Predictive Video Scaling – Adapting Source Coding to Early Network Congestion Indicators

Fabian Jäger[1]   Thomas C. Schmidt[1]   Matthias Wählisch[2]

`{fabian.jaeger,schmidt}@informatik.haw-hamburg.de`
`waehlisch@ieee.org`

University of Applied Sciences Hamburg[1]

Freie Universität Berlin[2]

September 03, 2012

Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Agenda

Motivation

- Quality in multimedia streaming applications is often limited by the network performance
- Network conditions are hard to predict, particularly in mobile regimes and heterogenous networks
- Ability to scale the video stream for each participant individually with SVC (Scalable Video Coding)
- **How can we control the SVC to avoid network congestions?**

Problem Statement

- General problem
    - The codec should be adapted to the available bandwidth of the path
    - Common bandwidth estimation approaches (like PGM or PRM) are slow and produce overhead
- Approach
    - Prediction of a congestion instead of bandwidth measurement
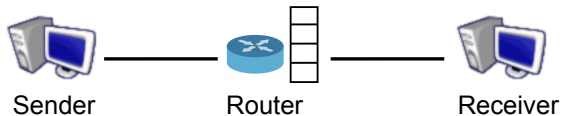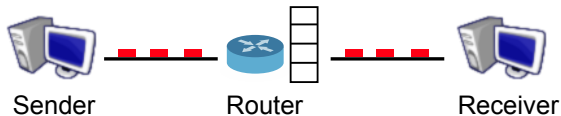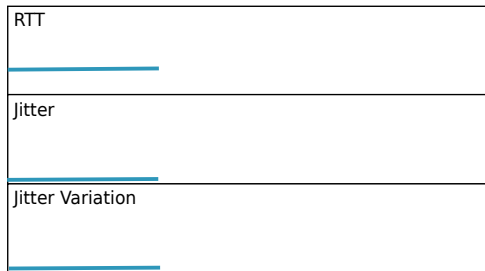   - The application is sender-sided and uses the congestion prediction to adapt the codec

Problem Statement

- General problem
  - The codec should be adapted to the available bandwidth of the path
  - Common bandwidth estimation approaches (like PGM or PRM) are slow and produce overhead
- Approach
  - Prediction of a congestion instead of bandwidth measurement
  - The application is sender-sided and uses the congestion prediction to adapt the codec
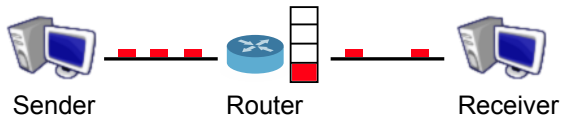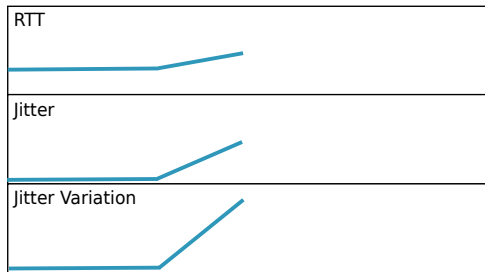
Network Congestion Indicators

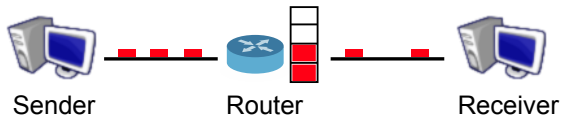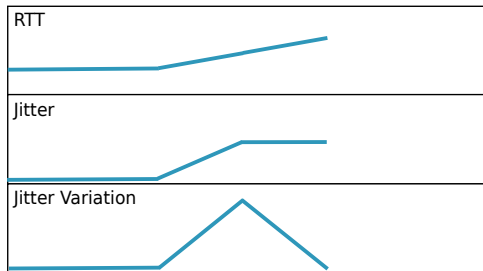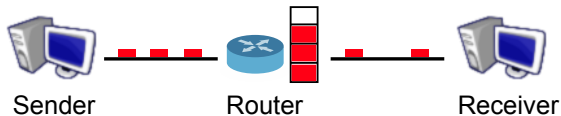| RTT |
| Jitter |
| Jitter Variation |

