



Adaptive MPEG4 Video Streaming using Bandwidth Estimation

Mario Gerla, Alex Balk, Medy Sanadidi
{gerla, abalk, medy}@cs.ucla.edu

Dario Maggiorini
dario@dico.unimi.it



Outline

- Problem Statement
- Current Streaming Research Directions
- Unique Feature of VTP: Bandwidth estimation
- Testbed Evaluation



Problem Background

- Internet multimedia streaming on the rise
- Most real-time video is still UDP (no end-to-end congestion control)
- UDP approach could potentially lead to congestion collapse
- Active research on congestion-controlled streaming protocols



Controlled Stream Approaches

- RAP (Rate Adaptation Protocol)
Mimics TCP (i.e., AIMD); multilayer video adaptation
- SR-RTP (Selective Retransmission-RTP)
“Binomial” rate control; selectively retransmits only certain packets that carry “key” video data.
- SCTP (Stream Control Transmission Protocol)
Mimics TCP; multistream;
- TFRC (TCP-Friendly Rate Control)
Mimics TCP via equation;

Limitations:

AIMD algorithm leads to rate oscillations
poor link utilization in random loss environments



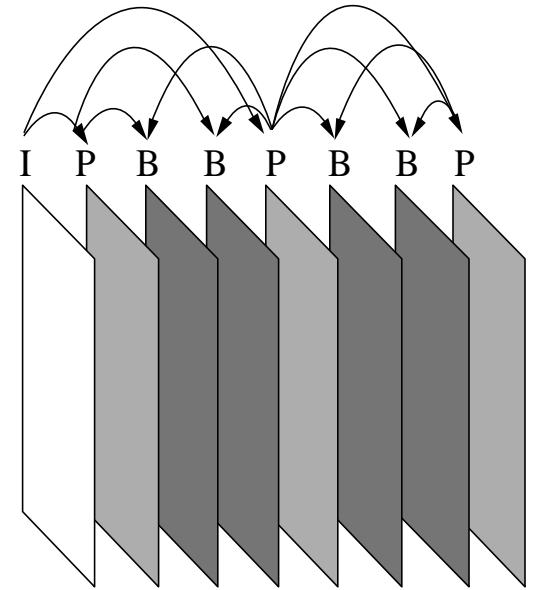
VTP: Video Transport Protocol

Key features:

- **bandwidth estimation** to adapt/reduce video stream
- MPEG specific adaptive **quantization levels** (while video frame rate is kept fixed to preserve perceived video quality).
- Inter-Protocol **fairness with TCP**.

Comments on MPEG-4

- **Intra-coded frames** (I-frames) are encoded independently, can be considered reference frames.
- **Predicted frames** (P-frames) depend on preceding I or P-frames, contain predicted motion data and error information.
- **Bi-directionally predicted frames** (B-frames) depend on both previous and next frames.
- VTP takes advantage of the wide **range of compression ratios** available in MPEG-4 to select appropriate video quality for streaming.





Bandwidth Estimation

- Receiver estimates available bandwidth: **unique feature** of VTP.
- Bandwidth estimation technique (inspired to TCP Westwood):

$$B_i = (\alpha)B_{i-1} + (1-\alpha)(b_i + b_{i-1})/2$$

B_i : bandwidth estimate

b_i : bandwidth sample (i.e., packet bits in packet/interpkt interval)

α : tunable coefficient

- Receiver sends back to source bandwidth estimates periodically (at least, each RTT)

Digression: TCP Westwood

- Congestion window control via **Achieved Rate Estimate** (RE)
- Sender estimates currently Achieved Rate RE by extracting/filtering *rate samples* from ACKs
- After packet loss (ie, 3 DUPACKs, or Timeout), RE estimate is used by sender to cut back **$cwnd$** as follows:

$$cwnd = RE \times RTT_{min}$$

Note: $cwnd = RE \times RTT_{min}$ is the min window required to achieve rate RE without causing congestion

