

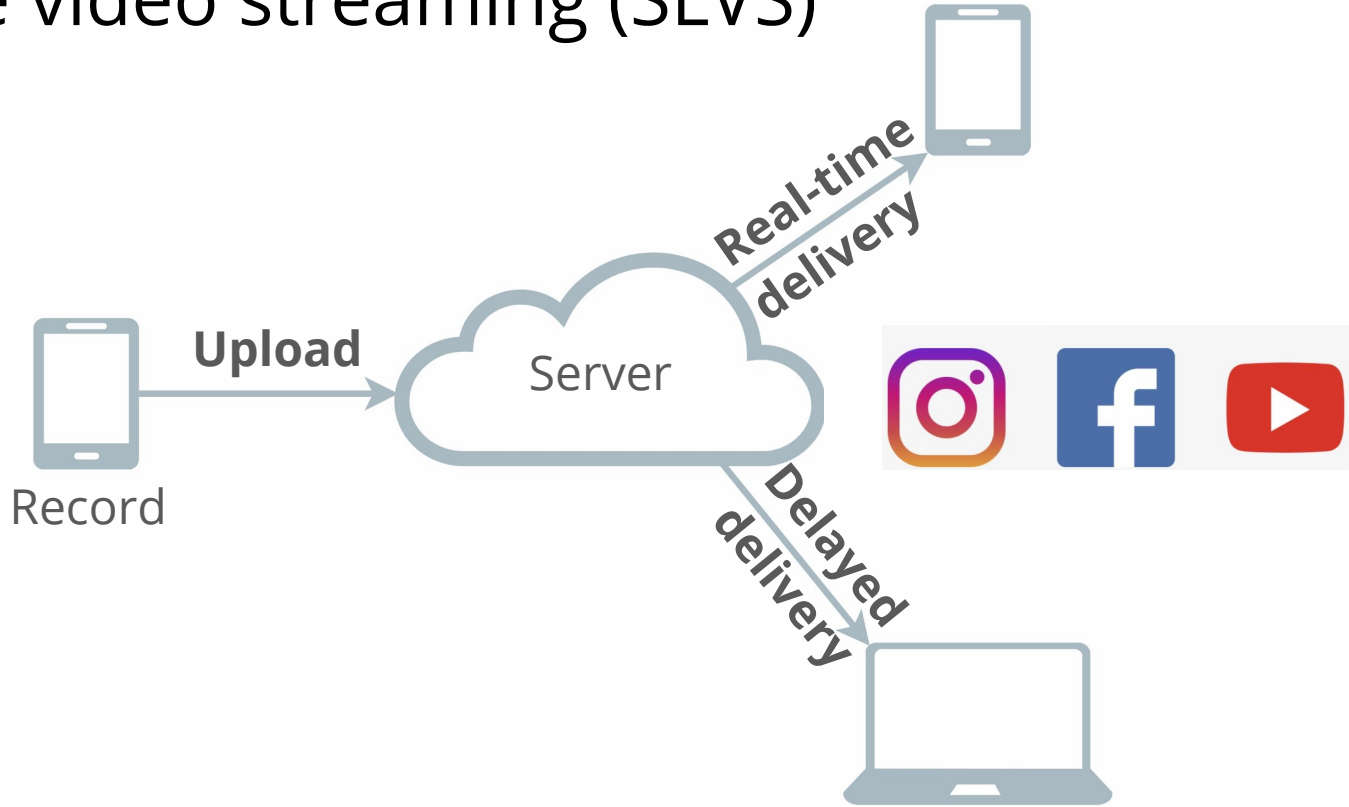
# Vantage

Optimizing video upload for time-shifted  
viewing of social live streams  
(SIGCOMM 2019)

**Devdeep Ray**, Jack Kosaian,  
Rashmi Vinayak, Srinivasan Seshan

**Carnegie Mellon University**

# Social live video streaming (SLVS)



# Challenges in mobile social live streaming

Limitations of current techniques

Vantage:

Key ideas

Design and implementation

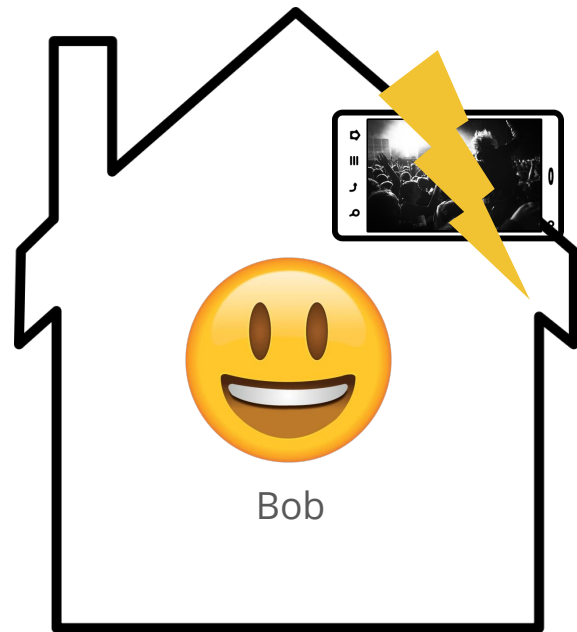
Evaluation



Alice



Network impairments tolerated by real-time viewers



Bob

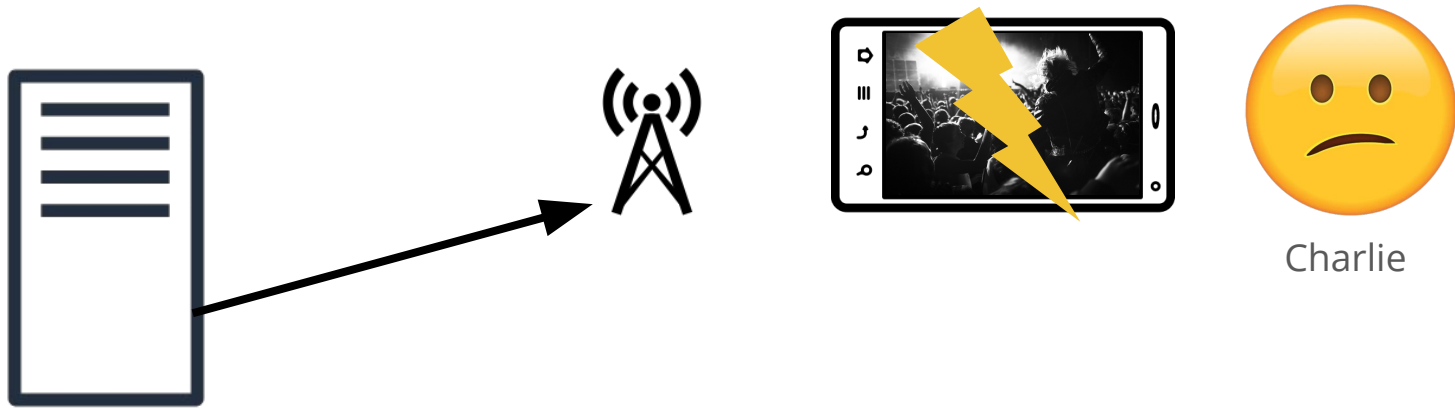


Charlie  
(Attending SIGCOMM during concert)



