USENIX ATC '25

MARC: Motion-Aware Rate Control for Mobile E-commerce Cloud Rendering

Yuankang Zhao, Furong Yang, Gerui Lv, Qinghua Wu, Yanmei Liu, Jiuhai Zhang, Yutang Peng, Feng Peng, Hongyu Guo, Ying Chen, Zhenyu Li, Gaogang Xie

Background

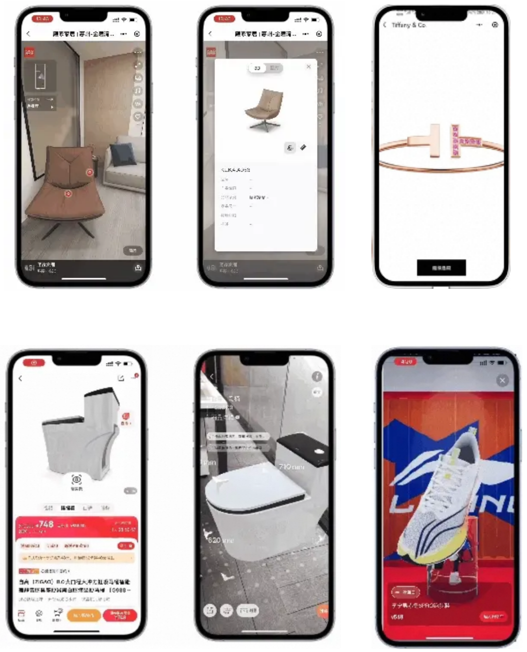
❖ Mobile cloud rendering: A key technology for immersive 3D experiences

Text/Images/Audio/Video

Improve immersion

On device rendering

Interactive 3D Environment



Offload

Cloud rendering

High-fidelity 3D renderings

Challenges:
Computing, Storage, Energy

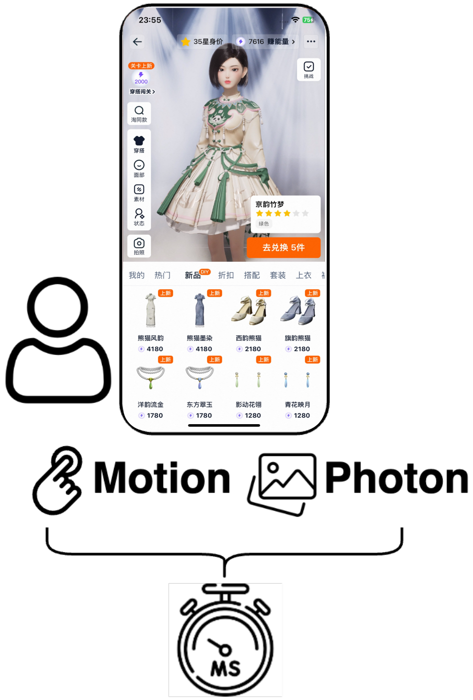
Conflicting QoE requirement

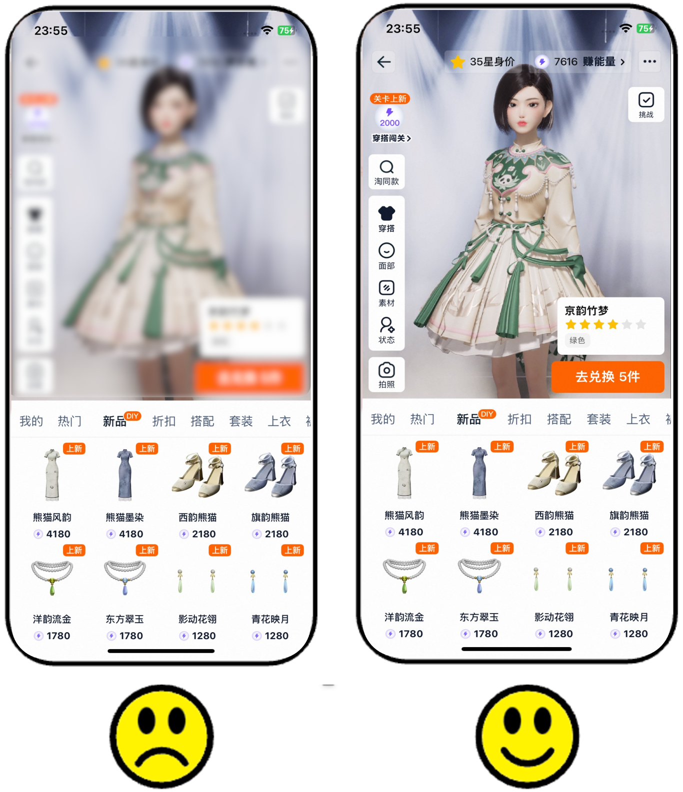
QoE (Quality of Experience) requirements in mobile cloud rendering

Low Latency
Motion-to-photon (MTP) latency < 150ms

vs.

