Internet Video Delivery Basics

ECE 50863 – Computer Network Systems

Types of Internet Video

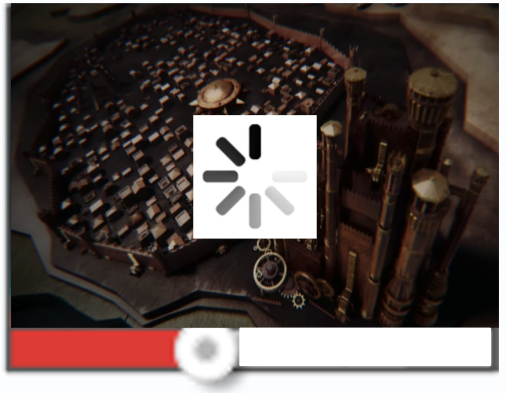
- Video Conferencing
  - Interactive: User must get video in 100-200 milliseconds.
- Video Streaming/ Video on Demand (VoD)
  - E.g., streaming a movie, recorded lecture or newsclip
  - Once video starts, playback should be “smooth”
  - Can prefetch content (tens of seconds or even a few minutes) in the video player buffer.
- Live video broadcast
  - E.g., tuning into a sports game
  - Typically 5-10 seconds of delay may be acceptable: not as interactive as video conferencing
  - Similar to VoD, but keep client buffer small (a few seconds).

Types of Internet Video

- Video Conferencing
  - Interactive: User must get video in 100-200 milliseconds.
- Video Streaming/ Video on Demand (VoD)  Our focus
  - E.g., streaming a movie, recorded lecture or newsclip
  - Once video starts, playback should be “smooth”
  - Can prefetch content (tens of seconds or even a few minutes) in the video player buffer.
- Live video broadcast
  - E.g., tuning into a sports game
  - Typically 5-10 seconds of delay may be acceptable: not as interactive as video conferencing
  - Similar to VoD, but keep client buffer small (a few seconds).

Key considerations for video streaming

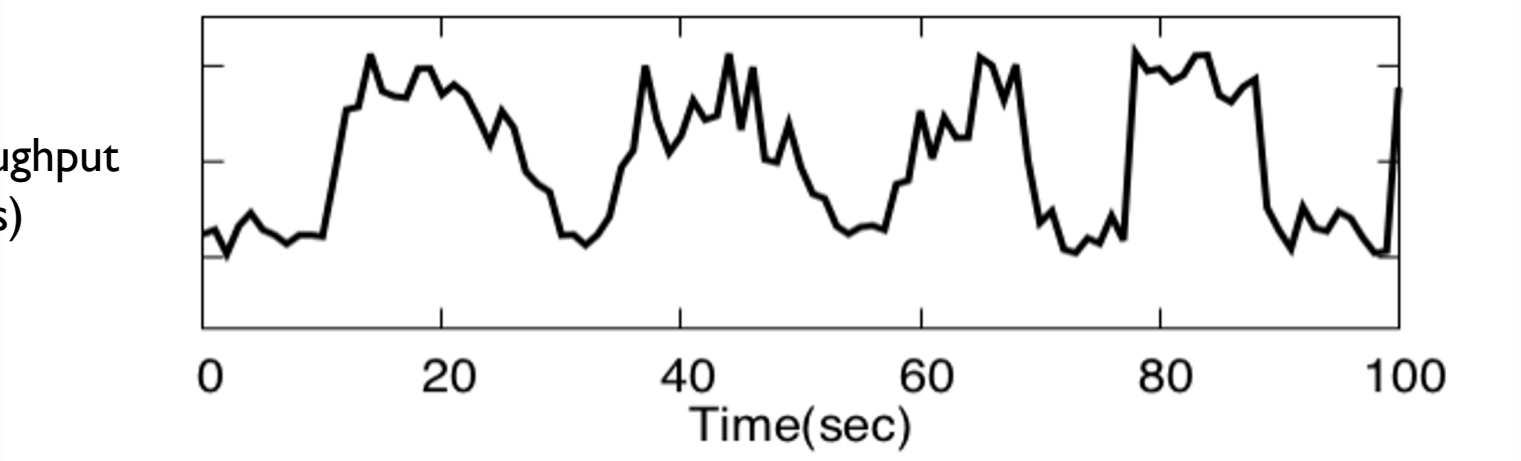
- Join Latency
  - How soon should video start playing back, once a user decides to play the video
- Avoid “rebuffering” events.
  - Once client starts playing video, avoid “stalls”.