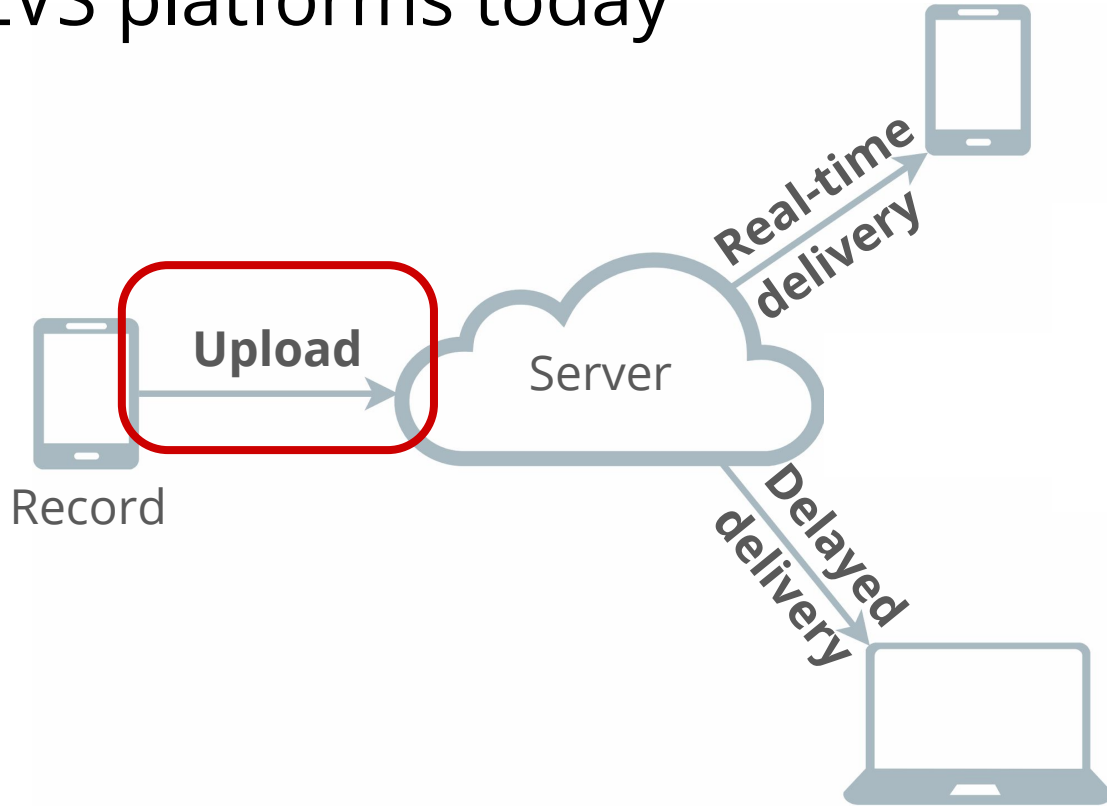


**One  
Eternity  
Later**



Delayed viewers also affected by network impairments

# Typical SLVS platforms today



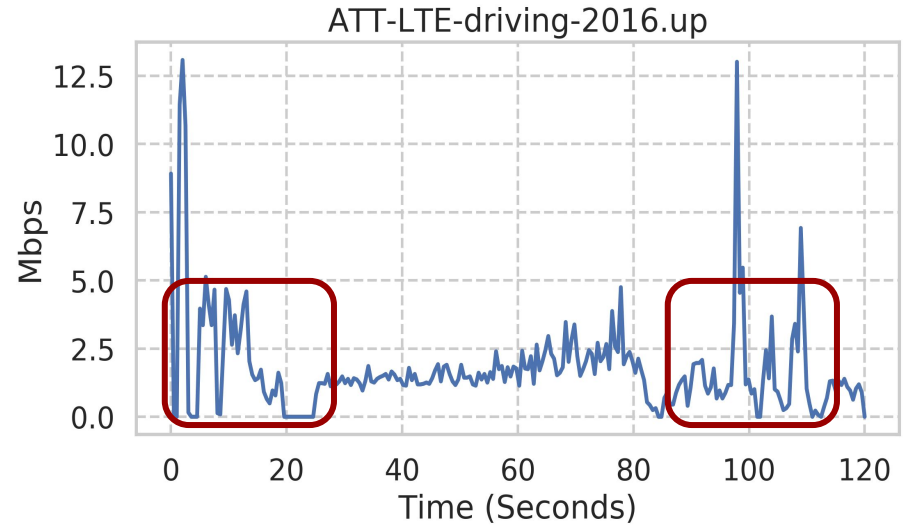
# Video upload path is critical

Mobile streaming common

Significant bandwidth variation on upload path

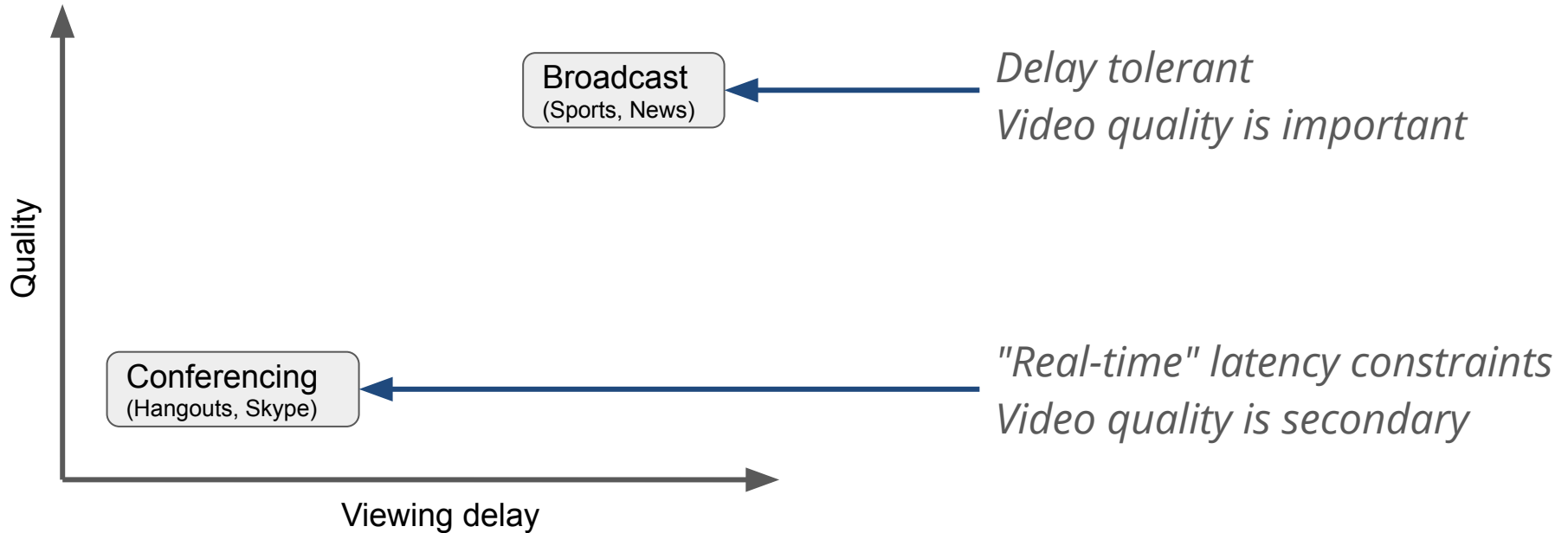
The uploaded video is a baseline for all viewers

Downstream optimizations are limited by upload quality



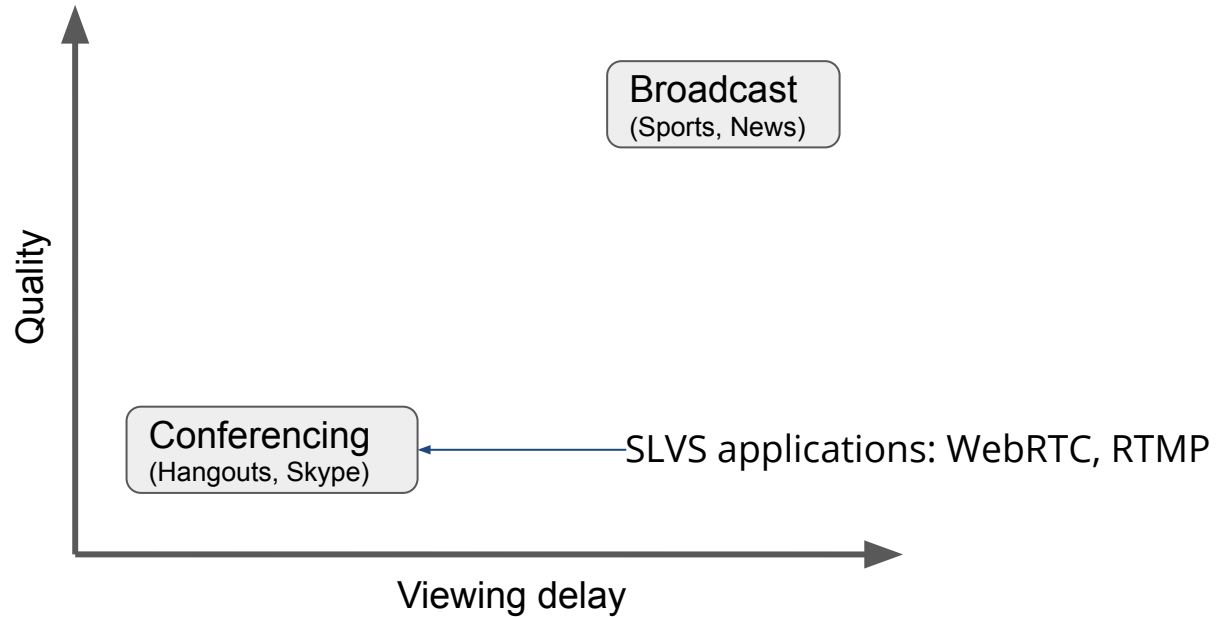
**Available network bandwidth**

# Live video streaming today

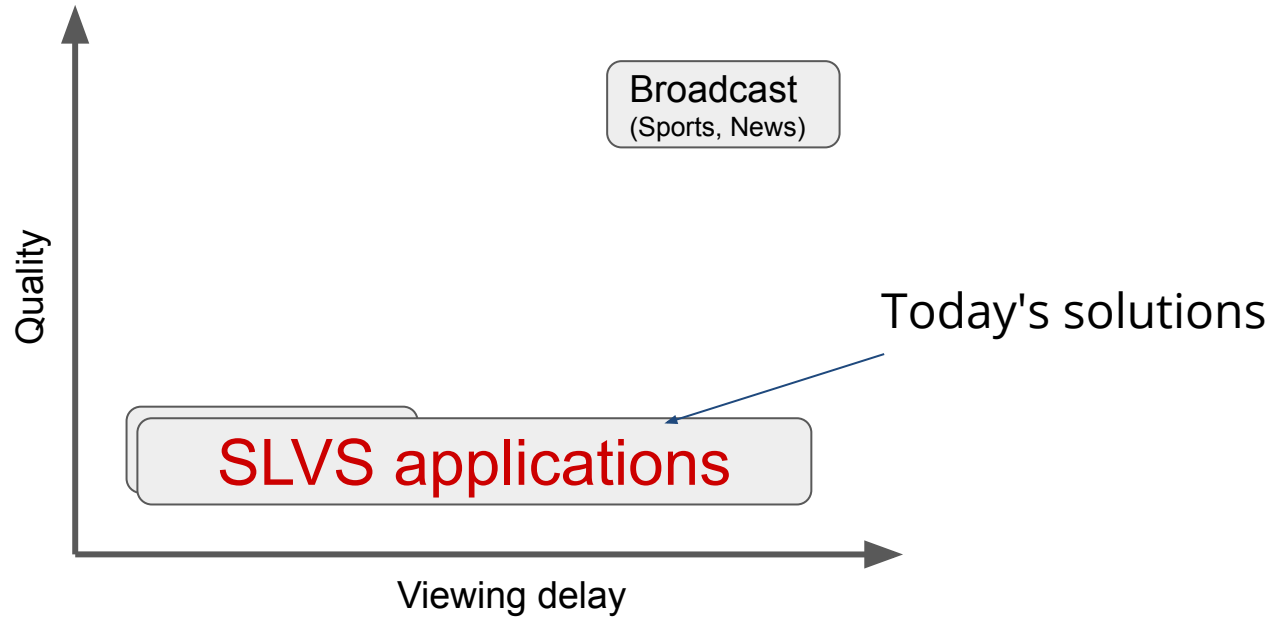


Social live streaming has **both** real time and delay-tolerant viewers *for the same session*

