Applications: Video

Introduction

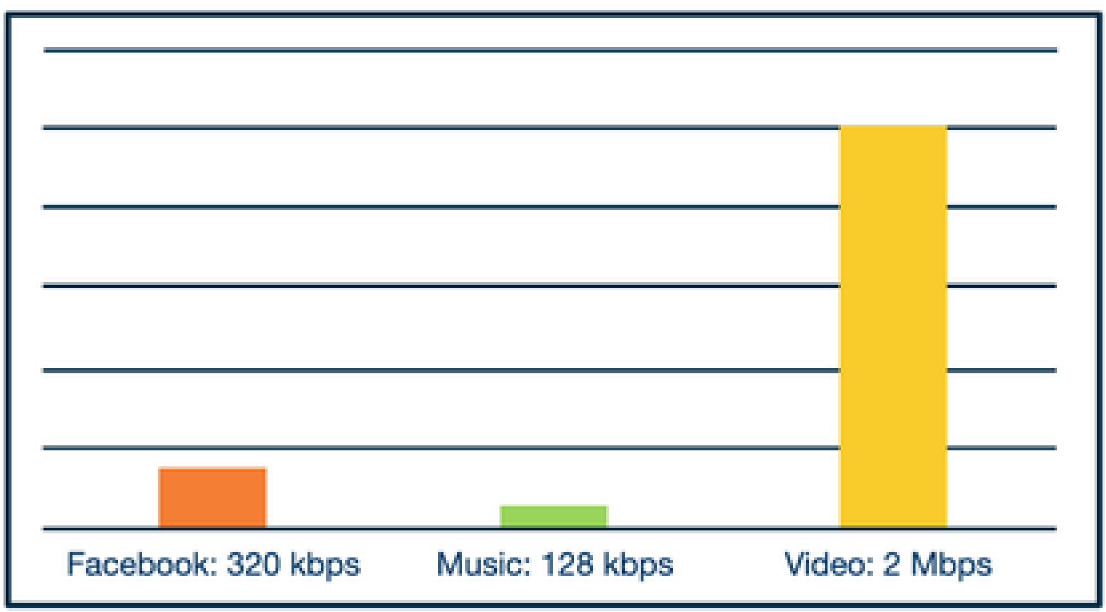
1. Video
   - Studying application layer
     - Voice and video applications
   - VoIP
   - Video compression
   - Bitrate adaptation

Multimedia Applications: A Perspective from the Network

1. Multimedia applications
   - Any network application that employs video or audio
   - Deal with interesting challenges
     - Video needs to play at the speed it was recorded
     - Can't have a delay in audio

Background: Video and Audio Characteristics

1. Video
   - High bit rate, between 100 kbps to over 3 Mbps
   - Can compress video, trading off compression for quality
2. Audio
   - Lower bit rate than video
   - Glitches are more noticeable than in video
   - Can also be compressed



Bitrate Comparison

Types of Multimedia Applications and Characteristics

1. Streaming Video
    - Video starts playing within a few seconds of receiving data instead of waiting for the entire file to download first
    - Interactive; can pause, fast forward, skip ahead, move back
    - Continuous playout: Shouldn't freeze up in the middle
    - Stored on a CDN rather than one data center
2. Streaming Live Audio and Video
    - Similar to stored audio and video with important differences
    - Many simultaneous users
    - Delay sensitive
3. Conversational Voice and Video Over IP
    - VoIP: Phone over Internet instead of traditional circuit-switched telephony
    - Highly delay sensitive
        - Short delay of 150 ms acceptable
        - Long delay of over 400 ms is noticeable
    - Loss tolerant
        - Can just ask the other side to repeat themselves

Quiz 1

1. When streaming stored audio and video, the content starts playing within a few seconds of receiving data, instead of waiting for the entire file to download first.
    - True
2. Streaming audio and video is interactive and should have a continuous playout.
    - True
3. Streaming live audio and video is usually not interactive and is delay-sensitive.
    - True
4. Conversational voice and video over IP is a service that uses traditional circuit-switched telephony networks.
    - False

How Does VoIP Work?