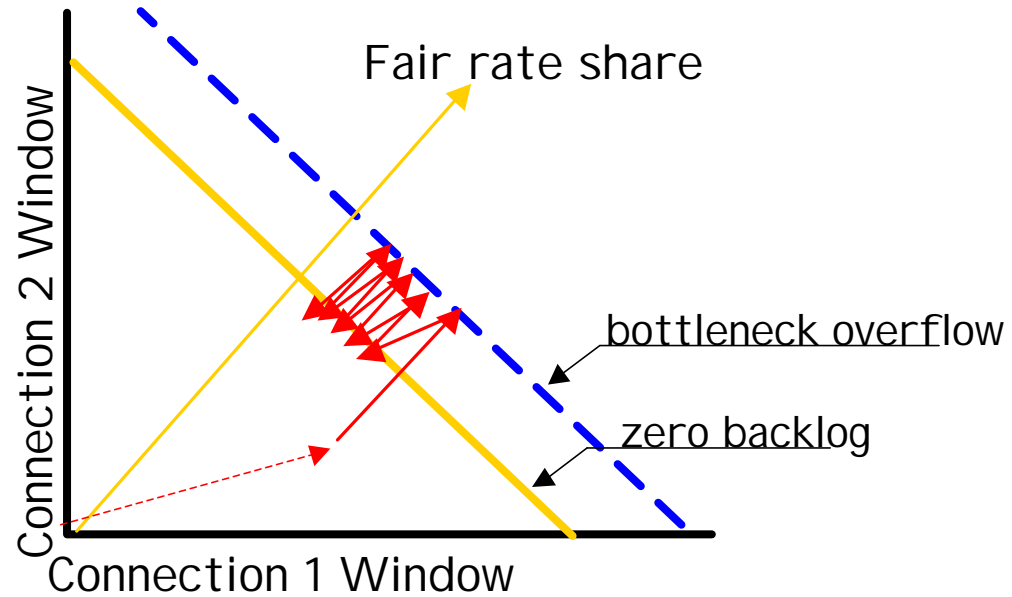
Additive Increase "Fair" Decrease (AIFD)

TCP Westwood: the control algorithm

- **When three duplicate ACKs are detected:**
 - set $cwin = RE * RTT_{min}$ (instead of $cwin = cwin / 2$ as in Reno) and $ssthresh = RE * RTT_{min}$ (instead of $ssthresh = cwin / 2$)
 - Go to **congestion avoidance**
- **When a TIMEOUT expires:**
 - set $ssthresh = RE * RTT_{min}$ (instead of $ssthresh = cwnd / 2$ as in Reno) and $cwin = 1$
 - Go to **slow start**

Note: RTT_{min} = min round trip delay experienced by the connection

TCP W converges to Fair equilibrium

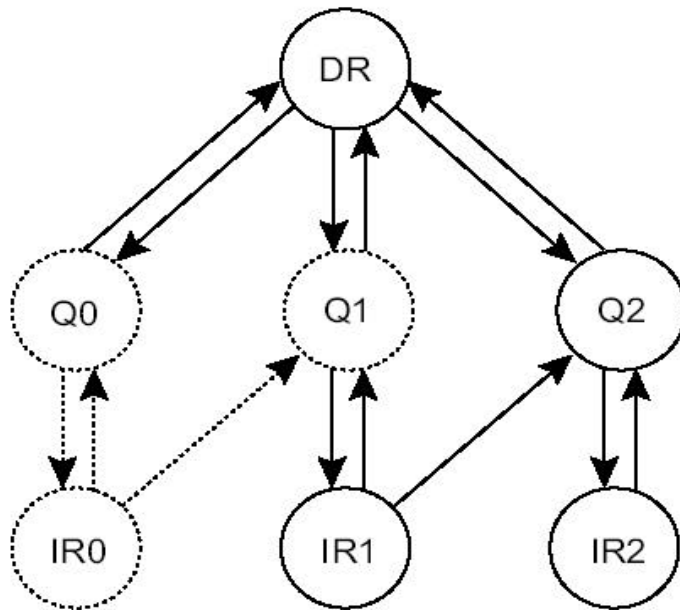




VTP algorithm

- **Multiple copies** of the video stream with different quantization levels are pre-computed and stored in the server (in the future, on line adaptive quantization will be explored)
- On the sender side: If estimated bandwidth feedback from receiver is **equal to (or larger than) sent rate**, gradually increase sending rate by one packet per RTT (probing phase)
- When bandwidth **estimate can support next quantization level**, switch to higher quality stream and higher bitrate.
- If bandwidth estimate falls below current sending rate, **switch to lower quantization level**.

