A Buffer-Based Approach to Rate Adaptation

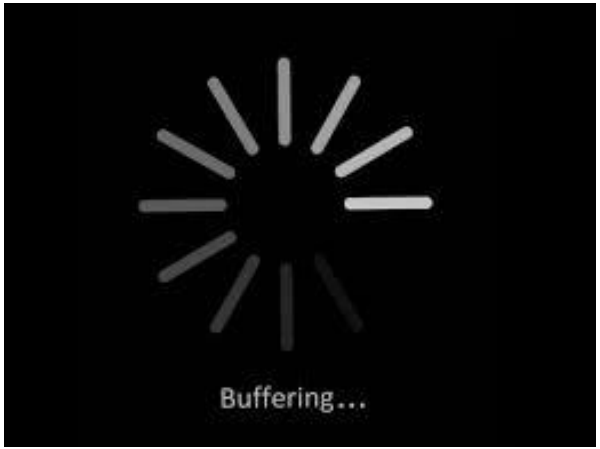
Huang et al. SIGCOMM 2014.

NETFLIX

Presented by: Sam White

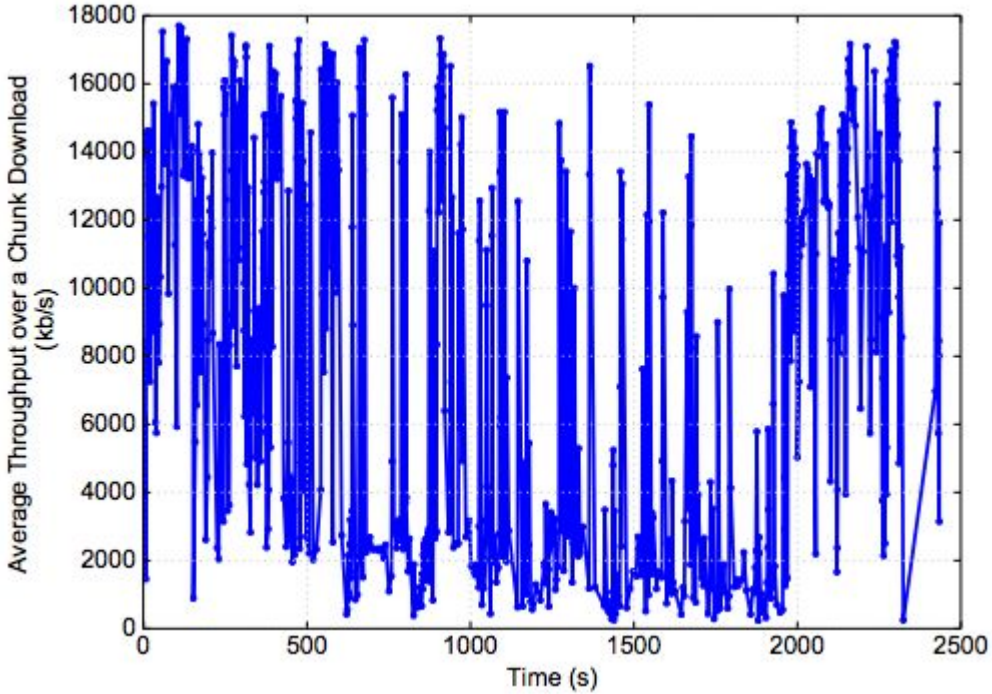
Background

- Video streaming services account for the majority of internet traffic in the US during peak hours
- Video streaming services are demanding
  - Sustained performance required
  - Daily fluctuations in usage



Background

- Clients experience high variability in throughput over time
  - 35x difference not uncommon
  - Causes unnecessary rebuffers (20-30%)



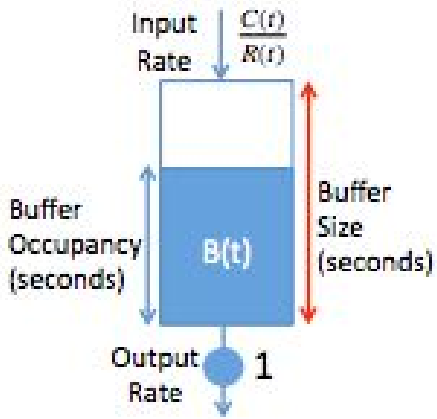
Motivation

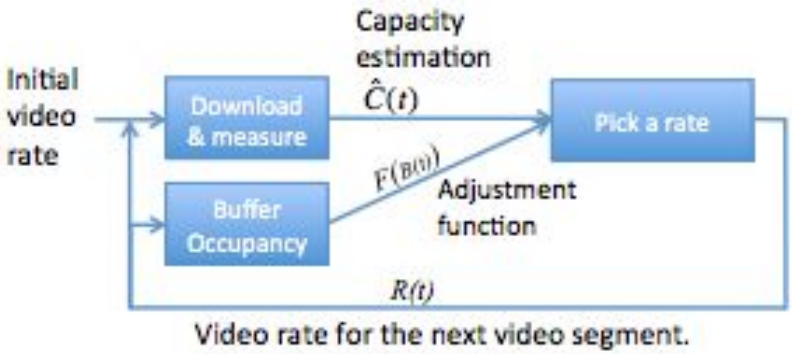
- Adaptive Bit-Rate (ABR) selection algorithms
  - Adjust the bit-rate based on estimated capacity of the network
  - Estimating capacity is difficult b/c of variability
  - Playback buffer occupancy can be used as a heuristic

