Consumer driven Adaptive Rate Control for Real-time Video Streaming in CCN/NDN

Takahiro YONEDA, Ryota OHNISHI,
Eiichi MURAMOTO(Presenter),
R&D Division, Panasonic Corporation
Jeff Burke, UCLA

Contact: muramoto.eiichi@jp.panasonic.com

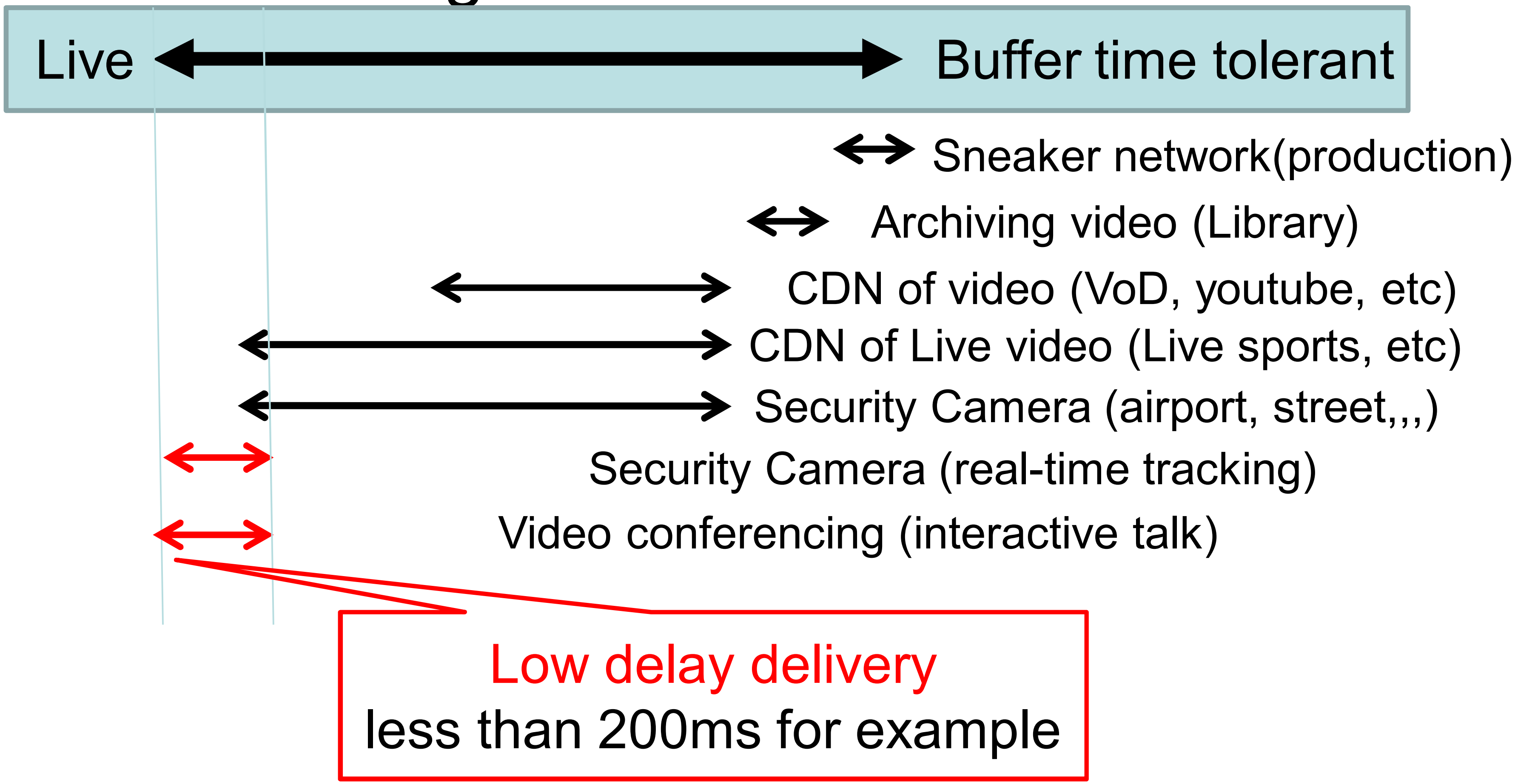
Paper will be to appear in IEICE (conditional acceptance)

Table of Contents

- Focused requirements and target applications
- Function of PIT, CS in CCN/NDN (background knowledge)
- 2 types of RTT variation
  - Source change, congestion
- Proposed method
  - Receiver driven, focusing on RTT fairness
- Simulation result
  - Single bottleneck (basic), multiple-RTT
- Implementation
- Conclusion

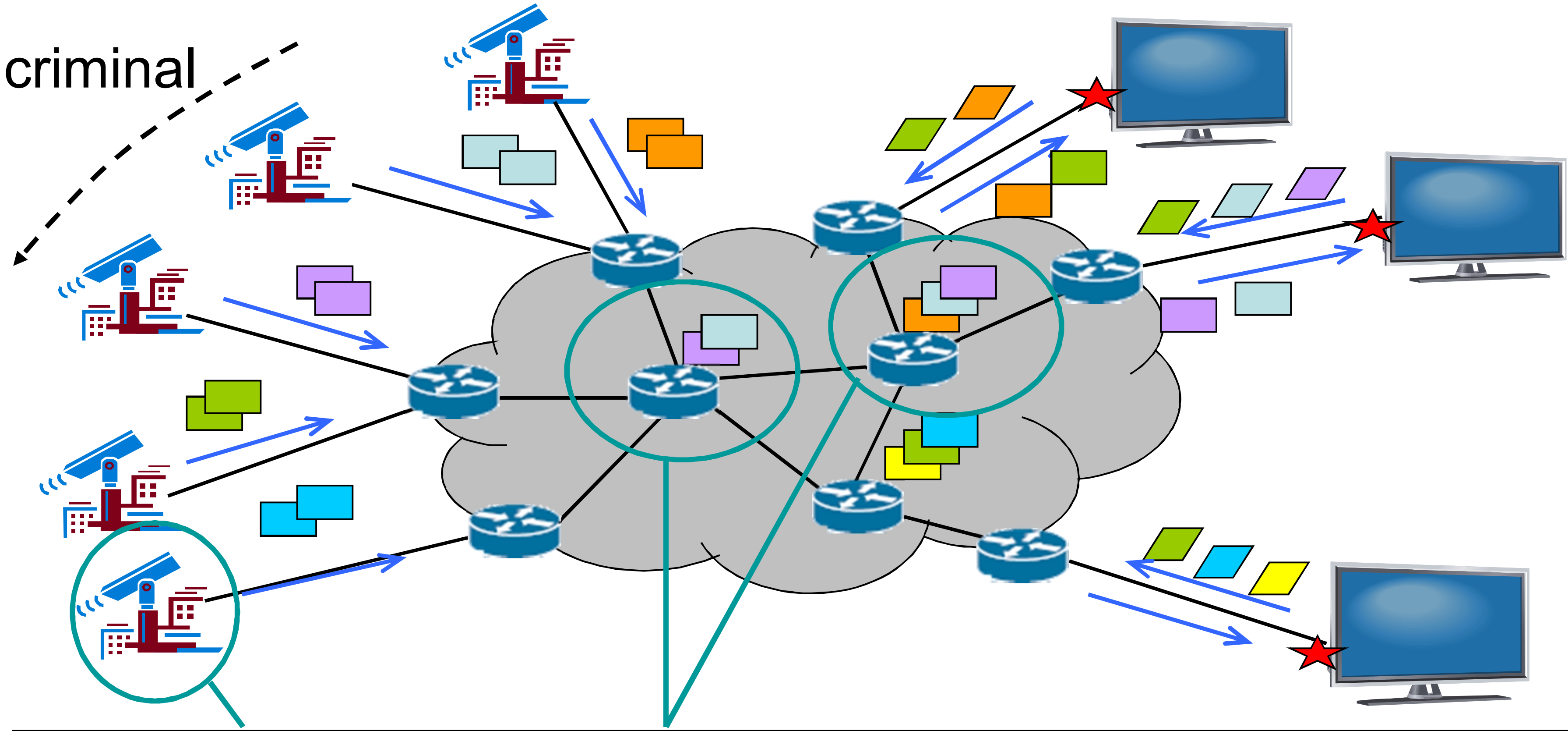
Requirement for the Real-time Adaptive Rate Controlling

