Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks

E. Kurdoglu, Y. Liu, Y. Wang ¹
Y. Shi, C. Gu, J. Lyu ²

¹NYU Tandon School of Engineering
²Tencent Co. Ltd.

May 11, 2016
1. Introduction and Problem Setting

2. Rebera Components
   - Measuring the Current Bandwidth
   - Predicting the Future Bandwidth
   - Determining the Sending Rate (Bit-budget)
   - Encoder & Video Stream Structure
   - Dynamic Frame Selection

3. Experiment Results
Video calls: very low-delay, interactive applications

1. Need to adapt transmission rate wrt network speed:
   \[ \text{Tx rate} > \text{Bandwidth} \Rightarrow \text{intolerable delays} \]
   \[ \text{Tx rate} < \text{Bandwidth} \Rightarrow \text{under-utilized link} \]
   - Accurate measurement and forecast of bandwidth is hard!

2. Need to adapt video encoding rate wrt to network speed
   - Accurate low-delay video rate control is hard!

3. Need resiliency against late or lost packets

Cellular links can be highly volatile: link speeds, delays...
A Video Call Scenario

- **Challenge**: measuring and predicting the e2e bandwidth
- **Assume**: cellular links are the only bottlenecks
- **Model**: cellular links as isolated queues (no cross-traffic)
  - uplink buffer in the device, downlink buffer in the BS

Kurdoglu, Liu, Wang, Shi, Gu, Lyu

NYU Tandon School of Engineering, Tencent Co. Ltd.

Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks
1 Introduction and Problem Setting

2 Rebera Components
   - Measuring the Current Bandwidth
   - Predicting the Future Bandwidth
   - Determining the Sending Rate (Bit-budget)
   - Encoder & Video Stream Structure
   - Dynamic Frame Selection

3 Experiment Results
We periodically insert I-frames

**Intra-period**: time between two successive I-frames (≈1 sec)
Measuring the Current Bandwidth

Observation: We can measure bandwidth by sending packet trains and measuring packet inter-arrival times at the receiver.

\[ m_n \triangleq \frac{z_2 + \cdots + z_p}{a_2 + \cdots + a_p} \triangleq \frac{Z_n}{A_n} \]

Key idea: Use the packets belonging to a frame as a packet train.

1. Estimate avg. ABW in the last intra-period
2. Feed it back to sender
3. Slide the window
**Predicting the Future Bandwidth**

**Observation:** Bandwidth measurement values are correlated.

**Key idea:** Forecast future bandwidth from past values with online linear adaptive filtering (e.g. RLS)

\[
\hat{c}_{k+1} = \sum_{i=0}^{M-1} w_i(k) \tilde{c}_{k-i}
\]

- \(\hat{c}_{k+1}\): number of filter taps
- \(w(k)\): filter tap vector of length \(M\)
- \(\tilde{c}_k\): measured capacity

Kurdoglu, Liu, Wang, Shi, Gu, Lyu

Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks
Determining the Sending Rate (≡ Bit-budget)

Observation: Forecast errors are inevitable.

Key idea: Estimate the (almost) worst-case scenario to find the (almost) safest max sending rate.

\[ r_k \quad \text{sending rate} \]
\[ \epsilon_k \quad \text{exact-to-predicted ratio (≡ } \frac{c_k}{\hat{c}_k} \text{)} \]

\[ \Pr(r_{k+1} > c_{k+1}) \leq \delta \Rightarrow \Pr\left(\epsilon_{k+1} < \frac{r_{k+1}}{\hat{c}_{k+1}}\right) \leq \delta \]

\[ r_{k+1}^{\text{safe}} = \hat{c}_{k+1} \times \delta\text{-quantile of } \epsilon \]
Determining the Sending Rate (≡ Bit-budget)

Observation: Forecast errors are inevitable.

Key idea: Estimate the (almost) worst-case scenario to find the (almost) safest max sending rate.

\[ r_k \quad \text{sending rate} \]
\[ \epsilon_k \quad \text{exact-to-predicted ratio (}= \frac{c_k}{\hat{c}_k}) \]
\[ q_k \quad \text{backlogged bits} \]

\[ \Pr(r_{k+1} > c_{k+1}) \leq \delta \Rightarrow \Pr\left(\epsilon_{k+1} < \frac{r_{k+1}}{\hat{c}_{k+1}}\right) \leq \delta \]

\[ r_{k+1}^{\text{safe}} = \hat{c}_{k+1} \times \delta\text{-quantile of } \epsilon \]

Bit-budget for intra-period \( k + 1 \)

\[ b_{k+1} = r_{k+1}^{\text{safe}} \times T - q_k \]
Observation: Resiliency against packet losses is essential.

Key idea: Use hierarchical-P temporal layering.

1. Enables UEP ✓
2. Complexity & overhead kept minimum ✓
3. DFS does not cause jitter. ✓

Kurdoglu, Liu, Wang, Shi, Gu, Lyu
NYU Tandon School of Engineering, Tencent Co. Ltd.

Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks
**Observation:** When video rate > target bitrate, we can discard higher layer frames and maintain a stable frame rate.

**Requirement:** Must make the “send/discard” decision on the fly.

- Decision for frame P1?
- Cannot wait for \{P2, P4, P6\}

Sequential decision making algorithm considers:

1. Qualities of layers
2. Decoding dependencies
3. Smoothness of frame intervals
Outline

1 Introduction and Problem Setting

2 Rebera Components
   ■ Measuring the Current Bandwidth
   ■ Predicting the Future Bandwidth
   ■ Determining the Sending Rate (Bit-budget)
   ■ Encoder & Video Stream Structure
   ■ Dynamic Frame Selection

3 Experiment Results
Our Testbed

- Implemented in C++ (platform: Linux)
  - Encoder: x264, modified to generate hierarchical-P w/ 3 TLs
  - Decoder: FFmpeg
  - Cellular links emulated by CellSim
- FaceTime used as a benchmark
- Performance metrics: ABW utilization, 95-percentile packet & frame delays
Evaluation under Cellular Bandwidth Traces

Kurdoglu, Liu, Wang, Shi, Gu, Lyu

NYU Tandon School of Engineering, Tencent Co. Ltd.

Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks
Evaluation under Cellular Bandwidth Traces

Rebera avg. util.: 55%, avg. 95-percentile frame delay: 865 msec
FaceTime avg. util.: 32%, avg. 95-percentile frame delay: 2.8 sec

Kurdoglu, Liu, Wang, Shi, Gu, Lyu
NYU Tandon School of Engineering, Tencent Co. Ltd.

Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks
Evaluation under Cellular Bandwidth Traces

Double-link scenarios

Rebera avg. util.: 55%, avg. mean frame delay: 295 msec
FaceTime avg. util.: 32%, avg. mean frame delay: 620 msec
Evaluation under Cellular Bandwidth Traces

Kurdoglu, Liu, Wang, Shi, Gu, Lyu

NYU Tandon School of Engineering, Tencent Co. Ltd.

Rebera: Real-time Bandwidth Estimation and Rate Adaptation for Video Calls over Cellular Networks
Conclusions

- Proposed a real-time bandwidth estimation for video calls → can be integrated with existing CC mechanisms
- Sequential linear prediction allows forecasting → can be further improved: ML, etc...
- Hierarchical-P structure enables easy rate control and some resiliency
- Proposed method integrated into last version of WeChat
Thanks!

Find the code and reproduce our work at

http://vision.poly.edu/

Q&A