1. Encoding
    - Analog audio is a continuous wave, but digital data is discrete
    - Audio signal is sampled thousands of times per second
        - Each sample is rounded to a discrete number in a particular range, called quantization
    - Pulse Code Modulation on an audio CD takes 44,100 samples per second, with each value being 16 bits long
    - Encoding schemes
        - Narrowband
        - Broadband
        - Multimode (can operate on either)
    - VoIP aims to minimize bandwidth while still being understandable
2. Signaling
    - Signaling protocol takes care of how calls are set up and torn down
    - Four major functions
        - User location
        - Session establishment
        - Session negotiation
        - Call participation management
    - VoIP uses signaling protocols, just like telephony, to perform the same fucntions
        - Session Initiation Protocol (SIP)

QoS for VoIP: Metrics

1. Three major QoS metrics
    - End-to-end delay
    - Jitter
    - Packet loss

QoS for VoIP: End-to-End Delay

1. Total Delay
    - Time it takes to encode audio
    - Time it takes to put it into packets
    - Normal network delays, such as queueing delays
    - Playback delay from receiver's playback buffer
    - Decoding delay to reconstruct the signal
    - End-to-end delay is the accumulation of all these sources
        - Because VoIP is so sensitive to delays, VoIP applications have thresholds, such as 400 ms, beyond which packets are discarded

---

Delay limits for one-way transmission according to ITU-T Rec.G.114.

End-to-end delay (ms)	Quality
0 - 150	Acceptable for most users
150 - 400	Acceptable but has impact
400 and above	Unacceptable

End-to-end Delay

QoS for VoIP: Delay Jitter

1. Jitter
    - Different buffer sizes, queueing delays, and network congestion levels can delay packets by different amounts, called jitter
    - Jitter is problematic for VoIP because it interferes with reconstructing the analog voice stream
        - Too many dropped sequential packets can make the audio unintelligible
    - Mitigated by maintaining a "jitter buffer"
        - Hides variation in delay between different received packets by buffering them and playing them out for decoding at a steady rate
        - Long buffer reduces lost packets, but adds to end-to-end delay

Quiz 2

1. The rounding of samples to a discrete number in a particular range is called quantization.

- True

2. The only consideration when encoding VoIP is to use as little bandwidth as possible.
   - False
3. When using VoIP, all packets are transmitted, regardless of any end-to-end delay, to make sure that no message is unaccounted for.
   - False
4. Which service maintains a jitter buffer as a mechanism for mitigating jitter?
   - VoIP

QoS for VoIP: Packet Loss

1. Packet loss is inevitable
   - VoIP operates on the Internet, which is a "best-effort" service
     - Could use TCP to retransmit, but packets that are received too late are no good
     - Congestion control can also drop transmission rate to be lower than the receiver's drain rate
     - Typically use UDP
   - Packet loss for VoIP: A packet is lost if it either never arrives OR if it arrvies after its scheduled playout
     - VoIP can tolerate loss rates between 1 and 20 percent
   - Methods for dealing with packet loss
     - Forward Error Correction (FEC)
     - Interleaving
     - Error concealment
2. Forward Error Correction
   - Transmit redundant data alongside the main transmission, which allows the receiver to replace lost data with the redundant data
   - Redundant data could be lower quality as a backup
   - Redundant transmissions use more bandwidth
3. Interleaving
   - Doesn't transmit redundant data so doesn't add extra bandwidth
   - Works by mixing chunks of audio together so that if one set of chunks is lost, the lost chunks aren't consecutive
     - Many smaller audio gaps are preferable to one large audio gap
   - Increases latency because receiver has to wait longer for consecutive chunks
   - Limited utility with VoIP, but useful for streaming stored audio
4. Error Concealment
   - Guessing what the lost audio packet might be
   - Similarity between really small audio snippets (4-40 ms)
   - Computationally cheap and generally works well
   - Can also interpolate around lost packet, but this is more computationally expensive

Quiz 3

1. Most of the time, VoIP uses UDP to transmit audio.
2. Which of the following applications has less delay tolerance (i.e. a lower threshold in terms of when a packet is considered lost)?
   - VoIP
3. Which of the following services is the least sensitive to network delays? (i.e. more tolerant to network delays).
   - File transfer
4. Which of the following services is the most tolerant to packet losses?
   - VoIP
5. Which of the following decreases the end-to-end delay when using VoIP?
   - UDP