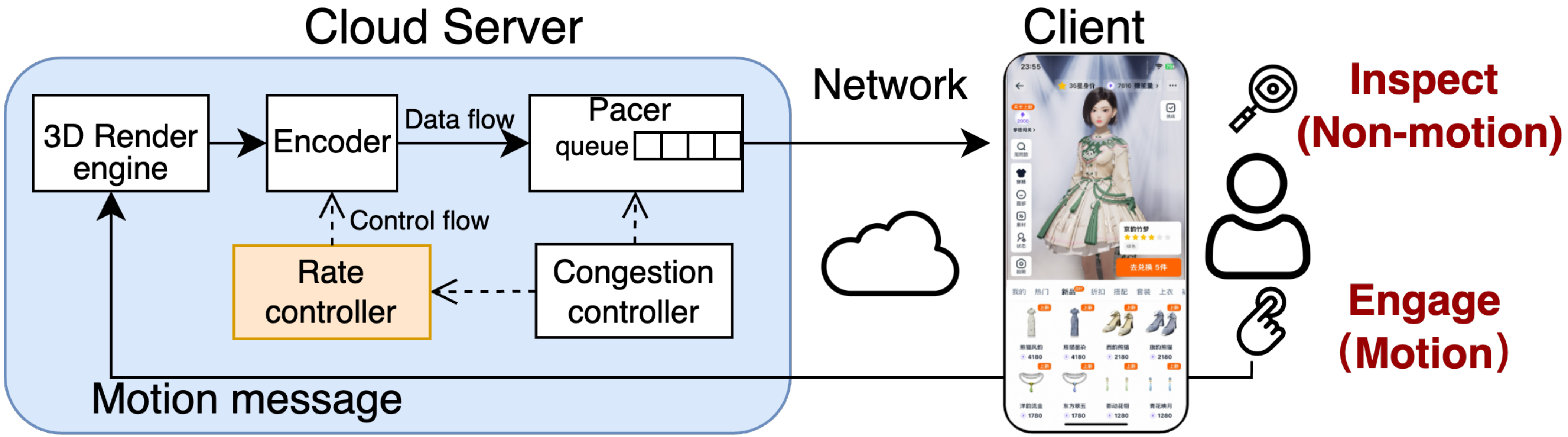
High Visual Quality
Higher video bitrate yields clearer frames





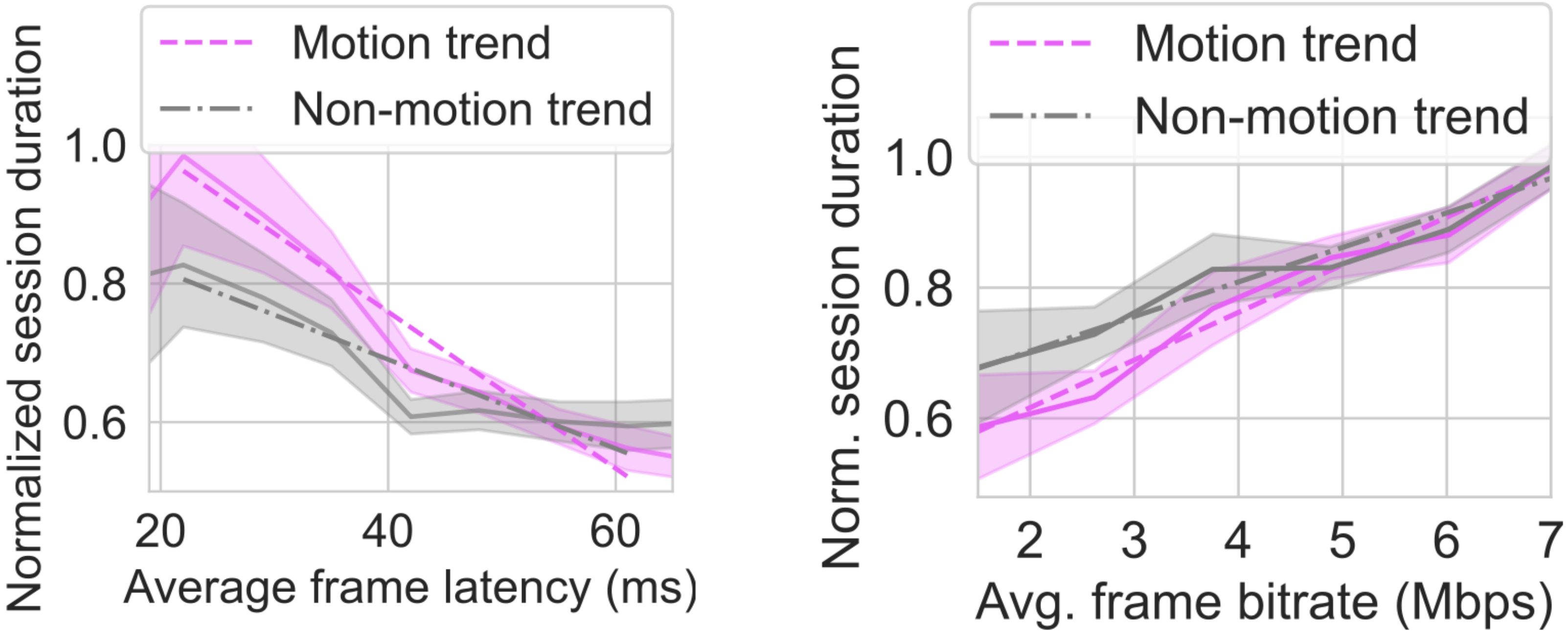
Dilemma: balancing the conflicting goals

Cloud rendering system architecture



Measuring user QoE preference

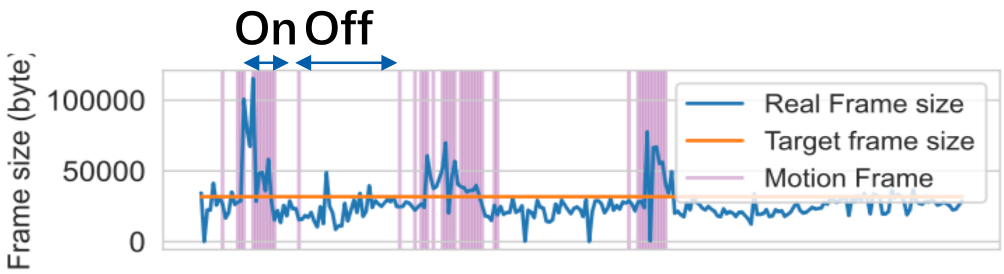
- ❖ Lower frame latency and higher frame bitrate → longer sessions
  - Engagement metric: **average session duration**
- ❖ Observation 1: In motion phases, frame latency has a greater impact on user engagement than bitrate
  - During user interaction, **latency sensitivity is 75.7 % higher** than in non-motion periods.



