On Adaptive HTTP Streaming to Mobile Devices

Guibin Tian and Yong Liu

ECE Dept.

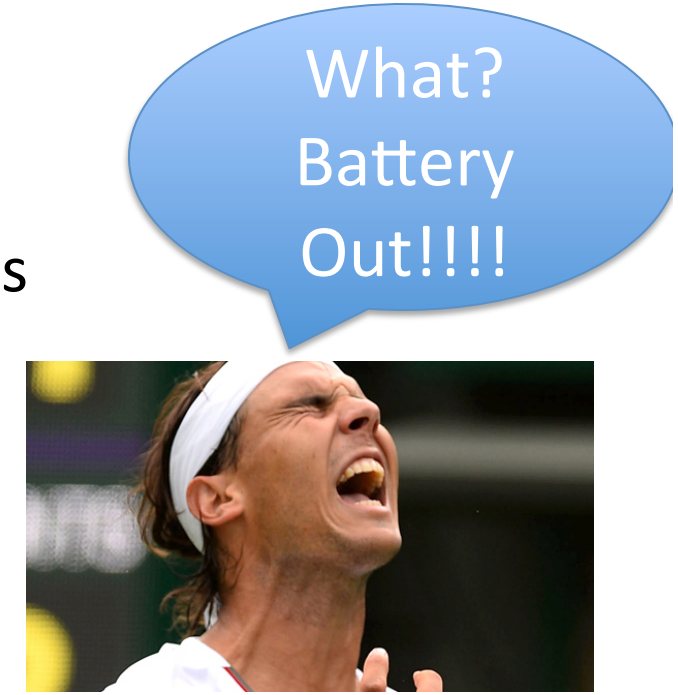
Polytechnic Institute of New York University

Outline

- Motivation & Background
- System Design
- Adaptation and Battery Consideration
- Performance Evaluation
- Conclusion

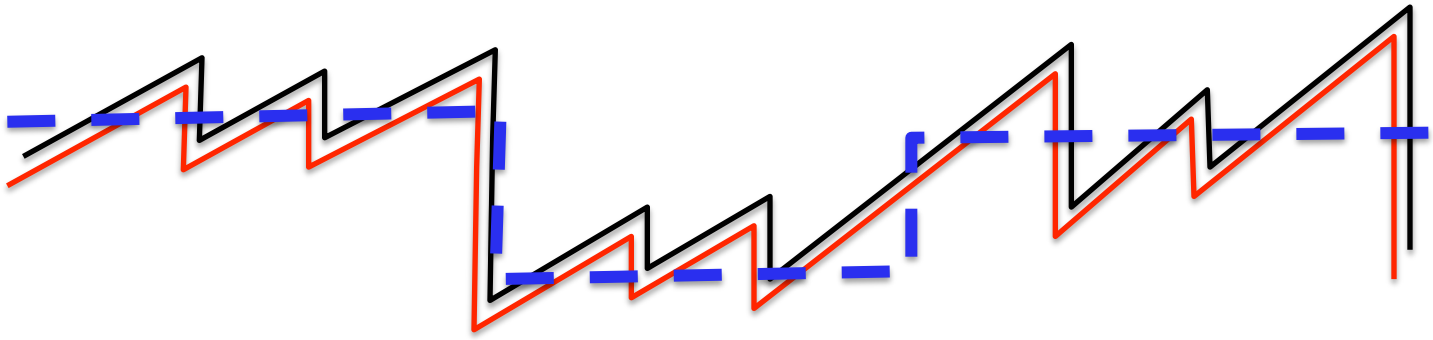
DASH over Mobile Devices

- DASH: Dynamic Adaptive Streaming over HTTP
  - single video streamed at multiple versions
    - address user heterogeneity: access/device
  - existing infrastructure optimized for HTTP
    - CDN, Caching, Firewall, etc
- DASH over Mobile Devices
  - wide adoption of mobile devices
    - smart phones, tablets
  - more limitations on mobile devices
    - wireless connection
      - Wifi, cellular
    - battery limitation



Video Adaptation

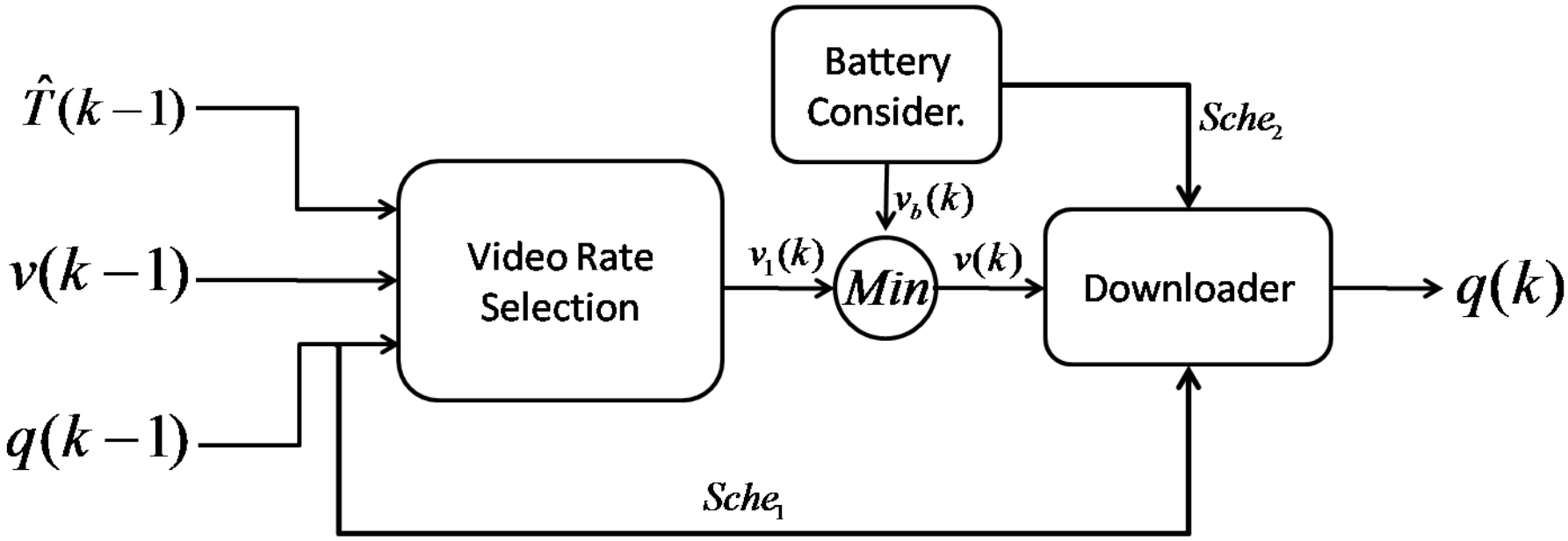
- Which Version to Download?
  - download rate controlled by TCP
  - version selection: application layer control
  - video rate should match TCP throughput
    - too aggressive: video freezes,
    - too conservative: low video quality
- TCP throughput prediction
  - TCP throughput has variations at different time scales
- Video rate smoothing
  - users sensitive to frequent quality changes



