University of Glasgow | School of Computing Science

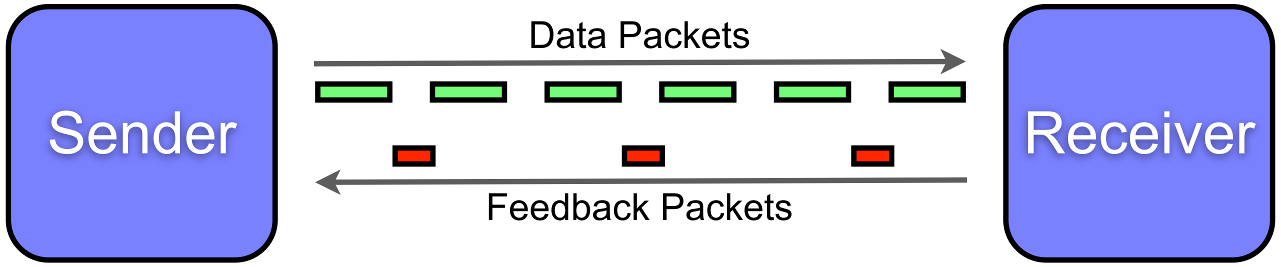
On the Use of RTCP Feedback for Unicast Multimedia Congestion Control

draft-perkins-rmcat-rtp-cc-feedback-00

Colin Perkins

Motivation

- Transport protocol provides a feedback loop



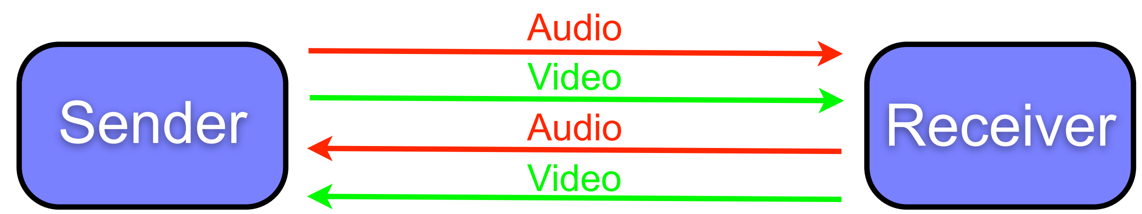
- Dynamics of congestion control depend on rate of feedback, and type of information returned
- RTCP provides a feedback channel for RTP-based applications – what sort of feedback can it provide?

Per-packet Feedback

- Per-packet feedback is ideal for congestion control
  - Effective ACK-clocking
  - Fast feedback on changes in RTT
- RTCP is not designed for per-packet feedback
  - RTCP reporting interval could be configured to match media data rate, but randomisation ensures control packets don't align with data packets
    - Reporting interval varies based on number of participants, number of active senders, average RTCP packet size, session bandwidth, and bandwidth fraction allocated to RTCP
    - Packet timing randomised ±50% to avoid synchronisation; reconsideration also impacts timing
  - RTCP packets are large, and sent as separate packets to RTP data – there is no mechanism to piggyback data and control packets
    - Media is not continually bi-directional in many scenarios – RTP header extensions don't work to piggyback feedback if there is no returning RTP flow
  - Potentially excessive overhead, depending on packet rate

Per-frame Feedback

- Consider simple WebRTC scenario:



  - One sender, one receiver, unicast video call
  - Four RTP flows (two audio and two video), all active essentially continually → four SSRCs in a single RTP session
  - Each SSRC sends RTCP reports for all other SSRCs
- RTCP reporting interval reduces to rtcp_interval = avg_rtcp_size * n / rtcp_bw
  - n = 4 SSRCs in the session
  - Configured rtcp_bw in octets per second
  - avg_rtcp_size = 156 octets

```
 UDP/IPv4 = 28
  RTCP SR =  8 header
            20 sender info
            72 report blocks (3 * 24)
RTCP SDES =  8 header + SSRC
            19 CNAME chunk (RFC 6222)
             1 padding
           ================
           156 octets total
```

- To report per frame, for 30fps video, want rtcp_interval = ~0.033 seconds
  - RTP/AVPF allows RTCP reporting intervals <5 seconds
  - rtcp_bw = avg_rtcp_size * n / rtcp_interval = 156 * 4 / 0.033 = 18,720 octets per second
  - If rtcp_bw configured as 5% of session bandwidth, then session bandwidth = 2.8Mbps (~1.4Mbps per video stream)
- If session bandwidth >= 2.8Mbps, all four SSRCs can report on every frame of video sent on average
  - Each report will convey RTT, packet sent, fraction lost, total lost, highest seqnum received, jitter – sufficient for congestion control?

Per-frame Feedback (cont’d)

- Assumptions: compound RTCP, cross-reporting
  - Ignores Jonathan Lennox’s optimisations to reduce RTCP cross-reporting
  - Ignores non-compound RTCP packets
  - Sending audio and video in separate RTP sessions, with different session bandwidth, would roughly half required session bandwidth for full reports
- If regular RTCP reports are not sufficient, can send additional RTCP packets in the compound packet
  - E.g., if an extra 20 octets feedback sent in each compound RTCP packet, required session bandwidth increases to 3.2Mbps for reporting per-frame at 30fps

Per-RTT Feedback

- Some congestion control protocols send feedback per RTT
  - RTT is usually longer than inter-frame interval
  - Arguments on previous slides apply, will give lower session bandwidth

Discussion

- RTCP can be suitable for congestion feedback
  - Not effective for per-packet feedback
  - Initial analysis: RTCP seems suitable for per-frame or per-RTT feedback in moderately high-quality sessions
    - Detailed analysis of other RTP topologies and scenarios needed
- If intended to work with RTP and RTCP as currently specified, congestion control should be designed to work with feedback per-video-frame or per-RTT, not per-packet