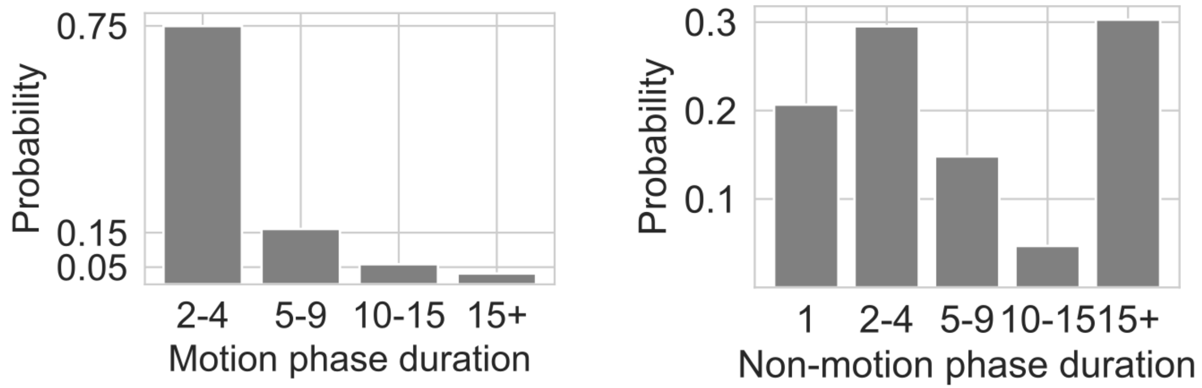
R1: Motion-aware rate control for **dynamic** user preferences

User motion characteristics

- Observation 2: User interactions exhibit an **On-Off pattern**
  - Motion phases are **short**: 74 % last only 2–4 frames (≈ 66–133 ms)
  - 70% of non-motion periods are brief pauses of less than 15 frames.



Time series of user motion

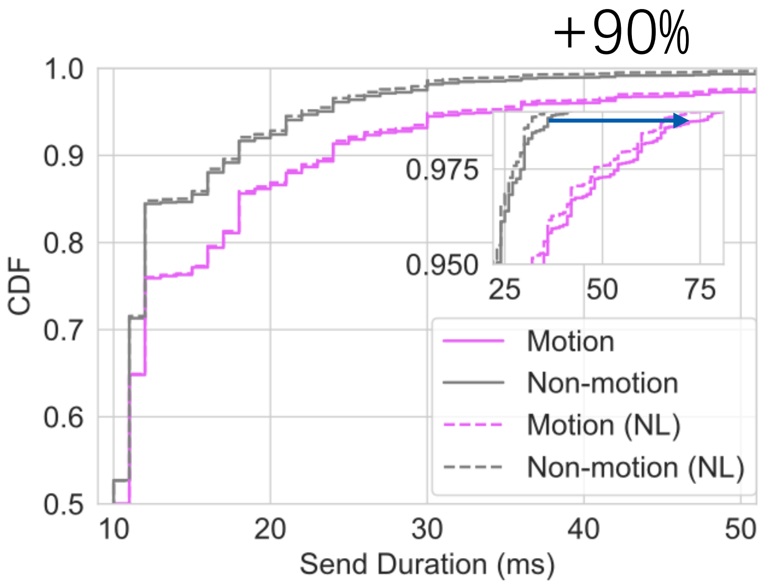


Length distribution of consecutive **motion or non-motion** frames

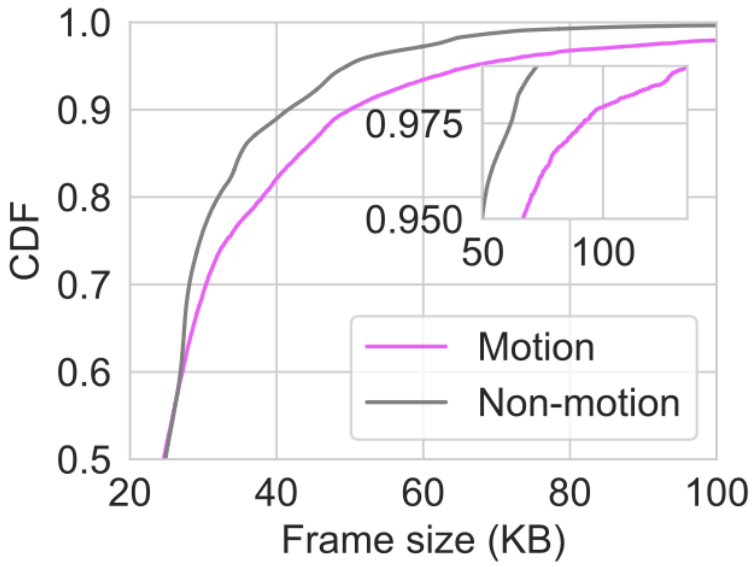
R2: Frame-level decisions are required to keep up with rapid state changes

Motion frames characteristics

- ❖ Observation 3: Motion frames are larger and incur higher latency
  - Motion frames +22% in size，P99 send duration +90%
  - Because changing content requires more bits to encode
- ❖ **Latency spikes often occur when users are most sensitive to interaction delays**



(a) CDF of send duration.



(b) CDF of frame size.