- Target: Live real-time streaming like conferencing



Example of the target application

- Security camera, Real-time tracking
- Multiple user access to the different sources

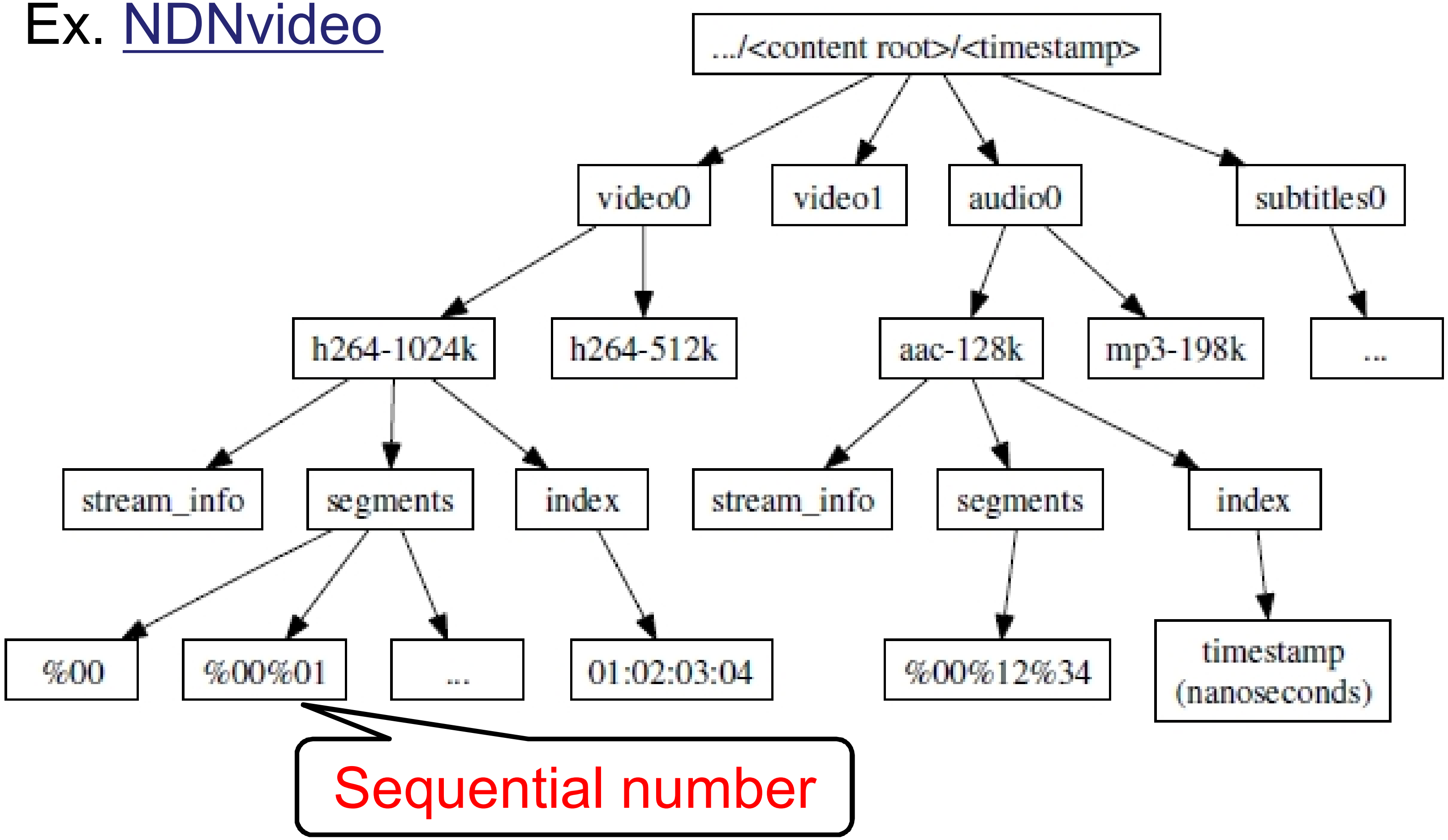


Some content (at certain bit-rate) might be cached on router

Assumption for the target application

- Data (frame data) is divided into a plurality of data chunk
- Each data chunk has sequential number (in its name)