Rebuffering

Challenges in streaming Internet video

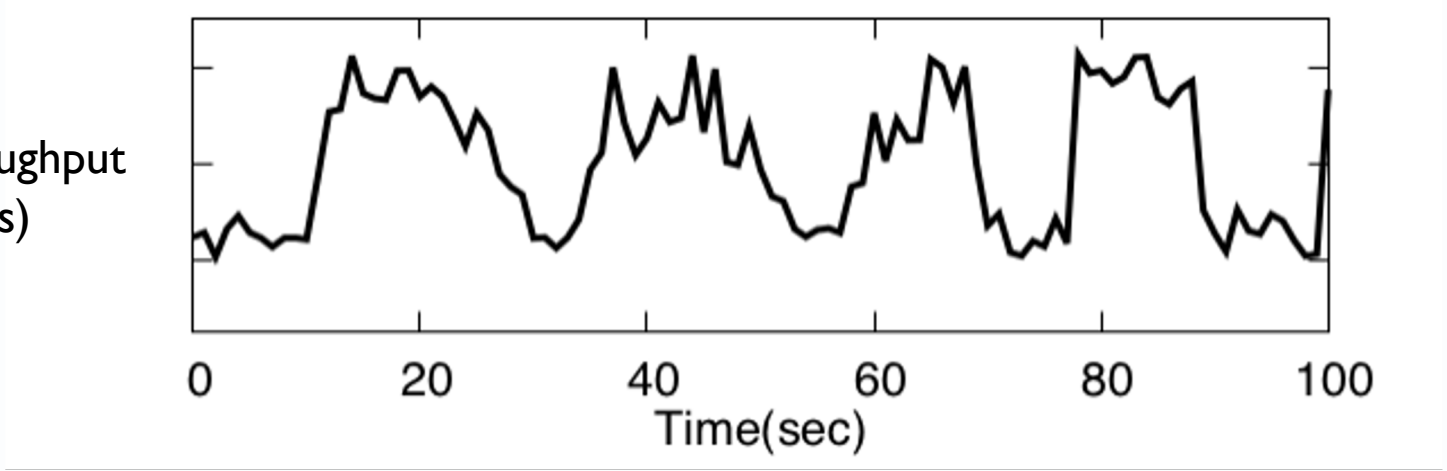
- Internet throughput is highly variable owing to congestion
- Internet video needs a certain throughput to avoid stall.
  - E.g., to send video at 1080p, typically need 5 Mbps
- How do we handle dips in throughput?



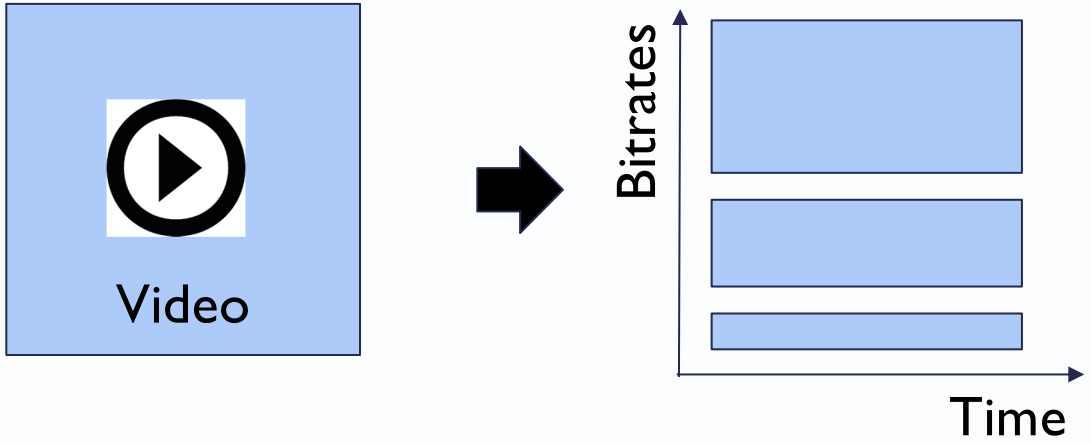
Approaches for streaming Internet video

- Static approach: Pick video rate conservatively at start of session
  - Low enough that it is highly unlikely the connection will see poorer throughput
  - Problem: May need to be very conservative, and even then cannot handle occasional dips when they arise.