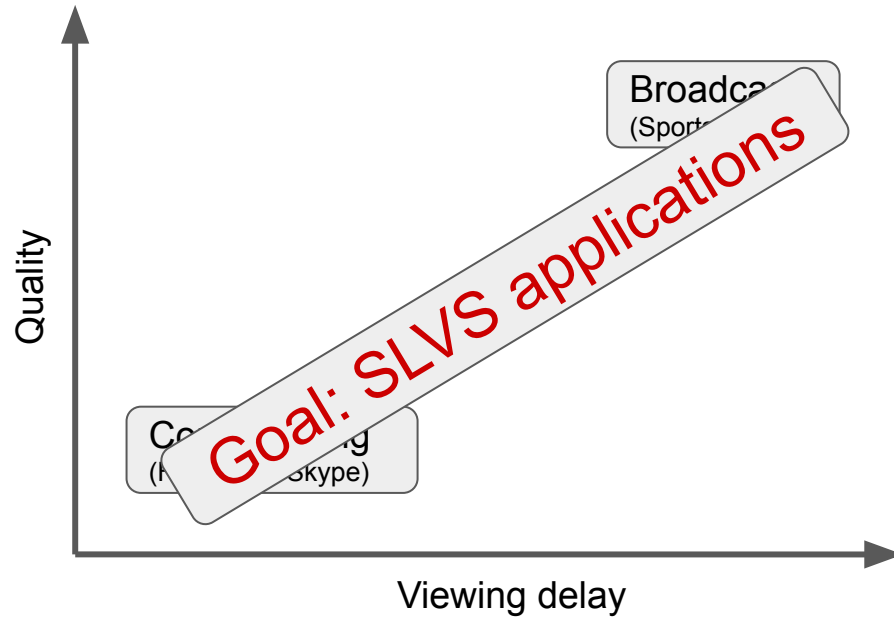
# SLVS applications use conferencing techniques



# SLVS today: Same video quality for all viewing delays



# Goal: Better quality for delayed viewers



# Challenges in mobile social live streaming

## **Limitations of current techniques**

Vantage:

Key ideas

Design and implementation

Evaluation

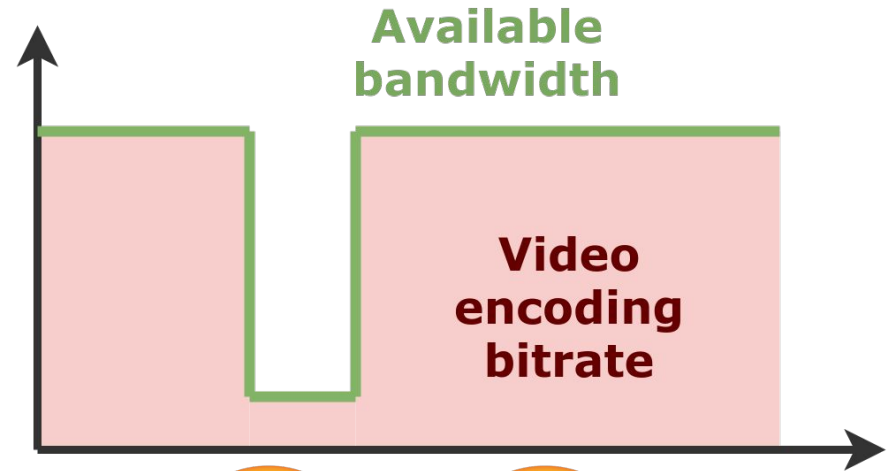
# Existing upload techniques: Real-time streaming

**Conferencing**  
(Skype, Hangouts, ..)

"Real-time" latency constraints

Bitrate closely matches available bandwidth

Sensitive to bandwidth variation



Bob  
(Real-time)



Charlie  
(Delayed)

# Existing upload techniques: Buffered streaming

## Broadcasting

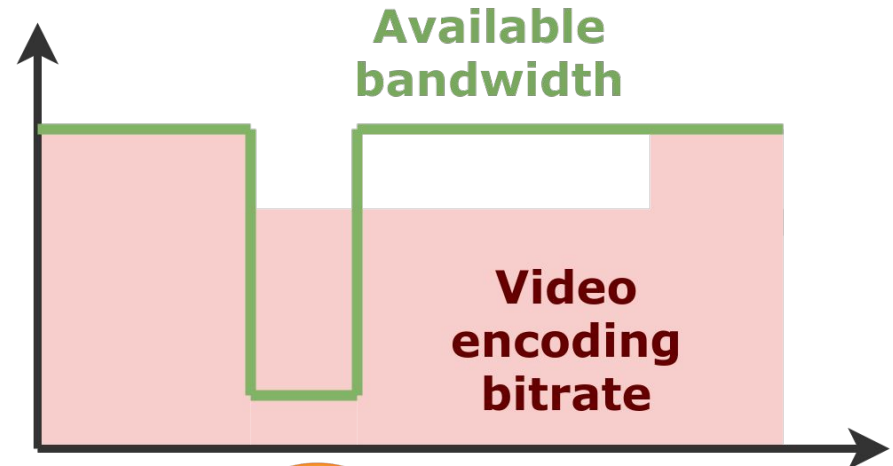
(Entertainment, News, ..)

Delay tolerant

Encode at ~ average bandwidth

Large sender-side buffers to absorb bandwidth variation

Higher video quality, no interactivity



Bob  
(Real-time)



Charlie  
(Delayed)

Existing upload techniques

**Inadequate for SLVS:  
Delayed + high quality video  
OR  
Interactive video**

Challenges in mobile social live streaming

Limitations of current techniques

**Vantage:**

**Key ideas**

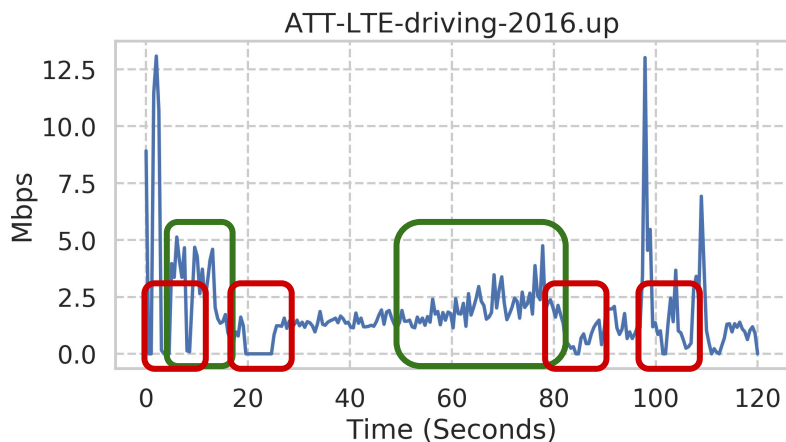
Design and implementation

Evaluation

# Observation #1: Bandwidth is highly variable

Analyzed traces from the Mahimahi \*\* project

Significant variations observed, with **extreme lows and highs**



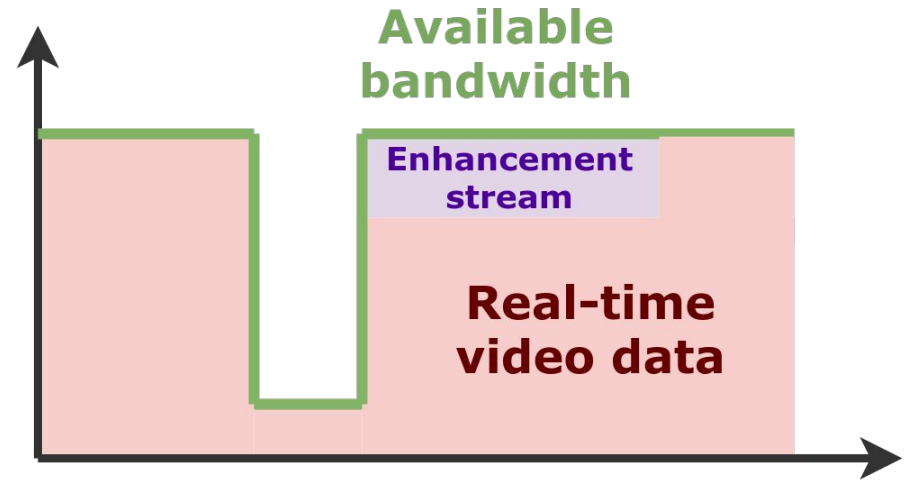
\*\* Netravali, Ravi, et al. "Mahimahi: a lightweight toolkit for reproducible web measurement." *ACM SIGCOMM Computer Communication Review* 44.4 (2015): 129-130.

# Catering to multiple viewing delays

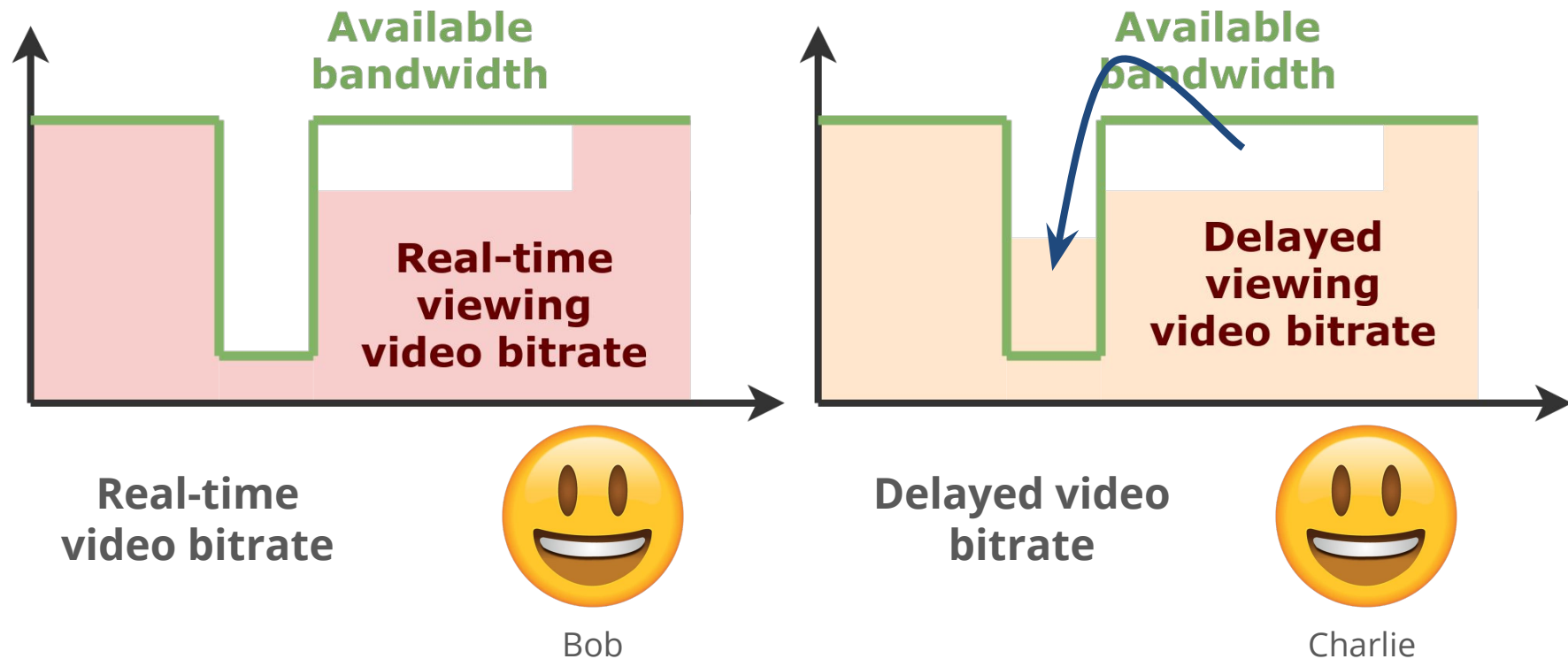
**Network is impaired:** Use real-time strategy