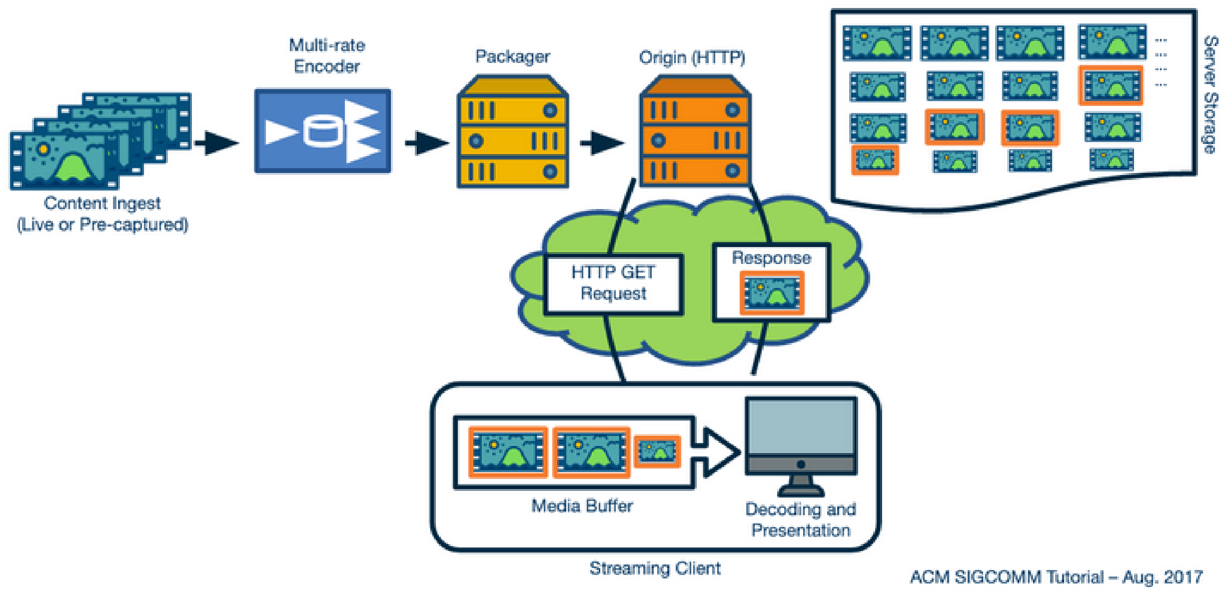
6. Which of the following is more likely to result in packet losses when using VoIP?
    - UDP
7. When using Forward Error Concealment, the redundant data that is transmitted alongside the main transmission could be a copy of the original data divided into chunks or could also be a lower-quality version.
    - True
8. An advantage of using Interleaving is that there is no added latency.
    - False
9. Error concealment can be computationally cheap if a lost packet is simply replaced with a previous packet.
    - True

Live/On Demand Streaming Introduction

1. Live Streaming
    - Account for 60-70% of Internet traffic
    - Enabling technologies
        - Bandwidth for core network and last-mile access links have increased tremendously over the years
        - Video compression technologies have become more efficient
        - Development of Digital Rights Management culture has encouraged content providers to put their content on the Internet
    - Two categories
        - Live streaming (sports, concerts)
        - On-demand (Netflix, Youtube)

Video Streaming Bigger Picture

1. Steps in Streaming
    - Video is created
        - Typically high quality
    - Compressed using an encoding algorithm
    - Secured using DRM and hosted over a server
    - End-users download the video content over the Internet
    - Content is decoded and rendered on the user's screen



Adaptive Streaming over HTTP

Sources of Redundancy in Video Compression

1. Compression
   - Lossy: Can not recover the high quality video
     - Gives higher savings in terms of bandwidth
   - Lossless: Original video can be recovered
   - Temporal redundancy: Consecutive images in a video are similar
   - Spatial redundancy: Nearby pixels in an image are similar

Image Compression

1. JPEG Compression
   - Transform image from RGC to color components (chrominance or Cb, Cr) and brightness (luminance or Y)
   - Operate on Cb, Cr, and Y independently
     - Divide the image into 8x8 blocks
     - Apply Discrete Cosine Transformation to each sub-image
   - Compress the matrix of coefficients using a pre-defined quantization table
     - Round to the nearest integer
   - Perform a lossless encoding to store the coefficients