- Q: Can we set the bit-rate based solely on the buffer?
  - Goals: 1) high average rate with 2) minimal rebuffering

Playback Buffer

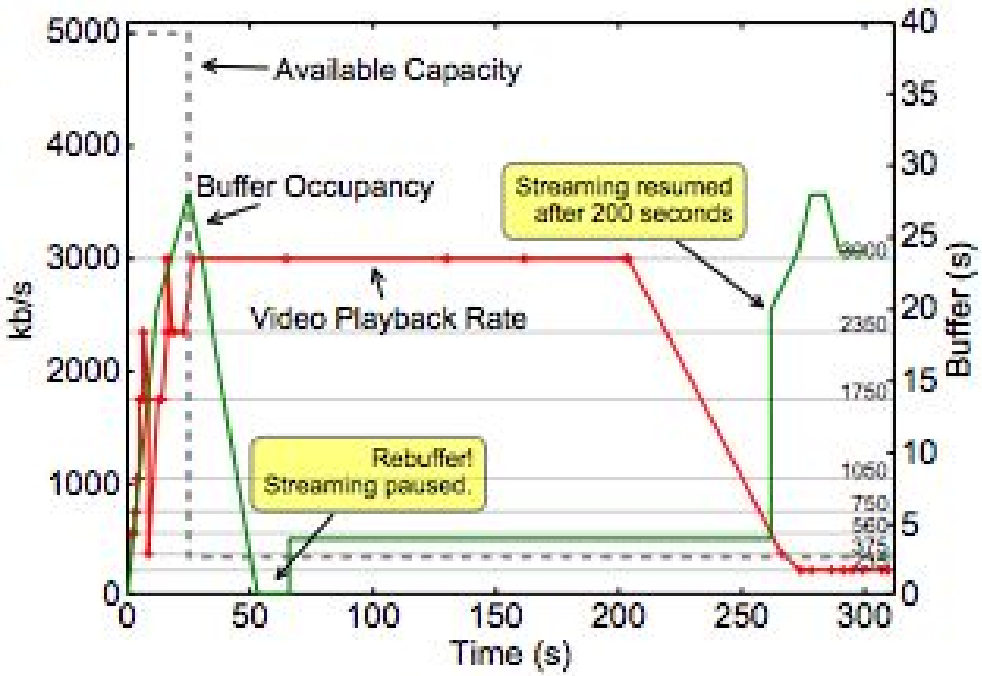
- ABR picks a video rate: R(t)
- If larger than system rate, C(t): C(t) / R(t) < 1
- Difficult to pick the right adjustment function when C(t) is highly variable in practice