Rate and code adjustment



DR = Decrease state
IR = Increase state

Q1, Q2, Q3: MPEG encoding states

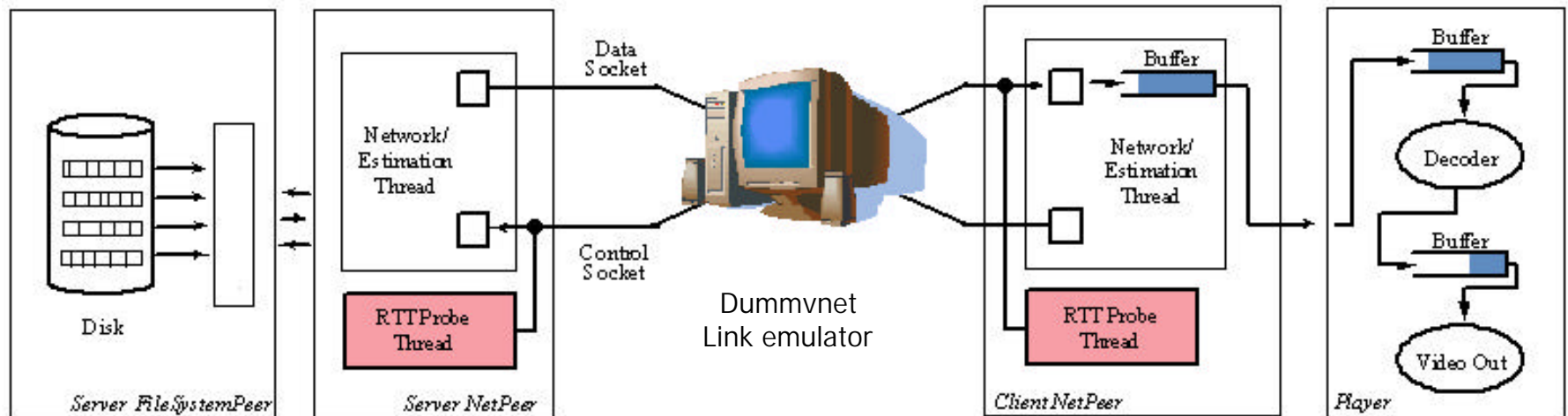
Example: suppose you are in Q1

If bandwidth estimate exceeds last value, move from Q1 to IR1.

Check if bandwidth is sufficient to support Q2. If not, increase rate and return to Q1. Else, move to Q2.

VTP: Linux testbed

- VTP was implemented and evaluated on the Linux operating system.





Software Architecture

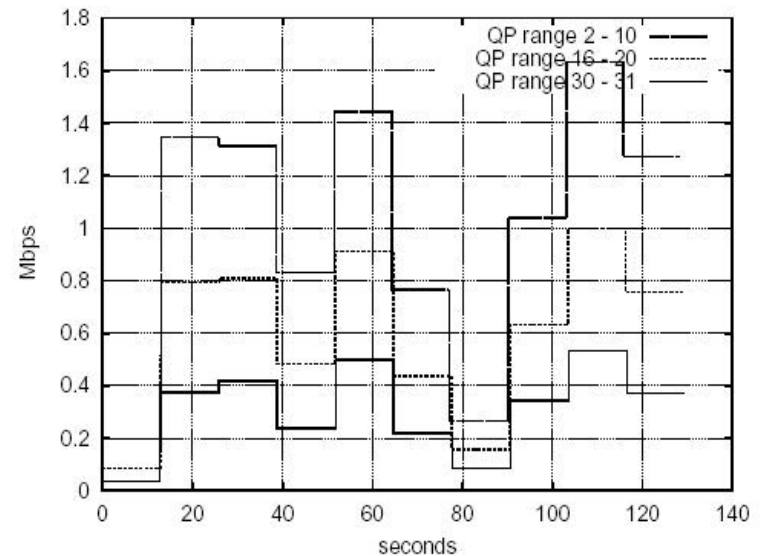
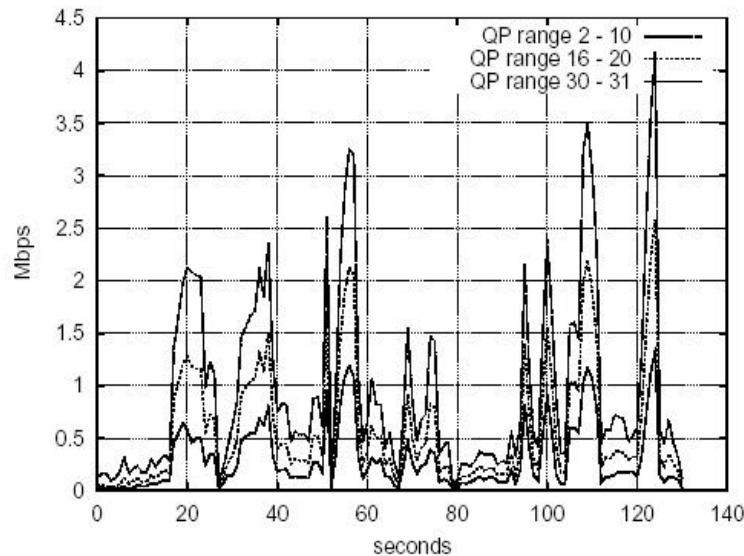
- VTP uses UDP to send both video packets and controls;
- Stream control and adaptation are done exclusively at application level;
- RTP and RTCP are not required, but can be integrated in the future (RTCP can be used for feedback)