Ex. NDNvideo



Panasonic

Background knowledge: CCN/NDN, CS and PIT

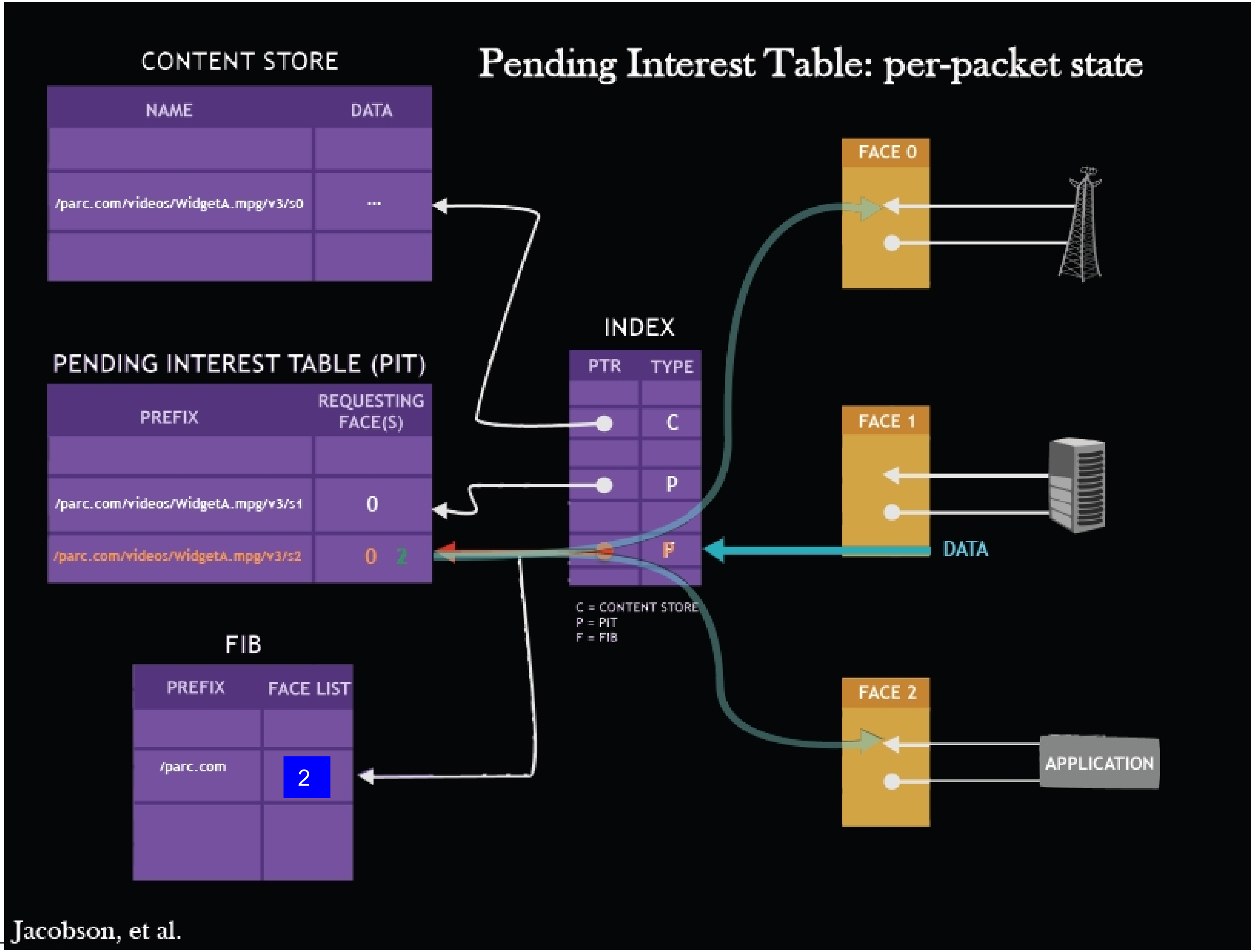
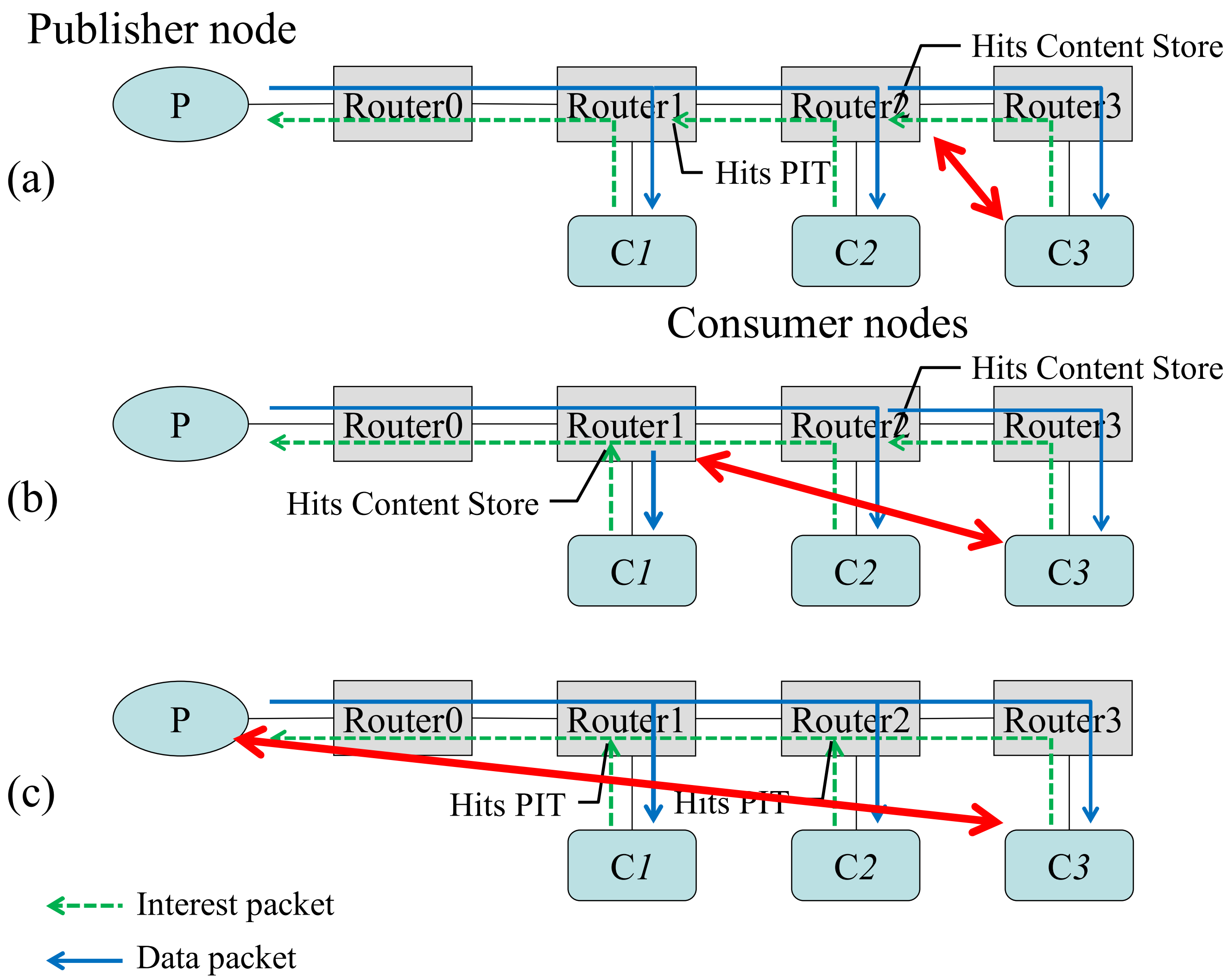


Fig: Presentation at Panasonic “Named Data Networking(NDN)”, Jeff Burke, June 2013