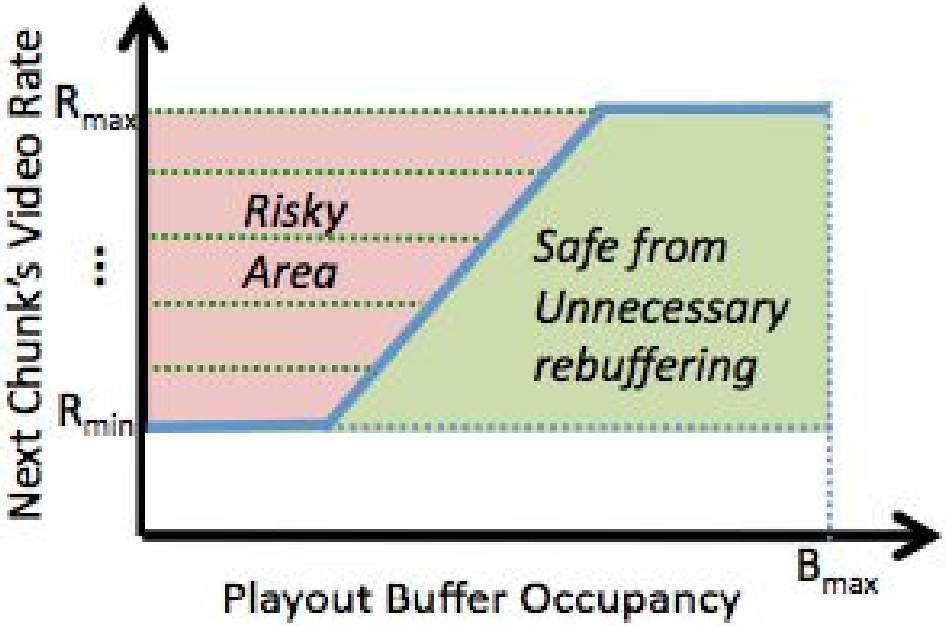
Playback Buffer

- Too aggressive a video rate can hurt performance



ARB Algorithm

- First step: pure buffer-based approach

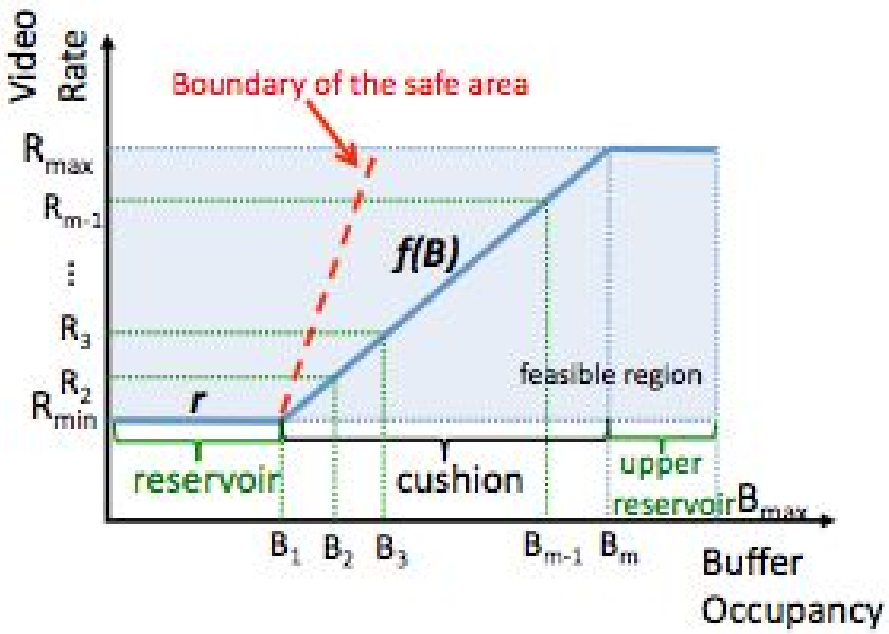


Algorithm

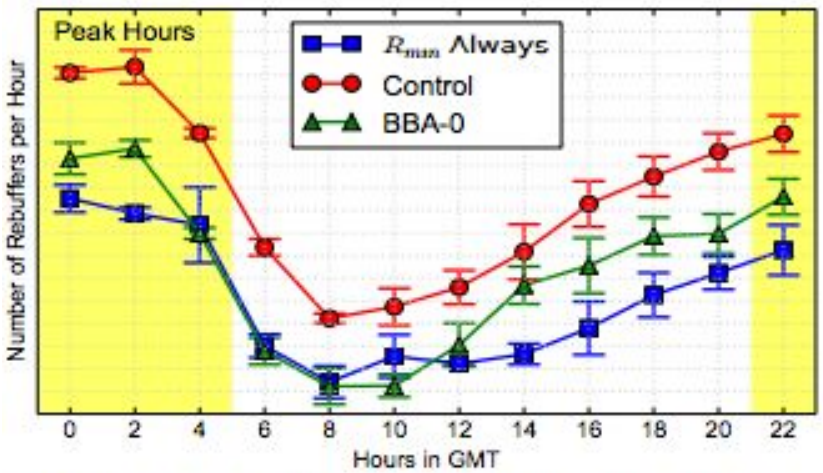
- Assumptions:
  - Chunk size is infinitesimal
  - Any video rate between R_{min} and R_{max} possible
  - Videos are encoded at a constant bit-rate (CBR)
  - Videos are infinitely long

BBA-0 Algorithm

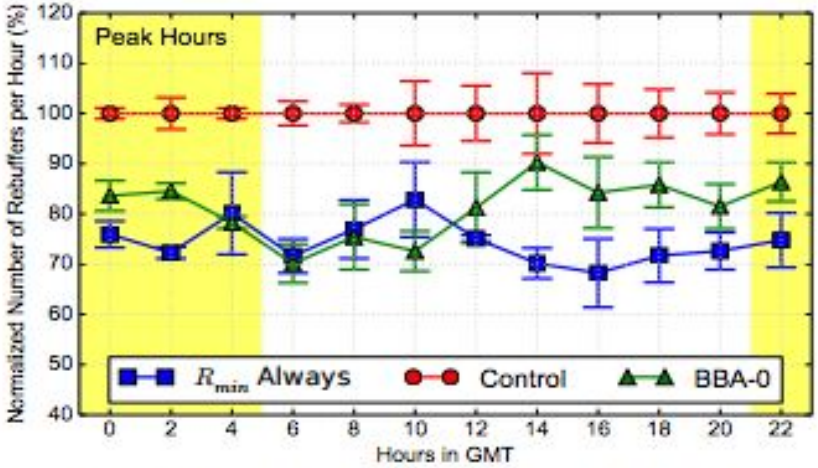
- Rate map for BBA-0 algorithm
- R_{max} is reached when the buffer is 90% full
- Adjust rate based on f(B), which stays in the safe "cushion" area
  - f(B) is a piecewise linear function



