TFRC for Voice:
VoIP Variant and Faster Restart.

Sally Floyd, Eddie Kohler.
November 2004

http://www.icir.org/floyd/papers/
draft-ietf-dccp-tfrc-voip-00a.txt,
not yet submitted
Issues for VoIP traffic:

• **Small packets:**
  – Fairness in packets per second (pps) or in bytes per second (Bps)?

• **Measuring congestion:**
  – Packet loss rates or loss event rates?

• **Restart after idle:**
  – Faster Restart.

• **Next steps.**
VoIP: fairness in Bps.

- Standard TFRC has the goal of fairness in pps with TCP flows using the same packet size.

- The VoIP variant of TFRC has the goal of fairness in Bps with TCP flows using 1500-byte packets, (following RFC 3714).

- The VoIP variant assumes optimistically that the network limitation is in Bps, not in pps.
VoIP: fairness in Bps.

- In the TCP throughput equation, use the measured loss event rate and a packet size of 1460 bytes.

- Reduce the allowed transmit rate to account for the fraction of the VoIP bandwidth that would be used by 40-byte headers:
  - \[ X \leftarrow X \times \frac{\text{TruePktSize}}{\text{TruePktSize} + \text{Header}} \]
  - TruePktSize = average segment size in bytes
  - Header = 40 bytes

- Enforce a Min Interval between packets of 10 ms.
TFRC only
Standard TFRC (bottom) and TCP (top)
VoIP TFRC (bottom) and TCP (top)
Measuring Congestion:

- The VoIP variant of TFRC uses the loss event rate.
  - RFC 3714 uses the packet drop rate.

- These are both affected by packet size and by the smoothness of the sending rate.

- The effect of packet size on the packet drop rate could use more investigation.
Measuring Congestion:

• Packet size in a Drop-Tail world:
  – Queue measured in bytes, packets, or in-between?
  – Smooth or bursty sending rates?
  – High or low levels of statistical multiplexing?

• RED in packet mode:
  – Same packet drop rate for big and small packets.
  – TFRC measures the loss interval in packets.

• RED in byte mode:
  – Same byte drop rate for big and small packets.
The VoIP variant of TFRC:

• As it stands now, it sometimes favors the VoIP TFRC flow over the large-packet TCP flow.

• This needs to be quantified and evaluated.
Faster Restart after Idle:

• The motivation:
  – The sender knows more after an idle period that it does when just starting up.
  – So it should be able to be more aggressive than the default slow-start.
Faster Restart after Idle:

• CCID 3 says not to reduce the sending rate below 4 packets per RTT because of an idle period.
  – Change this to 8 packets, or at most 4KB.

• Quadruple instead of double the sending rate each RTT.
  – Up to the old sending rate.

• Allow this just for VoIP TFRC flows restricted to at most one packet per 10 ms?
  – Or allow this for any TFRC flow?
The next step:

- Allow TFRC flows to send more than twice the reported receive rate under other circumstances, if:
  - Allowed by the allowed sending rate, and
  - That rate has been successfully sustained in the past, and
  - There has been no congestion in the recent past.

- Justification: You know more than a blank slate, so you should be able to be more aggressive than slow-start.
The state of TFRC in NS:

• Includes the VoIP variant.

• Includes RFC 3390 initial sending rates.

• More updating is needed.
  – Add RFC 3390 sending rates after idle periods.
  – Add Faster Restart.
  – Add overhead for packet headers.