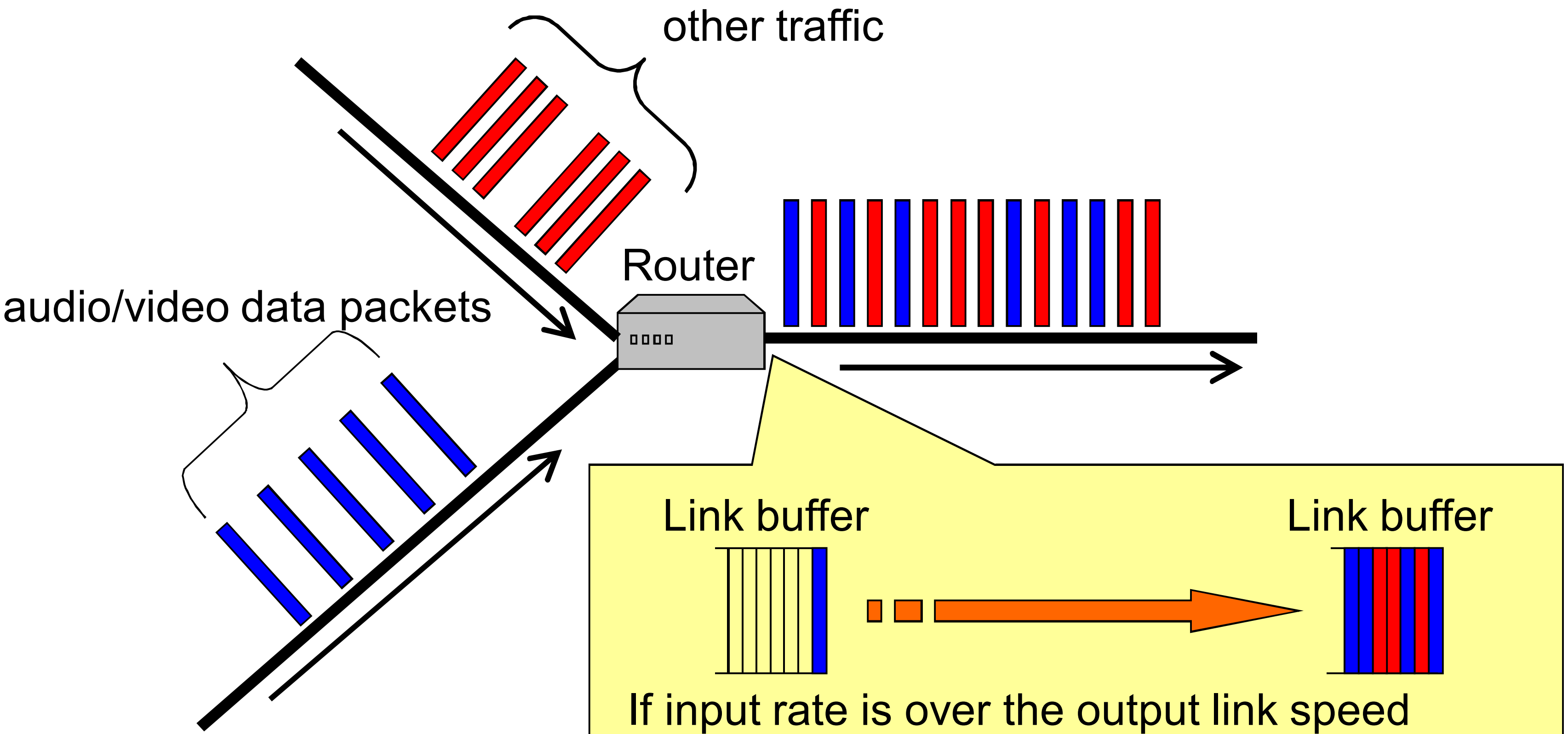
Panasonic

(1) RTT variation by source change (unexpected)



Panasonic

Queuing delay increasing



If input rate is over the output link speed incoming packets are stacked in the link buffer, so that network delay of each packet is increase.

Problem scope

Targets

- Keep low latency transmission & available best throughput
- Maintain RTT fairness (self fairness + RTT fairness)

Points

- Consumer-driven , (no router support)
- Network bandwidth estimation based on RTT variation & packet loss
- Control Interest sending rate according to the bandwidth estimation
- Select video stream bit-rate according to the bandwidth estimation
- Considering 2 types of RTT variation (unexpected or congestion)

Related works (1)

Consumer-driven approach

- AIMD based transport mechanism [1-3]
  - Low throughputs in large RTT environment
  - Easy to increase queuing delay
- Live video distribution [4,5]
  - fixed sliding window might be assumed?
  - No adaptability for network bandwidth variation

[1] Giovanna Carofiglio, et al. Icp: Design and evaluation of an interest control protocol for content-centric networking. INFOCOM NOMEN Workshop, 2012.
[2] Stefano Salsano, et al. Transport-layer issues in information centric networks. ACM SIGCOMM ICN Workshop, 2012.
[3] Somaya Arianfar, et al. Contug: A receiver-driven transport protocol for content centric networks. IEEE ICNP, 2010
[4] Ciancaglini V., et al. CCN-TV: A Data-centric Approach to Real-Time Video Services. Advanced Information Networking and Applications Workshops. 2013.
[5] Derek Kulinski, and Jeff Burke. NDNVideo: Random-access Live and Pre-recorded Streaming using NDN. In Technical Report http://named-data.net/techreport/TR007-streaming.pdf

Related works (2)

Router support approach

- Hop-by-hop Interest flow sharping mechanism [6]
  - Problem of deployment

[6] Giovanna Carofiglio, et al. Joint hop by hop and receiver-driven interest control protocol for content-centric networks. ACM SIGCOMM ICN Workshop, 2012.

Proposed method

Receiver driven

1. Measure RTT on receiving each Data packet

2. Calculate average RTT in each short period

3. Control Interest sending rate in each short period

- *AvgRTT* ≦ *(RTTmin + jitter_offset)* or *Consecutive AvgRTT decrease*

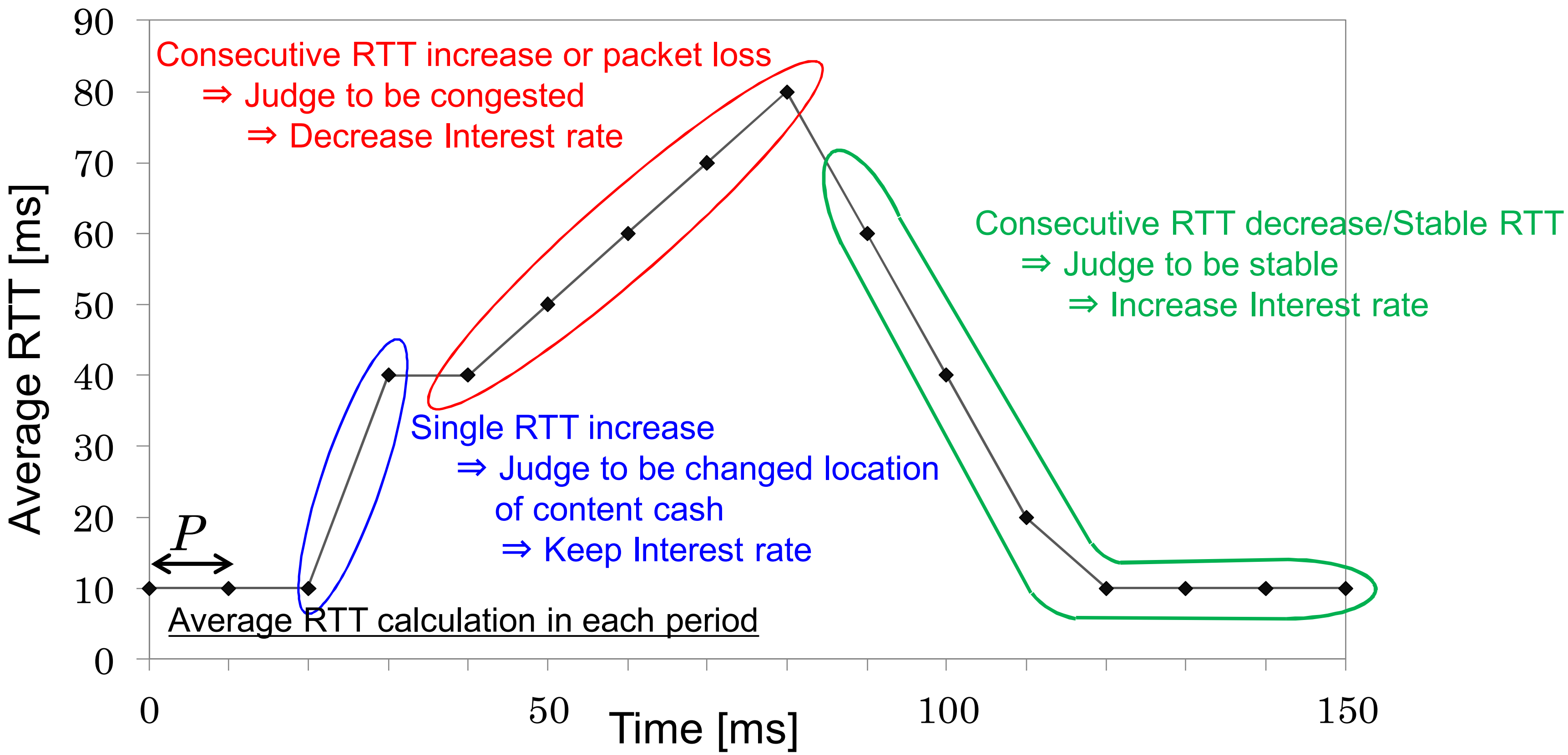
  $$pps_{now} \leftarrow pps_{prev} + \alpha / \sqrt{pps_{prev}} \qquad (\alpha \geqq 1)$$