- Adaptive approach: Adapt video rate (and quality) based on network conditions

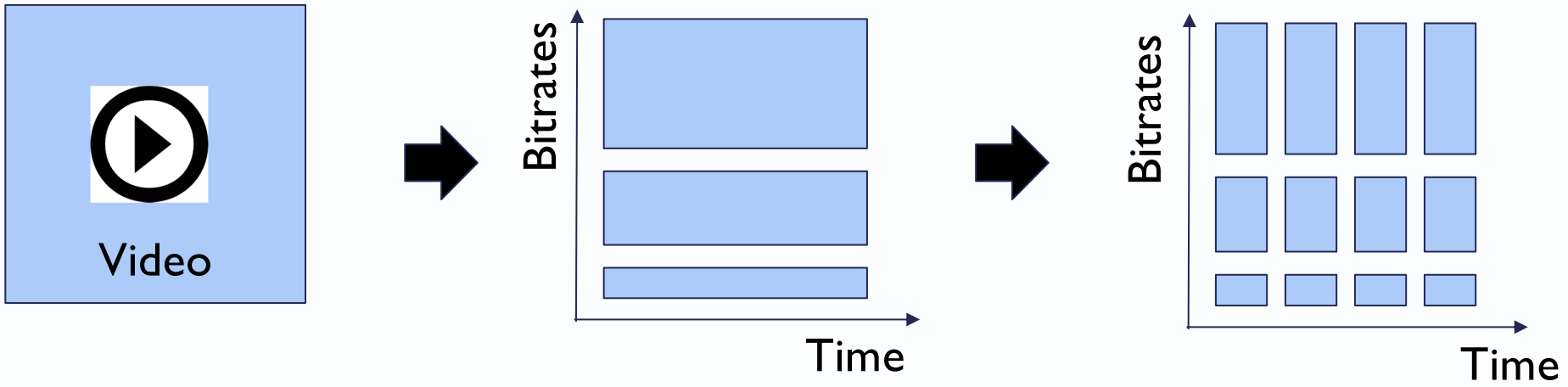


Adaptive Bitrate Streaming



A video clip is encoded
with multiple qualities (bitrates)

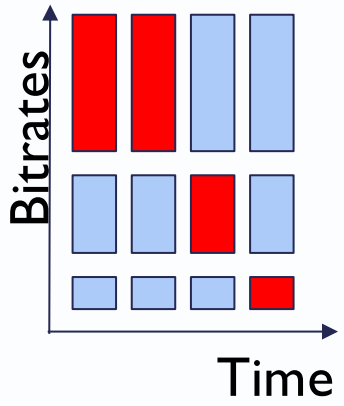
Adaptive Bitrate Streaming.



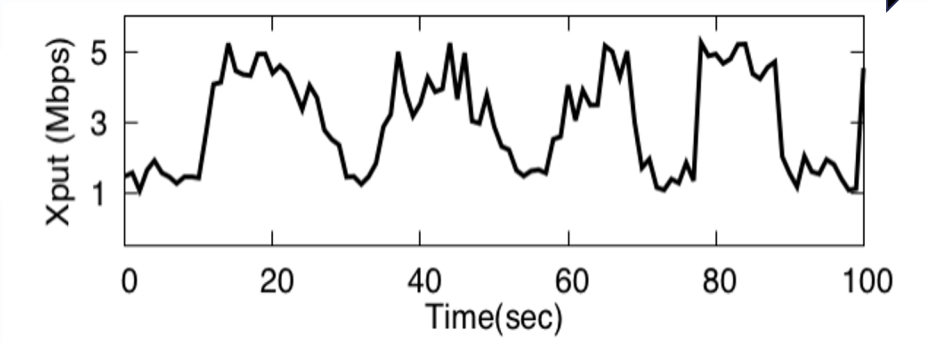
Video encoded at each bitrate is split into chunks

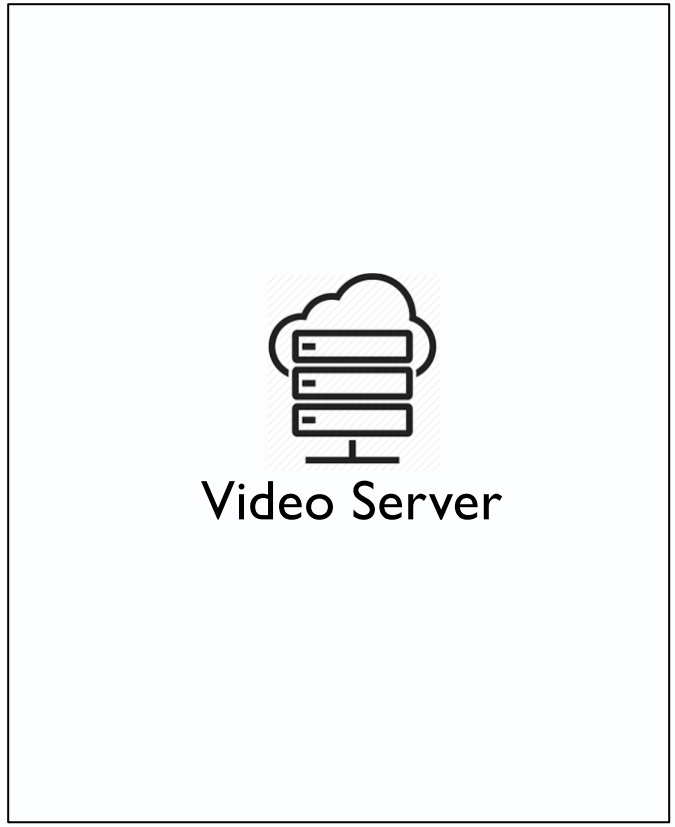
Adaptive Bitrate Streaming (2)