Video Compression and Temporal Redundancy

1. How do we encode similar frames in a video?
   - Instead of encoding each JPEG separately, encode one and then the differences between images
     - I-frame: Initial frame
     - P-frame: Predicted frame
   - Encode an I-frame every 15 images
   - B-frame: Bi-directional frame; Encode a frame as a function of the past and future frames

VBR vs CBR

1. Variable vs Constant Bit Rate
   - Constant bit rate: Output size of the video is fixed over time
   - Variable bit rate: Output size of the video remains the same on average, but varies
     - More computationally expensive than CBR
   - Video compression is expensive in general

UDP vs TCP

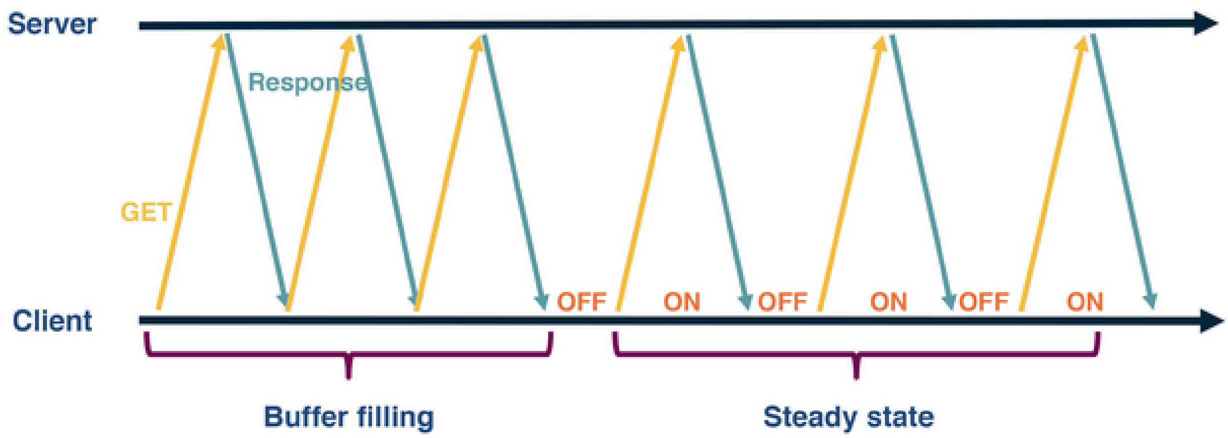
1. UCP or TCP for video delivery?
   - Video needs to be decoded at the client
     - Might fail if some data is lost
   - Content providers typically pick TCP for video delivery as it provides reliability
     - TCP provides congestion control which is required for effectively sharing bandwidth over the Internet

Why Do We Use HTTP?

1. What application protocol is used for video delivery?
   - Original vision was to have specialized video servers that stored the state of the client
     - When a client paused a video, it would signal to the server to stop sending video
     - Clients would have to do a minimal amount of work
     - Requires providers to buy specialized hardware
   - Another option is to use HTTP
     - Server is stateless, intelligence for downloading video is stored at the client
     - Major advantage is that content providers could use the existing CDN infrastructure
     - Easier to bypass middleboxes and firewalls as they already understood HTTP

Progressive Download vs Streaming

1. Downloading or Streaming
   - Download entire file through an HTTP GET request
     - Users could leave the video mid-way, wasting network resources
     - Would require storing the entire video in memory
   - Instead, send byte-range requests
     - Because Internet is best-effort, client pre-fetches some video ahead and stores it in a playout buffer
   - Filling state: Video buffer is empty and client tries to fill it as soon as possible
   - Steady state: Video buffer is full, so client waits for it to become lower than a threshold, then sends a request for more content



Progressive Download vs Streaming

How to Handle Network and User Device Diversity?