- *Consecutive AvgRTT increase* or *Packet loss*

  $$pps_{now} \leftarrow pps_{prev} - \beta \cdot \sqrt{pps_{prev}} \qquad (0 < \beta < 1)$$

AvgRTT : Average RTT in each short period
RTTmin : Minimum RTT
pps : Number of sending Interest packet per second

Distinguish consecutive RTT change and unexpected one



Increase Interest rate

$$pps_t \leftarrow pps_{t-P} + \alpha / \sqrt{pps_{t-P}}$$

Decrease Interest rate

$$pps_t \leftarrow pps_{t-P} - \beta \cdot \sqrt{pps_{t-P}}$$

Interest sending interval

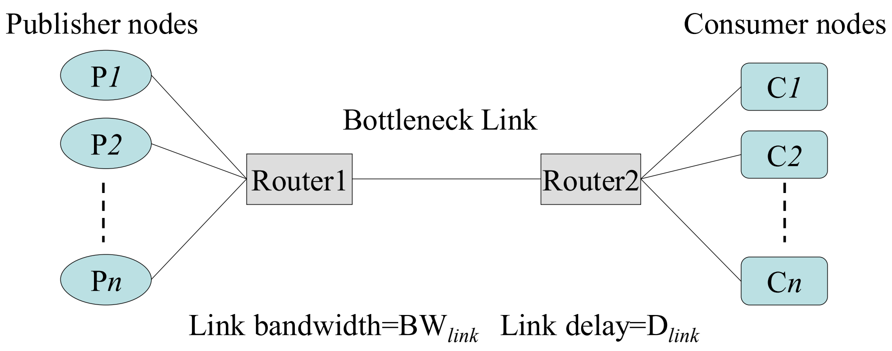
$$Interval = \frac{s}{x \cdot pps_t}$$

ppt_t	Number of Interest packet in one second
P	Constant period of estimation
s	Content chunk size [byte]
x	Pre-defined constant value

Panasonic

Simulation with ndnSIM (ns-3)

- Basic evaluation
  - on single bottleneck link
- Assumption
  - Each consumer node requests content with sequential numbering in the Content Name for each Interest packet
  - Each consumer node has determined the Content Name to fetch through other means
  - Each publisher node provides single video stream with variable bit-rate



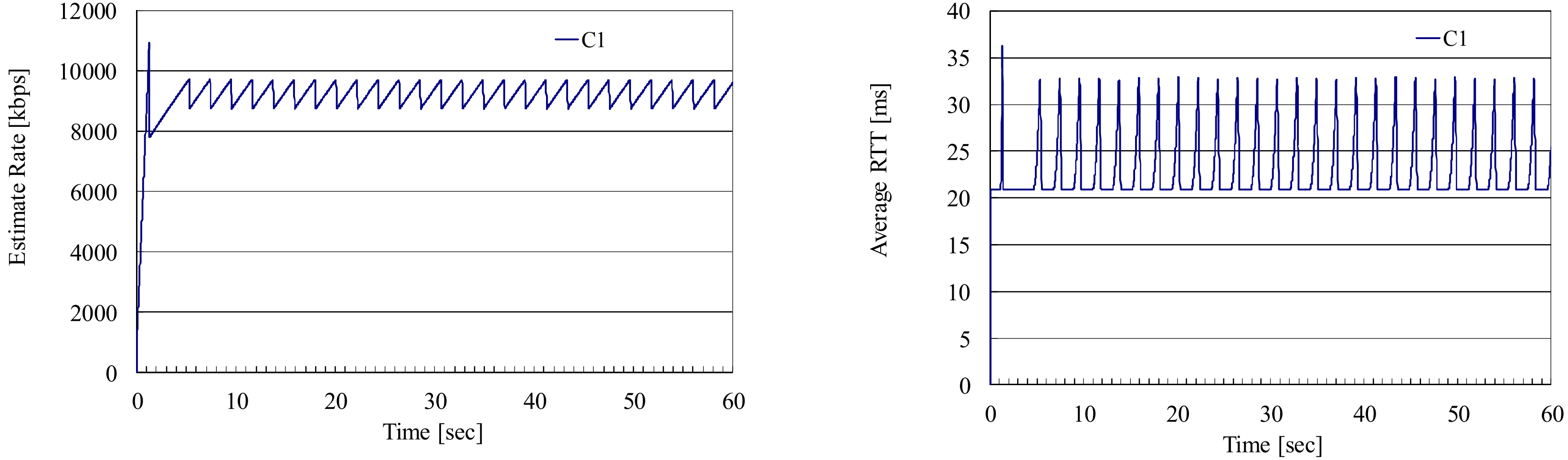
$BW_{Pn\text{-}R1}$	1Gbps
$D_{Pn\text{-}R1}$	1ms
$BW_{R2\text{-}Cn}$	1Gbps
$D_{R2\text{-}Cn}$	1ms
Queue	Droptail
Queue Size	50pkt