**When network recovers:** Use less than capacity for real-time

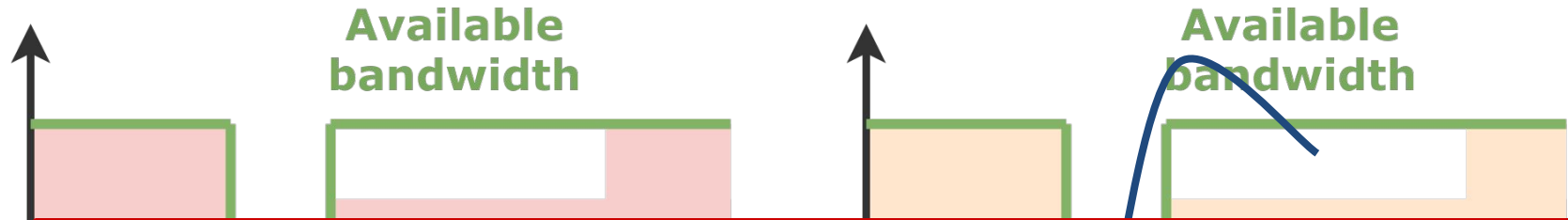
Excess bandwidth used to repair past segments



# Viewing quality for different delays



# Viewing quality for different delays



**Why is Bob  
(real-time viewer) okay with this?**

# Aside: Video quality metrics

**Video bitrate != video quality**

Vantage uses SSIM for measuring perceived video quality



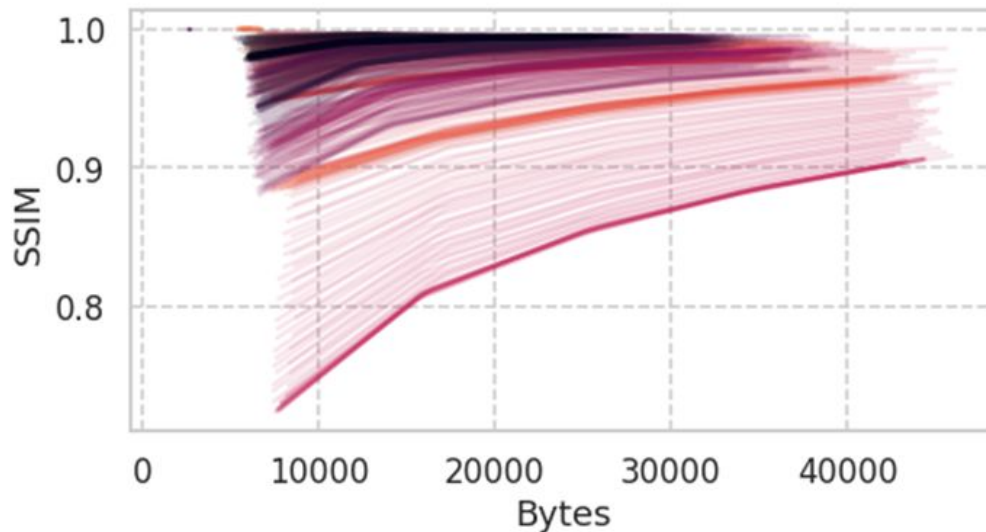
SSIM = 1.0



SSIM = 0.66

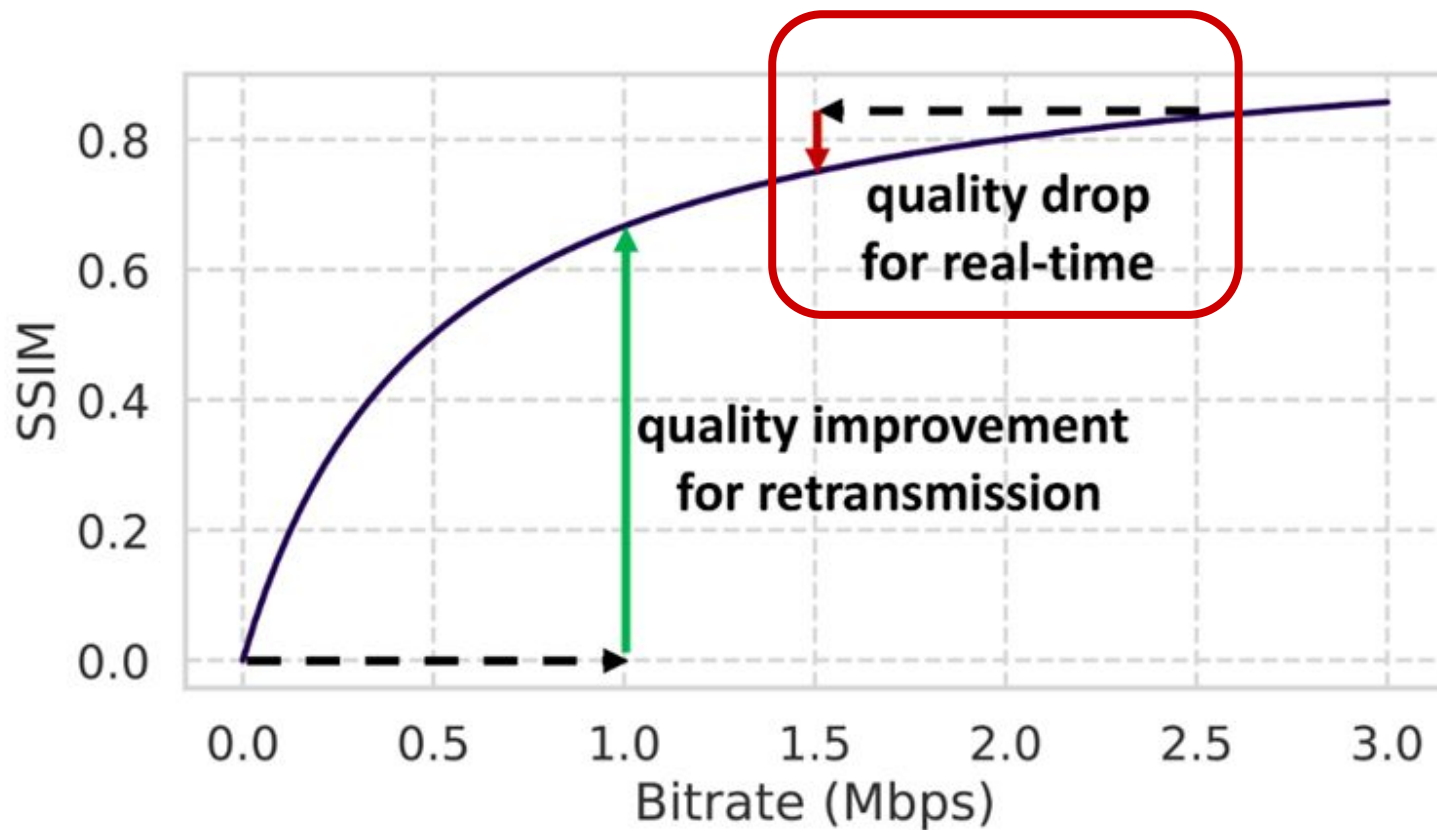
## Observation #2: Quality vs. Frame size is concave