Challenges

- Variability of TCP throughput
  - variability: built-in character of TCP
  - network congestion and server overload
  - wireless connection only on mobile devices
- Management of video playback buffer
  - small buffer: Low delay, high risk of freezing
  - large buffer: Safe but waste of resources (especially for cellular), impossible for live streaming
- Battery limitation on mobile devices
  - different energy consumption patterns on WiFi and cellular
  - what video rate should be chosen when battery is depleting?

Design Framework



$v_1(k)$: target rate based on buffer size, TCP throughput, previous rate

$v_b(k)$: target rate based on battery consideration

$v(k)$: requested rate Min($v_1(k)$, $v_b(k)$)

TCP throughput Based Adaptation

- Simple History-Based prediction($\hat{T}(k)$)
  - Proved to generate highly accurate prediction given real time throughput data
  - N = 10 (Number of historical data)
- Dynamic Margin (M)
  - target video rate lower than throughput estimate
  - rate margin depends on TCP throughput variability
    - higher variability → larger margin

$$SI(k) = \frac{1}{N-1} \sum_{i=k-N+1}^{k-1} \frac{|T(i) - T(i-1)|}{T(i)} \qquad M(k) = 1 - \frac{25 + 70e^{SI(k)}}{100e^{SI(k)}}$$

- $\hat{v}_1(k) = Q(\hat{T}(k)(1 - M(k))$ for smoothing

Smooth Video Adaptation Algorithm

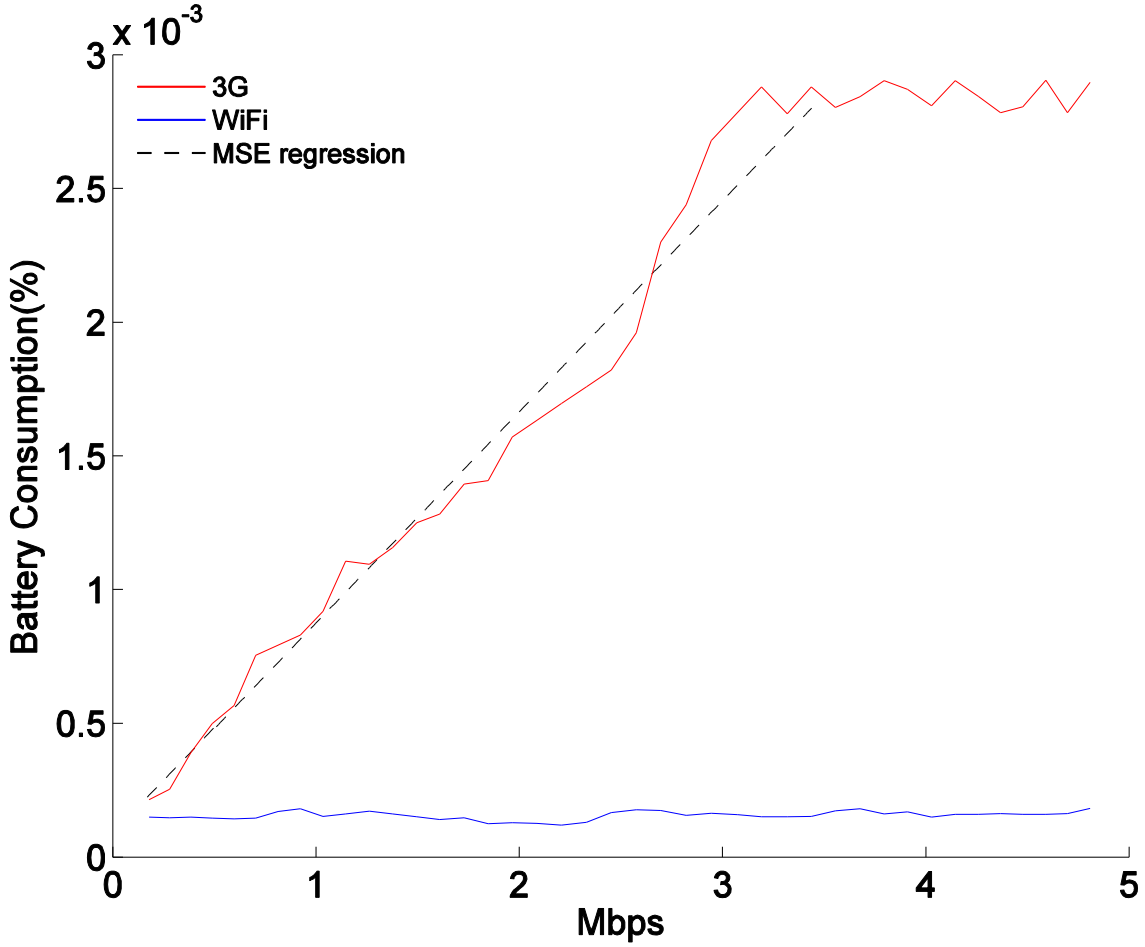
- Prompt Rate Decrease
  - if buffer becomes lower than a threshold
  - avoid buffer underflow (freeze)
- Conservative Rate Increase
  - only if target rate higher than the current rate for m consecutive steps
  - m calculated dynamically
  - avoid oscillations triggered by transient TCP rate increase
- Control Buffer Overflow
  - introduce idle time between chunk requests

Algorithm 1 Smooth Video Adaptation Algorithm.