Panasonic

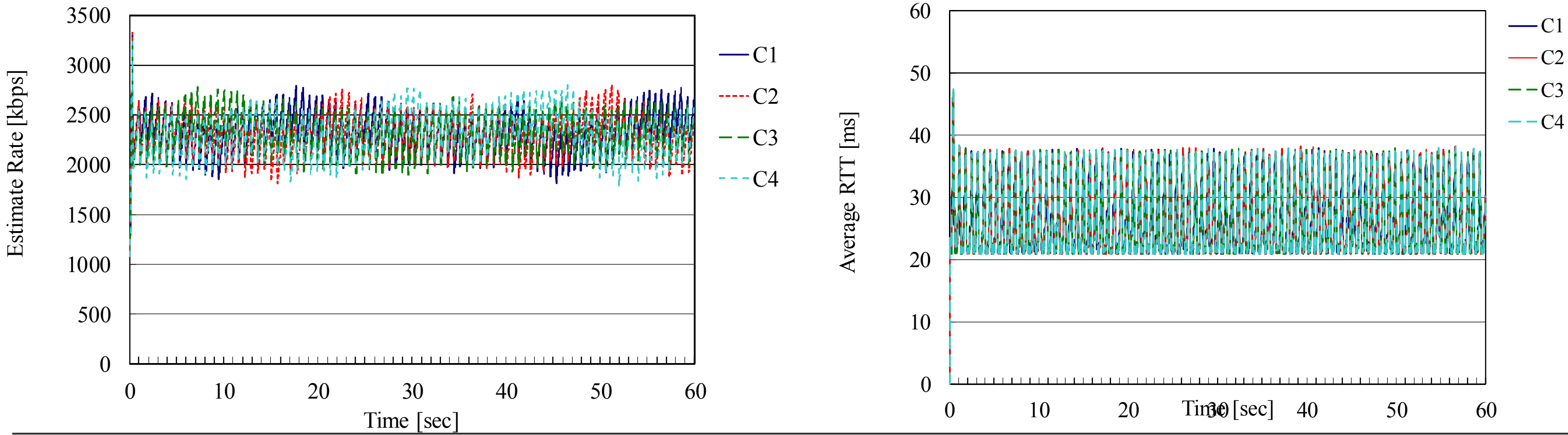
Basic simulation results (1-1)

Evaluation of bandwidth efficiency & transmission latency on the single bottleneck link

(n=1, $BW_{R1\text{-}R2}$=10Mbps, $D_{R1\text{-}R2}$=8ms)



(n=4, $BW_{R1\text{-}R2}$=10Mbps, $D_{R1\text{-}R2}$=8ms)



Comparison vs. AIMD

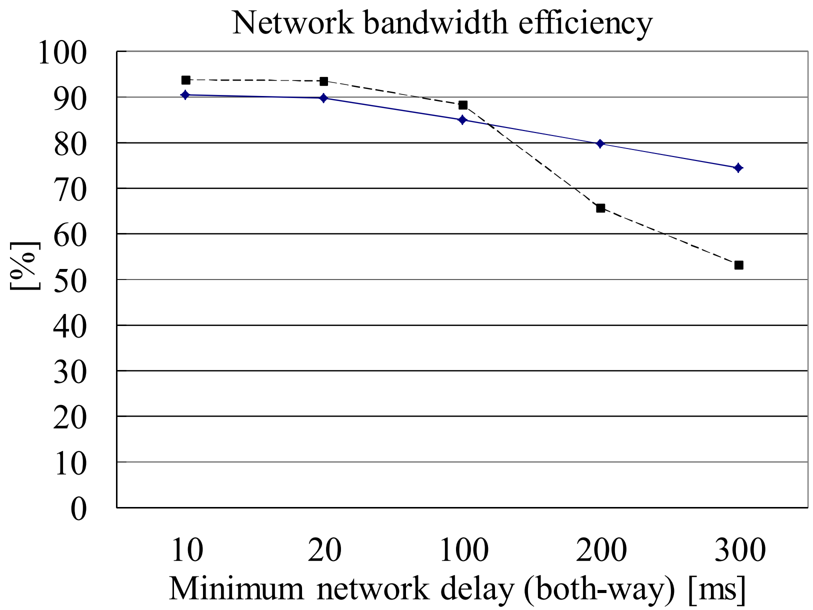
AIMD (Additive Increase/Multiplicative Decrease)

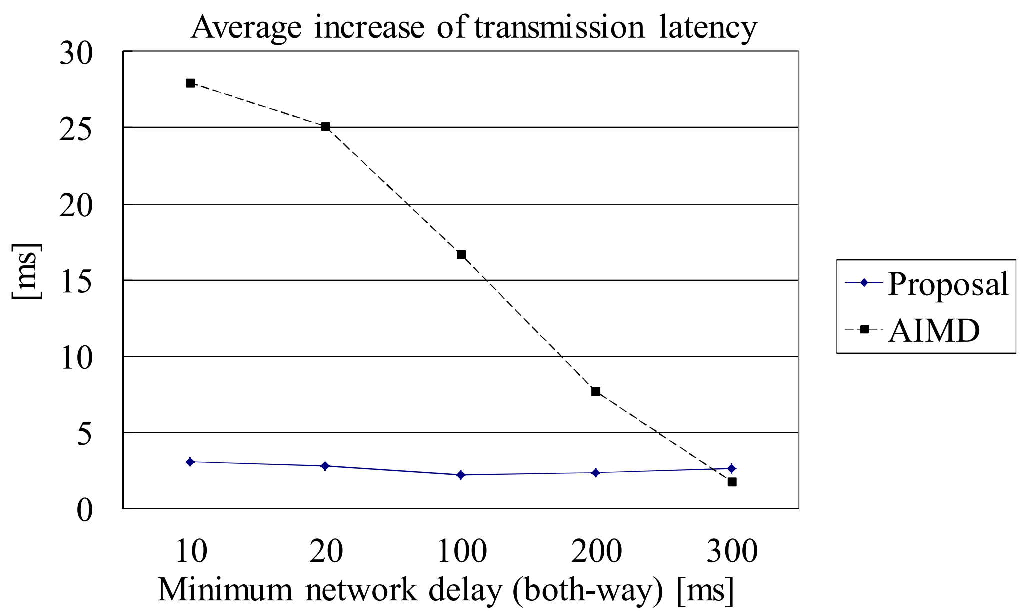
Decrease when packet loss, (duplicated ACK or time-out)

Proposed method:

- The throughput is more stable in various RTT
- Lower delay (especially in short RTT)

(n=1, $BW_{R1\text{-}R2}$=10Mbps, $D_{R1\text{-}R2}$=3-148ms)





