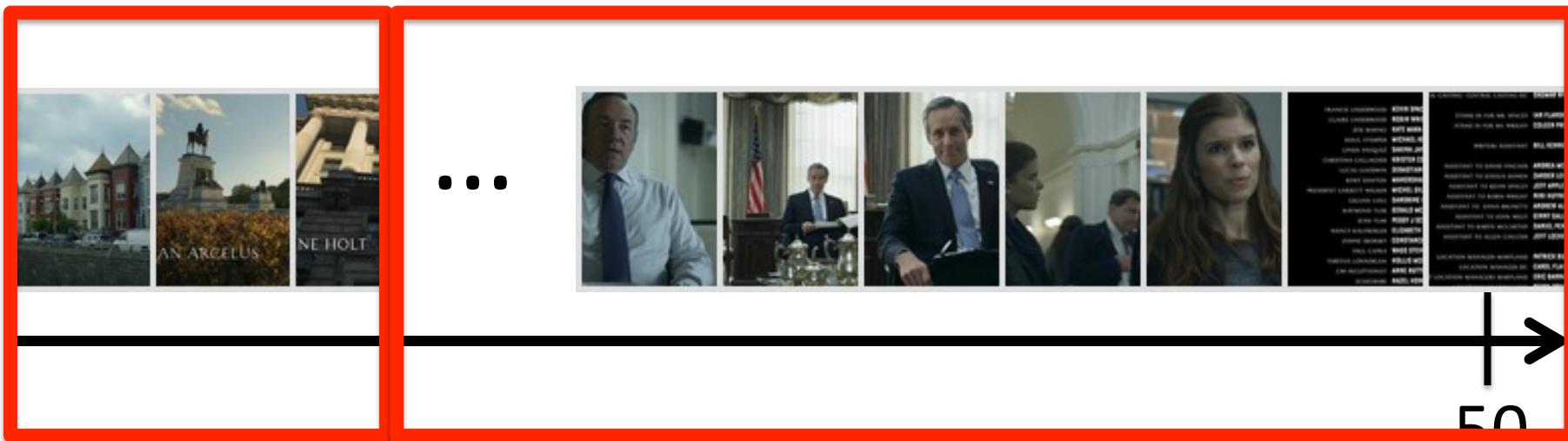
Yes, most of the time!

Capacity estimation
can be avoided in steady-state

90+% of a typical movie/TV show

Pick a rate as a function of buffer occupancy

Avoid Capacity Estimation in Steady-State



Start-Up

Steady-State

50

min

First few minutes Build up some idea about the network
No idea about the network encoded in the buffer occupancy
Simple estimation can be useful Capacity estimation can be avoided

Key Takeaway from Our Results

Using capacity estimation in steady-state is harmful

- 10–20% rebuffers is unnecessary

Pick a video rate as a function of buffer occupancy in steady-state

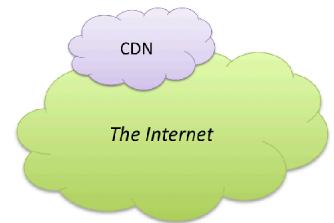
- Buffer occupancy encodes information about capacity

Capacity estimation can be useful in start-up

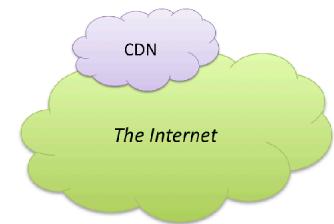
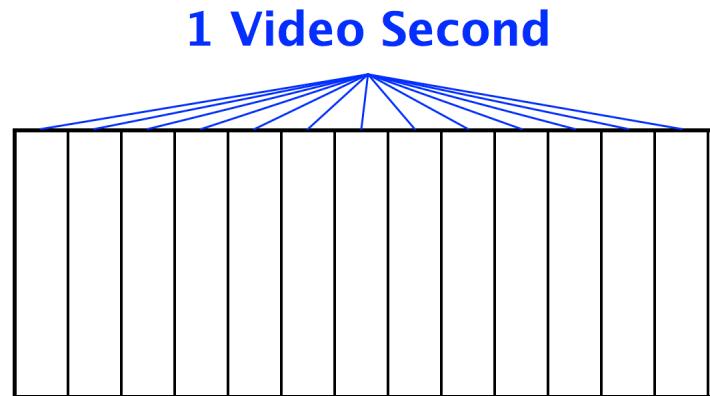
- Help us reach the steady state quickly

How is the network condition encoded in the occupancy of video playback buffer?

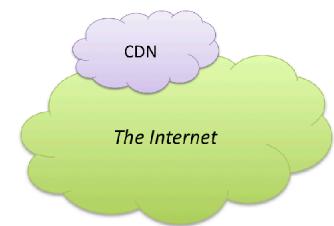
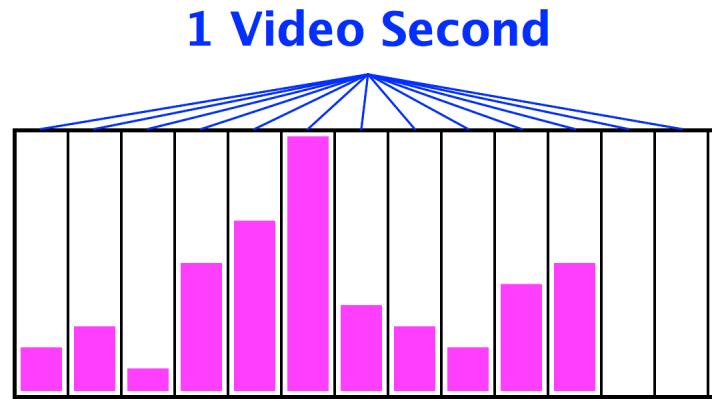
The relationship between **Buffer, Capacity, and Video Rate**



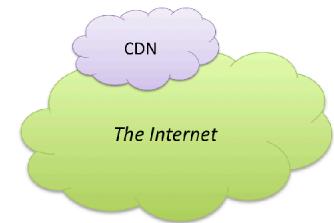
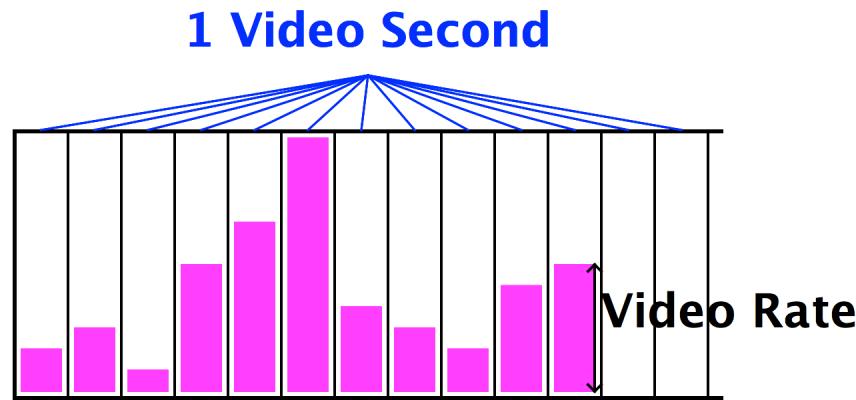
The relationship between Buffer, Capacity and Video Rate