1: **if** $q(k) < \frac{q_{th}}{2}$ **then**
2: $v(k) = Q\left(\bar{T}(k-1)(1-M(k))\right)$;
3: return;
4: **else**
5: $\hat{v}(k) = Q\left(\hat{T}(k)(1-M(k))\right)$;
6: **if** $\hat{v}(k) \geq v(k-1)$ **then**
7: $Counter++$
8: **if** $Counter > m$ **then**
9: $v(k) = \hat{v}(k)$;
10: $Counter = 0$;
11: return;
12: **end if**
13: **else**
14: $Counter = 0$
15: **end if**
16: **end if**
17: $v(k) = v(k-1)$;
18: **if** $q(k) - q_{cap} > 0$ **then**
19: $I = q(k) - q_{cap}$;
20: **else**
21: $I = 0$
22: **end if**
23: $Idle(I)$;
24: return;

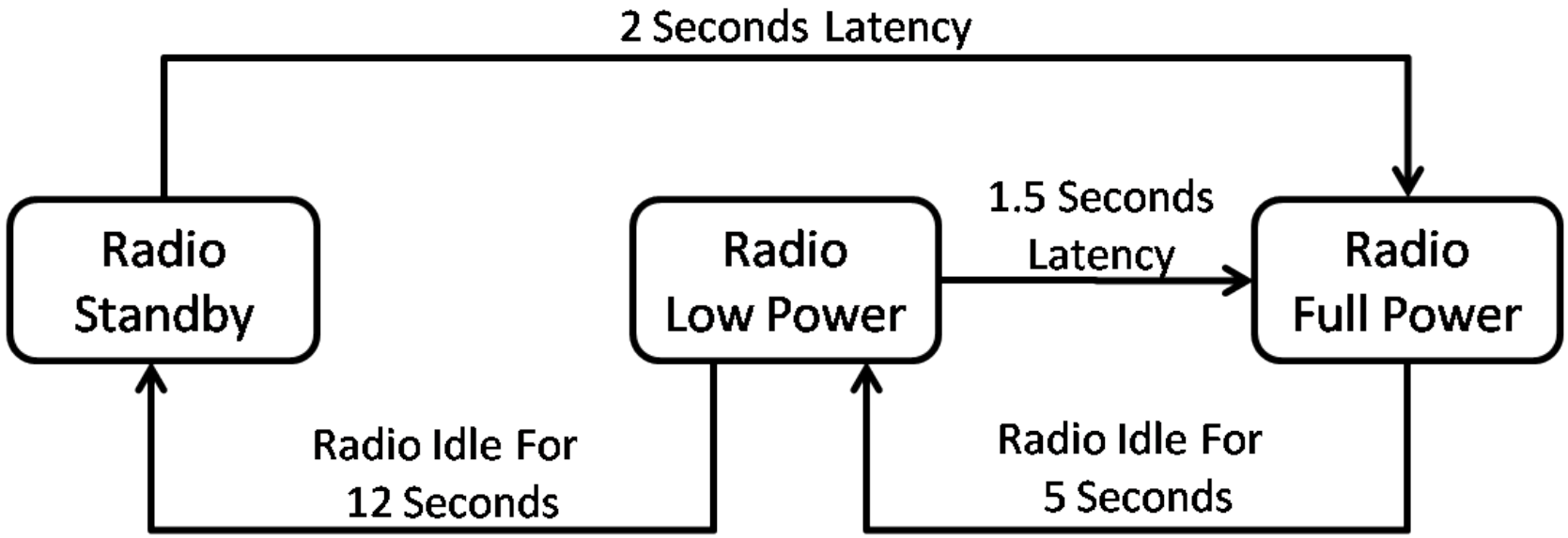
Battery Consideration

- 3G radio consumes more battery than WIFI



Battery Consideration

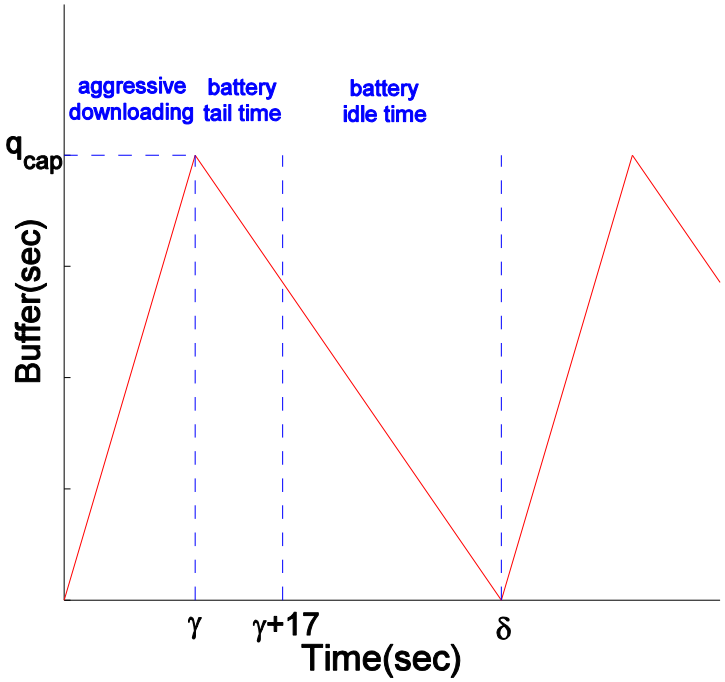
- 3G Battery State Machine (AT&T)



