MPEG-TFRCP: Video Transfer with TCP-friendly Rate Control Protocol

Department of Informatics and Mathematical Science,
Graduate School of Engineering Science, Osaka University, Japan
Masaki Miyabayashi
E-mail: miyabays@ics.es.osaka-u.ac.jp

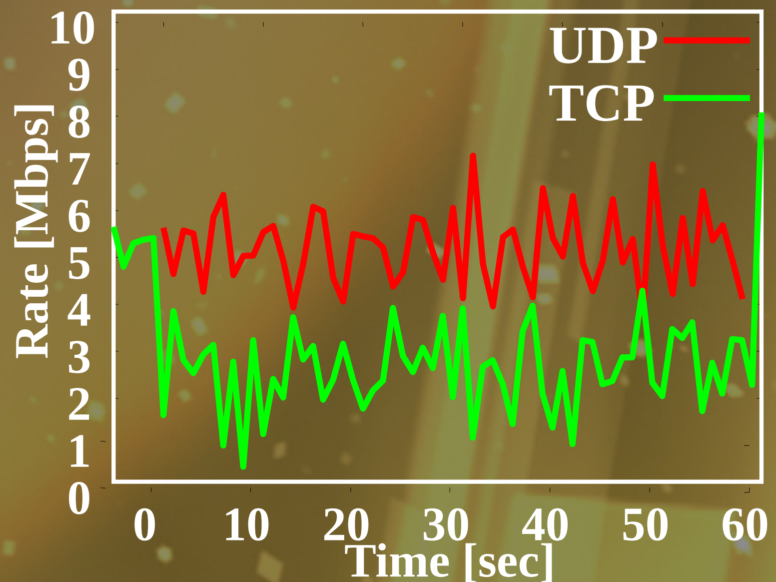
Introduction

- Unfairness: TCP vs.UDP
  - **TCP**: traditional data applications
    - Congestion control
  - **UDP**: real-time multimedia applications
    - No control mechanisms

Use of multimedia apps. increases

"Greedy" UDP degrades TCP performance

TCP, UDP co-exist

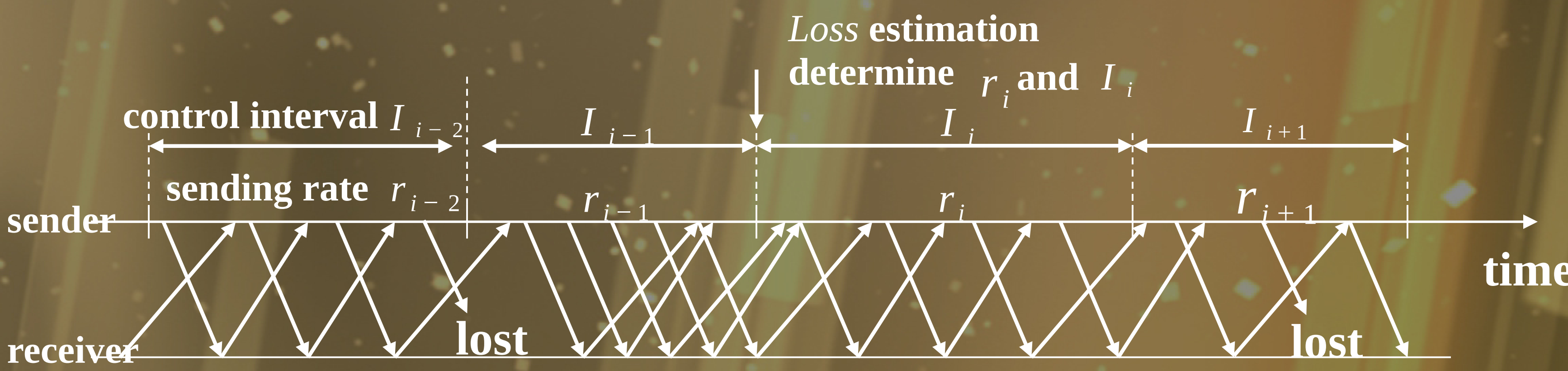


TCP-friendly rate control

"A non-TCP connection should receive the same share of bandwidth as a TCP connection if they traverse the same path."