1. Diversity
   - Device: Smartphone vs TV
   - Network environment: Ethernet/WiFi vs Cellular
   - Throughput: Family members using bandwidth
   - Due to this diversity, content providers encode their video at multiple bitrates chosen from a set of pre-defined bitrates
     - Higher bitrate means higher quality
   - Bitrate adaptation: Picking the best bitrate based on current circumstances
     - Client downloads a manifest file containing all the metadata about the video and associated URLs

Quiz 4

1. Video delivery is tolerant to packet losses. Reliability of packet delivery is not that important.
   - False
2. Content providers store all the intelligence to download the video at the server.
   - False
3. What is the first item downloaded by a client's video player?
   - A manifest file
4. Suppose that Alice is using a cloud service to listen to many MP3 songs, one after the other, each encoded at a rate of 128 kbps. Suppose that she downloads for 30 minutes (1800 seconds). How many Mbytes of data are transferred during the 30-minute session? Round your answer to the nearest Mbytes. (For simplification, assume 1 MB = 8,000 Kb).
   - 29 MB
5. Suppose that Bob is using a cloud service to watch video encoded at a rate of 2 Mbps. Suppose that his session lasts for 30 minutes (1800 seconds). How many Mbytes of data are transferred during the 30-minute session? Round your answer to the nearest MB. (For simplification, assume 1 MB = 8,000 Kb).
   - 450 MB

Bitrate Adaptation in DASH

1. DASH

- Dynamic Streaming over HTTP (DASH): Client dynamically adjusts the video bitrate based on the network conditions and device type
  - HLS and MPEG-DASH are the most popular implementations
- In DASH, a video is divided into chunks and each chunk is encoded in multiple bitrates
  - Each time a chunk is downloaded, the client calls the bitrate adaptation function f
  - Algorithm adapts the video bitrate based on its estimation of the network conditions

What are the Goals of Bitrate Adaptation?

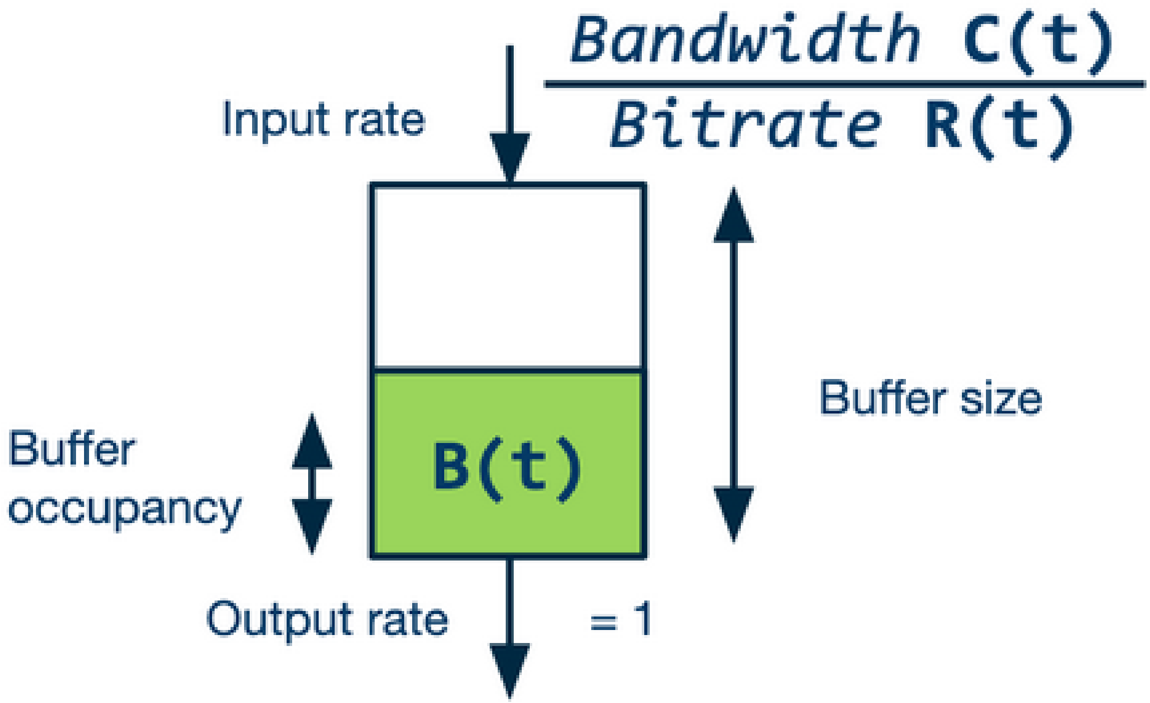
1. Goals of bitrate adaptation
   - Low or zero re-buffering: Users close a session if the video stalls often
   - High video quality: Better the video quality, better the user QoE. A higher video quality is usually characterized by high bitrate video chunk
   - Low video quality variations: Many quality variations reduce user QoE
   - Low startup latency: Time it takes for video to start playing should be short
   - Some of these metrics are at odds with each other; can't win them all
     - Goal is to consider tradeoffs and maximize user quality of experience