Performance measures

- Fairness
- Stability
- Adaptive compression (QP)
- Robustness to random errors/loss

Compression and smoothing



Left diagram: bandwidth required by a segment of the movie "Tron" for different compression parameters (QP = 31, max compression)

Right diagram: prerecorded segments are stored for only 3 QP values. The traffic is smoothed to reduce peak rate (from 4.2 Mbps to 1.6 Mbps)

Dynamic rate & QP

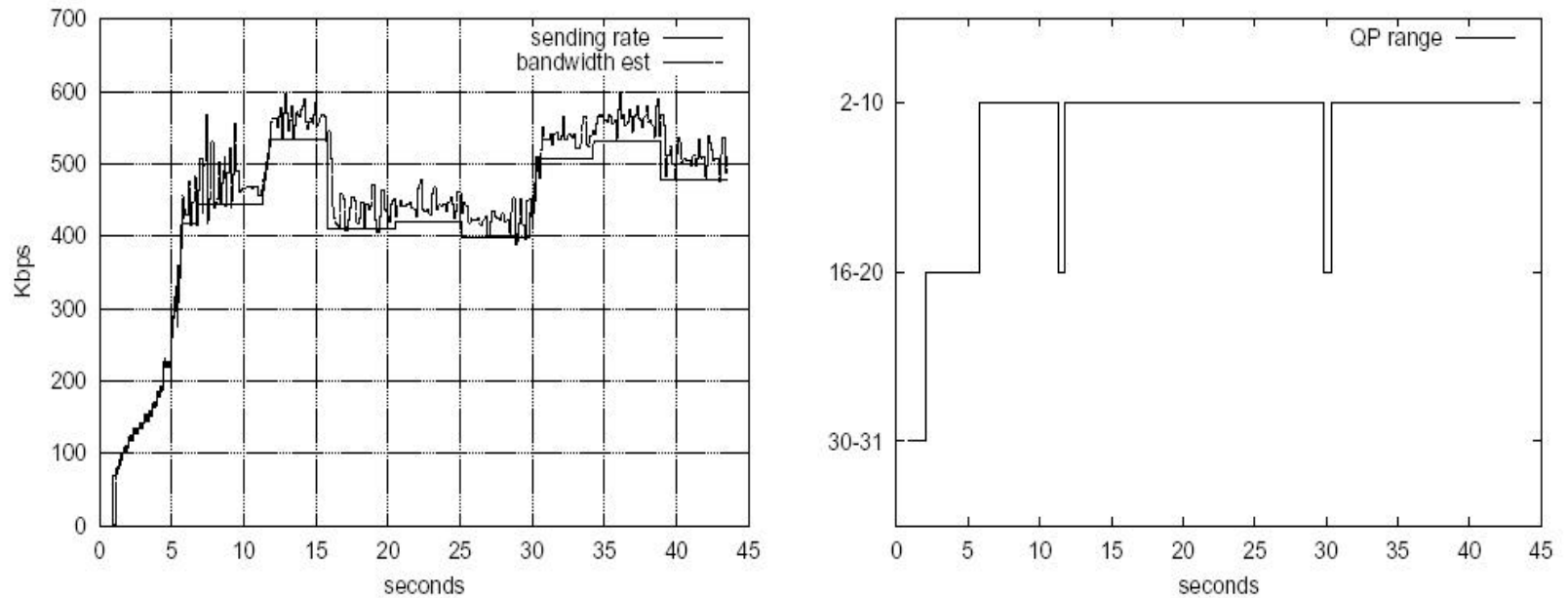
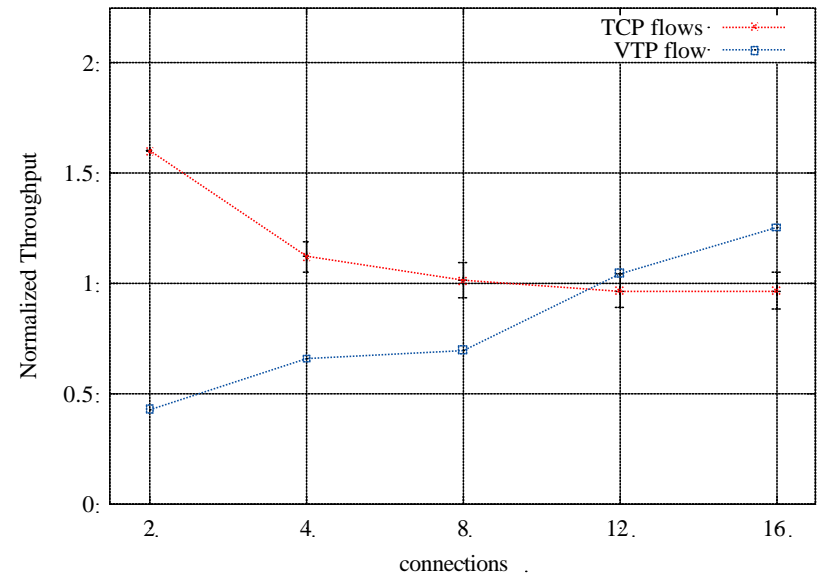
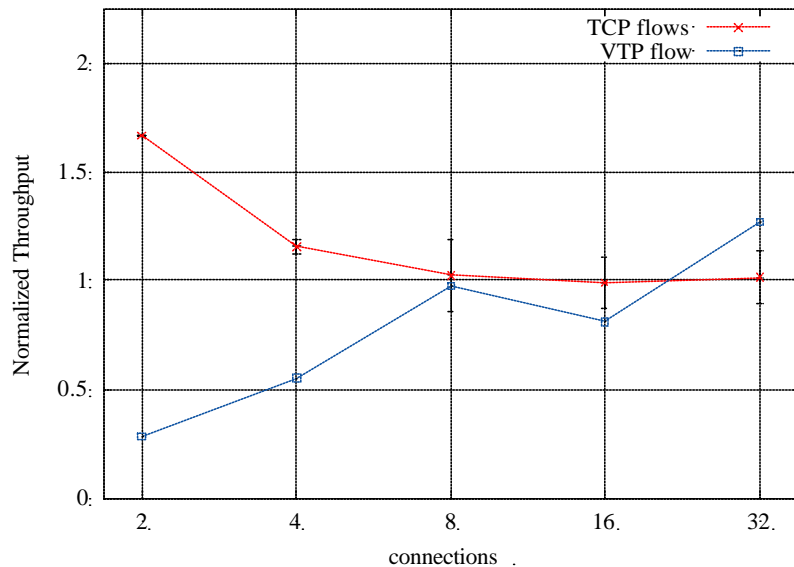


Figure 6: VTP isolated on a 10 Mbps, 10 millisecond RTT link.