Sender   Router   Receiver
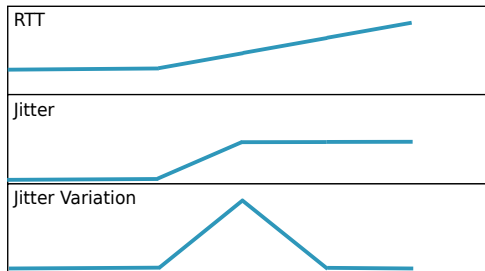
Network Congestion Indicators
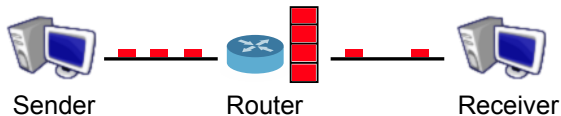
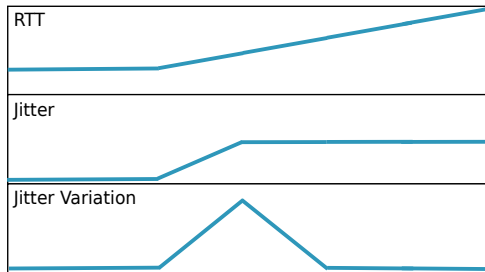# Network Congestion Indicators

Network Congestion Indicators

Network Congestion Indicators

Network Congestion Indicators

Video Codec Adaptation

- Collecting RTT and jitter samples requires a feedback-loop, which is commonly available through RTCP or HTTP/TCP.
- The temporal layer is used for rough scaling, while the quality (quantization factor) is used for fine scaling
- The quality increases if the jitter variation is stable and decreases if the jitter variation fluctuates

Decrease the Quality of the Video Stream

- The jitter variation is compared to the average jitter variation

```
IF  jitter_var > 5ms AND
    jitter_var > avg_jitter_var  THEN
        DecreaseQuality(jitter_var / avg_jitter_var)
END
IF  quality < 40% for 8 frames THEN
        RemoveTemporalLayer()
END
```
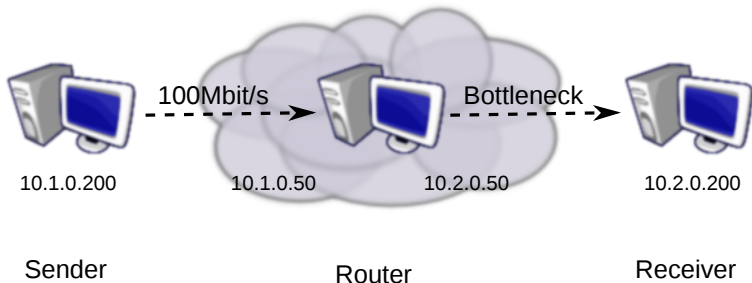
Increasing the Quality of the Video Stream

- More complicated, since we do not have a reliable indicator of an uncongested path

```
P = 8 frames
IF jitter_var <= avg_jitter_var for P frames THEN
      IncreaseQuality(current_quality)
      P = P - 1
ELSE
      P = 8 frames
END
IF quality > 60% for 8 frames THEN
      AddTemporalLayer()
END
```
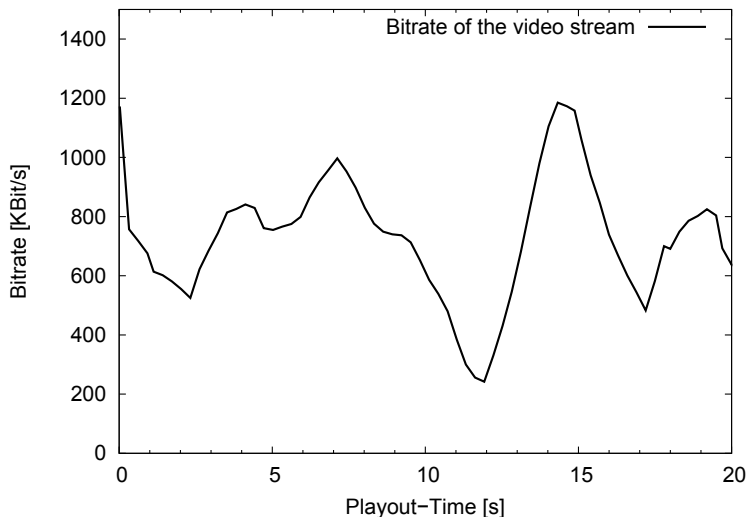
Testbed