- TFRCP (TCP-friendly Rate Control Protocol)
  - Equation-based control
    - Estimate TCP throughput
  - AIMD control
    (Additive Increase/Multiplicative Decrease)

MPEG-TFRCP mechanisms



1. Estimate network condition from feedback information
   - ✓ Estimation of RTT, Loss

➢ how to get feedback information?

2. Derive TCP throughput
   - i. No loss, $r_{i+1} = 2 \times r_i$
   - ii. Loss, $$r_{i+1} = \frac{MTU}{RTT\sqrt{\frac{2p}{3}} + T_0 \, min(1, 3\sqrt{\frac{3p}{8}}) p (1+32p^2)}$$

➢ applicability for real system?

3. Regulate the sending rate
   - ✓ adjusting MPEG-2 video rate

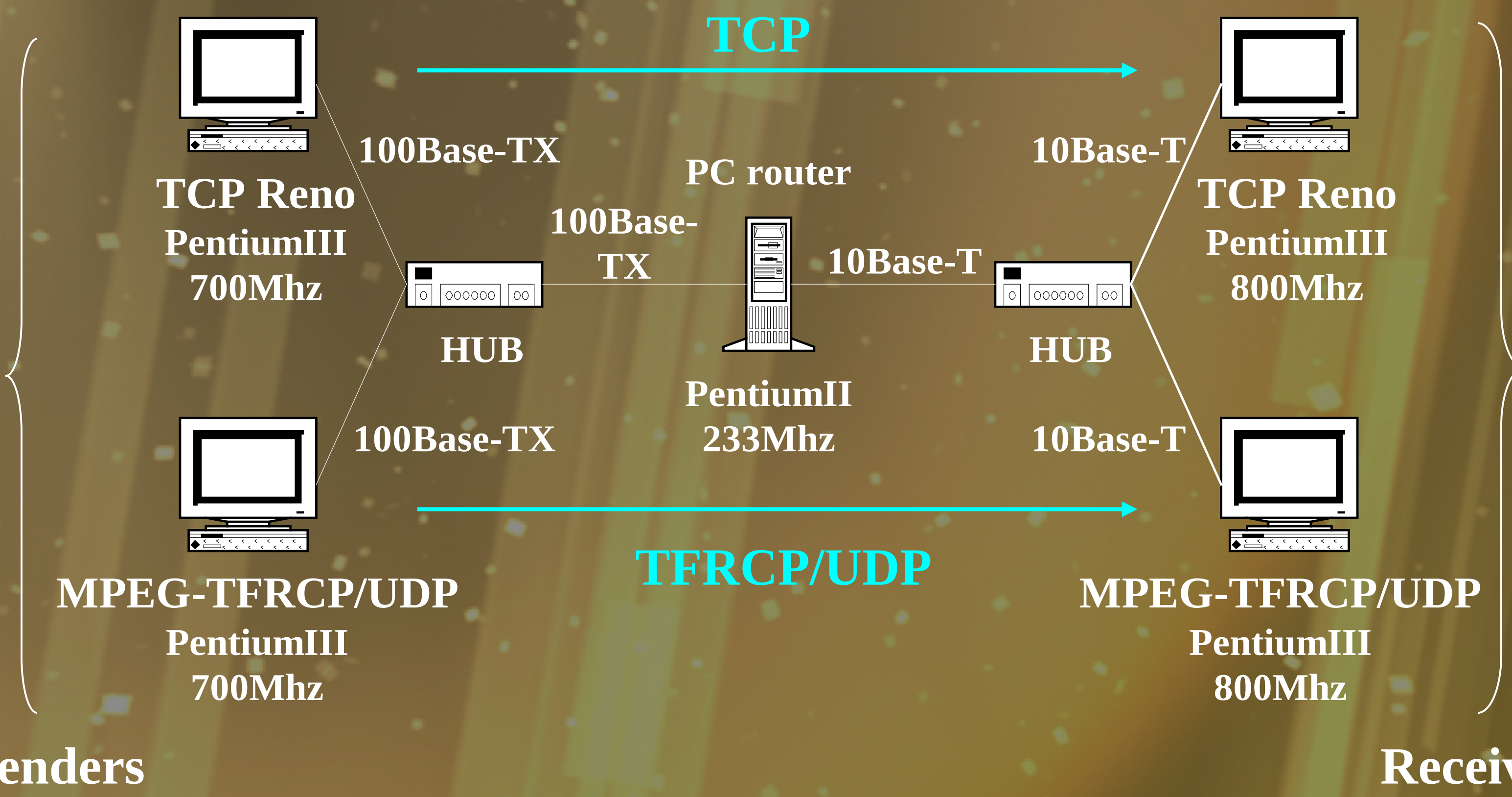
➢ perceived video quality?