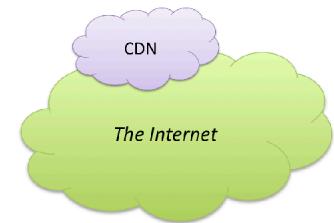
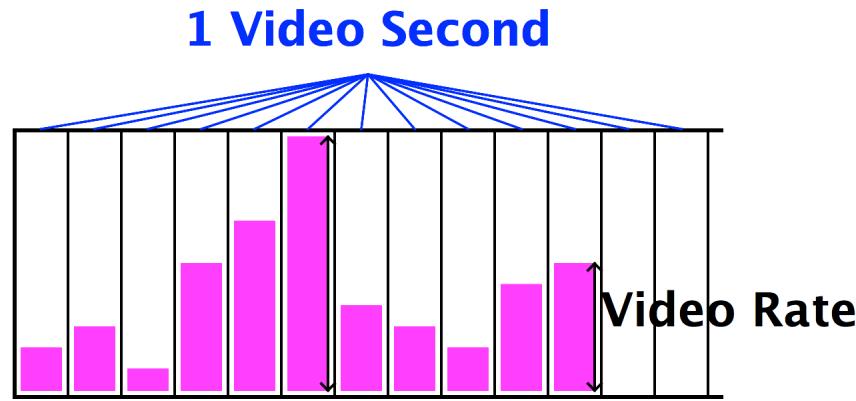
The relationship between Buffer, Capacity and Video Rate



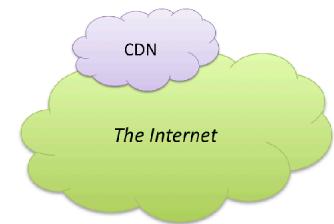
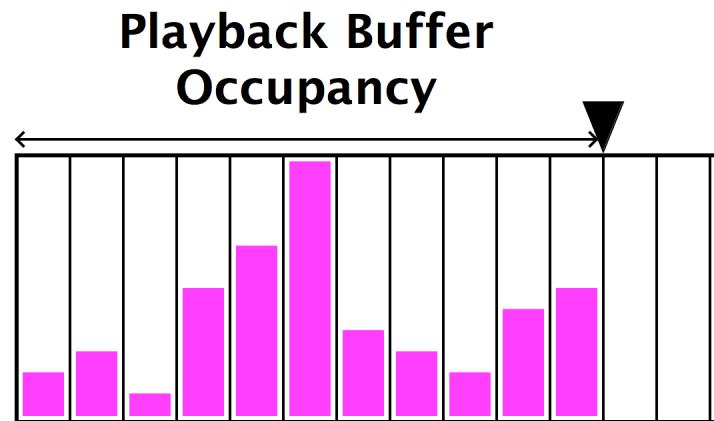
The relationship between Buffer, Capacity and Video Rate



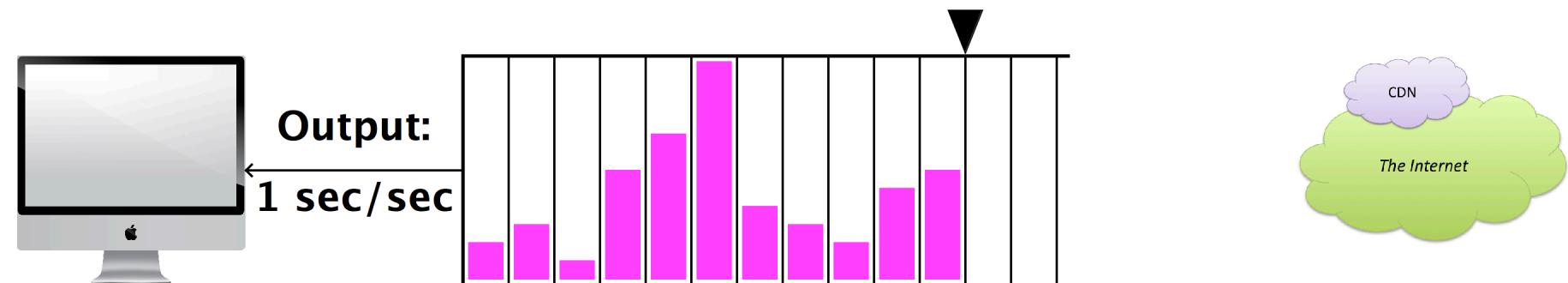
The relationship between Buffer, Capacity and Video Rate



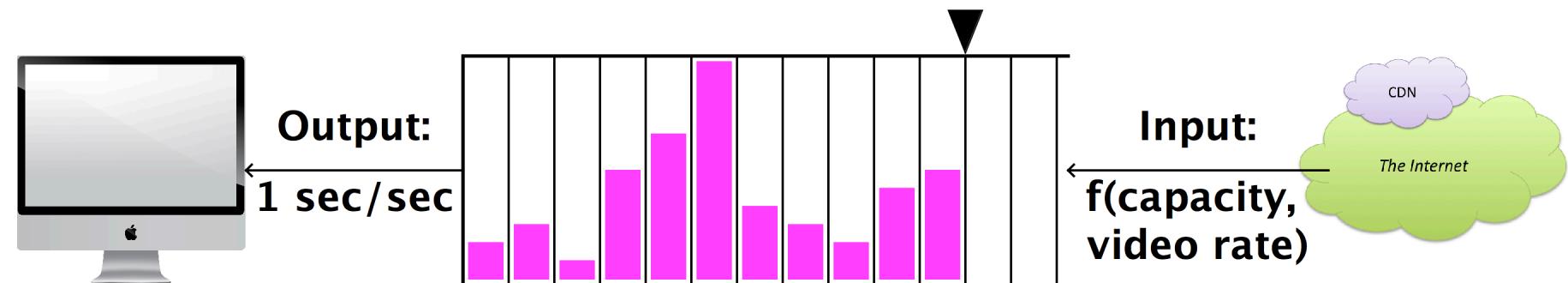
The relationship between Buffer, Capacity and Video Rate



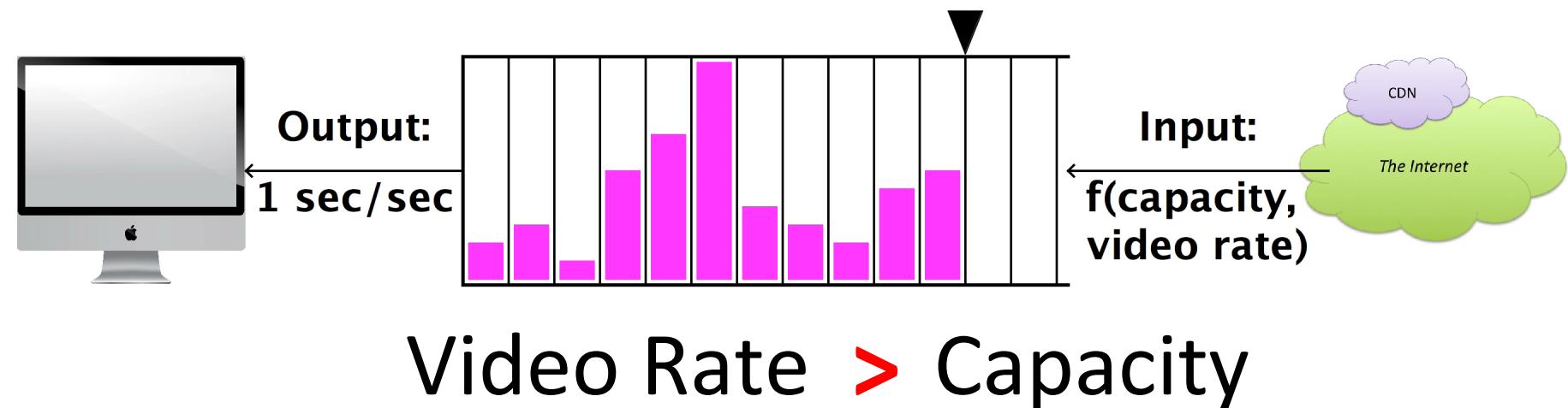
The relationship between Buffer, Capacity and Video Rate