Panasonic

Evaluation of RTT fairness

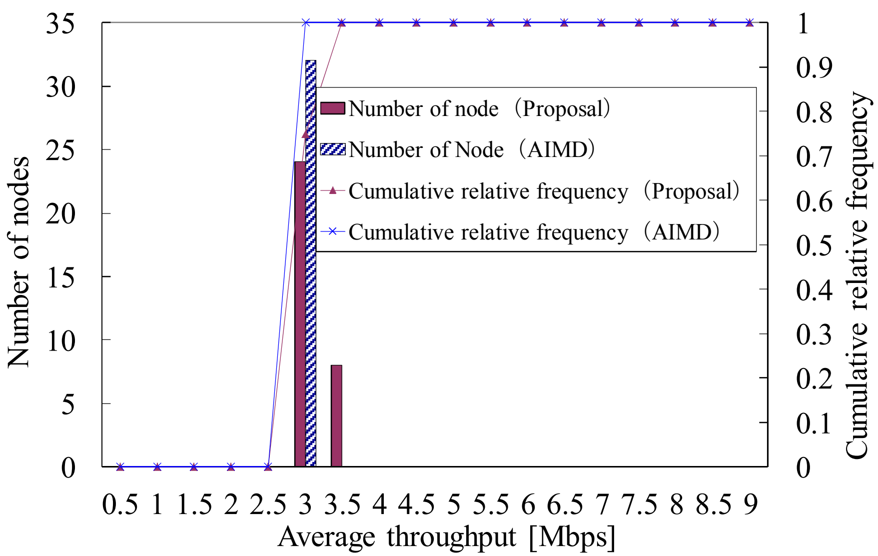
Proposed method

- each consumer gains almost same throughput

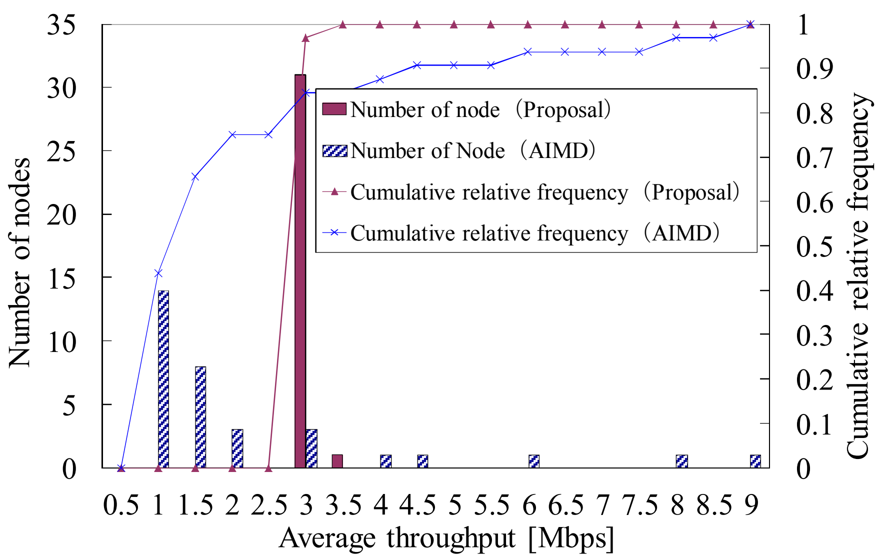
AIMD

- shorter RTT consumers gain more

$(n=32, BW_{R1\text{-}R2}=100\text{Mbps}, D_{R1\text{-}R2}=8\text{ms})$



$(Delay_{R2\text{-}Cn}=1\text{ ms})$

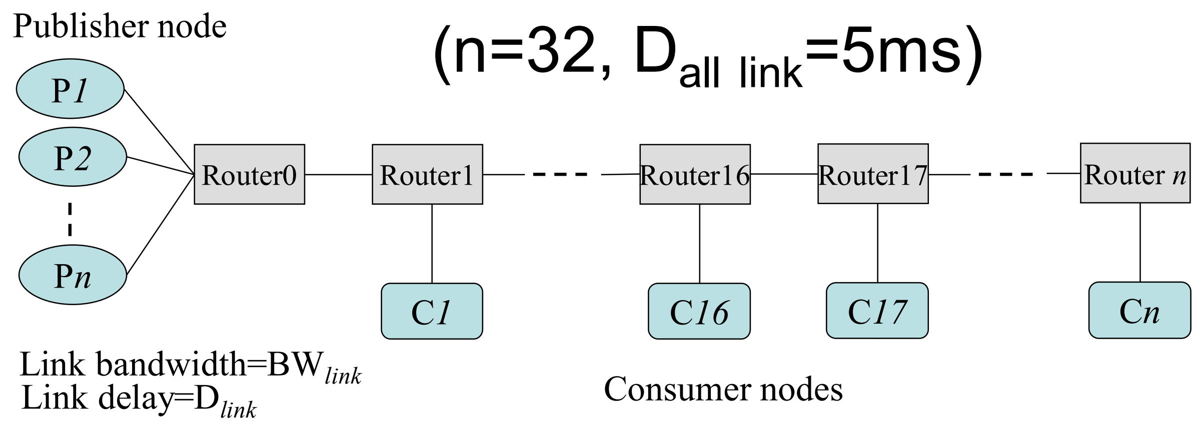


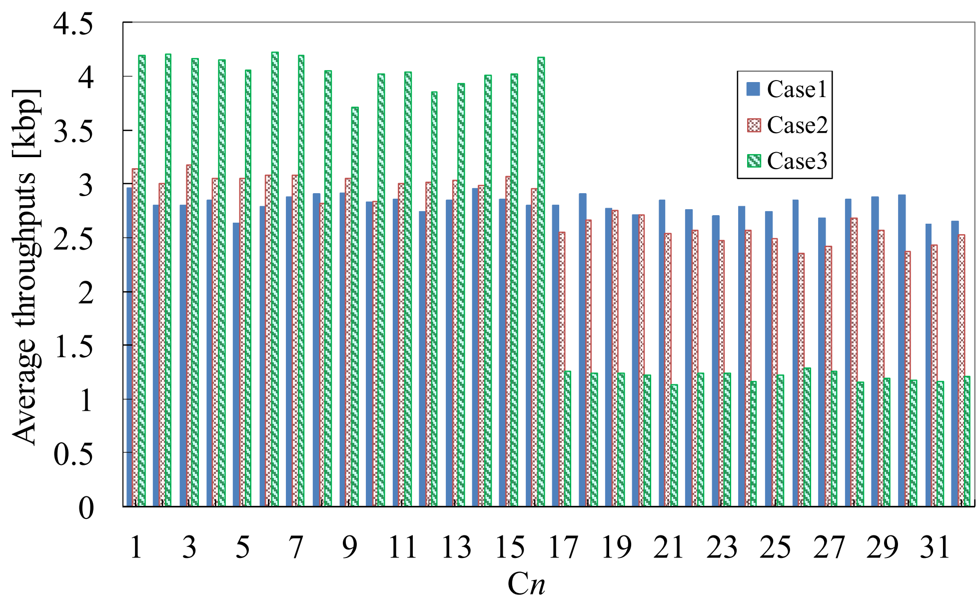
$(Delay_{R2\text{-}Cn}=5*n+1\text{ ms})$

Evaluation of RTT fairness on the multi-bottleneck link topology

Proposed method
- adapt to the narrowest bottleneck and fairly share the bandwidth

(n=32, $D_{all\ link}$=5ms)