Ref [10]: Naoki Wakamiya, Masayuki Murata, and Hideo Miyahara, "On TCP-friendly video transfer," in *Proceedings of SPIE International Symposium on Information Technologies 2000*, November 2000.

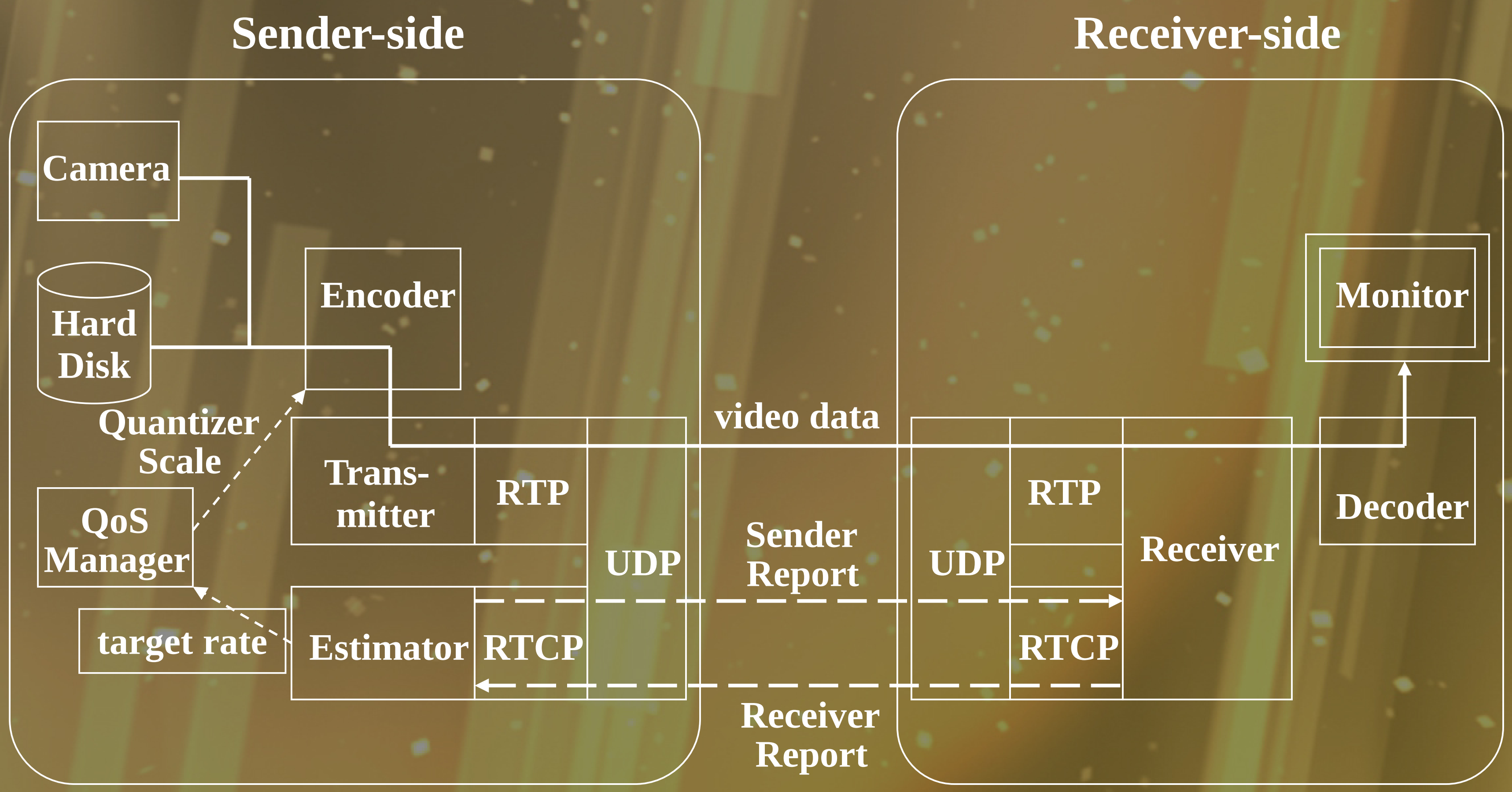
Research Targets

- Demonstrate applicability of MPEG-TFRCP to real system
  - Perceived video quality at receiver
    - MOS (Mean Opinion Score)
  - Observation of traffic on the link
    - Average throughput
    - Rate variation
- Improve MPEG-TFRCP
  - Rate control algorithm
  - Control interval