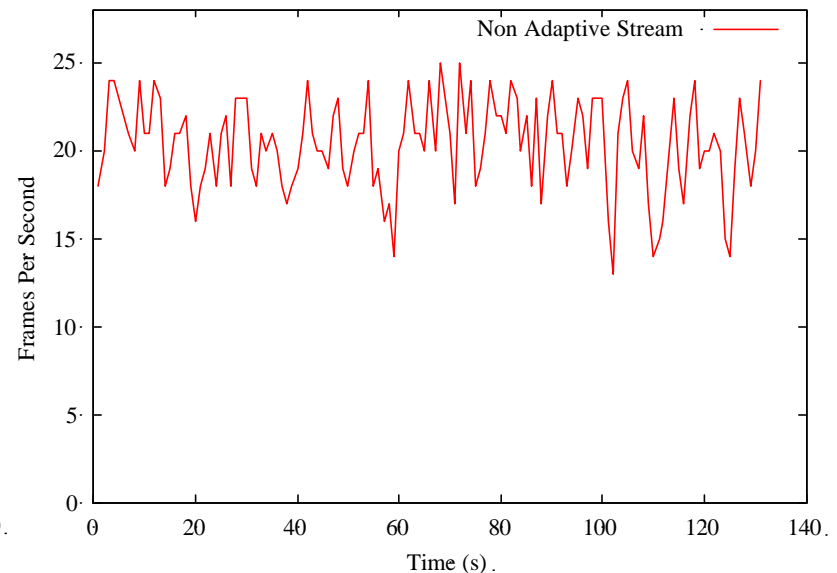
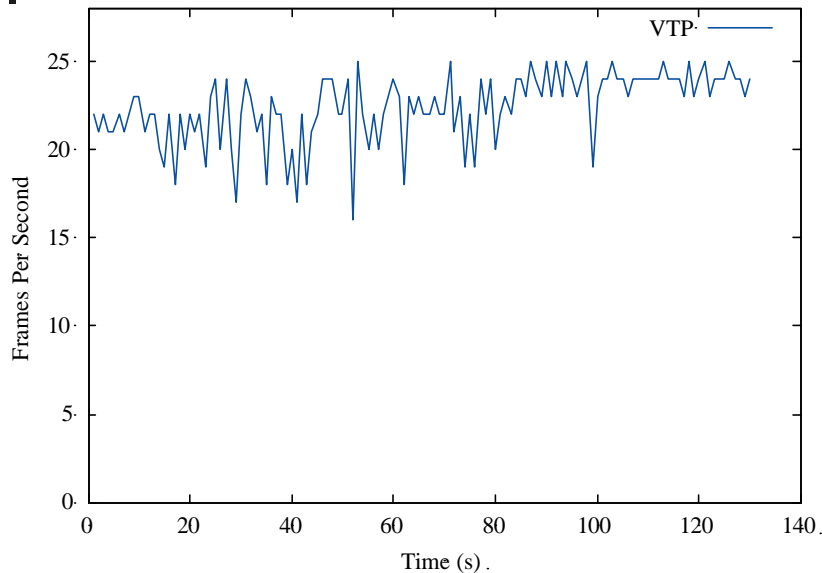
Fairness (1 VTP + N TCP Reno's)



- "Atlantis" segment encoded with the FFMPEG codec over a 3 Mbps, 10 ms RTT link.
- Normalized (wrt fair-share) throughput is shown; ideal fairness occurs when both protocols achieve "1" (note: single VTP requires $125\text{Kbps} < B < .45 \text{ Mbps}$)

- "TRON" segment encoded with the DivX codec over a 5 Mbps, 10 ms RTT link.
- Note, for Tron, bandwidth must be at least .5Mbps (at lowest quality) and at most 1.3 Mbps
- VTP uses its fair share of bandwidth or less in most cases.