Case1
$BW_{R0\text{-}R1}$=100Mbps

Case2
$BW_{R0\text{-}R1}$=100Mbps
$BW_{R16\text{-}R17}$=50Mbps

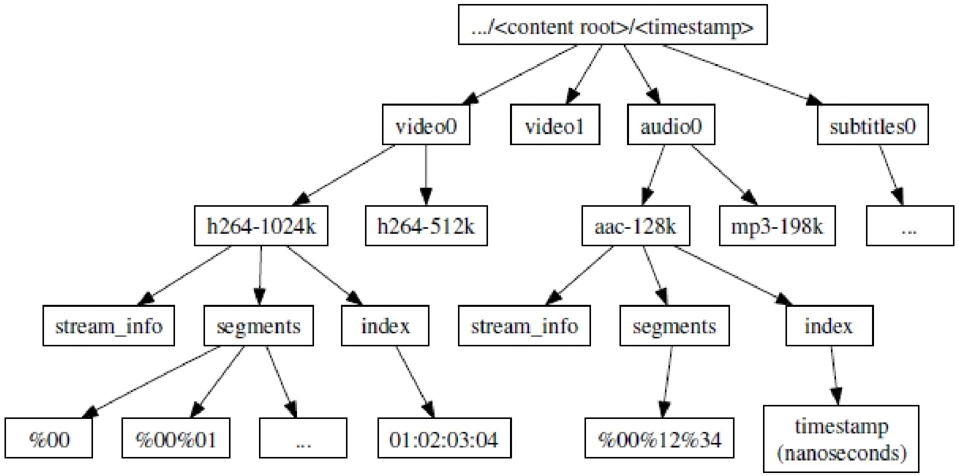
Case3
$BW_{R0\text{-}R1}$=100Mbps
$BW_{R16\text{-}R17}$=25Mbps

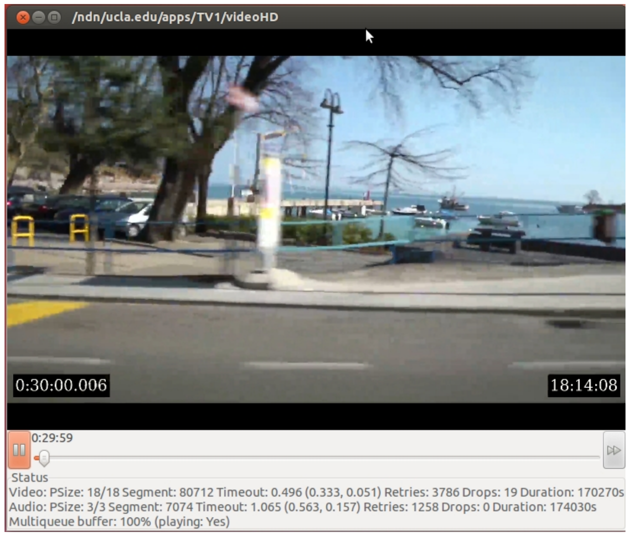
Panasonic

Feasibility for the implementation

On NDNvideo

- NDN-based live and pre-recorded video streaming (made by UCLA)
- random access to key frames using a time-code based namespace
- On-the-fly archival of live streams; identical playback approach for pre-recorded video

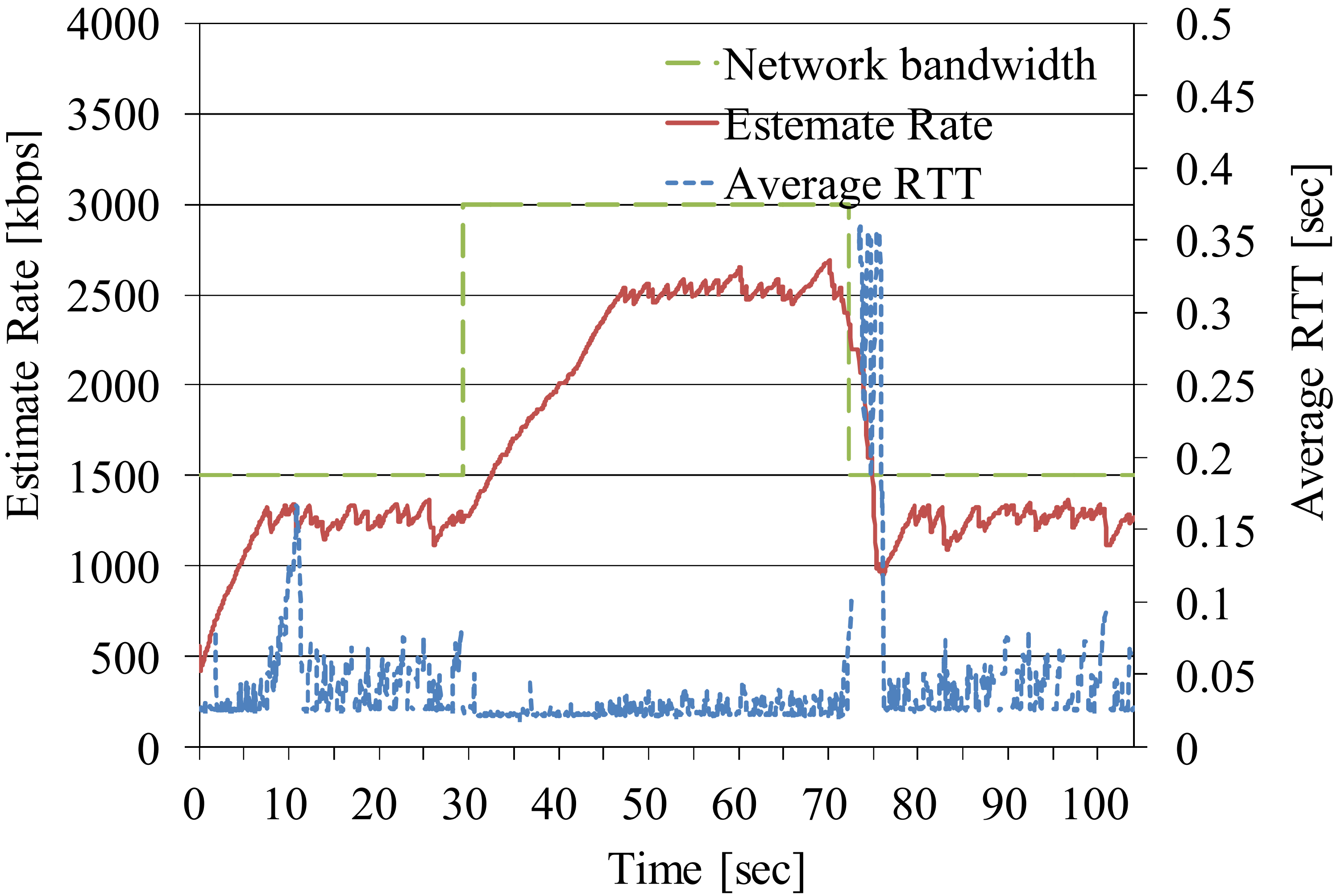




Panasonic

Implementation on NDNvideo (2)

Feasible to be implemented in the real-world application
(confirm the basic behavior of our implementation on NDNvideo)



Panasonic

Conclusion

- Focus on the Live real-time video streaming
- RTT fairness would be important
  - because it would be unexpectedly changed by source change in NDN/CCN
- Proposed method
  - Receiver driven (no router support)
  - Periodically (re-)compute PPS (not per RTT)
  - Use Short period average RTT (not EMWA)
- Simulation result
  - lower delay, more RTT-fair compared to AIMD
- Implemented on NDNvideo to show the feasibility
- Future work
  - Implementation on NDNRTC with UCLA
  - Supporting multi-source, multi-interface scenario