Video encoded multiple times at different bitrates

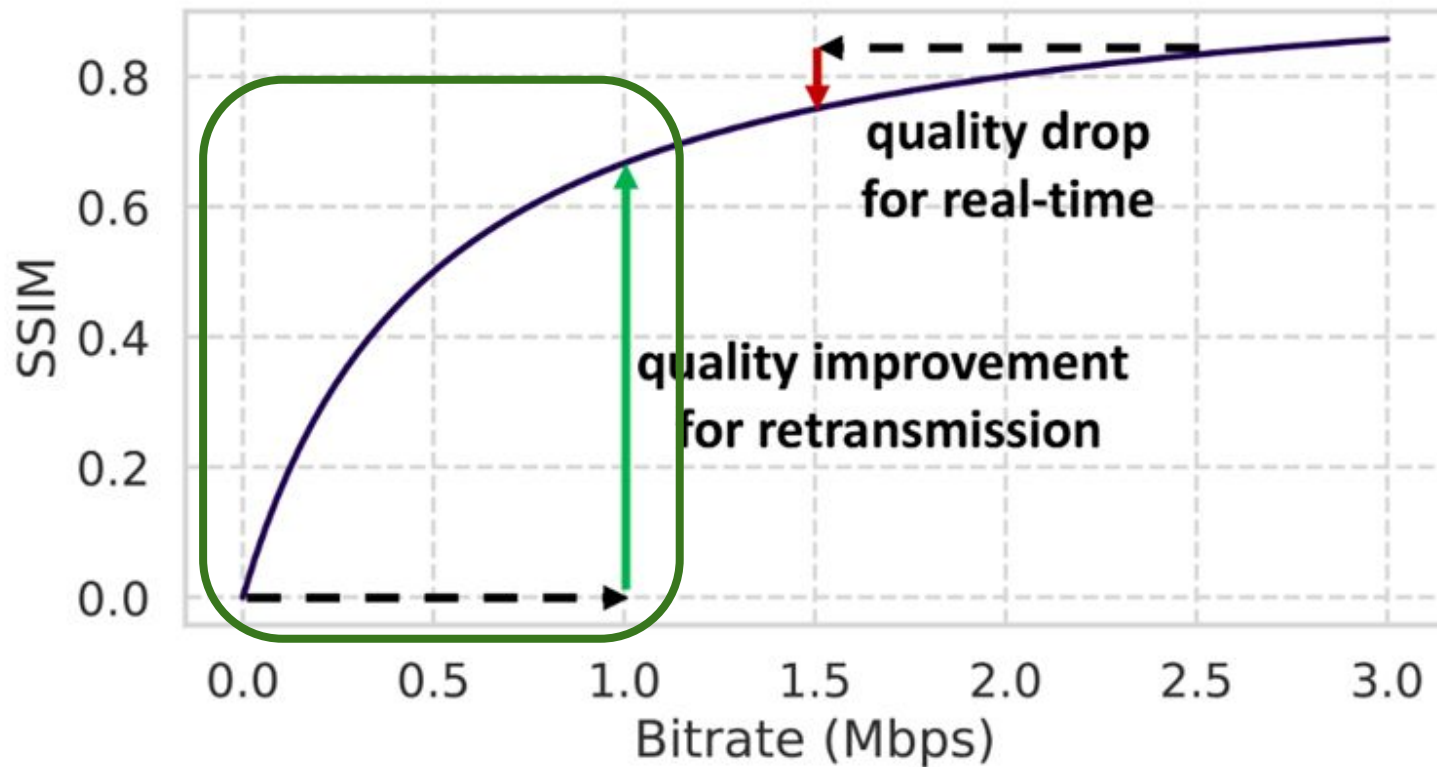


Size vs. SSIM plot for each frame

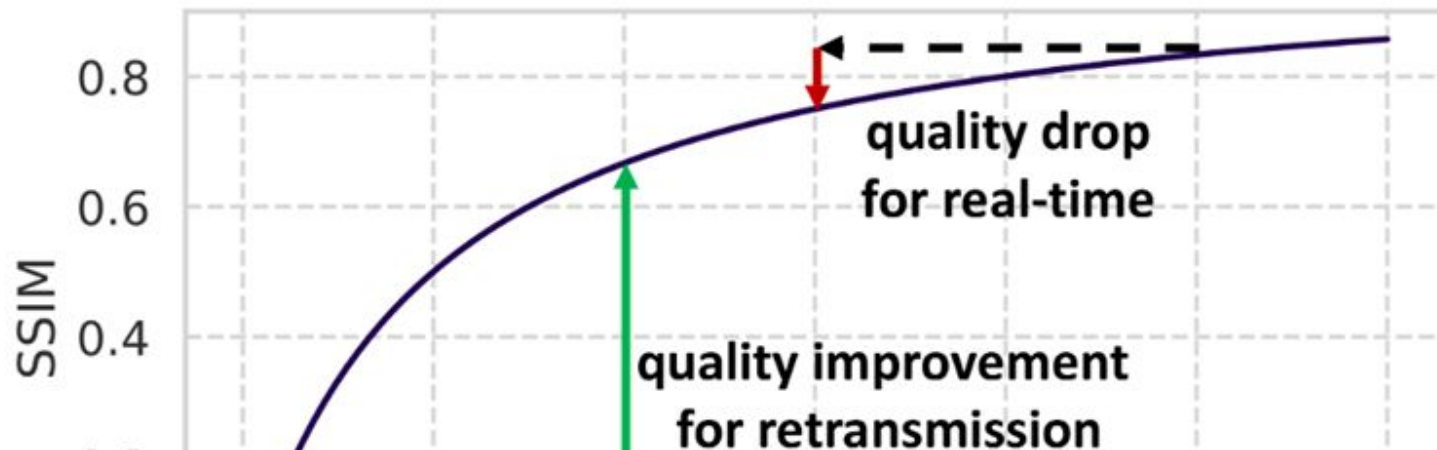
## Observation #2: Quality vs. Frame size is concave