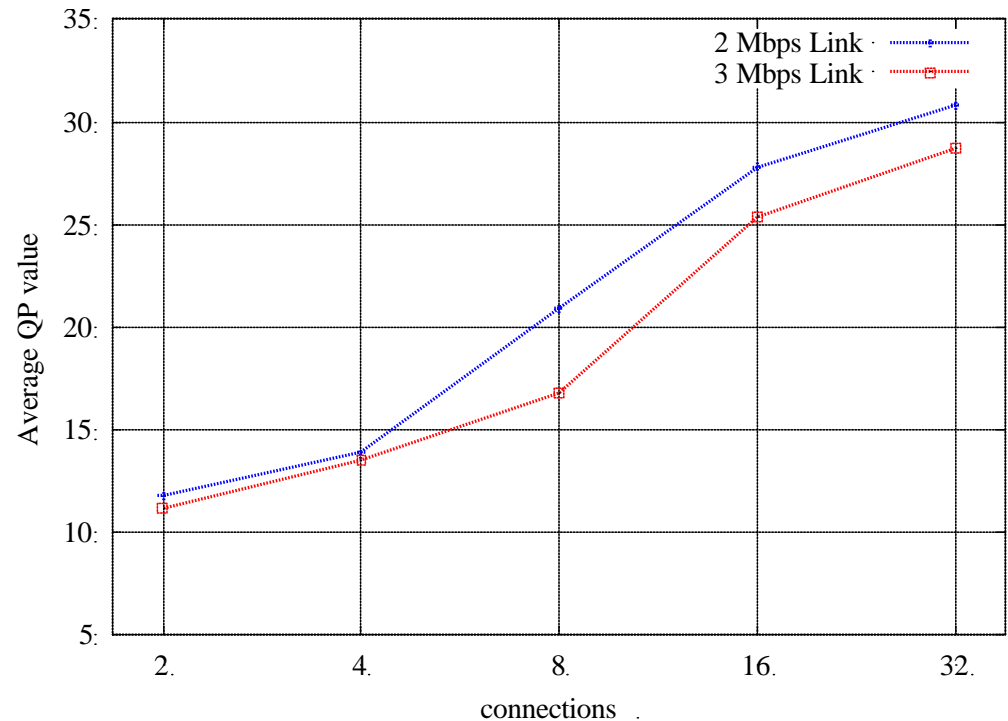
VTP Rate Stability (1 video + 11 TCP Reno's)



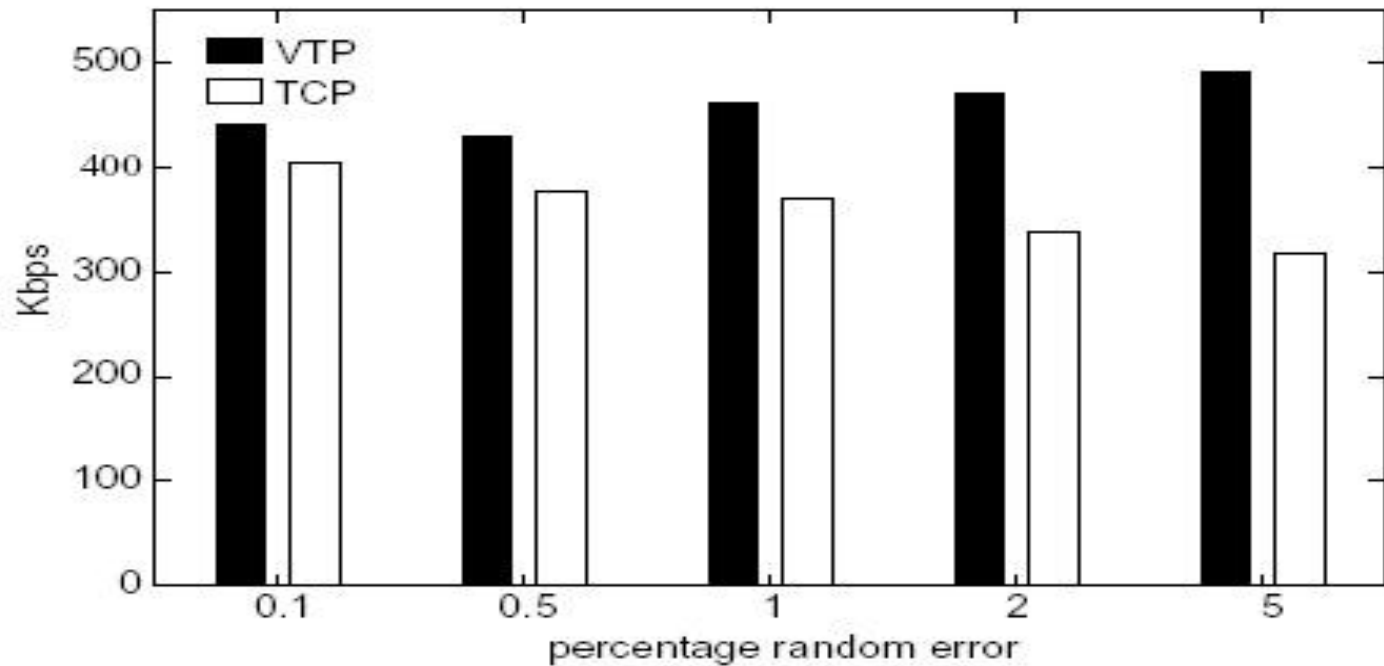
- Resulting frame rate of 1 monitored flow, either VTP or Non Adaptive Streaming, competing with 11 TCP connections.
- VTP frame rate stabilizes with time, as VTP discovers the available bandwidth. Non Adaptive Stream oscillates throughout the duration of the connection.