R3: Differentiate bitrate assignment for motion/non-motion frames

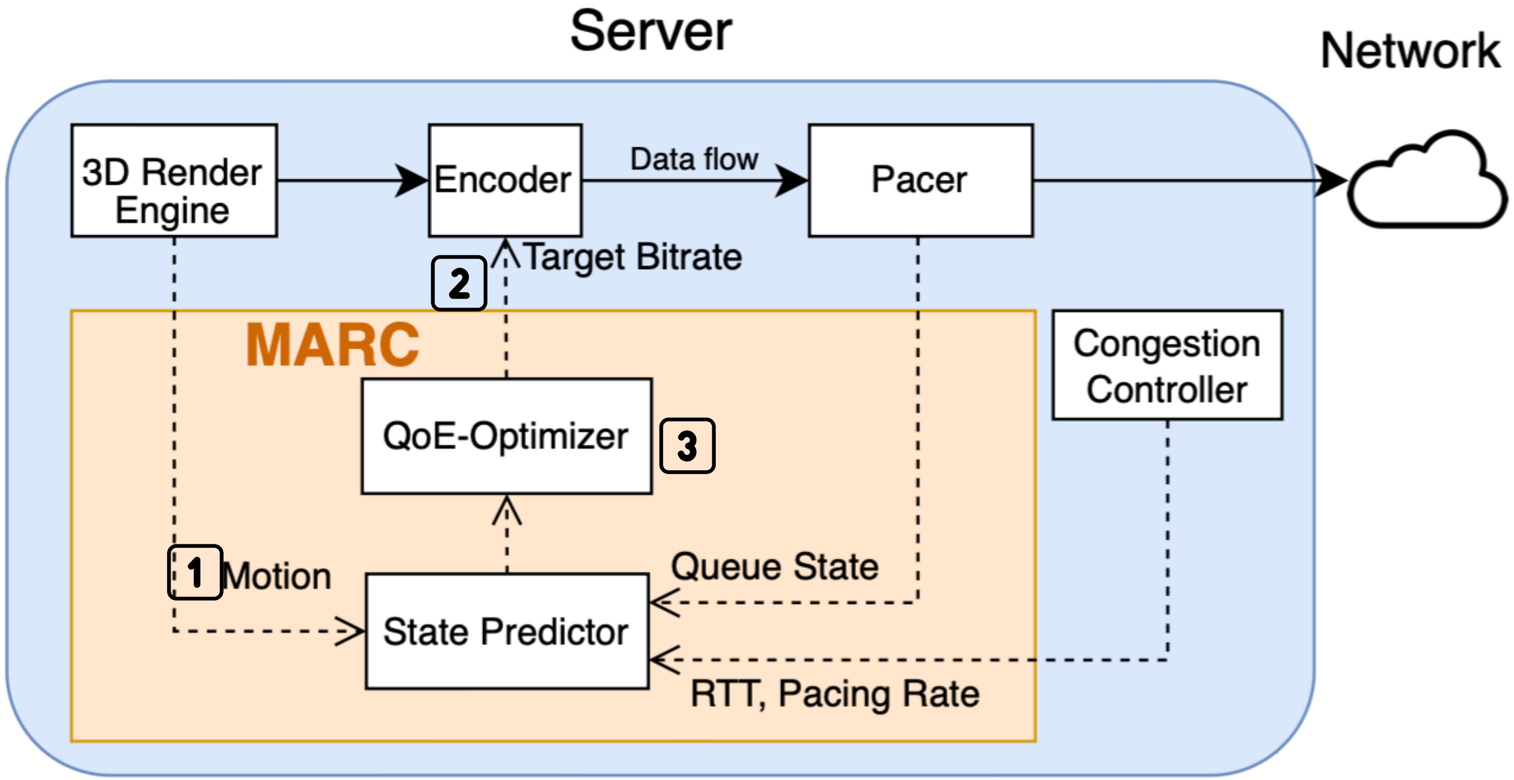
Problem and system design

Issues:

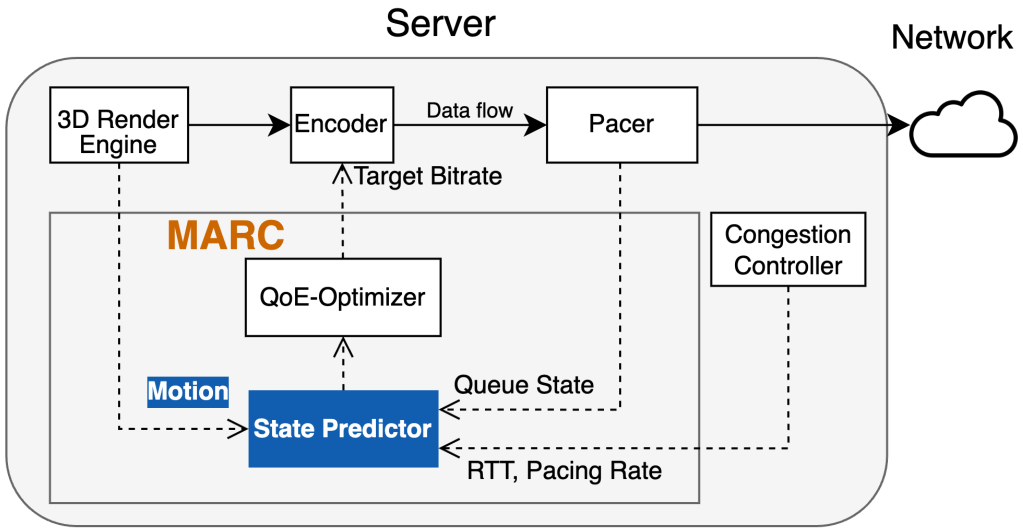
- Ignores QoE preference shifts
- Ignores motion state characteristics

Research goal： Motion aware rate control (**MARC**)

1 Awareness of user motion 2 Frame-level decision 3 Differentiated bitrate assignment



State predictor

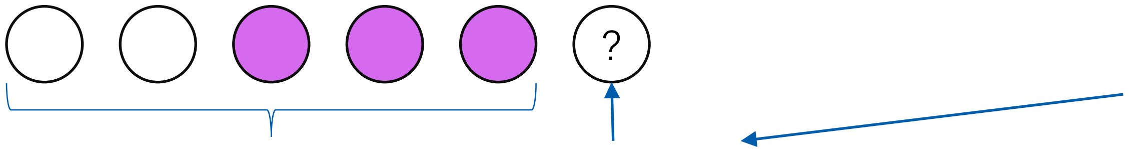


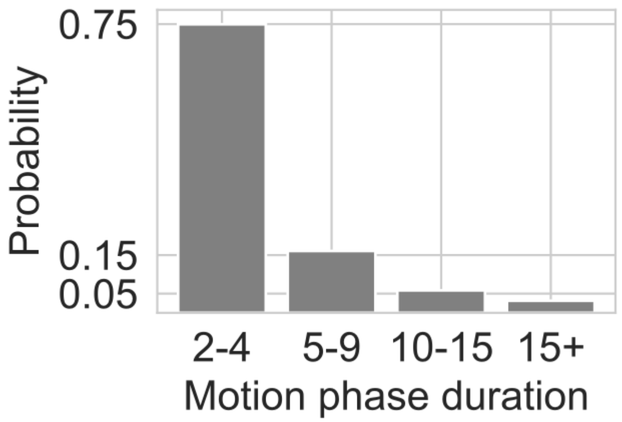
❖ User motion predictor

- The start of a user's motion is **random**
- However, once started, motion tends to be **continuous**