BBA-0 Results

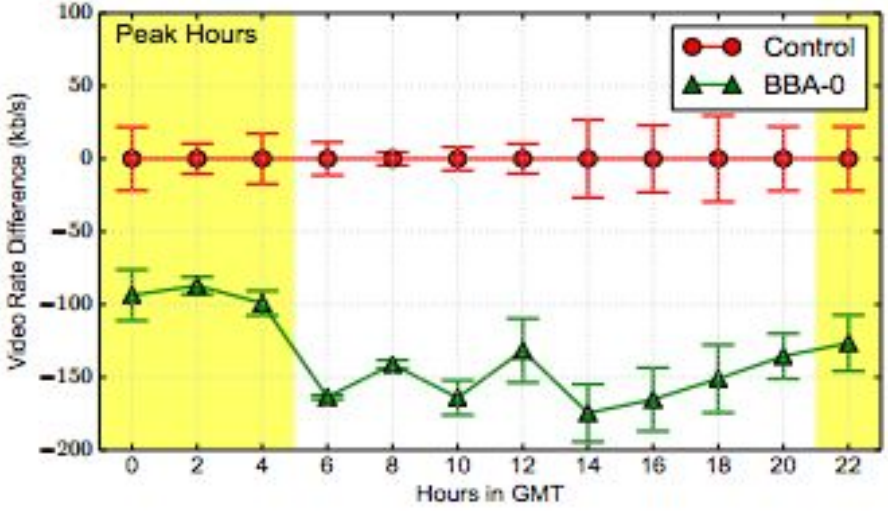


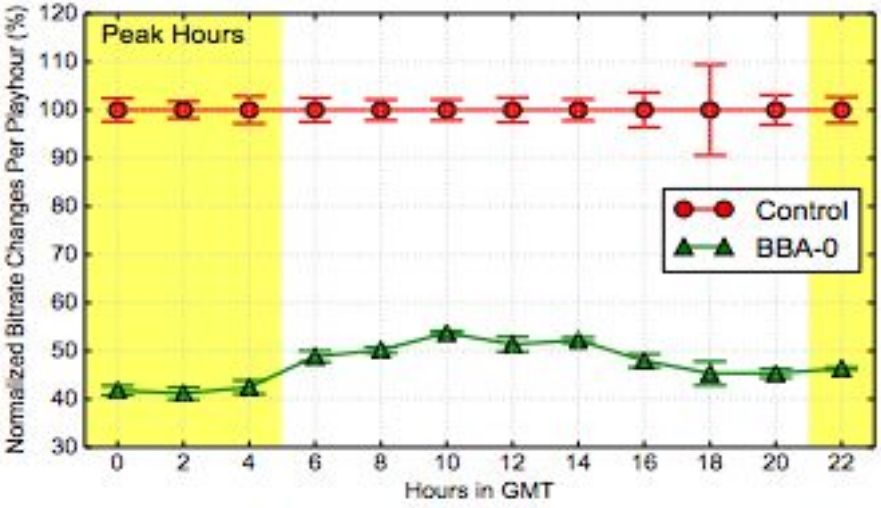
(a) Number of rebuffers per playhour during the day.



(b) Normalized number of rebuffers per playhour, normalized to the average rebuffer rate of *Control* in each two hour period.

BBA-0 Results



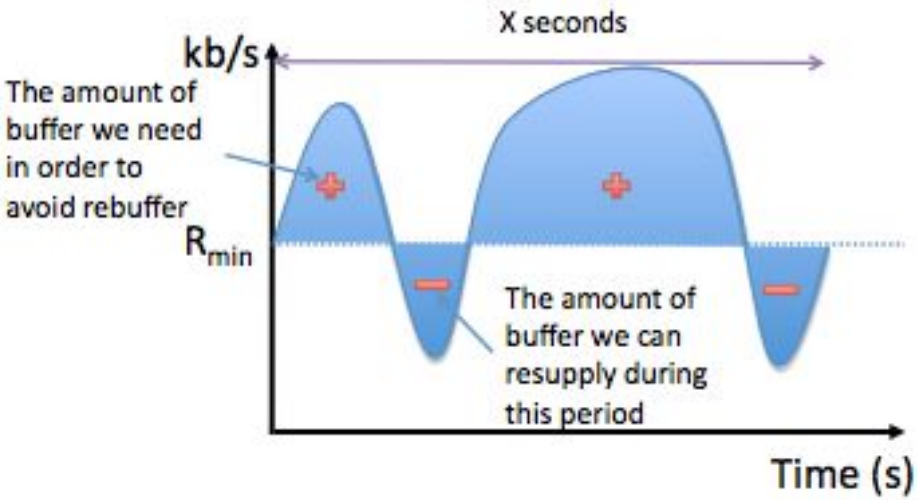


BBA-0 Results

- BBA-0 decreases the rebuffer rate significantly using solely the buffer's state
- BBA-0's video rate was significantly worse than the control algorithm
  - How can we increase the average video rate?