- Ubuntu 12.04.1 LTS
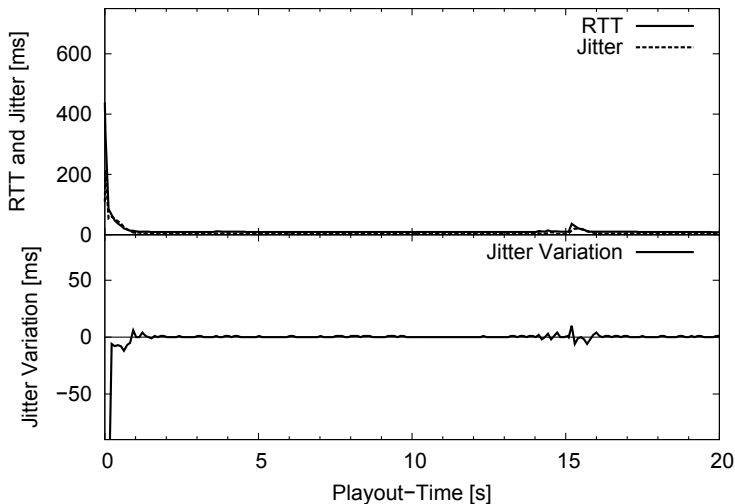- Wondershaper for traffic shaping

Bitrate Variation of the Test Video
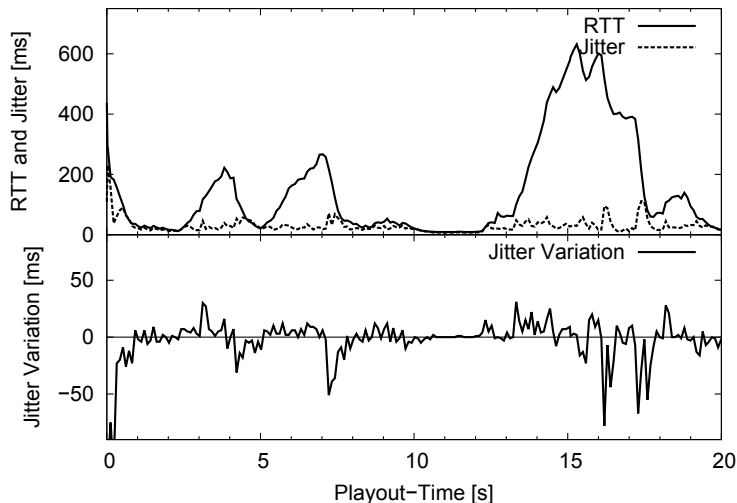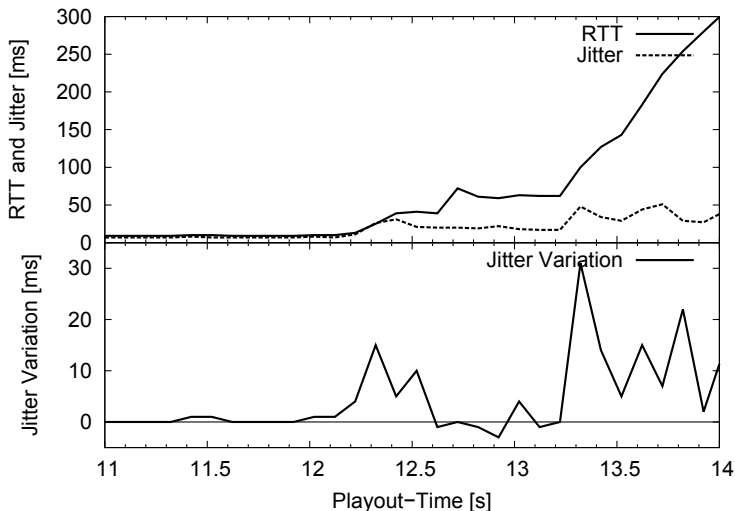## HHI Sequence TW 768x576