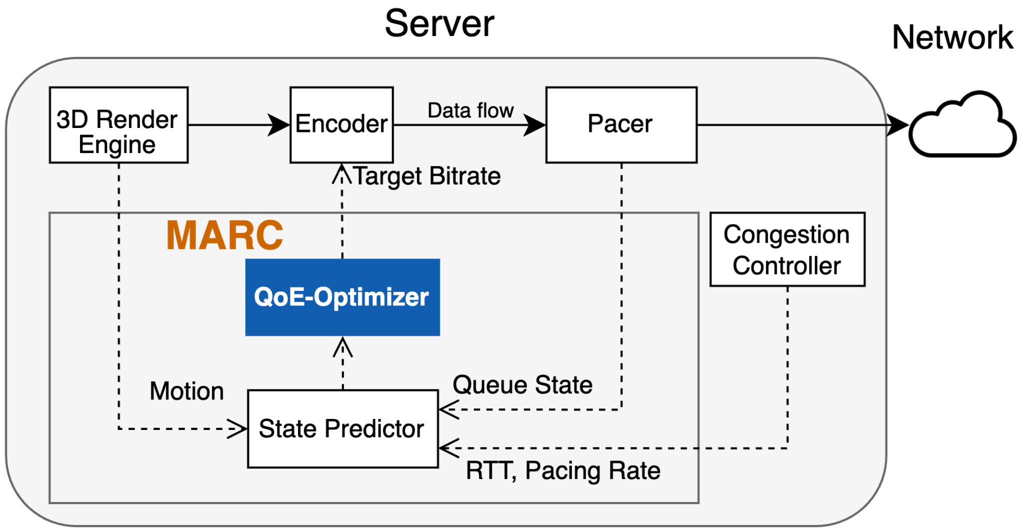
❖ A Markovian model to predict user motion

- Infer next-frame motion status from the previous N frames
- The model learns transition probabilities from large-scale data





MARC's QoE optimizer



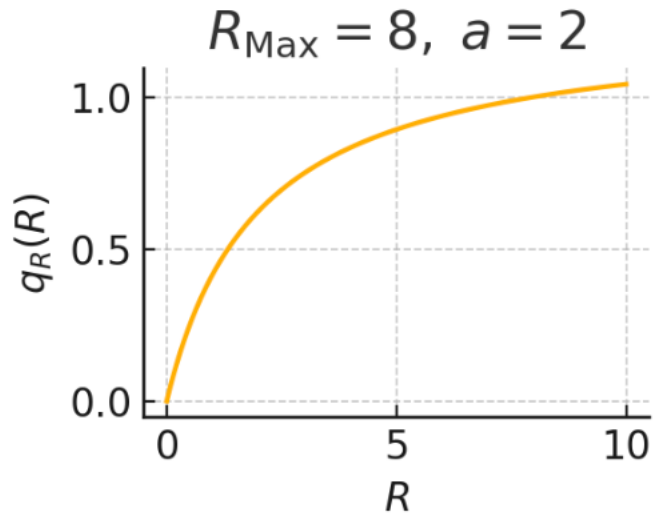
Motion indicator

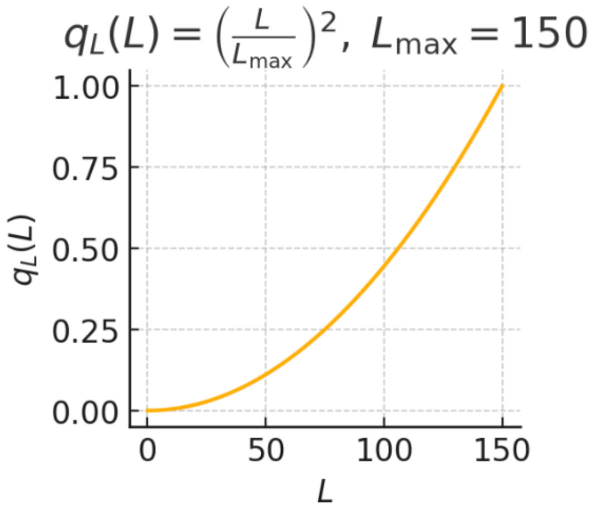
$$QoE_1^N = \sum_{i=1}^{i=N} q_R(R_i) - \sum_{i=1}^{i=N} (\lambda_s + \boxed{M(i)} \times \lambda_m) \times q_L(L(i))$$

Quality | Latency

$$^{*}\ q_R(R) = \left(1 - \frac{1}{R/a+1}\right) \times \frac{R_{Max}+a}{R_{Max}}$$