Bitrate Adaptation Algorithms

1. Inputs to bitrate adaptation algorithm
   - Network throughput: Want to select a bitrate that is less than or equal to the available throughput
     - Rate-based adaptation
   - Video buffer: Amount of video in the buffer can inform the video bitrate of the next chunk
     - Full buffer means we can afford to download high quality chunks

Throughput-based Adaptation and its Limitations

1. Rate-based adaptation
   - Buffer-filling rate is network bandwidth divided by the chunk bitrate
   - Buffer-depletion rate is 1
   - To have stall-free streaming, we need the buffer-filling rate to be greater than the buffer-depletion rate
     - $C(t)/R(t) > 1$ or $C(t) > R(t)$



Buffer-fill and Depletion Rates

Rate-based Adaptation Mechanisms

1. Adaptation Mechanisms
   - Estimation: Estimate future bandwidth by considering the throughput of the last few downloaded chunks
     - Smoothing filter (moving average or harmonic mean) is used over these throughputs to estimate the future bandwidth
   - Quantization: Continuous throughput is mapped to discrete bitrate
     - Select max bitrate less than the throughput estimate, including a factor in this selection
   - Why do we add a factor?
     - Want to be conservative in our estimate of future bandwidth to avoid re-buffering
     - If chunks are VBR-encoded, their bitrate can exceed normal bitrate
     - Additional application and transport-layer overheads associated with downloading the chunk
   - Client only requests next chunk when there is space in its buffer

Issues with Bitrate Adaptation

1. Issues
   - Rate-based adaptation can end up overestimating or underestimating the future bandwidth which can lead to selection of a non optimal chunk bitrate

Problem of Bandwidth OVER-Estimation with Rate-based Adaptation

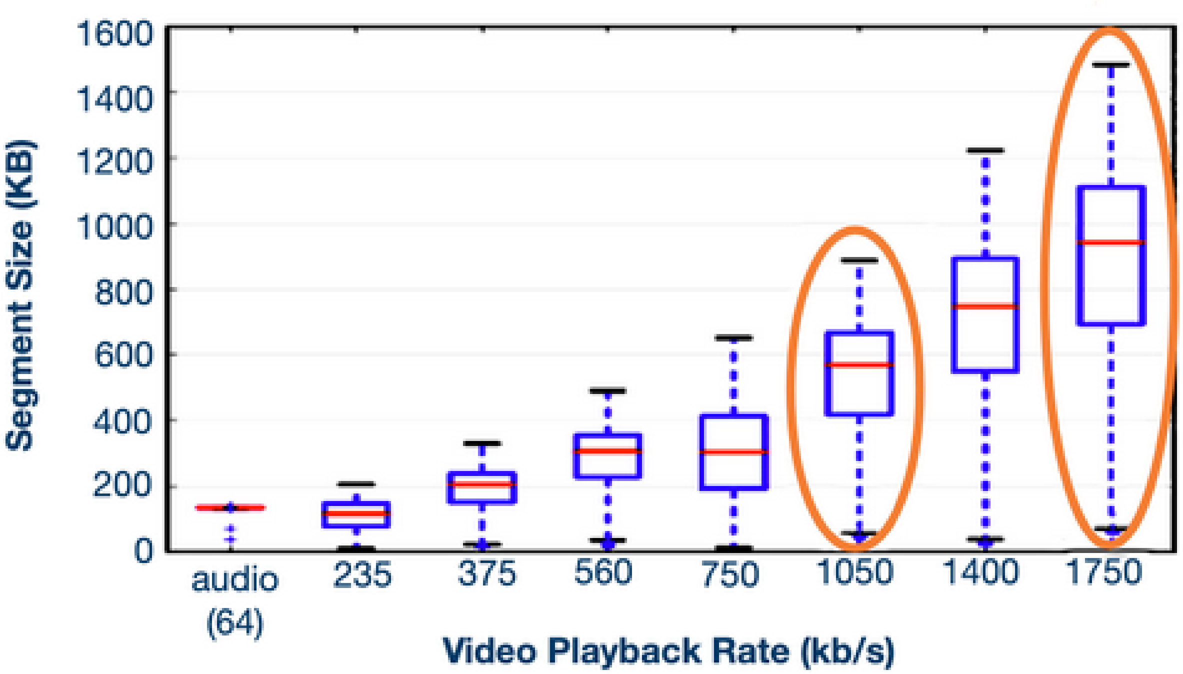
1. Over-estimation
   - When the bandwidth changes rapidly, the client has no way of knowing instantaneously
   - Takes time to converge to the right estimate of the bandwidth