## Observation #2: Quality vs. Frame size is concave



Observation #2: Quality vs. Frame size is concave



**Using high bandwidth to improve low quality frames very powerful!**

Challenges in mobile social live streaming

Limitations of current techniques

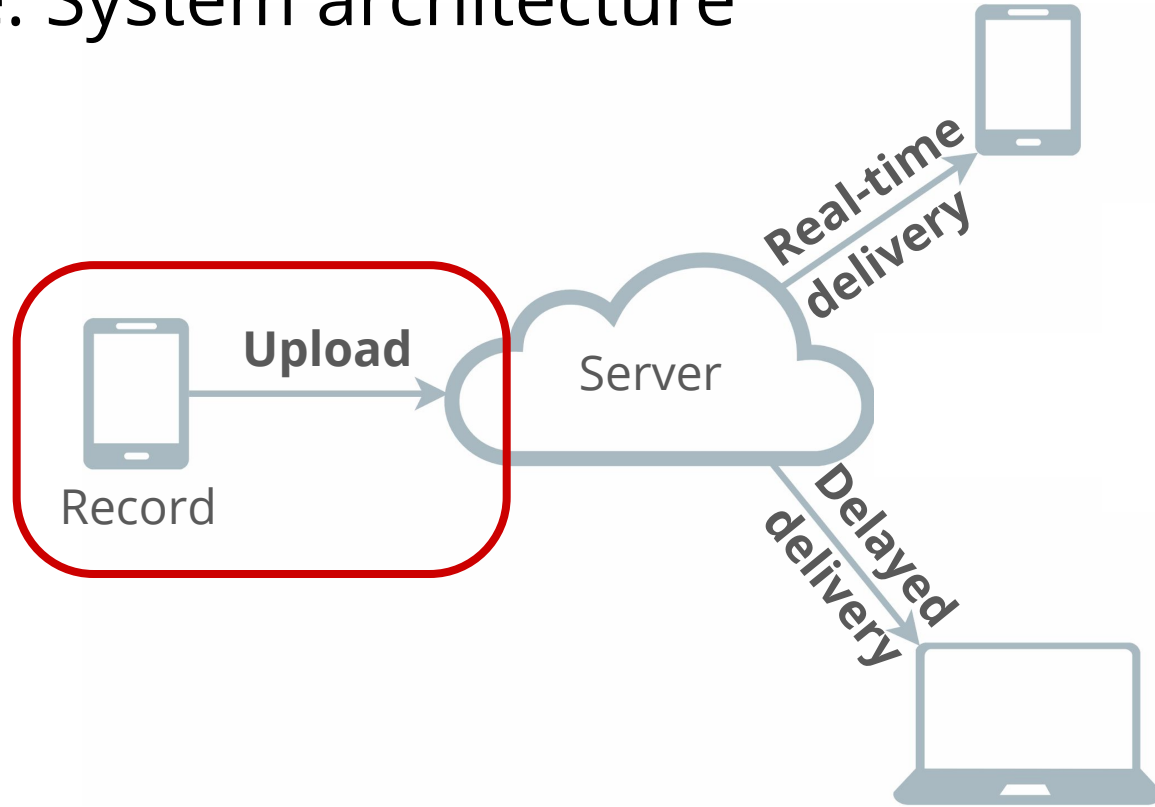
**Vantage:**

Key ideas

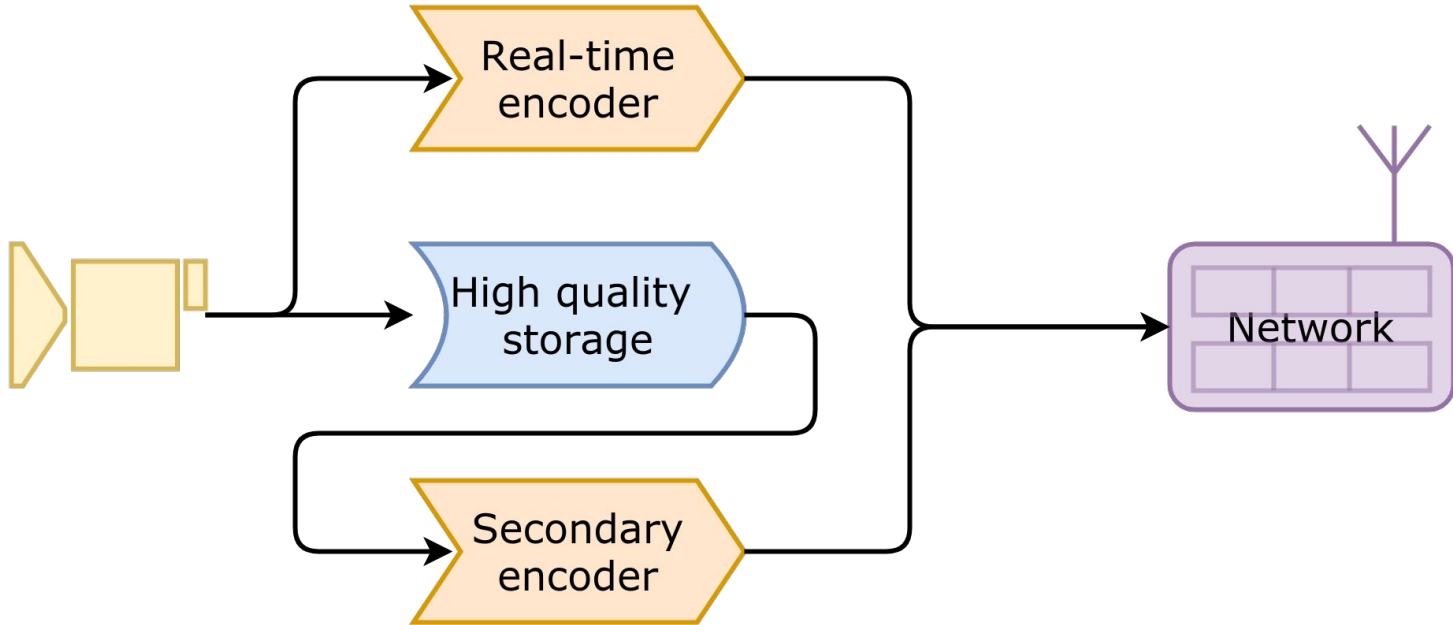
**Design and implementation**

Evaluation

# Vantage: System architecture



# Vantage: Streamer Architecture



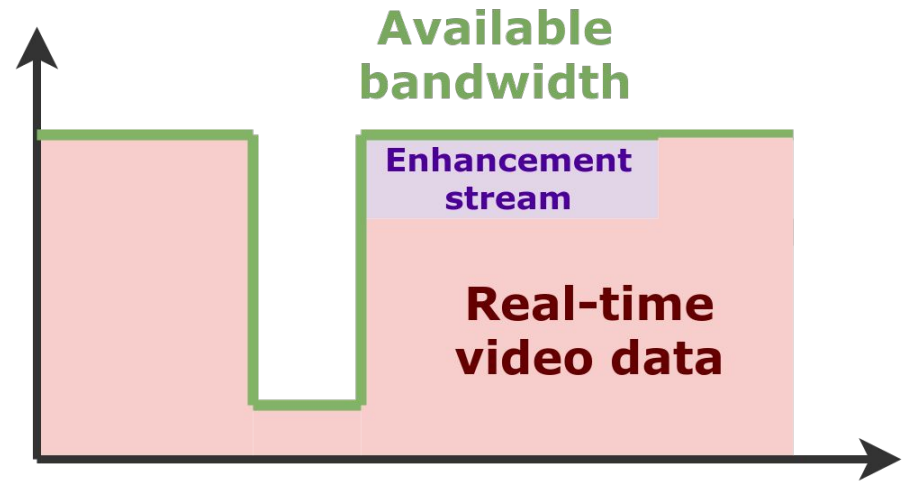
# Vantage: Key challenges

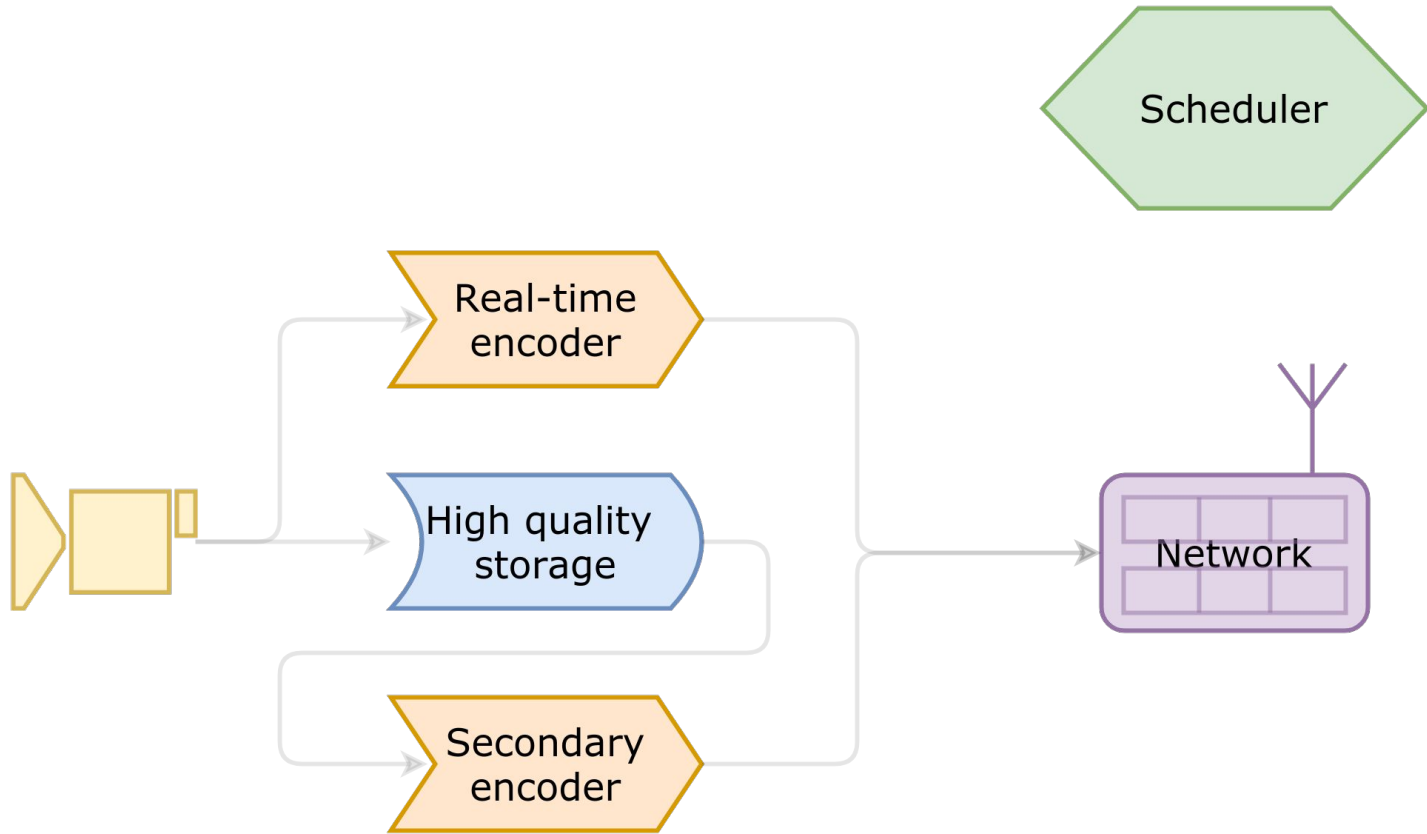
**Real-time decisions that optimize video quality for all viewing delays**