$$q_L(L) = \left(\frac{L}{L_{max}}\right)^2$$

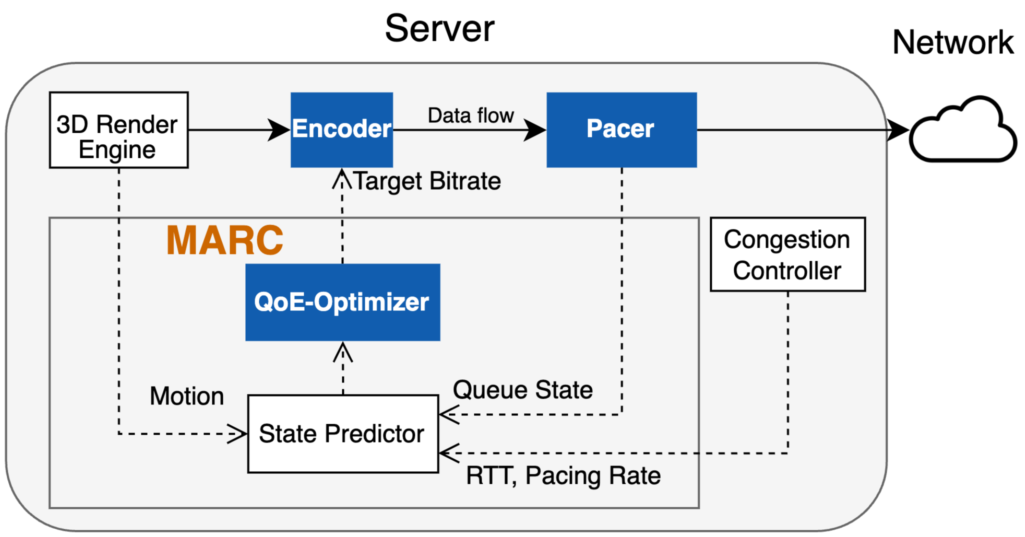


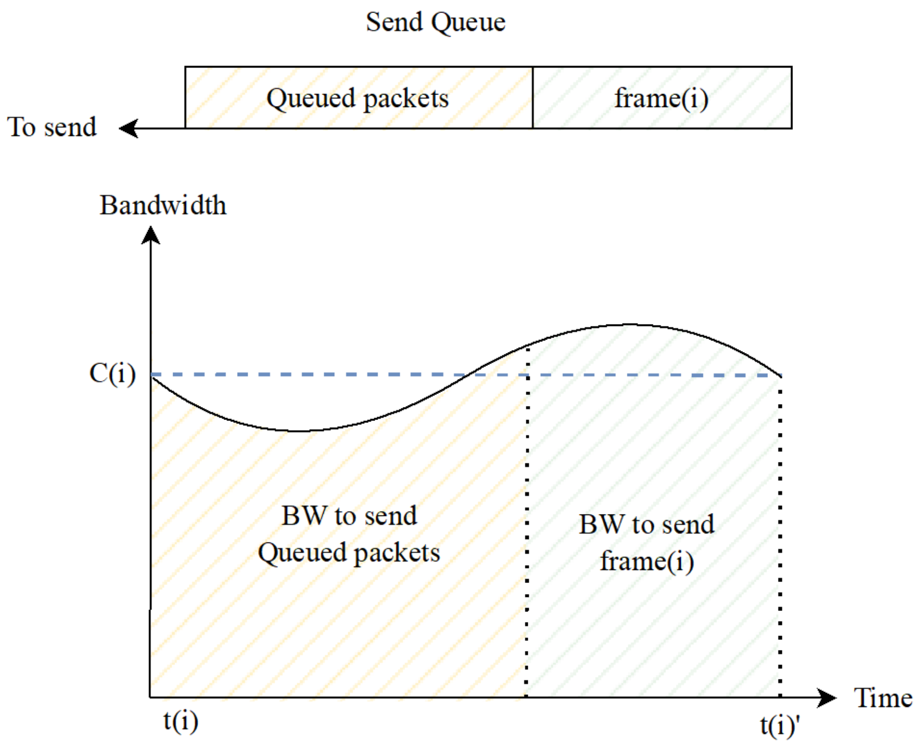


$q_R(R_i)$ is adopted from Ray et al. Vantage: optimizing video upload for time-shifted viewing of social live streams (Sigcomm 2019)