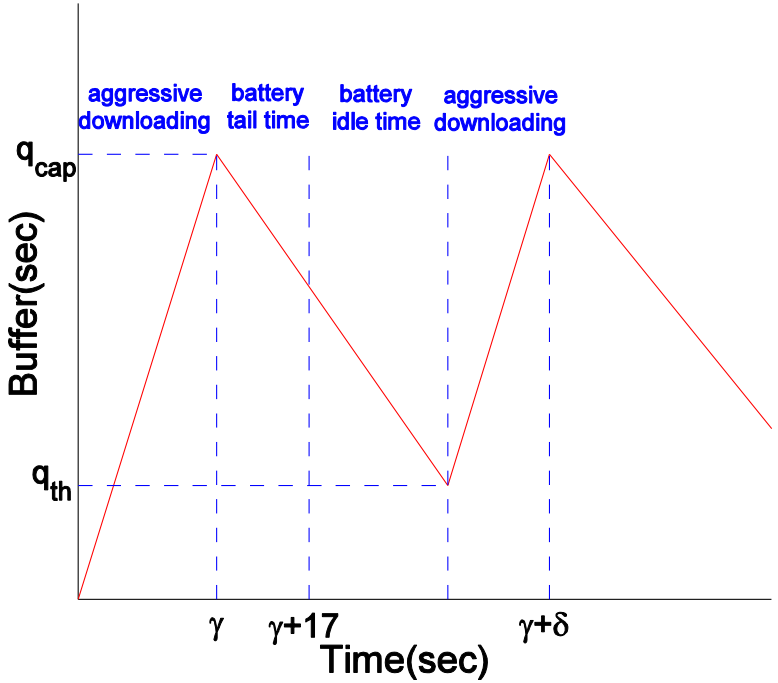
Different Scheduling can make huge difference

Scheduling Considering 3G Battery mode

- Download more chunks in one transaction given buffer size bounds



Ideal case



Practical case

$$\gamma = q_{cap} * \frac{v}{T - v}$$:Accumulating from 0 to cap

$$\sigma = q_{cap} * \frac{T}{T - v}$$: Loop Period

Battery Life Constraint

- Get the highest rate at which the batter can last for the whole movie

$$v_b(r,p) = \max(v_i, v_i : \frac{p}{E_{v_i}} > r)$$

E_{v_i} : Unit Time Energy Consumption for rate v_i

r : Remaining video time

p : Remaining battery percentage

- E_{v_i} calculated by MSE regression of 3G battery consumption data

- $v(k) = \min(v_1(k), v_b(k))$

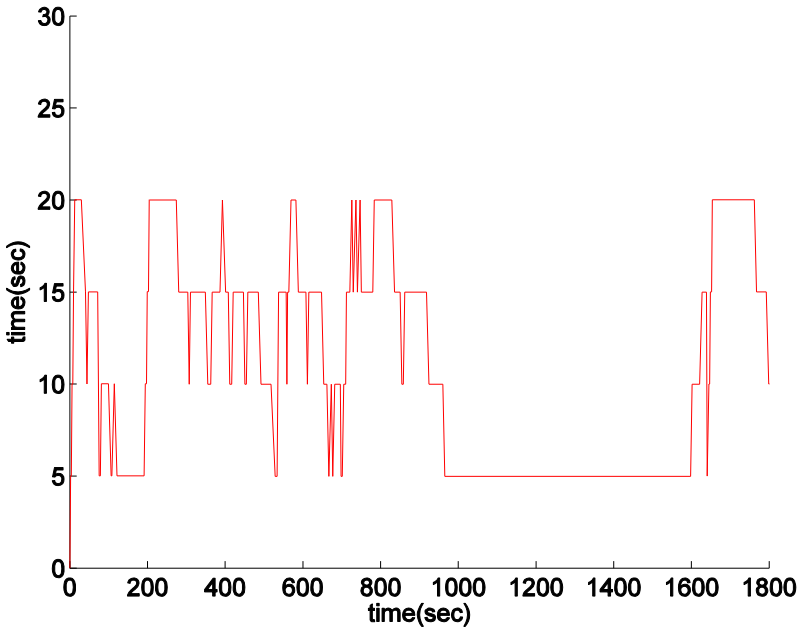
Evaluation: test platforms

- Vanilla apache HTTP server, Google Nexus 4 with Andriod 4.2 as client
- Video Rates
  - 100Kbps~4.1Mbps with even rate gap of 100Kbps
  - five-second video chunks
- Internet Experiments
  - Wifi connection: change the distance between laptop and router
  - 3G connection: in motion
  - Hand over between 3G and Wifi

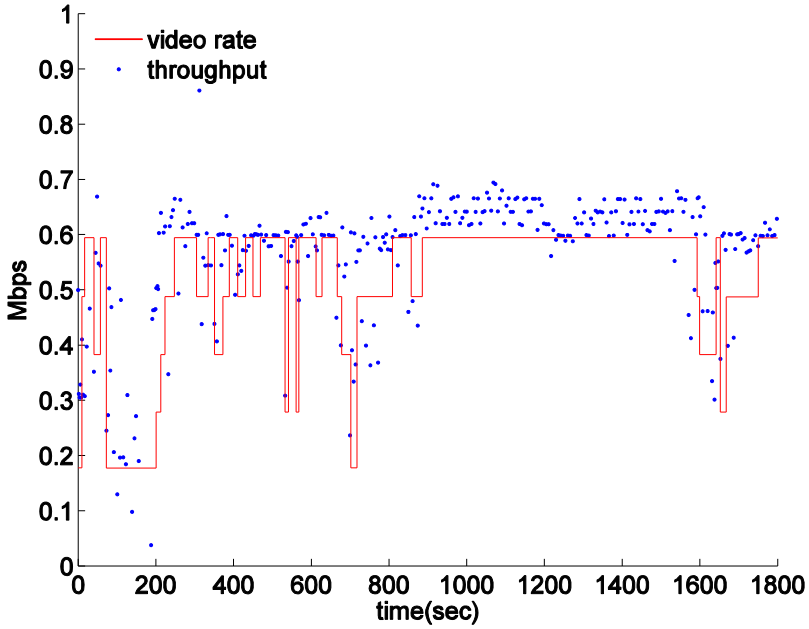
TCP Throughput Variability

(Mbps)	Wired	Strong WiFi	Weak WiFi	Static 3G	3G In motion
SI	0.004	0.074	0.336	0.151	0.214
Average Throughput	0.720	0.699	0.616	2.597	2.488
Variation	0.001	0.002	0.129	0.164	0.710