Video Client



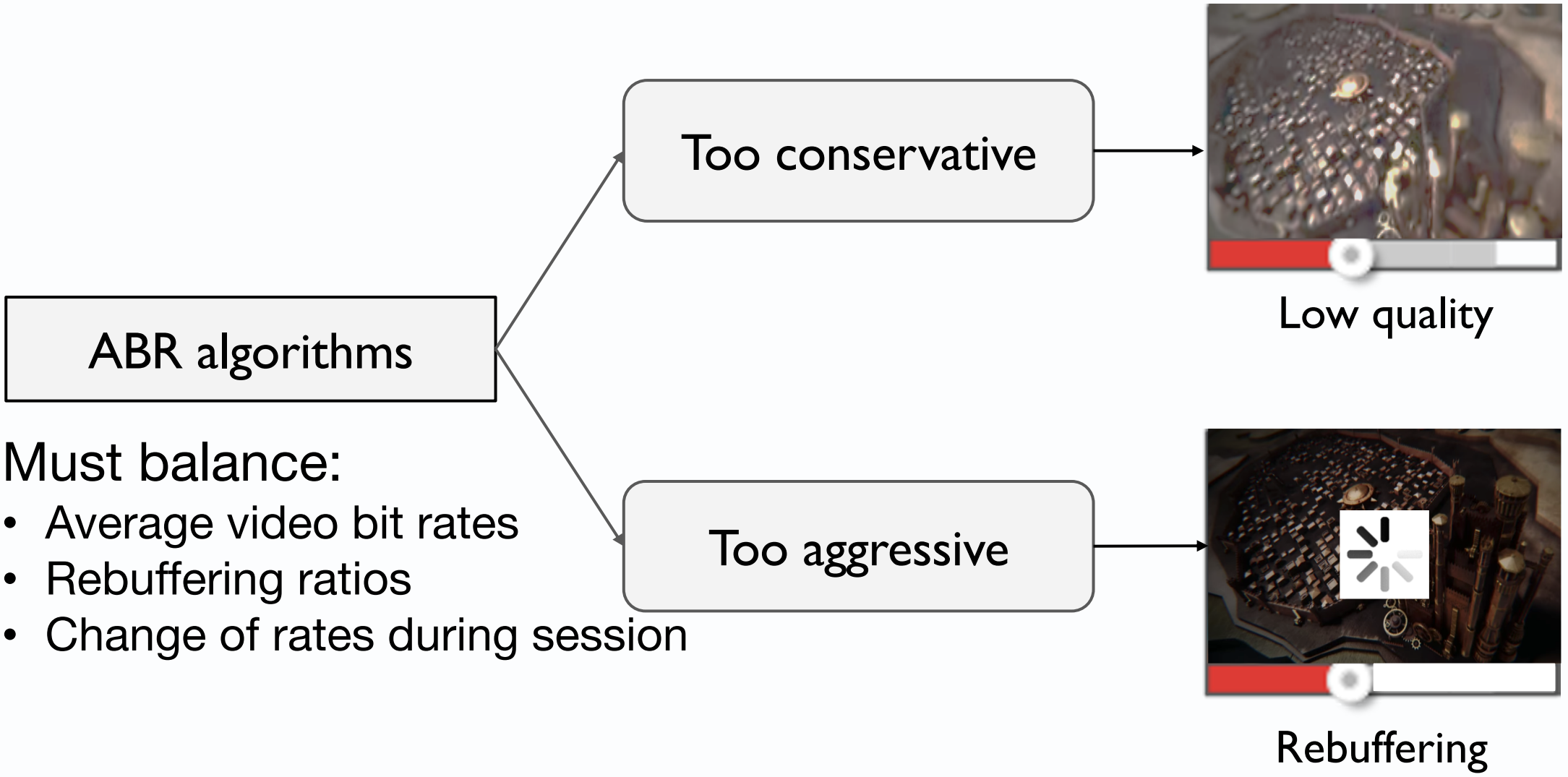
Request n*th* chunk at bitrate *r*





Adaptive Bitrate Algorithms(ABR)