Frame size-delay cascade

❖ Modeling the cascading effects of queueing





Queueing and sending of a frame

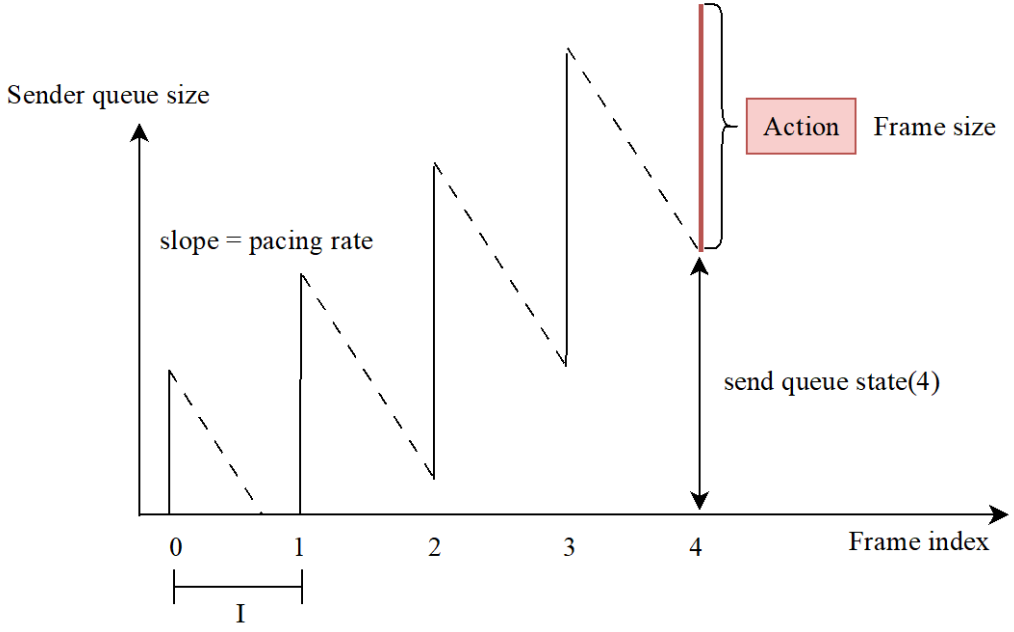
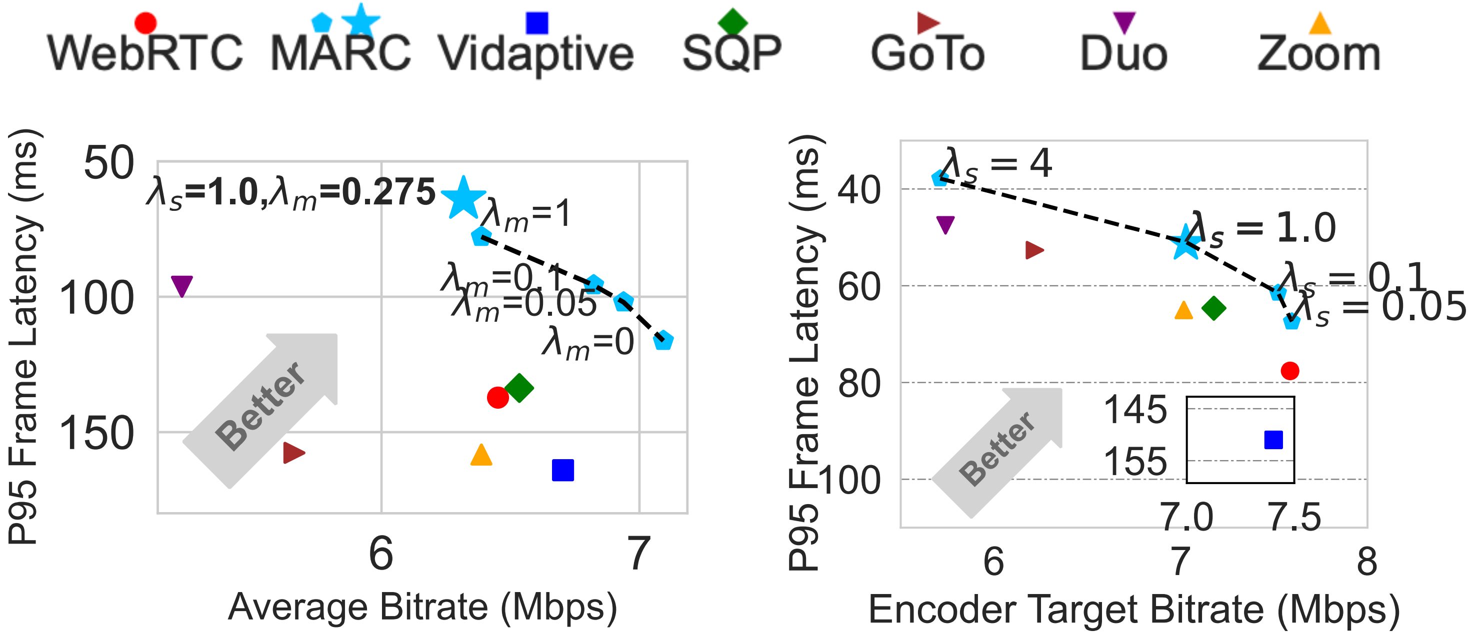


Illustration of send queue over time

Experiment: MARC performance validation

Platform: A simulation environment replaying real-world network and user motion traces from Taobao's production system.

Baselines: WebRTC, SQP, Vidaptive, and models of commercial apps*(GoTo, Duo, Zoom).



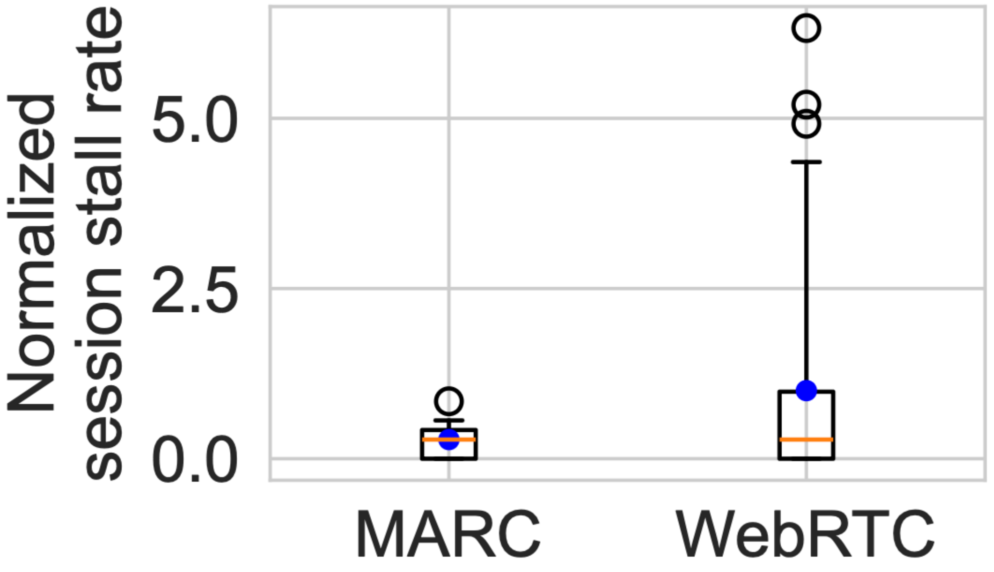
Tail latency for motion frames

Overall tail latency for sessions

*Lee at al. Demystifying commercial video conferencing applications (MM 2021)

Online A/B test results

- An A/B test was conducted on Taobao's platform with over **1 million** user sessions.
- Online results (MARC vs. WebRTC)
  - Average session stall rate was reduced by **71%**
  - User interaction time increased by **20%**
  - Average user session duration: increased by **9%**



- Performance overhead
  - **Client-side: zero** overhead
  - **Server-side: 1.3%** computation overhead increase per session.