Strong WiFi without margin, q_{cap}= 20



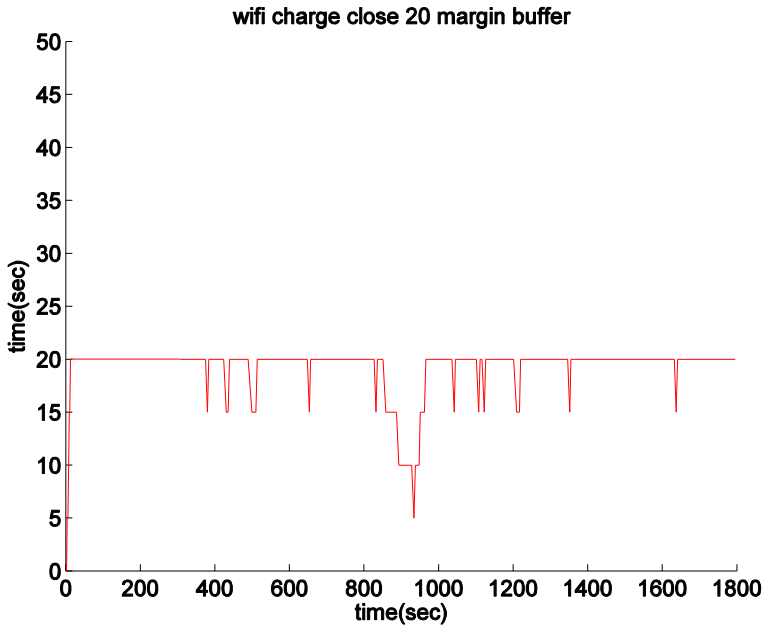
video buffer length



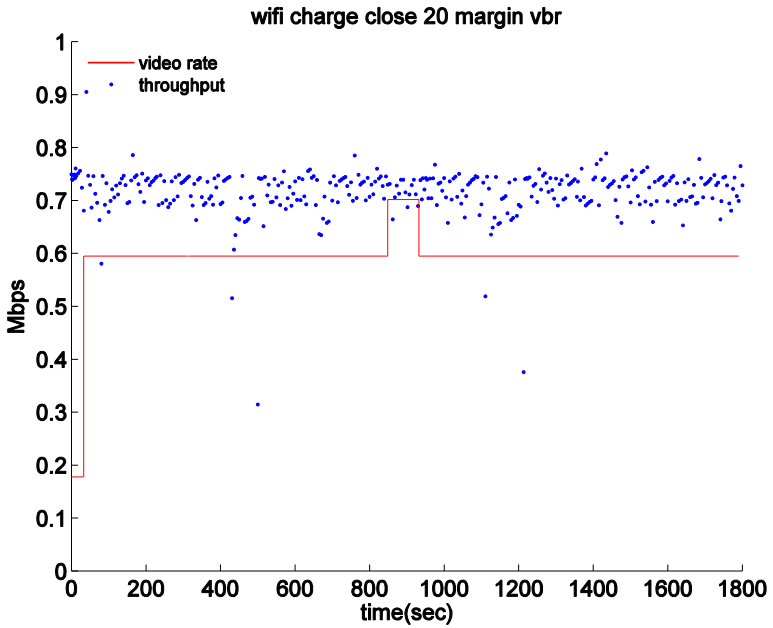
video rate vs. TCP throughput

- Video rate follows TCP throughput tightly, but too much oscillations for users.
- Buffer fluctuates a lot
- Bad User QoE