The relationship between Buffer, Capacity and Video Rate

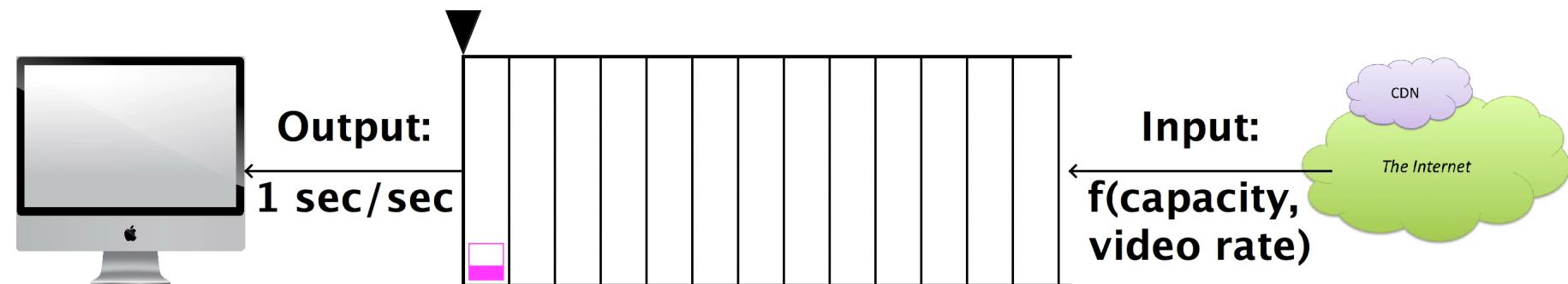


The relationship between Buffer, Capacity and Video Rate



The relationship between Buffer, Capacity and Video Rate

Capacity is not fully utilized!
Keeps going into slow-start!



Video Rate < Capacity

Natural State Variable: Buffer Occupancy

When buffer occupancy is decreasing

- We can not sustain the current video rate
- Decrease the video rate

When buffer occupancy is increasing

- We can stream a higher video rate
- Increase the video rate

Pick a rate as a function of buffer occupancy

Nice Properties of Buffer-Based Algorithm

Theorem #1: Avoid unnecessary rebuffers

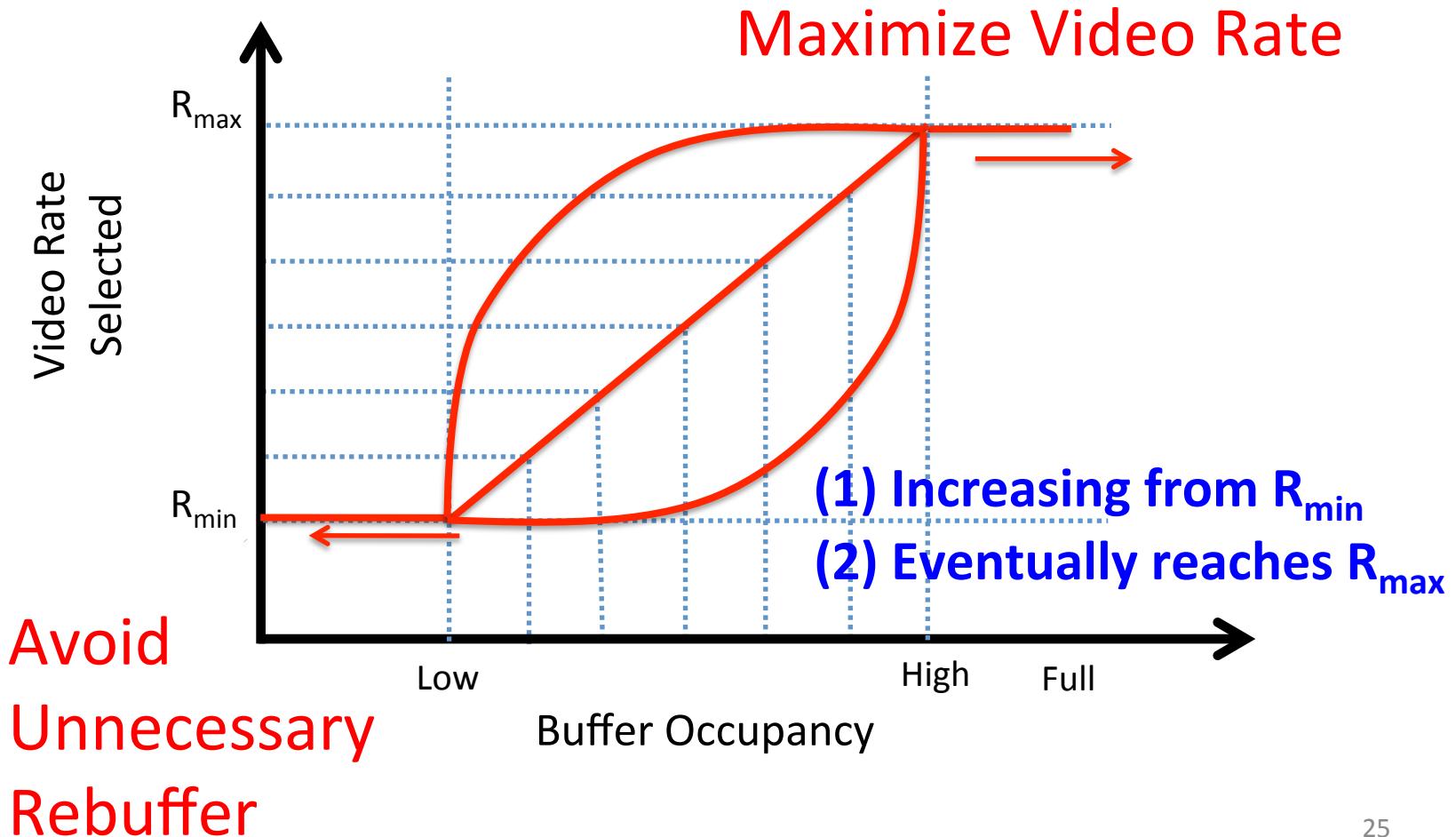
- When buffer approaches empty
- Request R_{\min}
- As long as Capacity > R_{\min}
- Buffer will grow