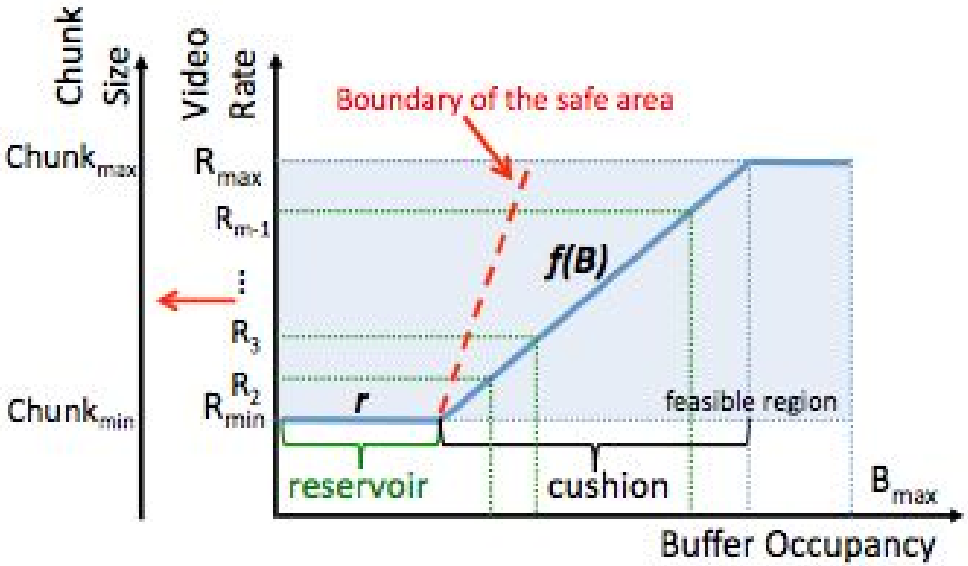
BBA-1 Algorithm

- We assumed CBR, but in reality VBR is used
  - Chunk sizes vary dramatically in time
  - Ex: opening credits