Strong WiFi with margin, q_{cap}= 20



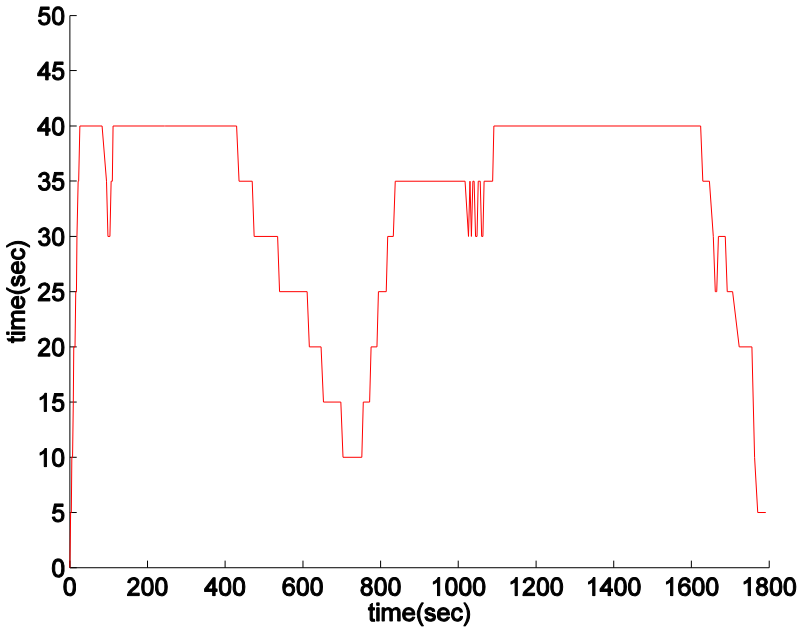
video buffer length



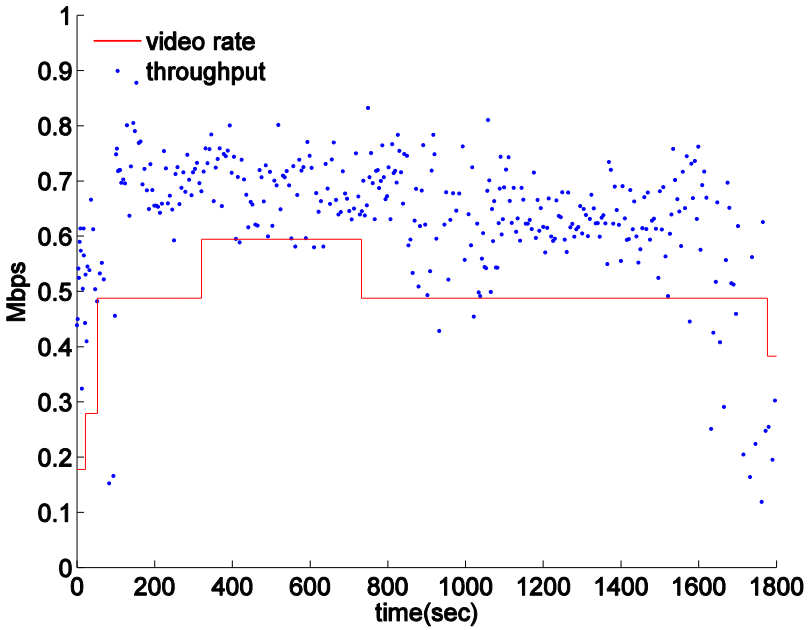
video rate vs. TCP throughput

- Video rate slightly lower than throughput
- Stable buffer and video rate
- Good user QoE

Weak WiFi with margin, q_{cap}= 40



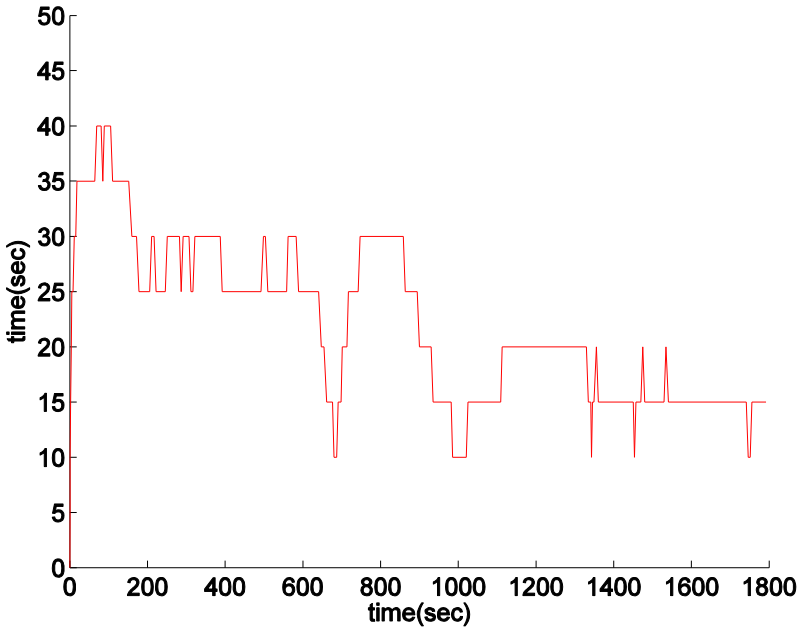
video buffer length



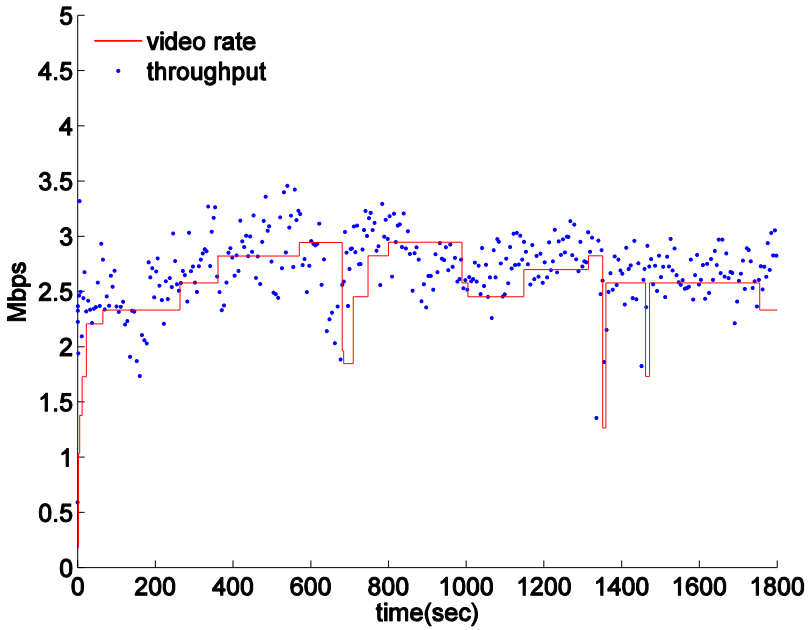
video rate vs. TCP throughput

- Buffer fluctuates but never hit zero (transparent to users)
- Stable video rate, good user QoE
- Lower buffer cap value won't work