Stream on an uncongested Path
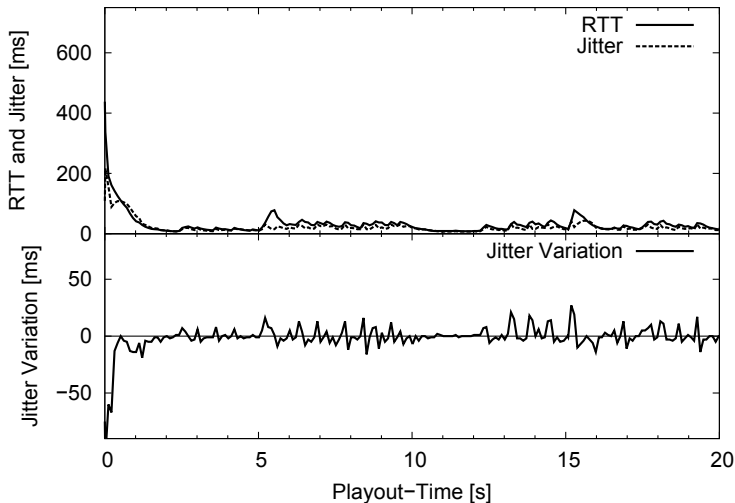
Unscaled SVC Stream