Quiz 5

1. One of the goals of quality of experience is to have high re-buffering.
    - False
2. Assume the available network bandwidth is 15 Mbps and the bitrate of the chunk is 3 Mbps. Determine the buffer-filling rate and the buffer-depletion rate.
    - Filling rate: 5
    - Depletion rate: 1
3. Calculate how long it takes to download a 5-second chunk.
    - 50 seconds

Problem of Bandwidth UNDER-Estimation with Rate-based Adaptation

1. Under-estimation
    - As bitrate decreases, chunk size also reduces
    - In the presence of a competing flow, a smaller chunk size would lower the probability for the video flow to get its fair share
    - Client ends up further underestimating the network bandwidth and picks up an even lower bitrate until it converges
    - Occurs because of the ON-OFF behavior in DASH
        - Two competing TCP flows would get their fair share as TCP is fair



Problem of Underestimation

Rate-based Adaptation Conclusion

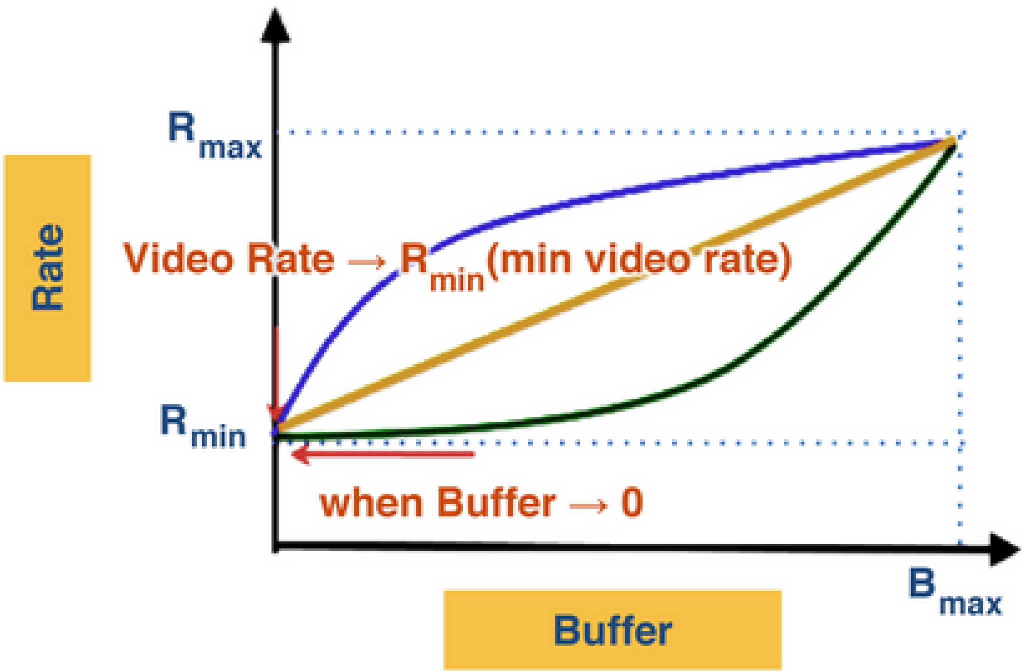
1. Rate-based adaptation
    - Pick the chunk bitrate based on estimation of available network bandwidth
    - Actual available bandwidth is unknown and variable, so it uses past throughput as a proxy for the available bandwidth