3G without margin, q_{cap}= 40, q_{th}=20



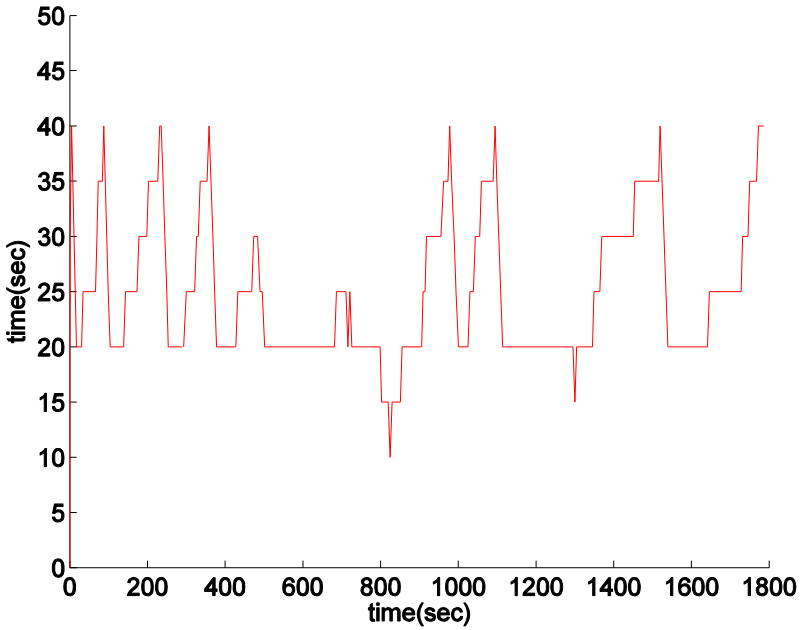
video buffer length



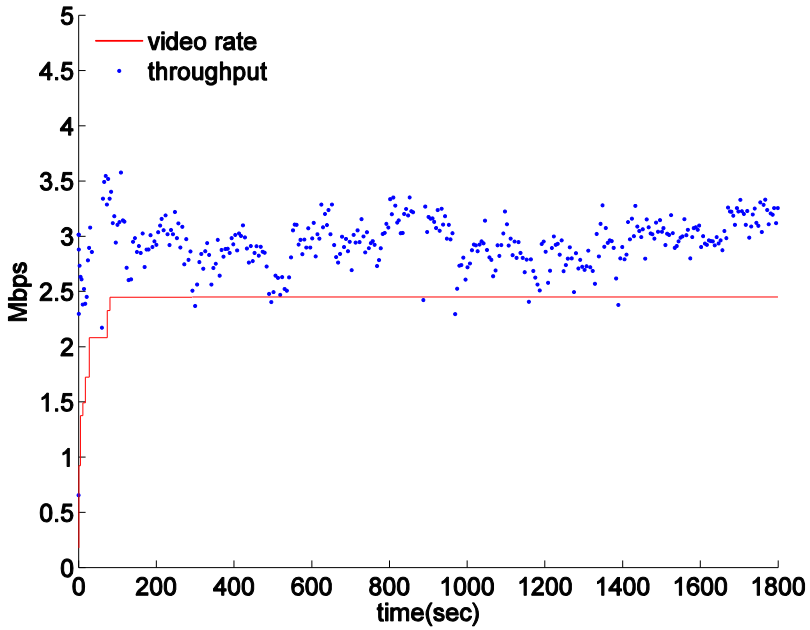
video rate vs. TCP throughput

- Buffer and video rate fluctuates
- Bad user QoE
- Pretty high throughput

3G with margin, q_{cap}= 40, q_{th}= 20



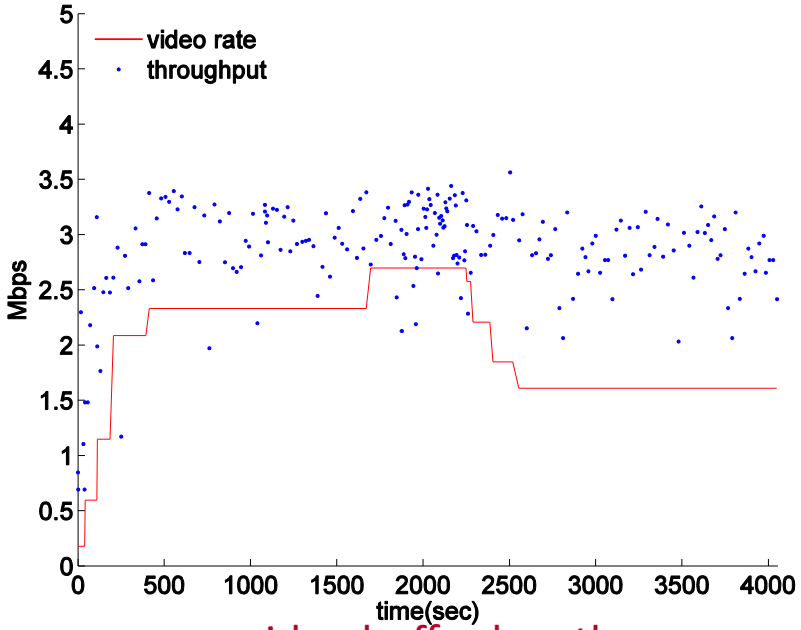
video buffer length



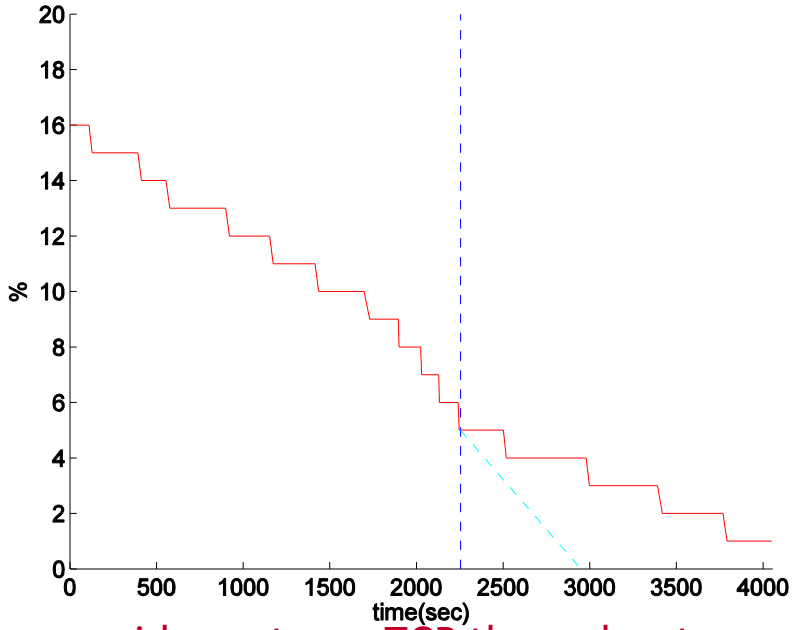
video rate vs. TCP throughput

- Buffer fluctuates as scheduled through bundled chunk downloading mechanism
- Stable Video rate
- Good user QoE

