Enabling High Quality Real-Time Communications with Adaptive Frame-Rate

Zili Meng, Tingfeng Wang, Yixin Shen, Bo Wang, Mingwei Xu, Rui Han, Honghao Liu, Venkat Arun, Hongxin Hu, Xue Wei
Real-Time Communications (RTC) are increasingly popular.

*Slide taken from Salsify [NSDI’18].
Frame-rate (≥60fps) and resolution (≥1080p) increase simultaneously.

---

**Legacy RTC**

- Frame-rate: 5fps~24fps
- Resolution: 240p~720p

**High-quality RTC**

- Frame-rate: 60fps~120fps
- Resolution: 2K~8K
Motivation
Latency Variation

Emerging RTC applications asks for extremely low stall ratios!

A 0.3 second stall  

0.1% Stall rate

Such a 0.3 sec stall happens every 300 secs (5 min)

*Video source: https://www.youtube.com/watch?v=hfySDsMW8BU
Motivation
Decoder queue overload

Problem identification: Latency comes from the video client

➢ For cloud gaming with short RTT, the latency at the client device might be unimaginably high.
➢ Contribute to **57% of end-to-end stutters** in Tencent START cloud gaming!

<table>
<thead>
<tr>
<th>Root cause of a stutter event</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>44%</td>
</tr>
<tr>
<td>Client device</td>
<td>57%</td>
</tr>
<tr>
<td>Server</td>
<td>&lt;1%</td>
</tr>
</tbody>
</table>
Motivation
Decoder queue overload

Problem identification: Increased video quality overloads the video client.

- Decoder queue *between the network and decoder* is not for low latency.
- A queue will be formulated at the client between the application and network stack.
Motivation
Overload is increasingly severe!

Problem identification: Increased video quality overloads the video client.

- Decoder queue *between the network and decoder* is not for low latency.
- More and more common in RTC

Hardware capacity doubles every 27 months

Application demands of Internet video double every 20 months
Insight: *adapt the frame-rate* to alleviate transient decoder overloads.

- The decoding speed (px/s) depends on the *resolution (px/frame)* and *frame-rate (fps)*.

- Existing work usually *adapt the bit-rate (or resolution)*, which will incur traffic bursts for commercial video codecs.

- We therefore *adapt the frame-rate* to alleviate the overload.
Insight: *adapt the frame-rate* to alleviate transient decoder overloads.

- The decoding speed (px/s) depends on the *resolution (px/frame)* and *frame-rate (fps)*.

**Challenge: achieve an ultra-low queueing delay**

- Existing queue management mechanisms in computer networks *reactively* control the queue length around a target.

![Diagram of control target and internet flow](image)
Challenge: achieve an ultra-low queueing delay

- Existing queue management mechanisms in computer networks reactively control the queue length around a target.
- Decoder queue is at the granularity of video frames (with an interval of $O(10 \text{ ms})$).
- Even a queue of one frame will incur $O(10 \text{ ms})$ delay.
Design
Adaptive Frame-Rate

Solution: *Predictive* frame-rate adaptation.

➢ Predict the queueing delay based on arrivals and departures rather than queue states.

![Graph showing decoding delay and queue length](image-url)
Use Kingman’s formula to be aware of both arrivals and departures.

$$\mathbb{E}(\tau_{\text{queue}}) = \left( \rho \frac{\rho}{1 - \rho} \right) \cdot \frac{c_a^2 + c_s^2}{2} \cdot \mu_s$$

- Adaptive to current decoding speed
- Robust to absorb fluctuation

$$\rho = \frac{\text{arrival rate}}{\text{departure rate}}$$

Adaptive to rate mismatch (average)
Various factors can all lead to transient fluctuations.

➢ Solution: Pattern modelling and matching / filtering

- Decoder degradation
  ➢ Frequency downgrades

- Burst network arrivals
  ➢ Wireless throttling

- Sudden decoder stalls
  ➢ Decoder failure

Stationary controller (queueing theory)

Transient controller (queue length)

Transient controller (head sojourn time)

Please refer to the paper for details!
Evaluation
Experiment Setup

Large-scale trace-driven simulations.

- Simulation traces collected from Tencent START cloud gaming
  - Network RTT, decoding delay, etc.
  - 42k hours (playing time), 38k user sessions.

- Baselines
  - DropTail, FrameSkip [HotEdgeVideo’21]
  - qWait-, qLen-, txRate-based AFR
Evaluation
End-to-end Delay Improvement

Metric: Ratio of *end-to-end delay > 100 ms* (how we define stutter).

Reduce by 55%
Reduce by 19%
➢ The **increased video quality** overloads the client decoder queue.

➢ AFR **adapts the frame-rate based on network / decoder conditions**.

➢ AFR is **deployable** with current video codec.

➢ AFR improves the application performance by **34% in production**.

![Diagram showing the process of increased video quality overloading the decoder queue.](Diagram)
Enabling High Quality Real-Time Communications with Adaptive Frame-Rate

Thank you!
Zili Meng
https://transys.io/afr/