Theorem #2: Maximize video rate

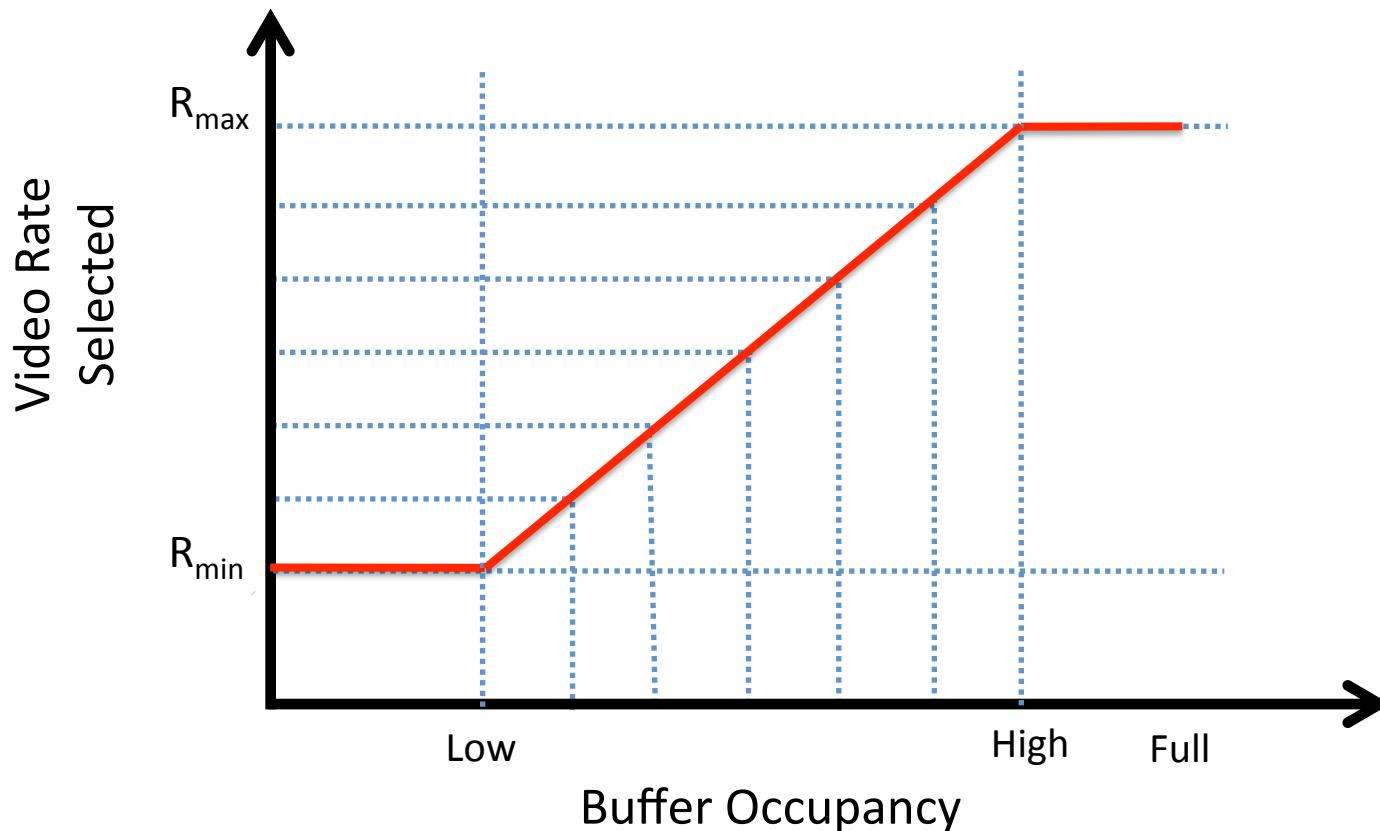
- When buffer approaches full
- Request R_{\max}
- As long as Capacity < R_{\max}
- Buffer will not be filled
- Fully utilize the capacity

Encoding scheme
Available video rates
Chunk size

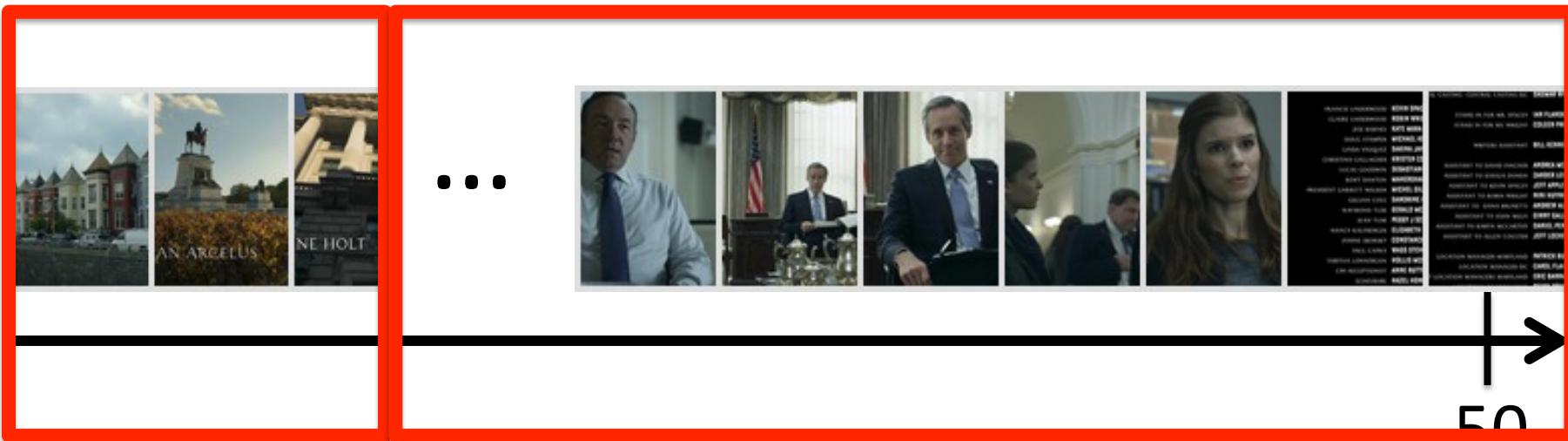
Pick a Video Rate as a Function of Buffer Occupancy