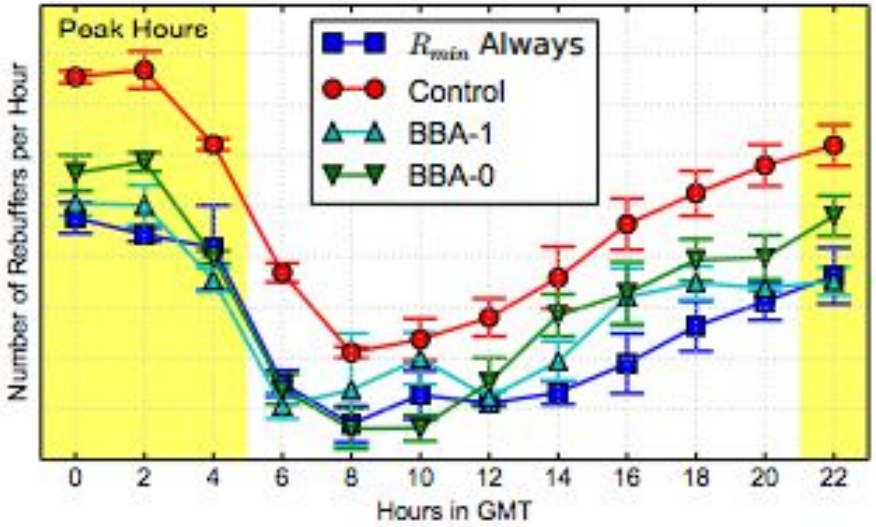


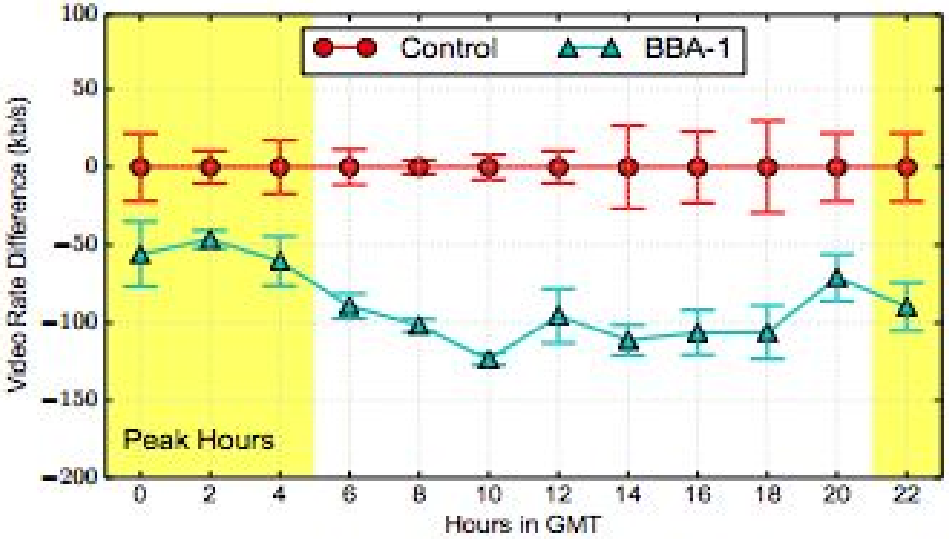
BBA-1 Algorithm

- For VBR: buffer dynamics depend on upcoming chunk size, instead of video rate



BBA-1 Results



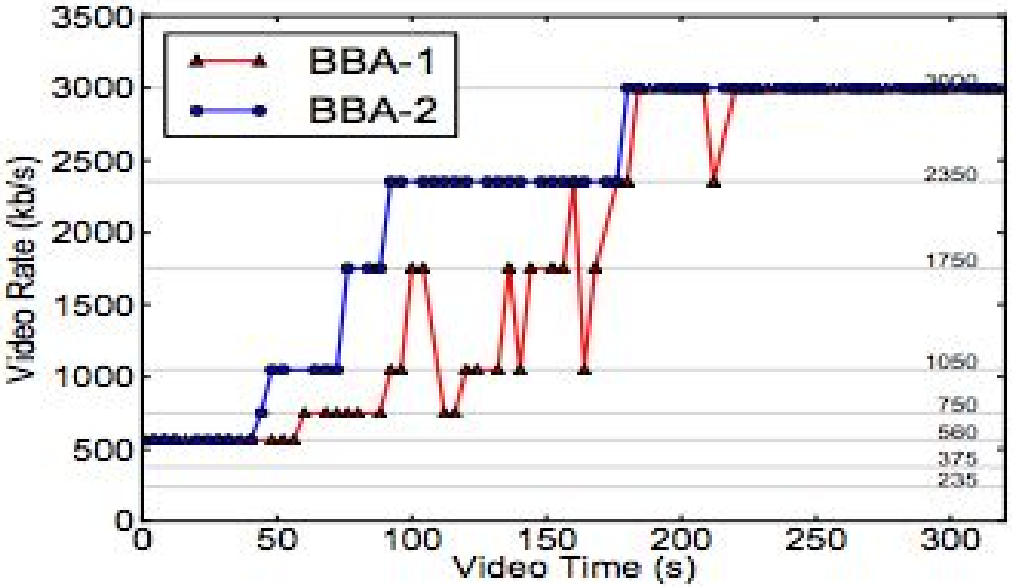


BBA-2 Algorithm

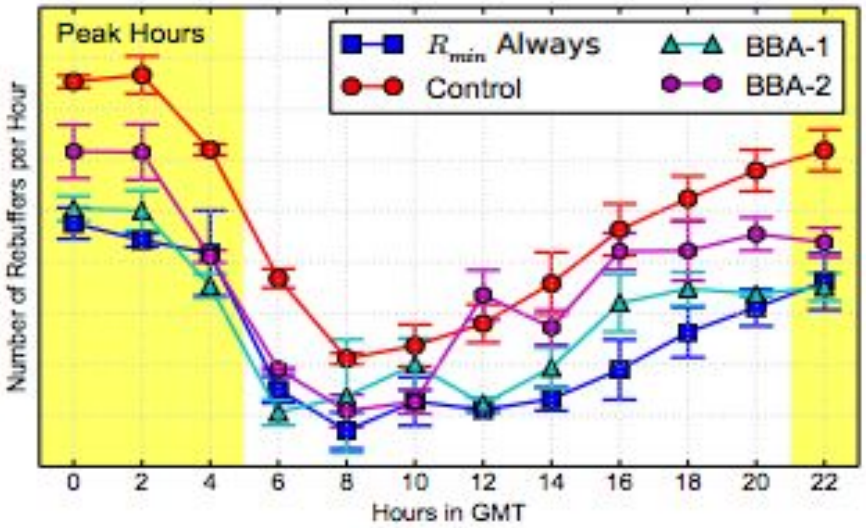
- BBA-1 still suffers from poor video rate while the buffer is still filling up initially
  - The algorithm is too conservative in startup
  - Chunk map is useful for steady-state
  - Capacity estimate is still useful for startup
    - Estimate based on the throughput of the last chunk