Real-time video stream bitrate

Enhancement frame selection

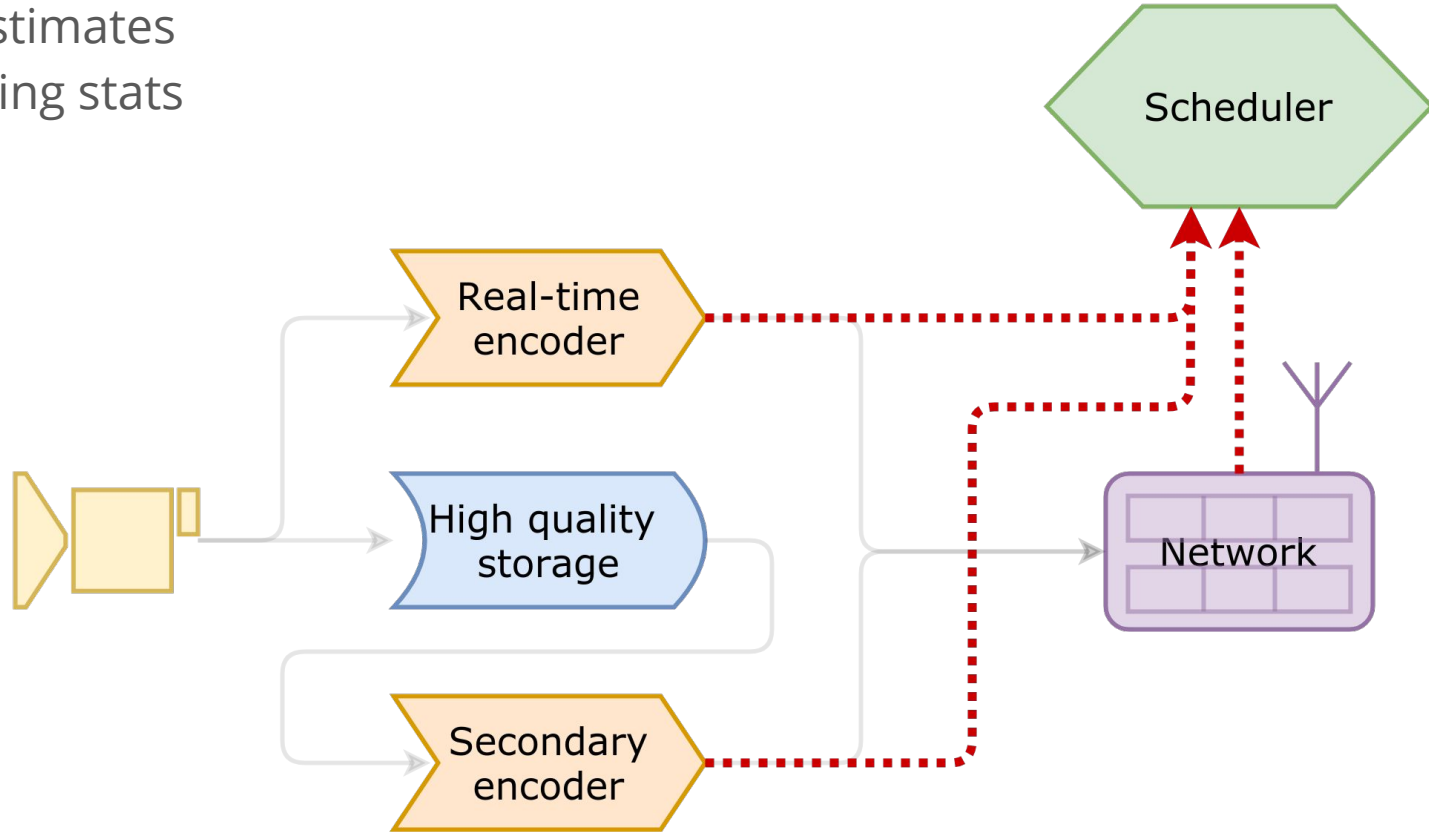
Video enhancement stream bitrate





## Input:

1. Bandwidth estimates
2. Frame encoding stats

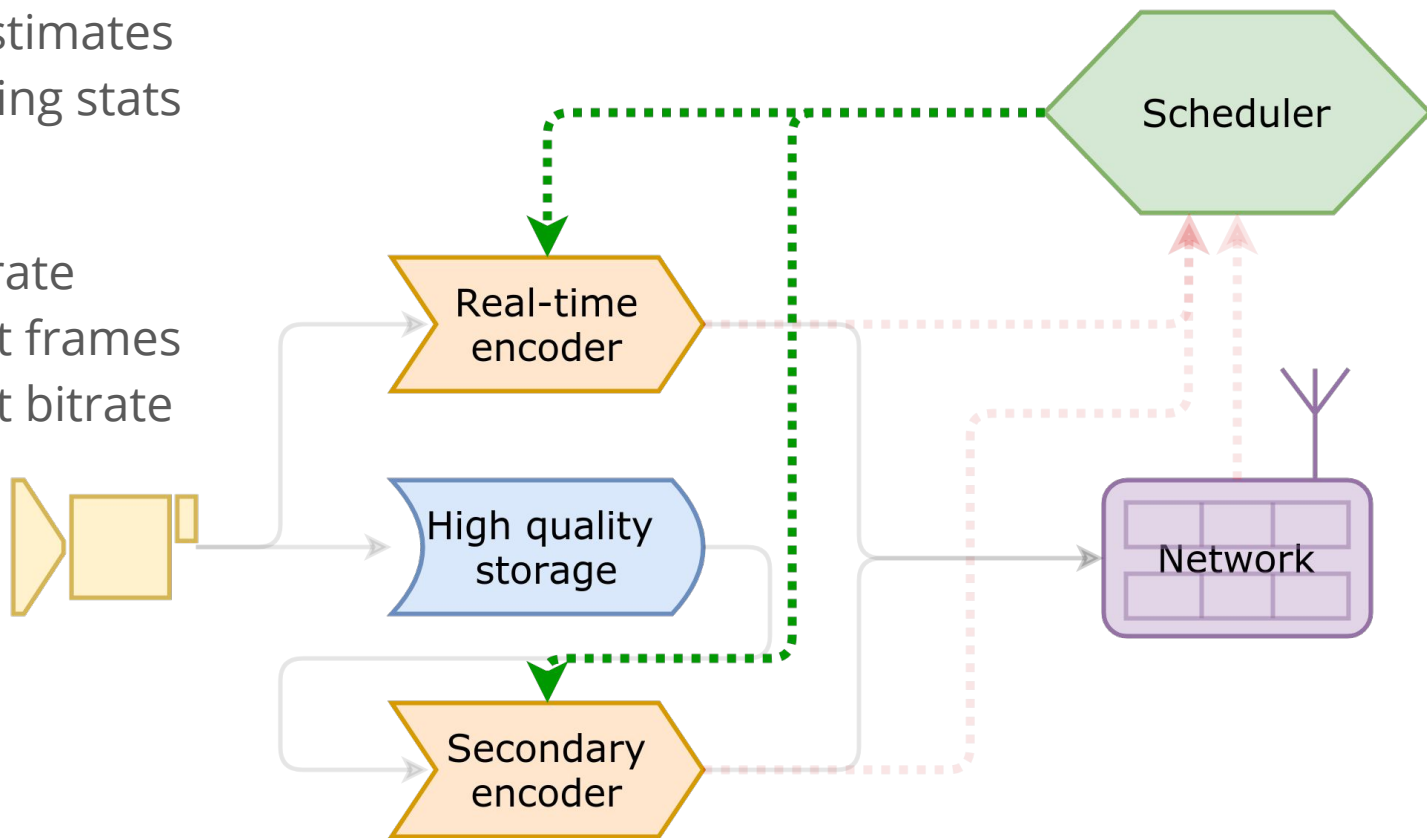


## Input:

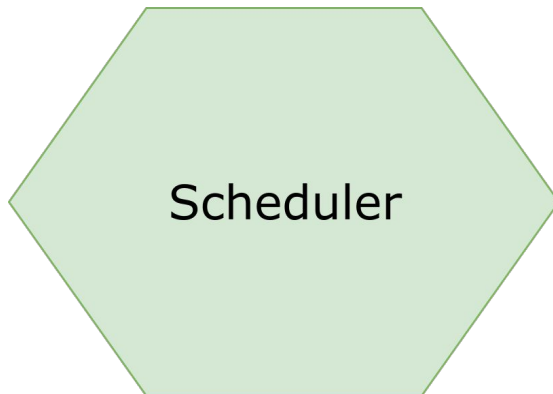
1. Bandwidth estimates
2. Frame encoding stats

## Output:

1. Real-time bitrate
2. Enhancement frames
3. Enhancement bitrate



# Scheduling goals



- 1. Constrain encoded bits to the available bandwidth**
- 2. Optimize video quality across multiple viewing delays**

# Vantage scheduler

Mixed integer program (MIP) maximizes quality for multiple viewing delays

Periodically generates video encoding schedule

## **Key challenges:**

### **Handling stale bandwidth estimates**

Mapping frame sizes to quality

# Scheduler period trade-offs

**Short time period**

**Accurate** bandwidth estimates

**Short sighted** scheduling

**Long time period**

**Stale** bandwidth estimates

**Long term** optimal scheduling

# Handling stale network estimates

**Dual approach: Long term (MIP) + short term (Execution engine)**

Vantage: MIP generates schedule every 2 seconds

Fallback strategy: Execution engine prioritizes real-time

# Vantage scheduler

Mixed integer program (MIP) maximizes quality for multiple viewing delays

Periodically generates video encoding schedule

## **Key challenges:**

Handling stale bandwidth estimates

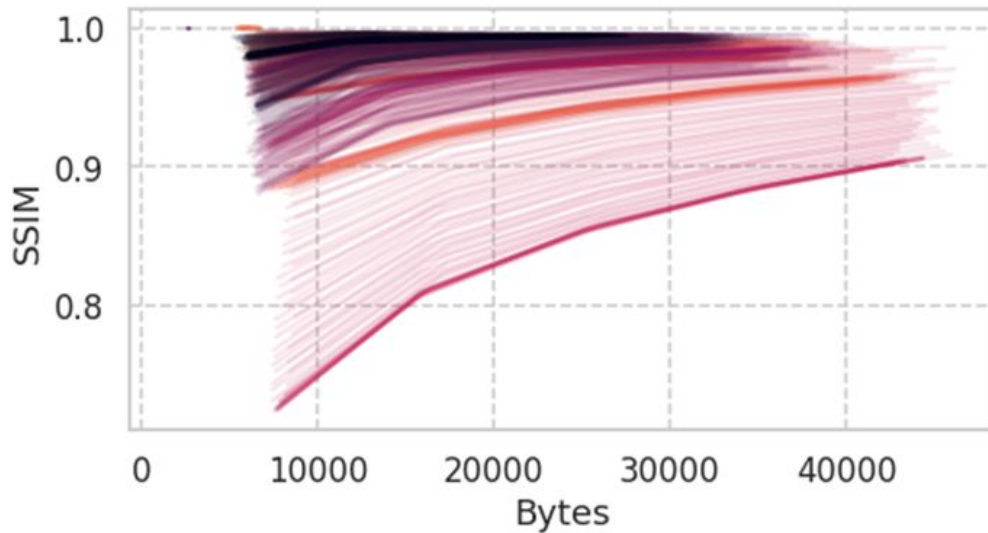
**Mapping frame sizes to quality**

# Frame size vs SSIM estimation

Frame size vs SSIM curve  
needed for optimization

Statistics from encoders drive  
estimation

Simple non-linear model: works  
well



**Size vs. SSIM plot for each frame**

Challenges in mobile social live streaming

Limitations of current techniques

**Vantage:**

Key ideas

Design and implementation

**Evaluation**

# Vantage: Evaluation

Designed to work with existing congestion control protocols for real-time video

Emulated transport layer that provides network estimates from traces

Evaluation performed on different combinations of videos and network traces

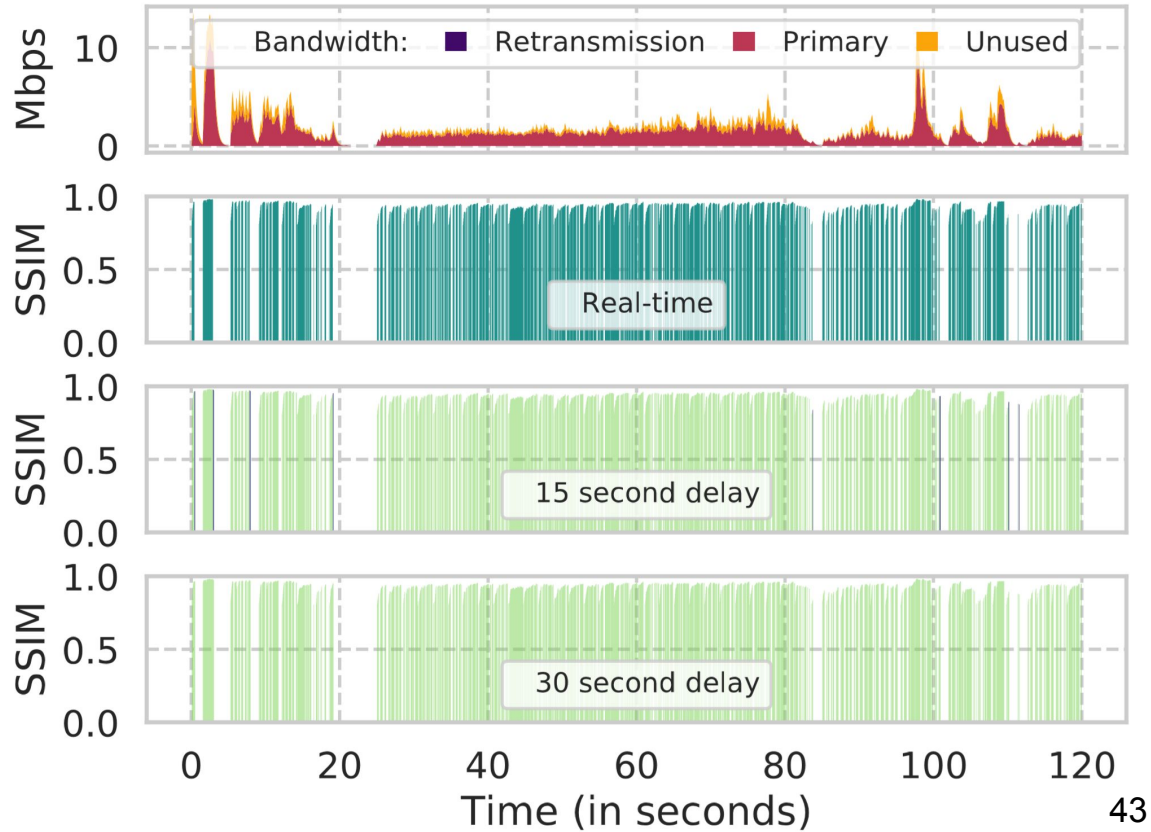
Videos: Animated, talking head, drone footage

Network traces: LTE (Verizon, ATT), UMTS (T-Mobile) \*\*

\*\* Netravali, Ravi, et al. "Mahimahi: a lightweight toolkit for reproducible web measurement." *ACM SIGCOMM Computer Communication Review* 44.4 (2015): 129-130.