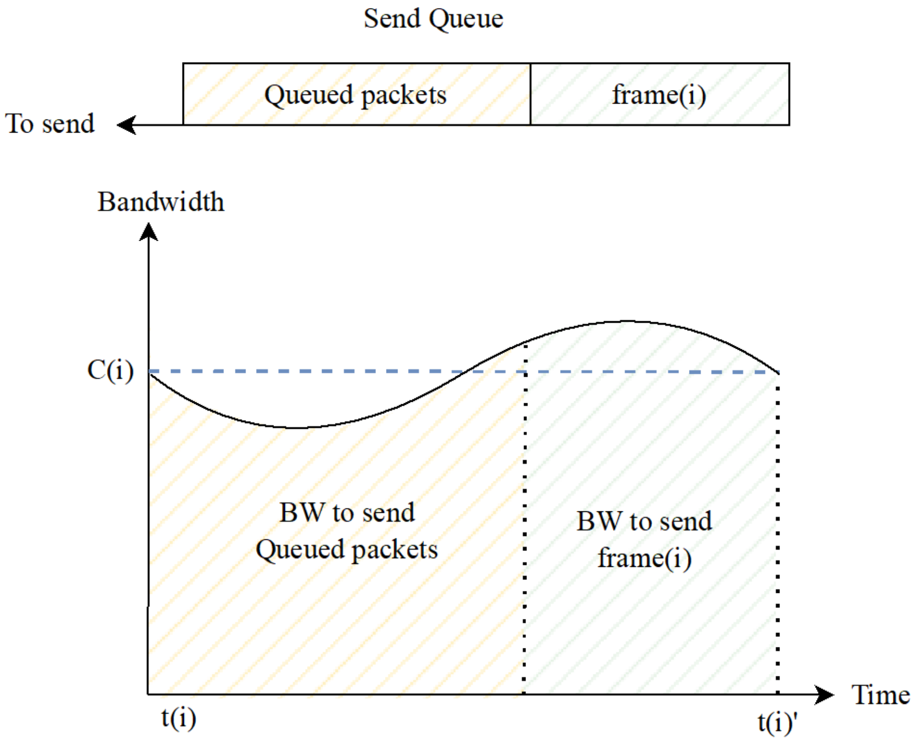
Takeaway

- **Discovery**: user QoE preference evolves dynamically with user motion
  - Users are most sensitive to latency during interaction, which is precisely when existing systems deliver the worst performance.
- **Solution**: We proposed **MARC**, a motion-aware rate control framework.
  - MARC dynamically optimizes a QoE objective that balances quality and latency according to real-time user behavior.
- **Impact**: MARC was deployed in a large-scale production environment
  - MARC **reduced session stalls** and **improved user engagement**, demonstrating its effectiveness.

Thanks for listening / Q&A

yuankangzhao@gmail.com

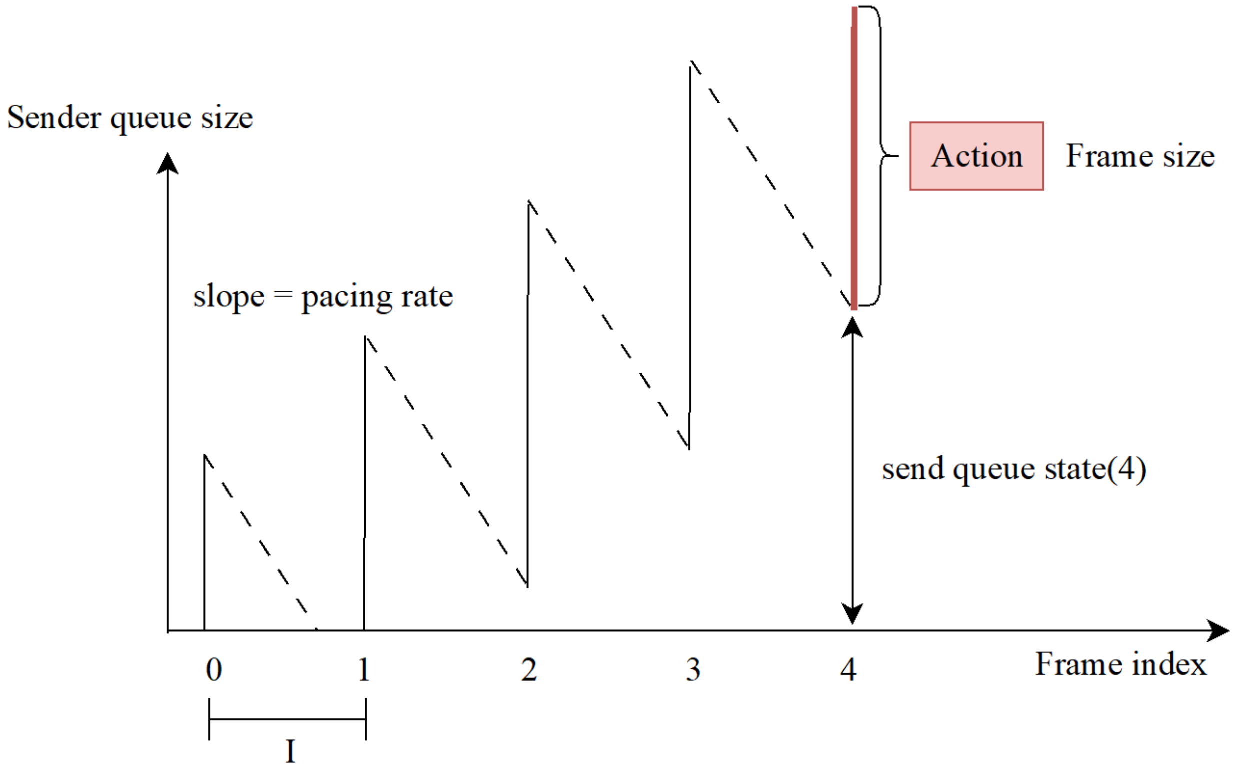
Appendix



$$t(i)' = t(i) + s(i)$$

$$s(i) = \frac{b(i) + d(R_i)}{C(i)}$$

$$C(i) = \frac{1}{t(i)' - t_i} \int_{t_i}^{t(i)'} C_t dt,$$



$$b(i) = \left(b(i-1) + d(R_{i-1}) - \int_{t_{i-1}}^{t_i} C_t dt \right)_+$$