Pick a Video Rate as a Function of Buffer Occupancy



The Buffer-Based ABR Algorithm



Start-Up

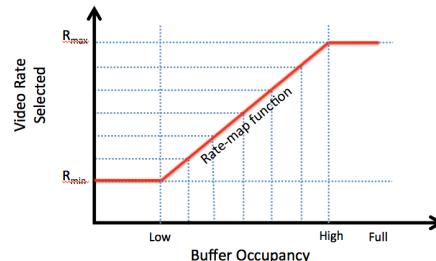
Steady-State

50

min

Buffer is still growing
Pick a rate as a function **buffer occupancy**

Pick a rate based on the
immediate past throughput



Verification by Netflix Deployment

Deployed in Netflix's browser-based player

500,000 users for a weekend in September, 2013

To compare the performance

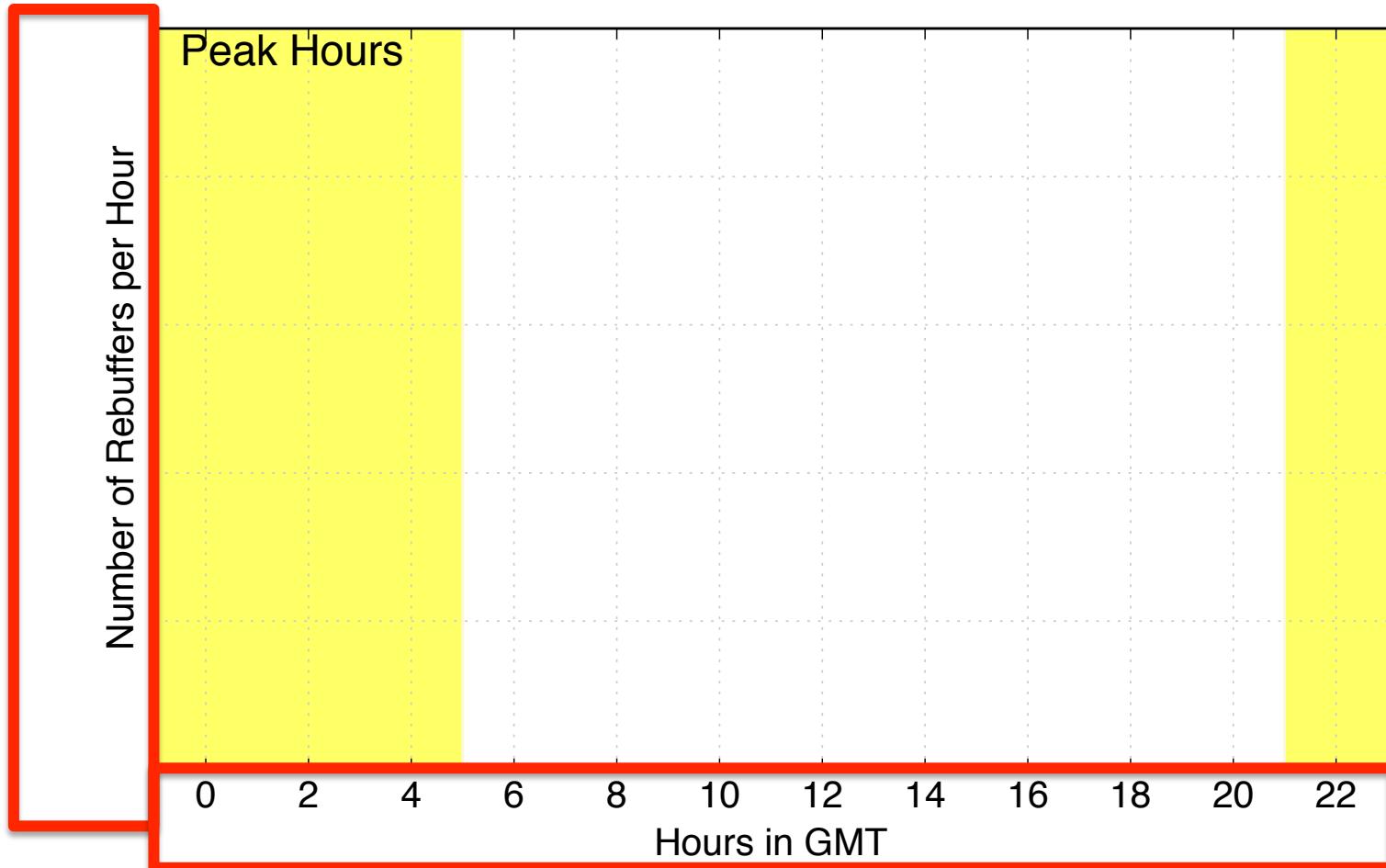
Group 1: Netflix's then-default “**Control**” algorithm

Group 2: Lowest possible rebuffer rate “**Lower bound**”

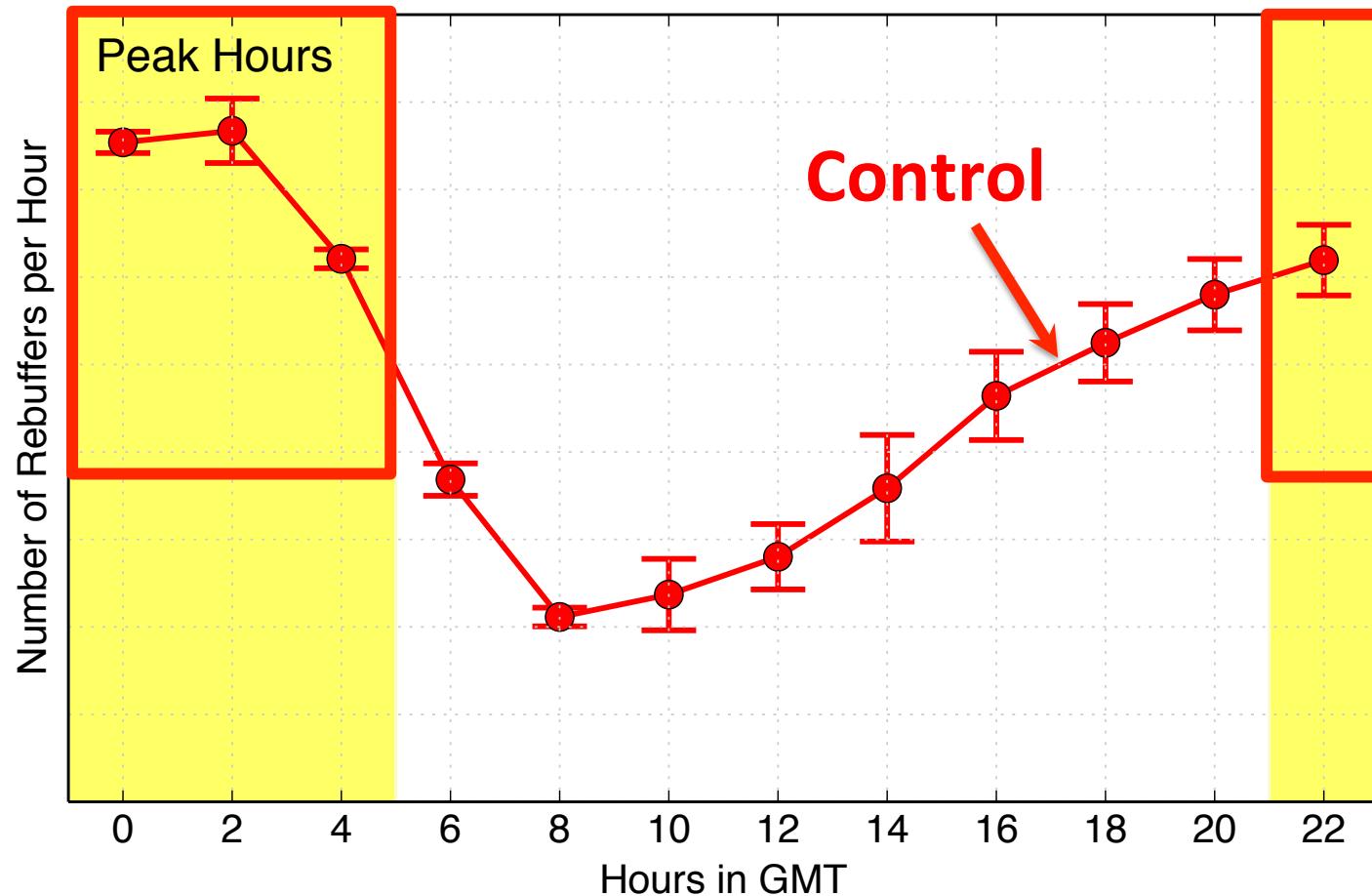
- Always streams at the lowest video quality
- Empirical lower bound on rebuffer rate

Group 3: Our buffer-based algorithm “**BBA**”