# Baseline: Real-time (conference style) streaming

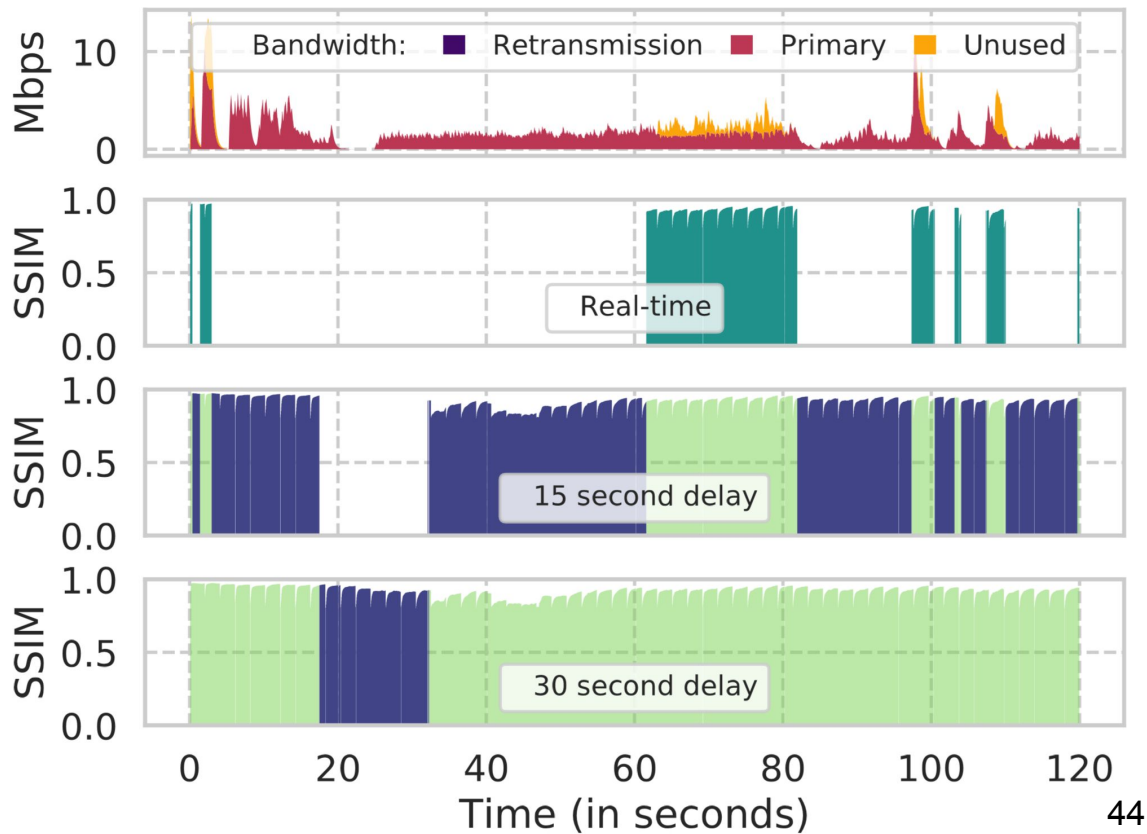
All viewers affected by network variations



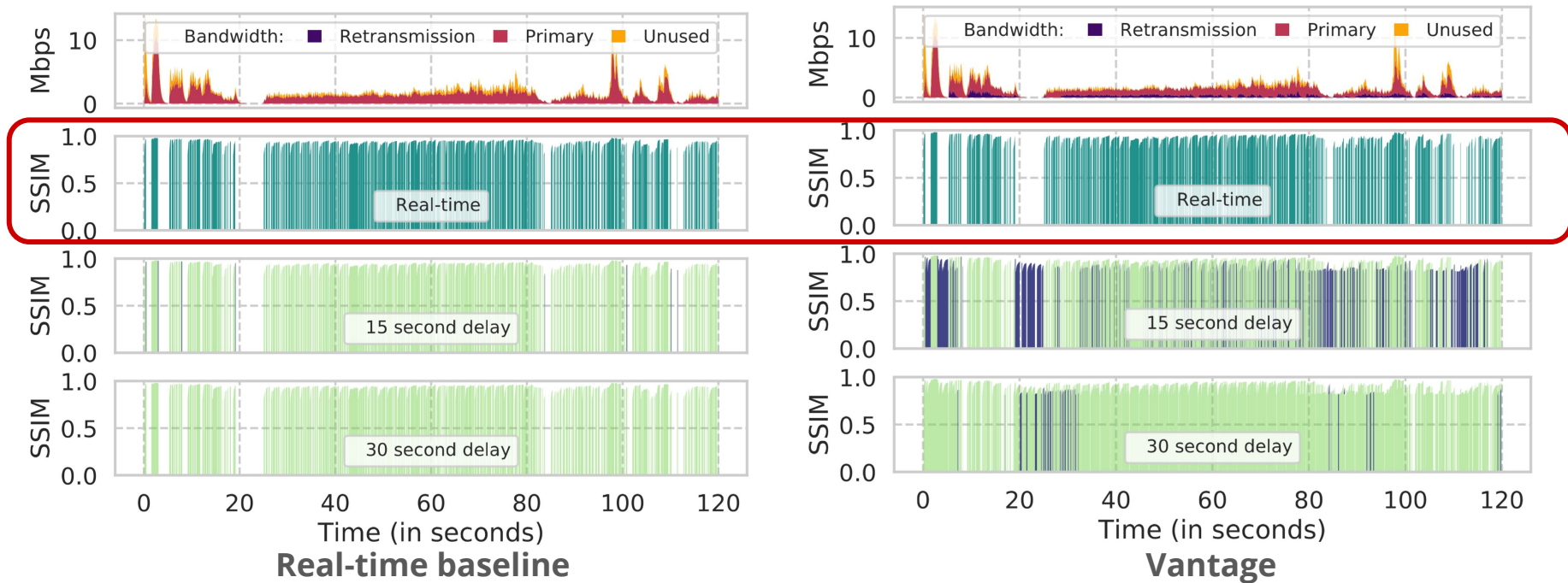
# Baseline: Buffered streaming

High quality for delayed viewing

Real-time viewing infeasible

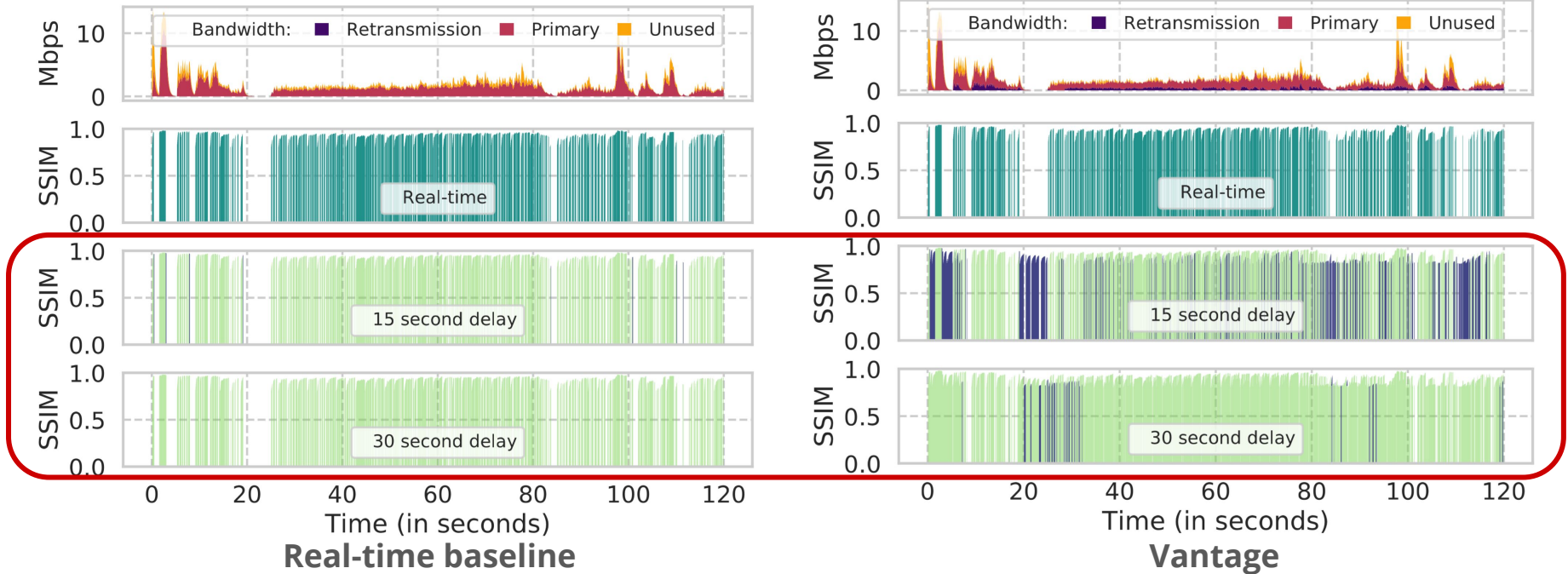


# Vantage: Quality enhancing retransmissions



**Real-time quality almost as good as real-time baseline**

# Vantage: Quality enhancing retransmissions



Real-time quality almost as good as real-time baseline

**Delayed viewing quality significantly better!**

# Other results in the paper

Multiple traces + videos, detailed results in paper

Varying delay distributions

Sensitivity analyses of optimizer period and bandwidth estimation error

Ablation studies comparing Vantage with naive solutions

# Results summary

**Up to 42.9% (average 19.9%) higher delayed video quality  
(Charlie = Happy)**

**At most 7% (average 3.3%) drop in real-time quality  
(Bob = Still Happy)**

# Summary

SLVS applications present new and unique challenges

New paradigm of watching videos: Time-shifted viewing

Upload path variability is important to address

Vantage: Mitigates upload path variability to improve quality for time-shifted viewing

**Thank you for listening!**

# Our research group



Devdeep  
Ray

devdeepr@cs.cmu.edu



Jack  
Kosaian

jkosaian@cs.cmu.edu



Rashmi  
Vinayak

rvinayak@cs.cmu.edu



Srinivasan  
Seshan

srini@cs.cmu.edu

**Carnegie Mellon University**