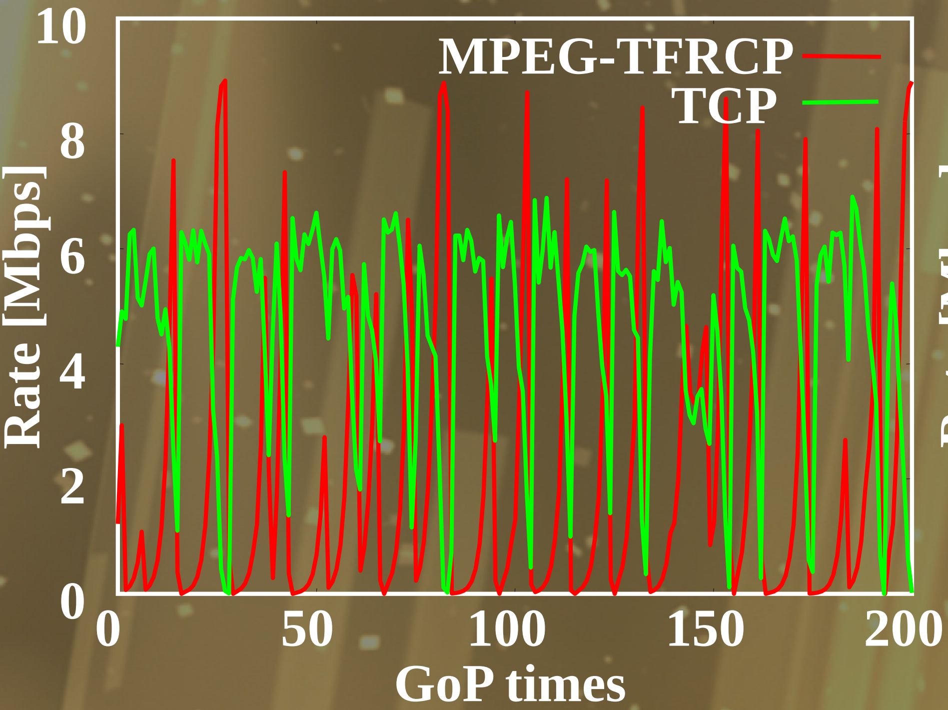
System configuration

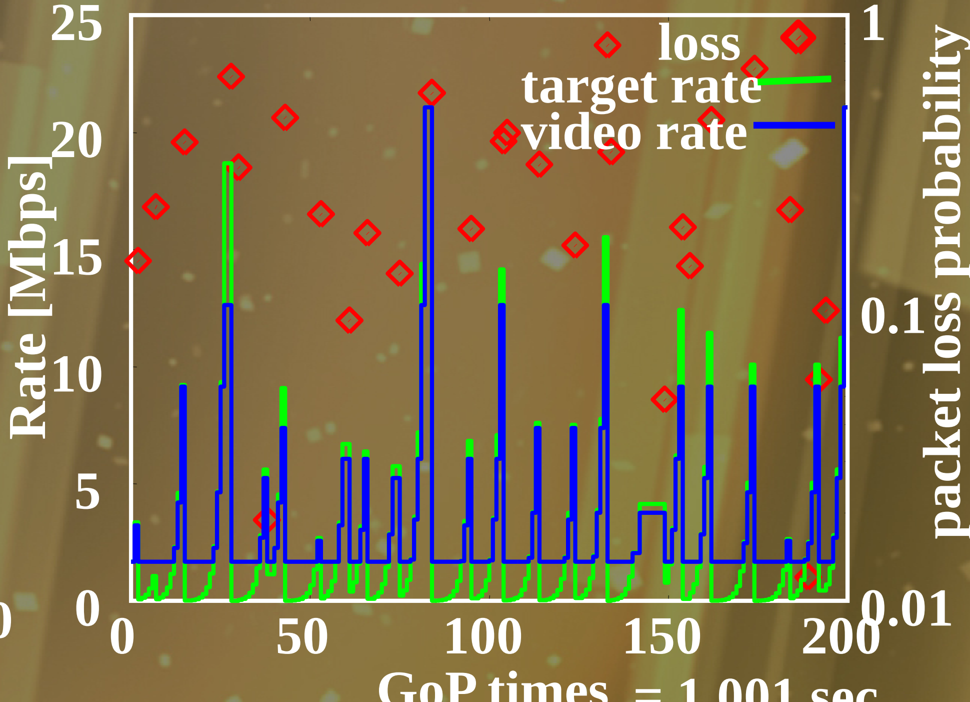


MPEG-TFRCP sender & receiver



Original MPEG-TFRCP





= 1.001 sec

- Drastic rate variation
  - Increasing exponentially, decreasing extremely
- Average throughput: TCP 4.4 [Mbps], TFRCP 2.0 [Mbps]