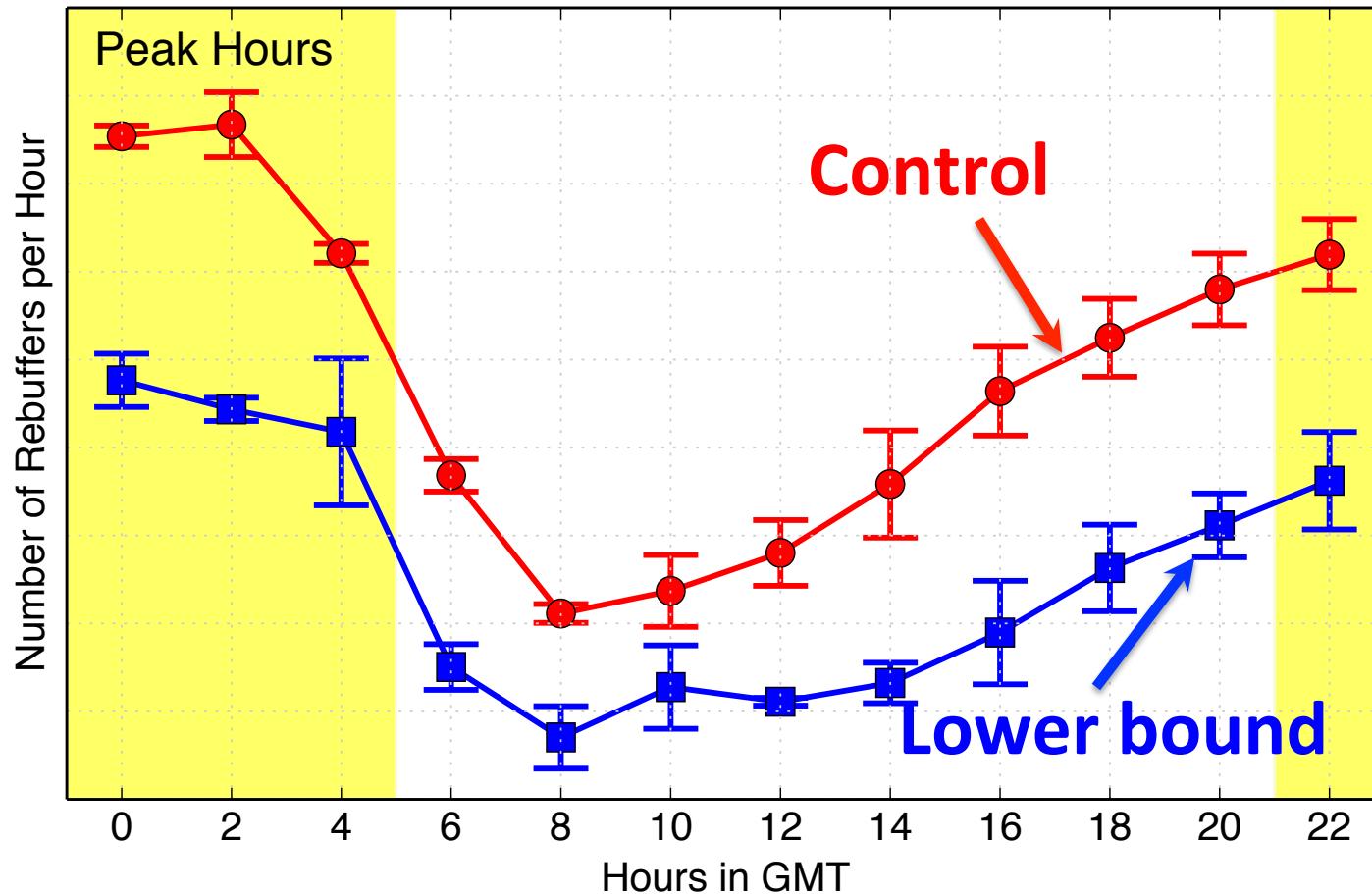
Performance: Rebuffers Rate



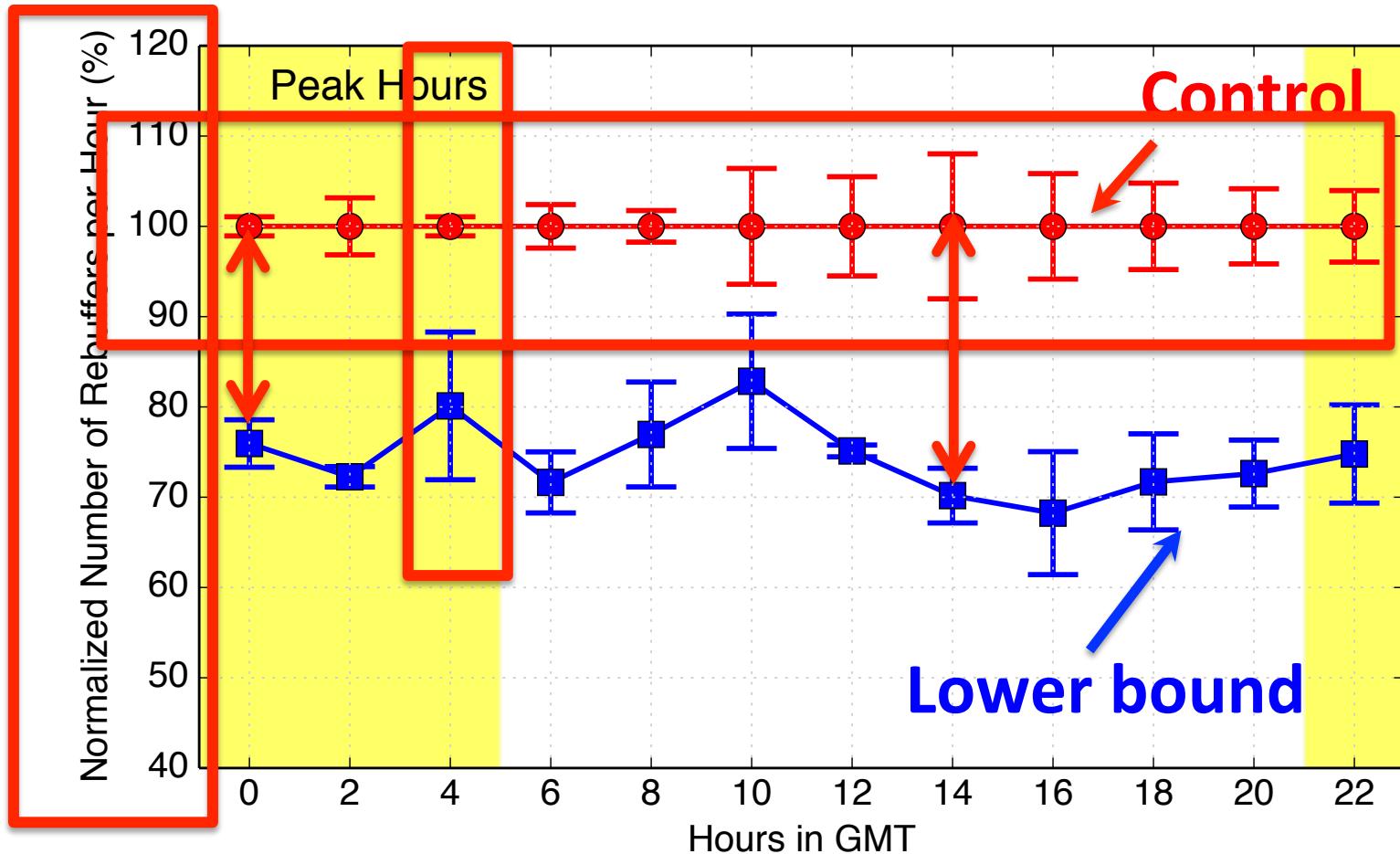
Performance: Rebuffers Rate



Performance: Rebuffers Rate