Adaptive Bitrate Streaming



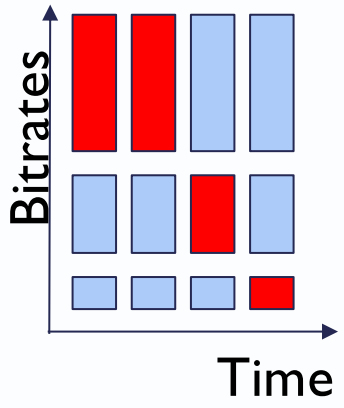
Must balance:

- Average video bit rates
- Rebuffering ratios
- Change of rates during session

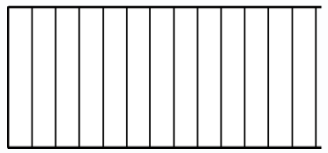
Examples of Adaptive BitRate Algorithms

ECE 50863 – Computer Network Systems

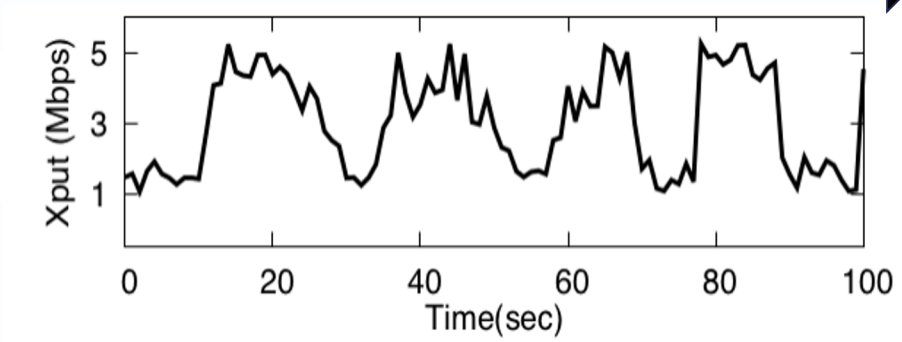
Recap: Adaptive Bitrate Streaming



Video Client



Request n*th* chunk at bitrate *r*



Video Server

Adaptive Bitrate Algorithms(ABR)

How do we design ABR algorithms?

- Many different algorithms used in the industry.
- Active area of research in networking community
- Discuss two example research algorithms
  - Do not accept research publications as gospel!
  - Critique them, question their claims, ask if you can do better

Decision that an ABR algorithm needs to make

4 sec of chunks in the player buffer

Bitrate Decision

Given I have a certain amount of data in my client buffer, which quality should I download next chunk?

What if we could perfectly predict download times?

4 sec of chunks in the player buffer

Bitrate Decision

2.1 sec ✗ **Low quality**

3.5 sec ○

5.3 sec ✗ **Rebuffering!**

Example Algorithm 1

- ***"A buffer-based approach to rate adaptation: evidence from a large video streaming service", Huang et al., ACM Sigcomm 2014.***
- Do not try to predict throughput (and hence chunk download times)
  - Prediction is hard and often erroneous
- Simply consider the occupancy of the client buffer in the decision process
  - Low occupancy => lower bit rate, High occupancy => higher bit rate
- Three versions of the algorithm, each considers additional enhancements not considered in the previous version.

Example Algorithm 2

- ***"A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP", Yin et al., ACM Sigcomm 2015***
- Predict throughput, and consider buffer occupancy
- Makes decisions considering multiple factors:
  - Average video bit rates
  - Rebuffering ratios
  - Change of rates during session
- Optimizes "QoE", a composite metric that combines the above three metrics

How the control theoretic algorithm works

- Lets assume chunks ***1, 2, … i*** have been downloaded.
- Algorithm considers a lookahead of ***k*** chunks
- Predicts throughput for this period of time.
- Pick quality for next K chunks based on permutation with best QoE score.
  - Q different qualities (q1, q2,q3 etc.) => Q^K possible permutations
- Download chunk i+1, adjust predictions and go back to first step.

i+1	i+2	….	i + K
q1	q1	…	q1
q3	q3	….	q3
q1	q3	…	q1
q2	q1	…	q3

Questions to think about..

- How do we predict throughput based on throughput of last few chunks?
- Does a purely buffer-based approach suffice? When can throughput prediction help?
- How could we be robust to prediction errors?
- How well can the algorithm cope with “sudden and prolonged” bandwidth dips?
- Strengths and weaknesses of each algorithm?