  - Not TCP-friendly
- Lower subjective video quality: MOS 1.25

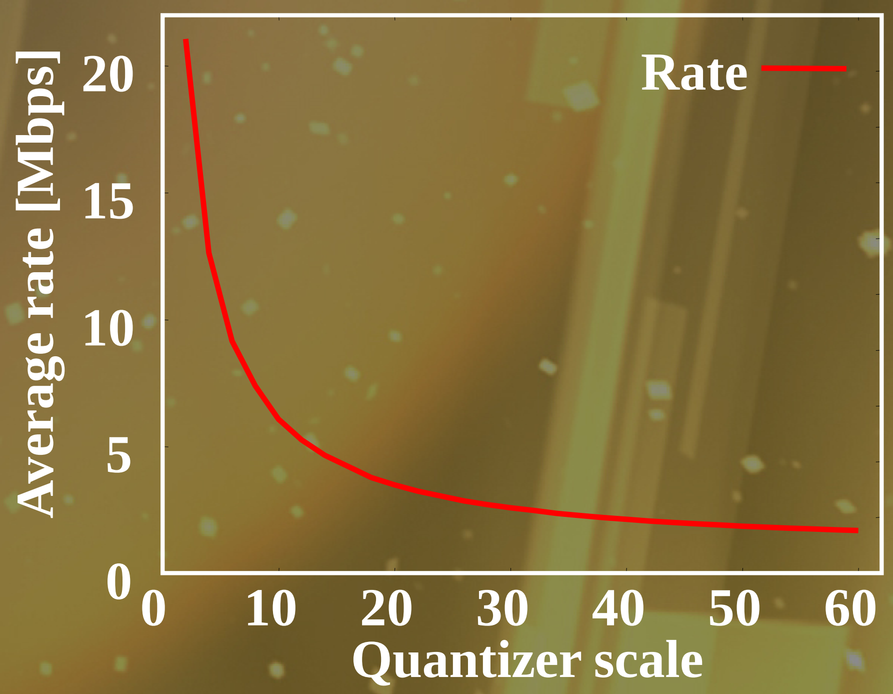
Improving rate control algorithm

- Quantizer-scale-based Additive Increase algorithm (QAI)
  - When no loss occurs,
    - Increase sending rate with regard to quantizer scale