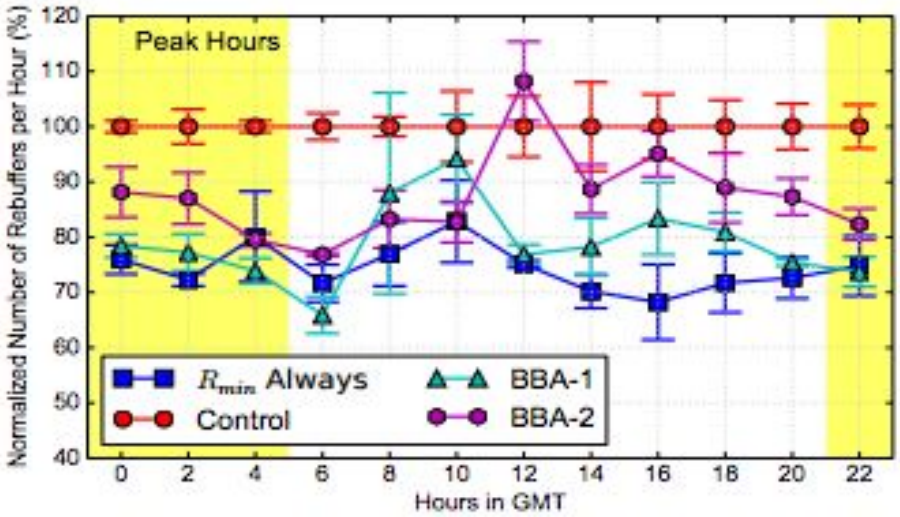
BBA-2 Results

- BBA-2 ramps up to steady-state faster than BBA-1
- BBA-2 switches to BBA-1 after the buffer is built up



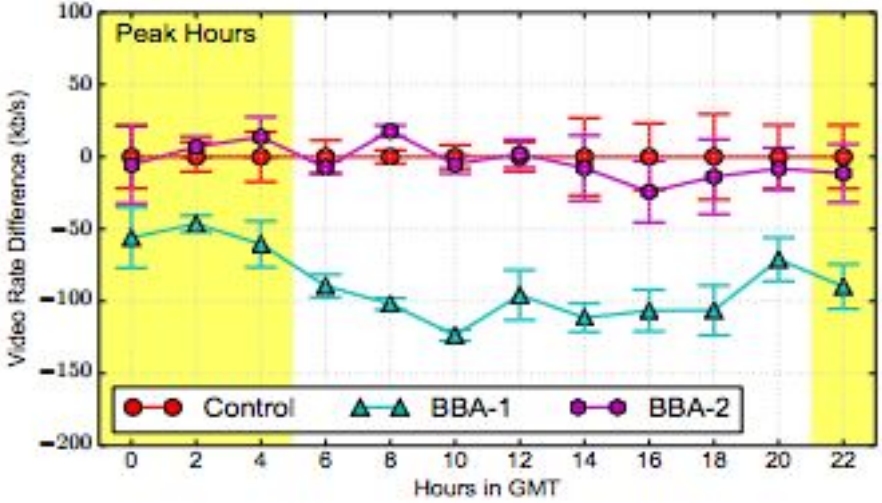
BBA-2 Results





BBA-2 Results

- BBA-2 has higher video rate than Control
- BBA-2 suffers slightly more rebuffers than BBA-1, but less than Control

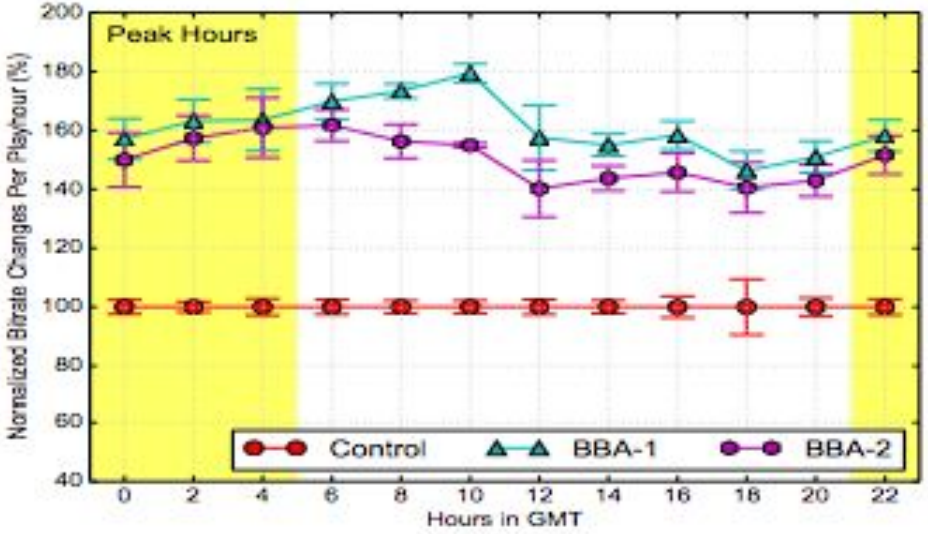


Further Optimizations

- Outage Protection:
  - Handle transient network outages by shifting the chunk map to maintain increased occupancy
- Video switch rates:
  - Chunk map approach can change the video rate for every chunk request
  - Handle by looking at multiple future chunks