Battery Consumption and Video Rate Adaptation over 3G Connection



video buffer length

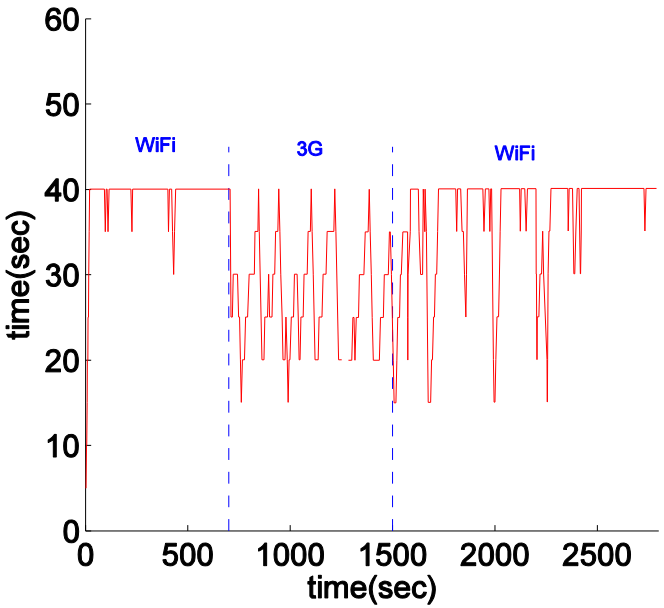


video rate vs. TCP throughput

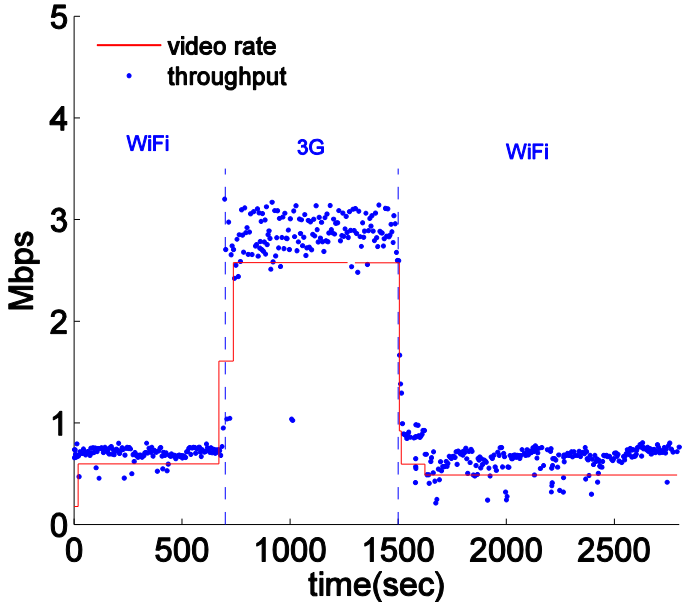
- Video rate drops because of battery limitation while throughput still high
- Battery consumption rate decreased to finish the whole video

Handover between WiFi and 3G connection

q_{cap}=40, q_{th}= 20



video buffer length

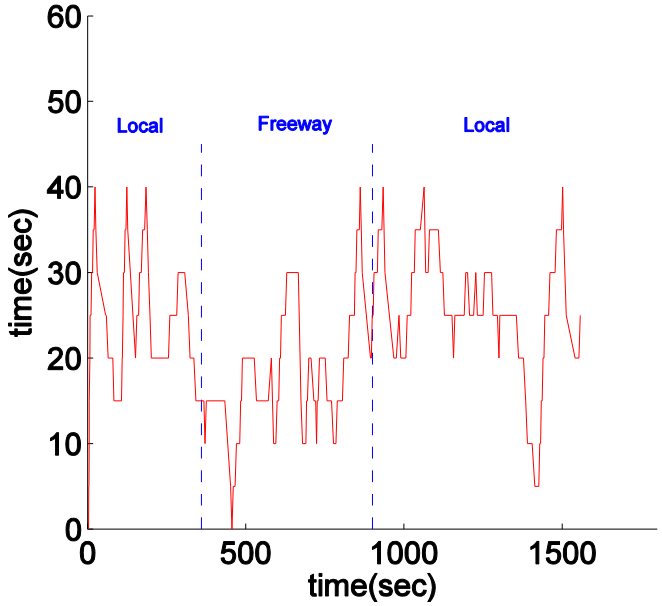


video rate vs. TCP throughput

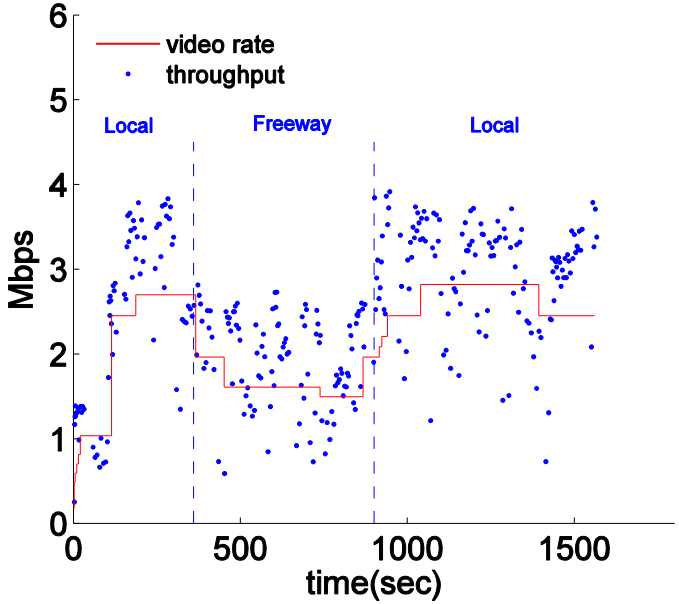
- Buffer fluctuates as scheduled in 3G and stable in WiFi
- Stable Video rate with responsive handover
- Good user QoE

3G Mobile Experiments in a Moving Car

q_{cap}= 40, q_{th}= 20



video buffer length



video rate vs. TCP throughput

- Buffer fluctuates a lot (Hits zero once!)
- Highly fluctuating throughput at high speed (30~70mph)
- But still acceptable QoE

Conclusion

- Client-side video adaptation algorithm
  - TCP throughput highly dynamic
  - Dynamic rate margin leads to smooth video rate
  - Video scheduling with battery considerations (3G on-off pattern and available battery level)
  - Extensive evaluation on real systems
- Future Work
  - Considering cooperation between mobile devices

Thanks!

Q&A