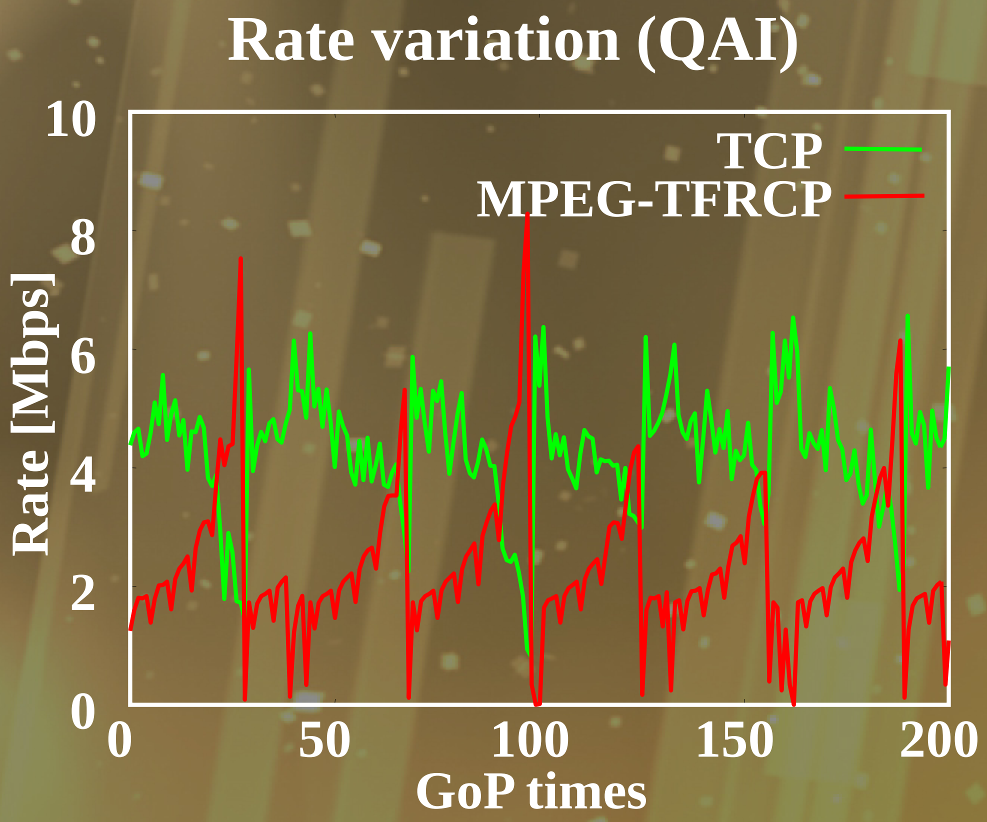
      ⇒ Decrease quantizer scale by two

      Initially set at 60
  - When loss occurs,
    - Original algorithm

$$r_{i+1} = \frac{MTU}{RTT\sqrt{\frac{2p}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3p}{8}}\right) p \left(1+32p^2\right)}$$



Evaluation of QAI MPEG-TFRCP



- ○ Rate variation becomes relatively smaller
- × Not TCP-friendly
  - TCP : 4.3 [Mbps]
  - TFRCP : 2.3 [Mbps]
- × Not attain high-quality video transfer (MOS)
  - UDP : 4.25
  - Original : 1.25
  - QAI : 2.50

Variants in packet loss probability derivation

- Original

$$\text{Loss} = \frac{\text{Number of Lost packets}}{\text{Number of transmitted packets}}$$, within each control interval