- Reactive estimation can lead the player to sometimes underestimate or overestimate the bandwidth in different scenarios

Bitrate Adaptation Algorithm: Buffer-Based

1. Using buffer occupancy to inform bitrate selection
   - If buffer occupancy is low, player should download a low bitrate chunk and increase the chunk quality of the buffer occupancy increases
2. Benefits
   - Avoids unnecessary re-buffering
   - Fully utilizes the link capacity and does not suffer from bandwidth underestimation

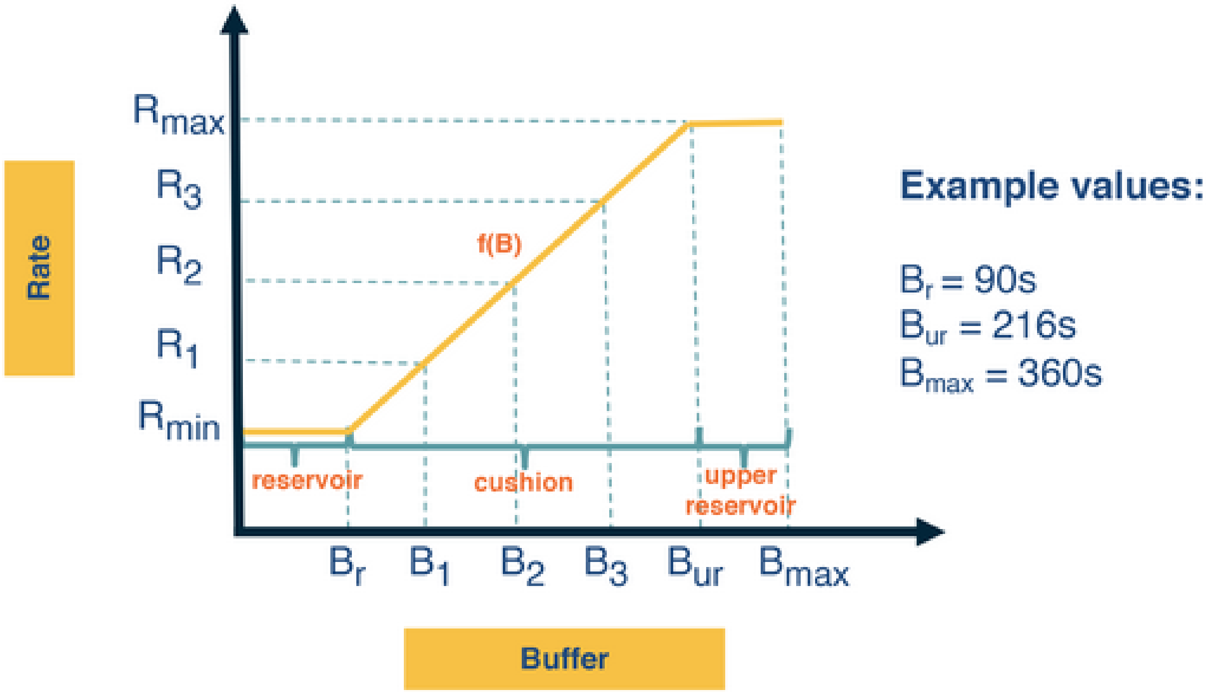
---



Bitrate Adaptation Algorithm

Buffer-Based Adaptation Example

1. Buffer-based function
   - Reservoir region corresponds to low buffer occupancy
     - Player always selects minimum available bitrate
   - Player selects highest available bitrate in the upper reservoir region
   - In the cushion region, bitrate is a linear function of the buffer occupancy



Bitrate Adaptation Algorithm Example

Issues with Buffer-based Adaptation

1. Issues
   - In startup phase, buffer occupancy is 0, meaning the player will download low quality chunks (maybe unnecessary)
   - Can lead to unnecessary bitrate oscillations
   - Requires a large buffer to implement the algorithm efficiently

Bitrate Adaptation Conclusion

1. Conclusion
   - Area of active research
   - Most video players use both signals