700 kbps available bandwidth

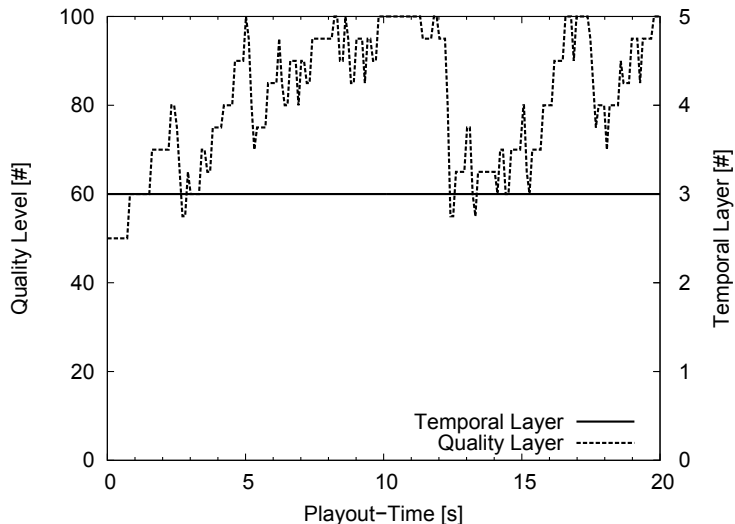# Unscaled SVC Stream

700 kbps available bandwidth - Detail

Scaled SVC Stream

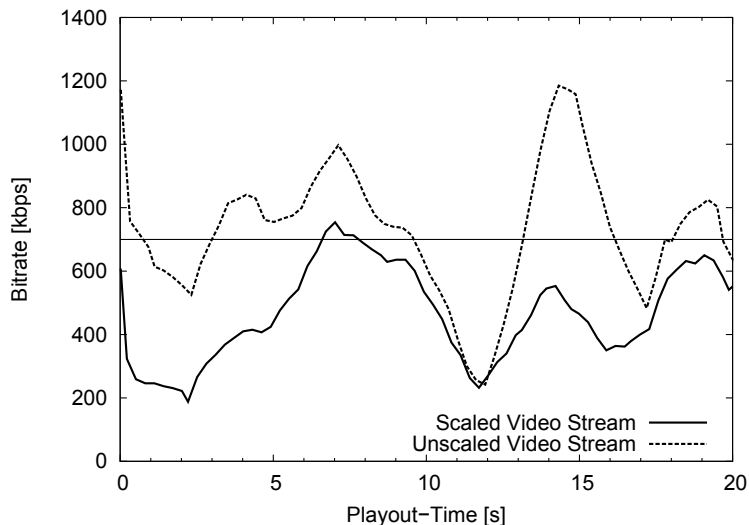**700 kbps available bandwidth**

Quantization and Temporal Layer
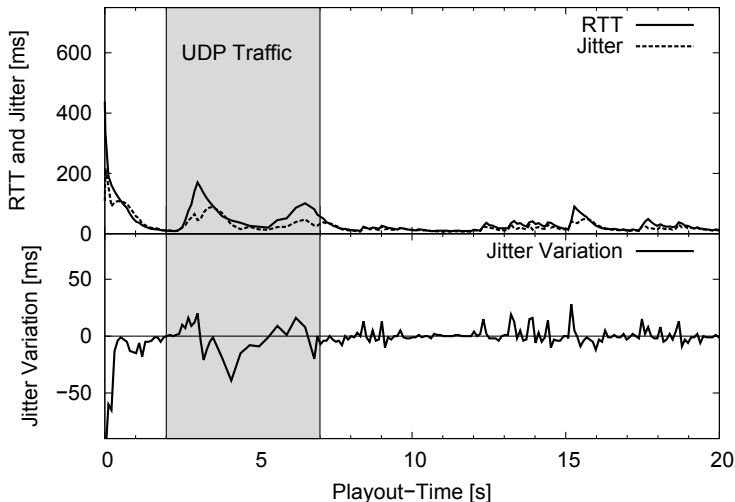
700 kbps available bandwidth

Bitrate of the Scaled and Unscaled SVC Stream

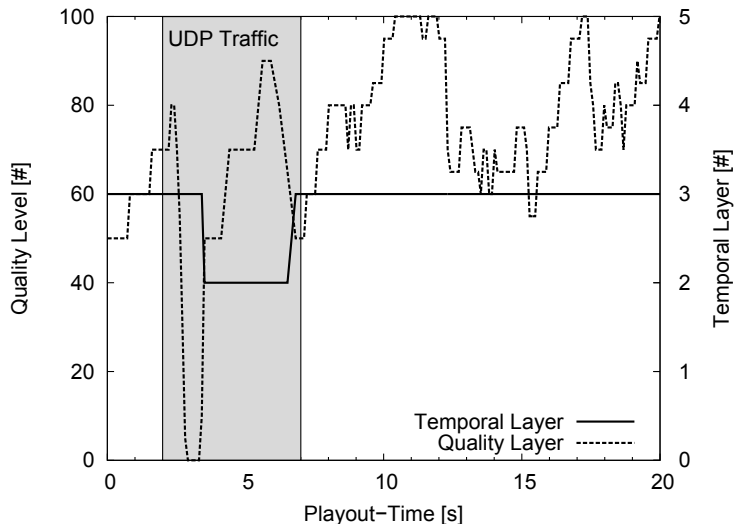**700 kbps available bandwidth**

Scaled SVC Stream

700 kbps available bandwidth and 100 kbps competing UDP traffic

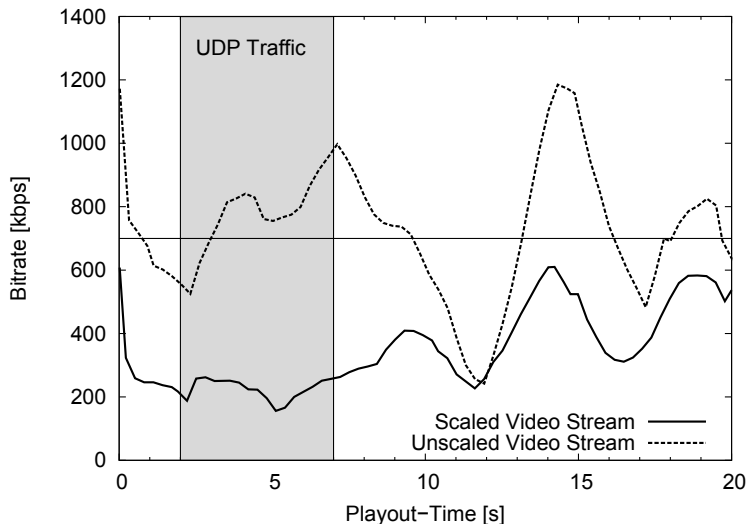# Quantization and Temporal Layer
700 kbps available bandwidth and 100 kbps competing UDP traffic

Bitrate of the Scaled and Unscaled SVC Stream

700 kbps available bandwidth and 100 kbps competing UDP traffic

Conclusion and Outlook

- **Proposal:** Sender-based, fast and lightweight video adaptation
- **Key idea:** The jitter variation is used as a fast indicator for changing network conditions

Future Work:

- Current measurement shows a conservative available bandwidth estimation, which often underestimates
- Usage of the spatial and quality enhancement layer and QoE optimization
- Testing the approach on a real network (PlanetLab)

Thank you for your attention

- Website of our research group:
  http://inet.cpt.haw-hamburg.de