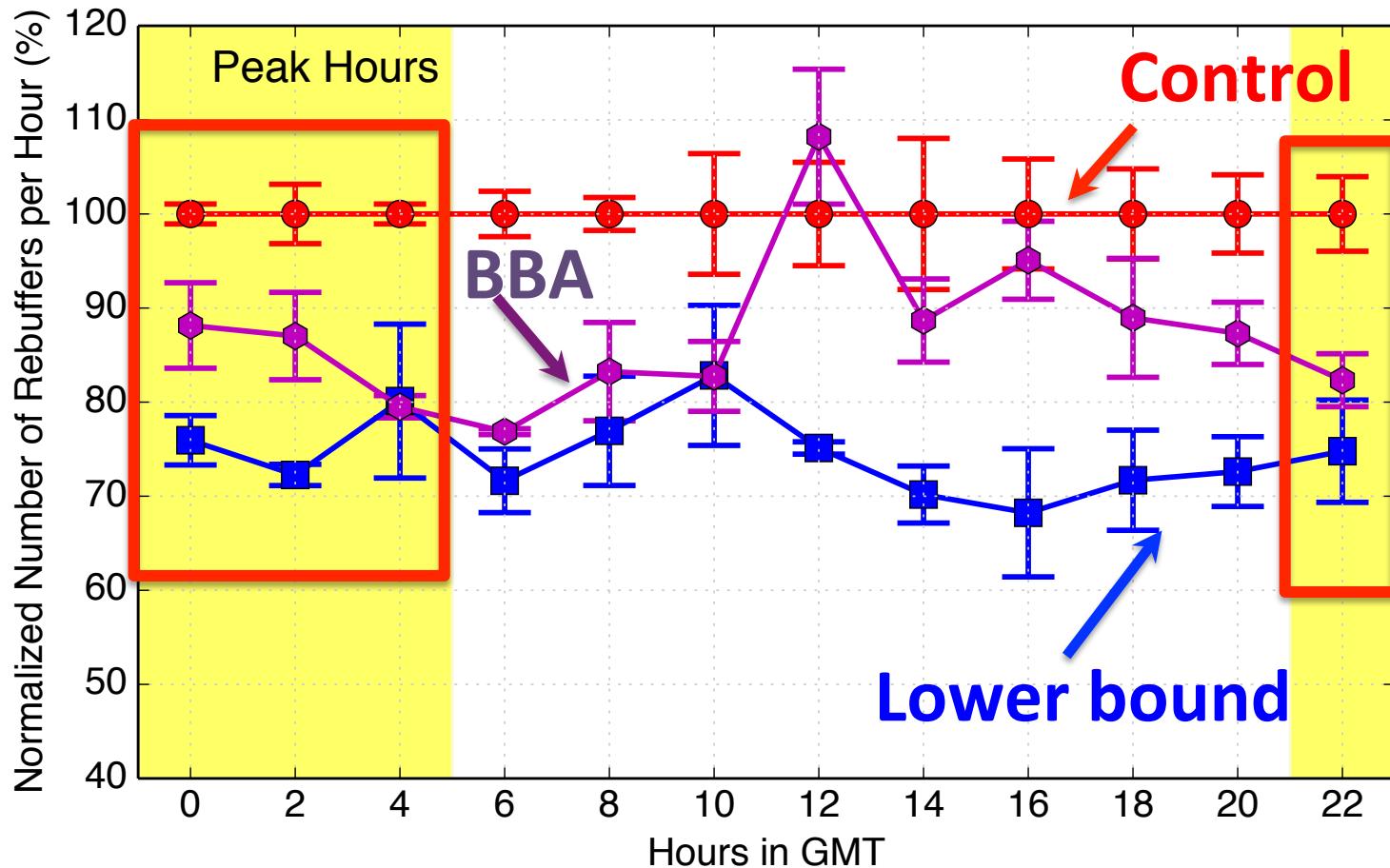


Performance: Rebuffer Rate



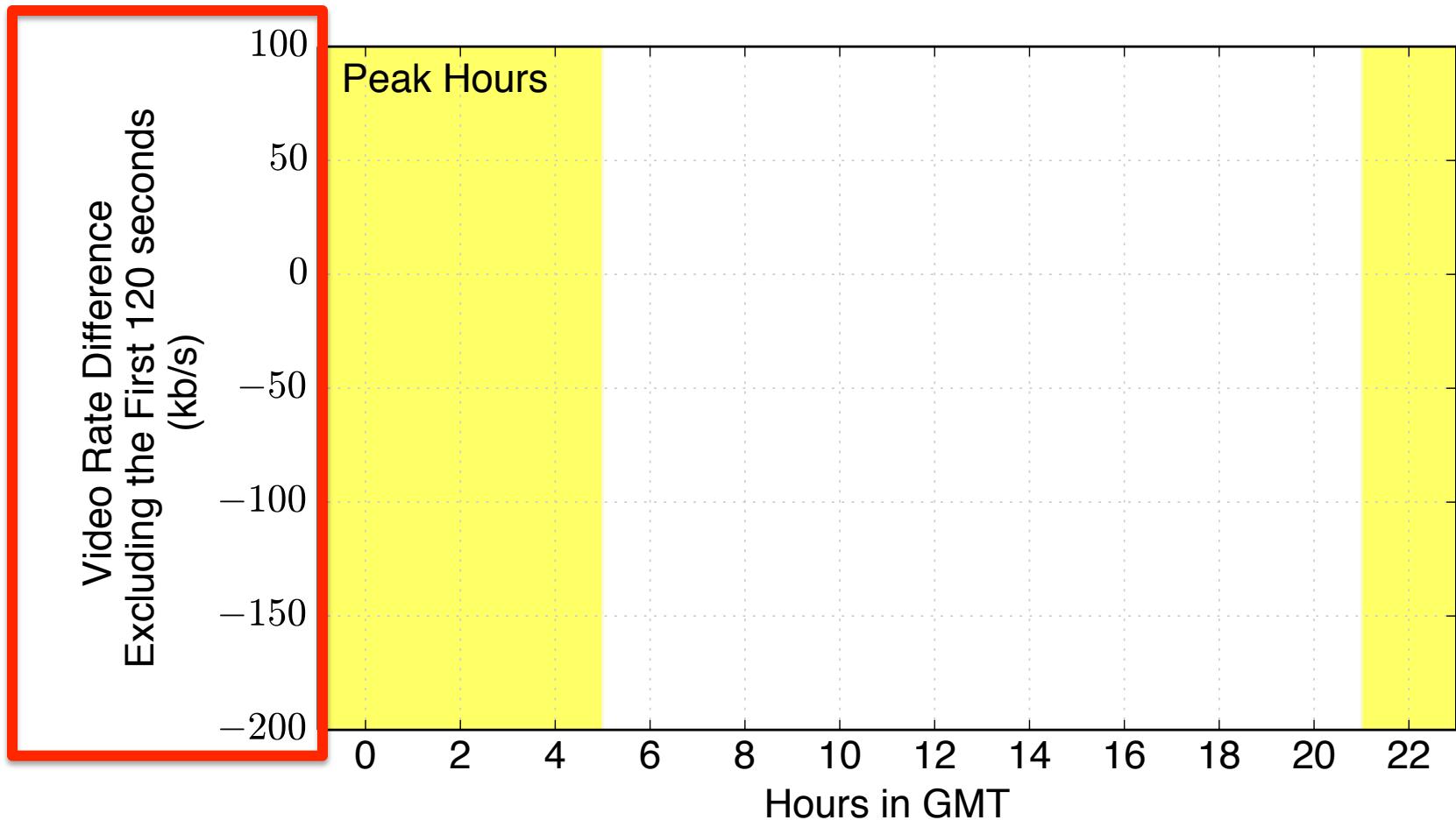
20 - 30% of rebuffers can be eliminated.