Adaptive Compression

- “QP”: quantization parameters (lower QPs imply more compression and lower video quality)
- As the number of connections increases, VTP transmits more compressed video streams using less bandwidth.

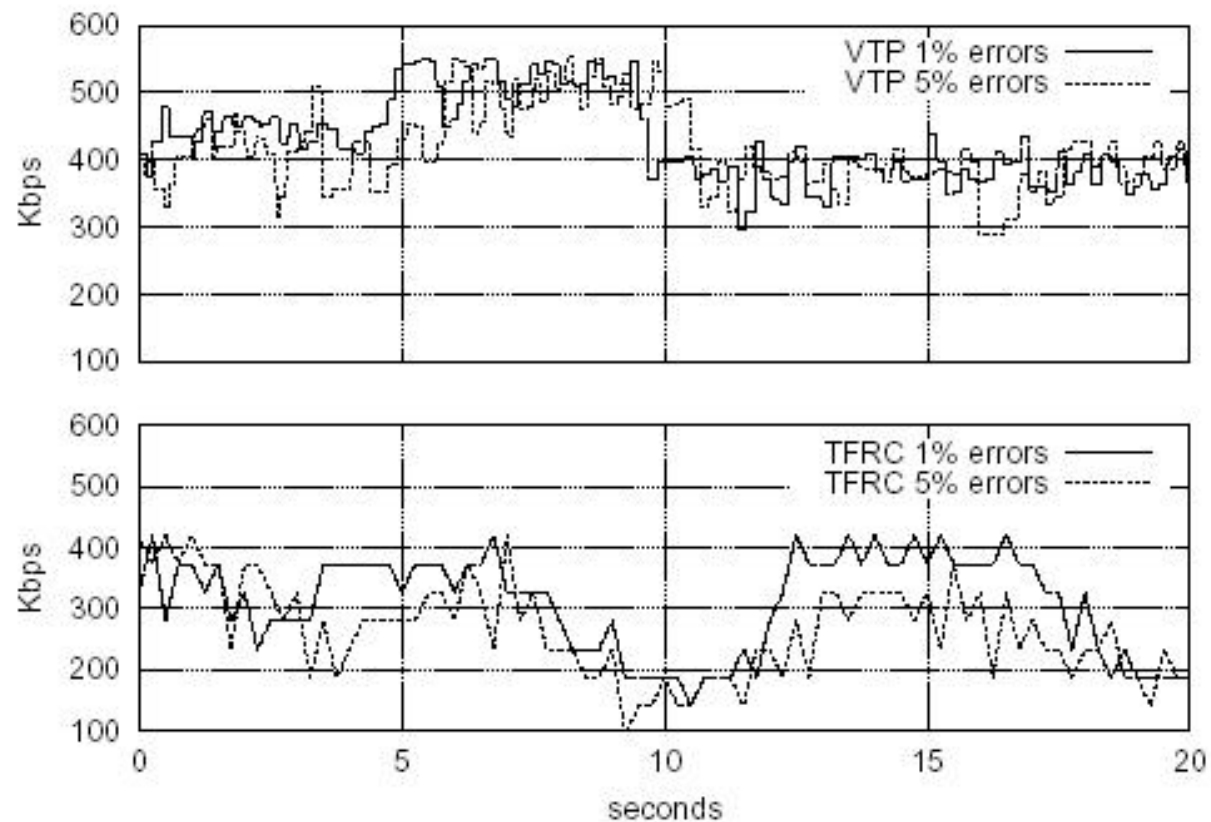


VTP & TCP Reno: random loss channel



VTP and TCP sharing a random loss channel

VTP vs TFRC in random loss



VTP vs TFRC behavior in random loss – same video trace for both



Conclusions

- VTP streams yield more stable frame rates than non-adaptive streaming.
- VTP fair to TCP
- VTP robust to random loss
- On going work:
 - Adaptive adjustment of receiver feedback reports
 - Investigation of other bandwidth estimation filters (e.g., AB probe, CAP-probe, etc)
 - Object oriented adaptive encoding