  – React so quickly against short-term congestion

  ⇨ Extreme rate fluctuation

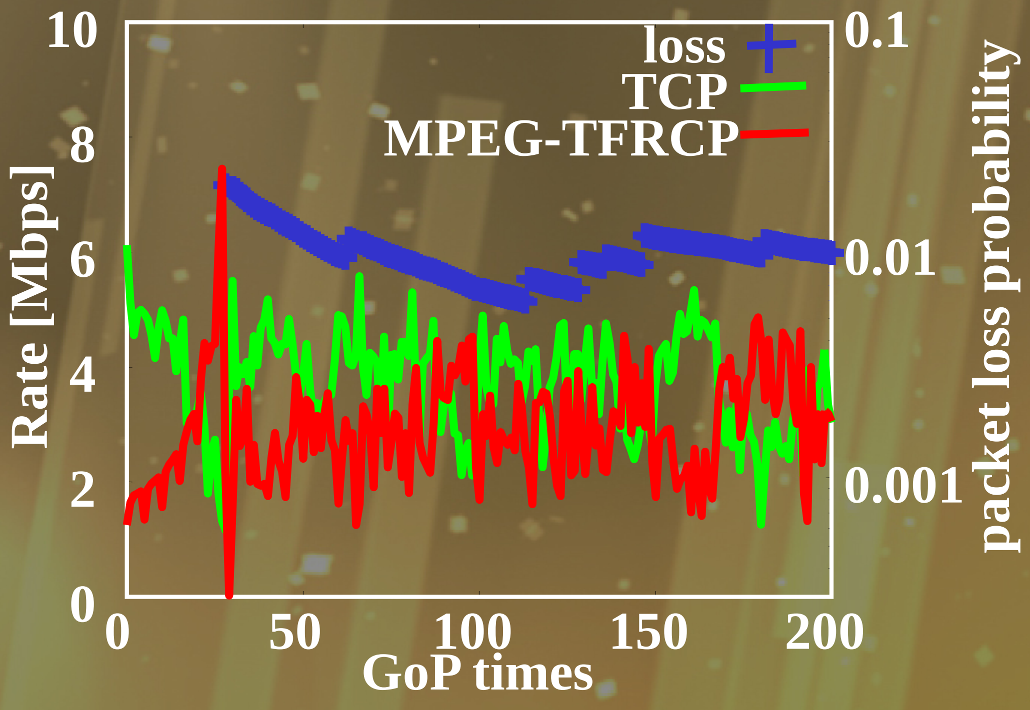
- Cumulative packet Loss probability (CL)

$$\text{Loss} = \frac{\text{Total number of Lost packets}}{\text{Total number of transmitted packets}}$$

, from beginning of the session

Evaluation of QAI-CL

Rate and packet loss probability variations (QAI-CL)



- Rate variation becomes relatively small
- Improve video quality
  - Original : 1.25
  - QAI : 2.50
  - QAI-CL : 3.00
- accomplish reasonable TCP-friendly control
  - TCP : 3.7 [Mbps]
  - QAI-CL : 3.0 [Mbps]

Election of the control interval

- When interval is too short,
  - Perceived video quality becomes unstable
  - Cannot estimate network condition precisely
- When interval is too long,
  - Cannot follow changes of network condition

$$Interval = \left\lceil \frac{32 \times RTT}{GoPtime} \right\rceil \times GoPtime$$

- **32-RTT: proposal**
- **8-RTT:**, **16-RTT:** — **shorter**
- **64-RTT:**, **96-RTT:** — **longer**

Several settings of control interval

	Throughput [Mbps]		Friendliness	MOS value
	TFRCP	TCP		
8-RTT	3.10	3.53	0.878	2.25
16-RTT	2.87	3.71	0.774	3.25
32-RTT	2.97	3.70	0.802	3.00
64-RTT	2.51	4.06	0.618	3.33
96-RTT	2.33	4.29	0.543	2.50

16-RTT or **32-RTT** control interval is appropriate

Conclusion

- Conclusions
  - Evaluated applicability of proposed method to real system
  - Improving the TCP-friendliness and perceived video quality by our method (QAI-CL MPEG-TFRCP)
- Future work
  - Larger scale network
  - Consider RTT variation