Performance: Rebuffers Rate

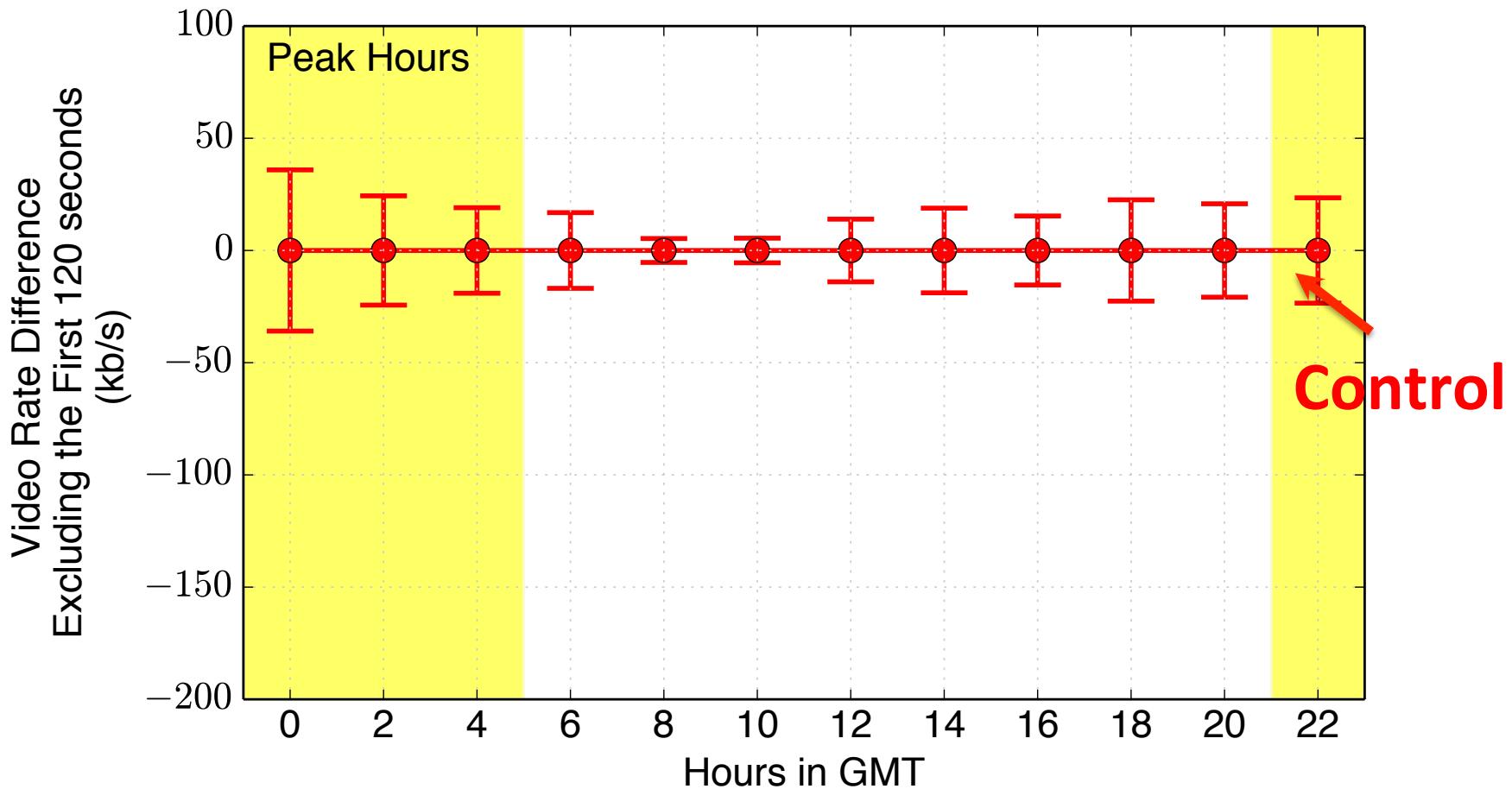


10 – 20% less rebuffer than Control

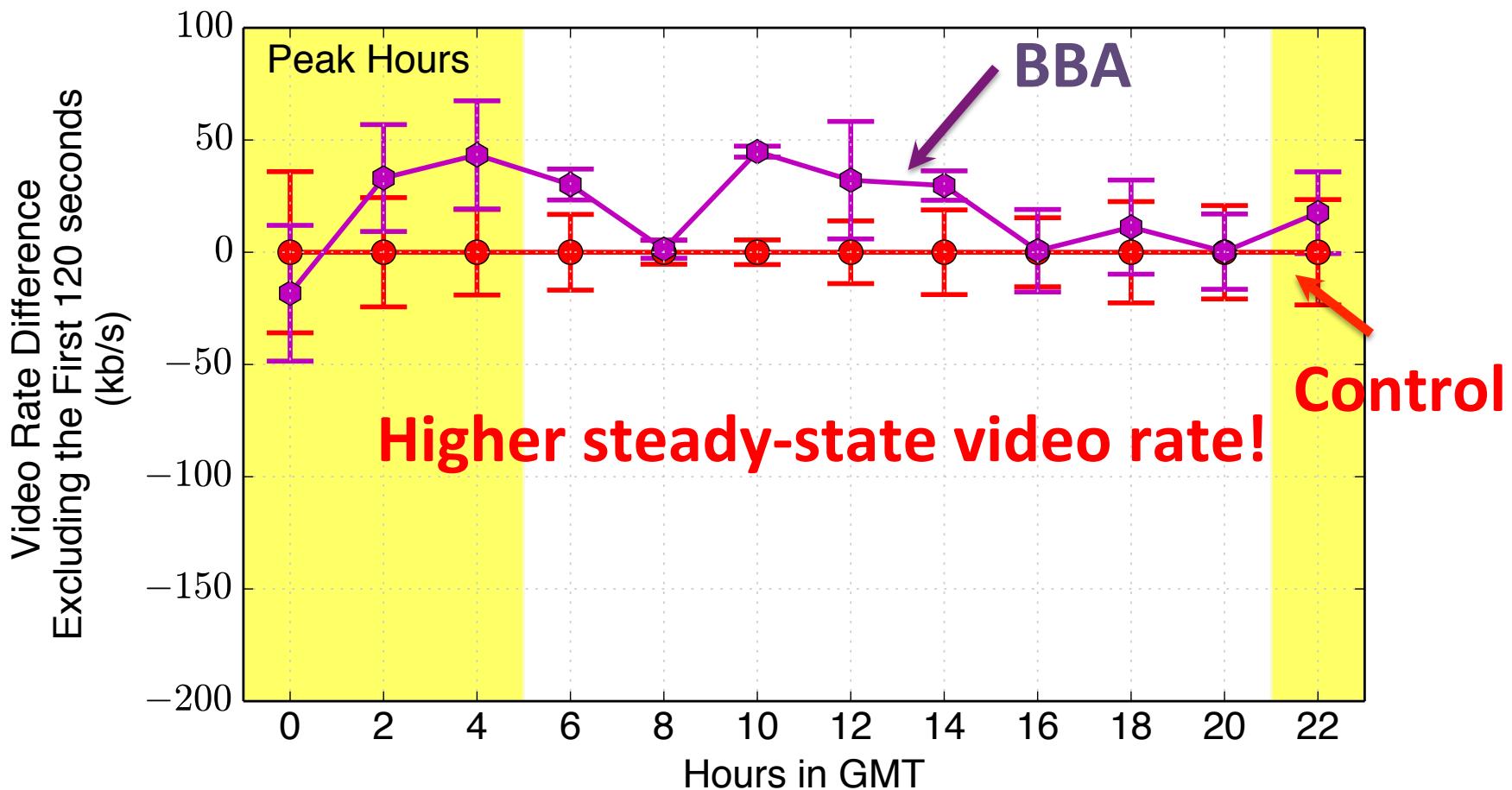
Performance: Steady-State Video Rate



Performance: Steady-State Video Rate



Performance: Steady-State Video Rate



Conclusion

- Capacity varies wildly in practice
 - Capacity estimation is unreliable
- Capacity estimation is still helpful in start-up
- Avoid capacity estimation in steady-state
- Pick a rate as a function of buffer occupancy
 - Buffer occupancy is the natural state variable

Focus on the buffer occupancy allows you to:

Avoid unnecessary rebuffers

Maximize video rates