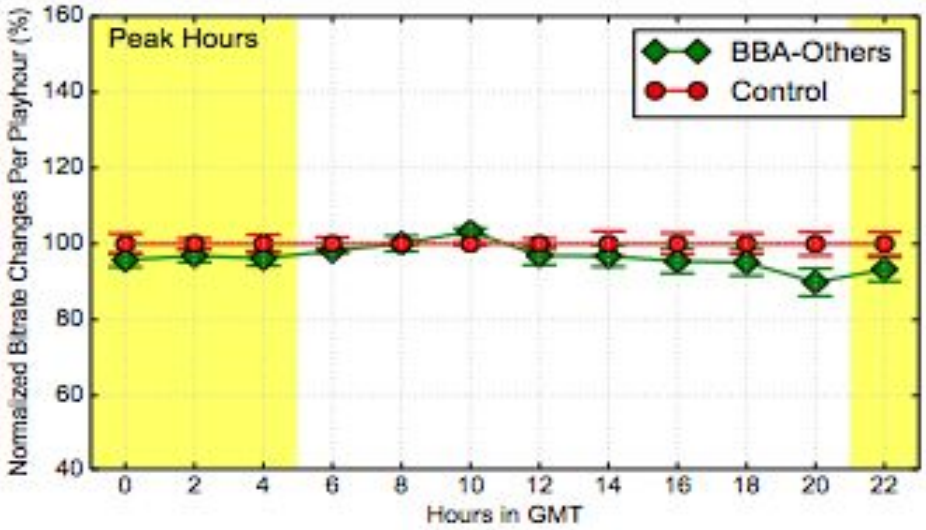
Video Switch Rates

- BBA-1 and BBA-2 dynamically change the buffer rate ~60% more than Control
- Does this hurt the end user's experience?



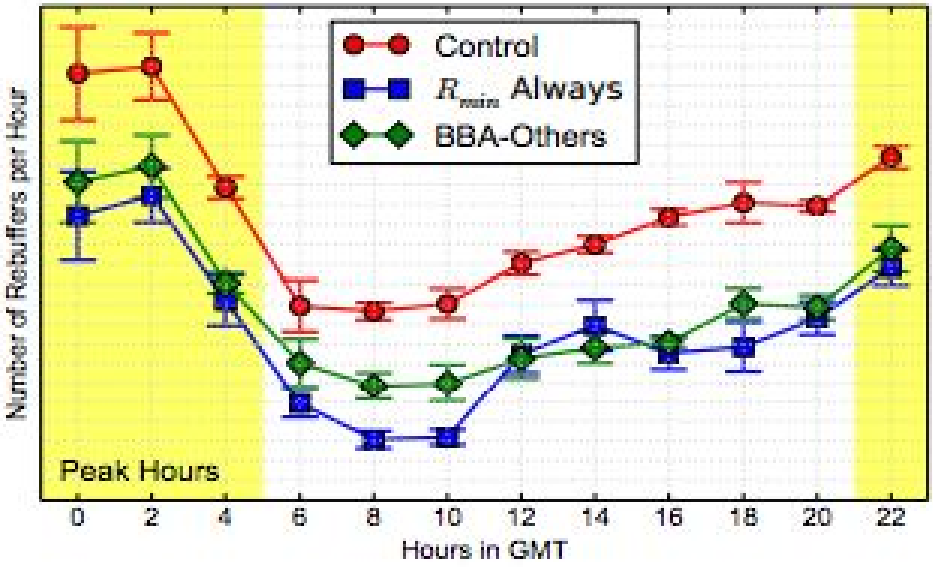
BBA-Others

- BBA-Others looks ahead at the same number of chunks as are in the buffer



BBA-Others Results

- Still low rebuffering rates compared to Control



Conclusions

- ABR algorithms can and should be based on the buffer's state when possible/productive
  - In steady-state, buffer is the best metric
  - In start-up, capacity estimation is better metric
- BBA-Others decreases rebuffer rate by ~20%, and off-peak video rate by 30 KB/s

Discussion

- Authors did a good job addressing the assumptions made for BBA-0
- Comparisons to Control are difficult to make without knowing more about the Control algorithm
- Utility of video switch rates as a metric?

Questions?