Error Resilient Internet Video Transmission

Ciro A. Noronha, Ph.D.
Director of Technology, Compression Systems
Cobalt Digital
Juliana W. Noronha
University of California, Davis
Motivation

• There are a number of protocols in use today to transport Video over IP.
• Since the “I” in IP stands for “Internet”, the Internet can (potentially) be used to transport Video over IP.
  ▪ Low-cost contribution links!!
• However, not all Video over IP protocols are suitable for transporting Video on the Internet because:
  ▪ The Internet drops packets
  ▪ Video over IP is compressed and needs every bit
  ▪ Video over IP cannot take packet drops
  ▪ The Video over IP protocol has to handle this issue
So, where are packets really lost?

Congestion!
What is an “acceptable” packet loss?

• Video compression works by removing redundancy from the content
  – Every bit of compressed video is very important
• There is a simple way to look at the effect of packet loss:
  – Assume that every packet that is dropped by the network causes a noticeable glitch in the video
    ▪ A block of packets dropped together causes one glitch
  – Decide how many glitches per (day/hour/minute) is acceptable to you
Some numbers

Assume a 4 Mb/s stream, with 1316-byte packets

<table>
<thead>
<tr>
<th>Dropping one packet in</th>
<th>Produces a glitch every</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,000</td>
<td>2.6 seconds</td>
</tr>
<tr>
<td>10,000</td>
<td>26 seconds</td>
</tr>
<tr>
<td>100,000</td>
<td>4 minutes 23 seconds</td>
</tr>
<tr>
<td>1,000,000</td>
<td>44 minutes</td>
</tr>
<tr>
<td>10,000,000</td>
<td>7 hours 19 minutes</td>
</tr>
</tbody>
</table>

In order to achieve reliable operation on the Internet, a network protocol is needed to “recover” in some way the packets that have been lost.
Protocols Considered

• SMPTE-2022 FEC
  – Transmit redundant information with the packets
  – Losses may be recovered from received packets and redundant information

• Retransmission (ARQ)
  – If a packet is lost, receiver will request a retransmission
RTP plus SMPTE-2022 FEC

- Basic idea:
  - Transmit the video using RTP
    - That gets you timestamps and sequence numbers
    - Sequence numbers let you know when packets were dropped
  - Transmit “extra” FEC packets
  - If packets are lost in the network, it may be possible to rebuild them from the received packets and FEC packets:
    - For each N packets send 1 FEC packets
    - If there is one loss in this set of N+1 packets, it can be corrected
  - Use a matrix arrangement to deal with burst losses
Some FEC Numbers

<table>
<thead>
<tr>
<th>Columns</th>
<th>Rows</th>
<th>Recovery Capability</th>
<th>Overhead</th>
<th>Latency @ 2 Mb/s</th>
<th>Latency @ 10 Mb/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>5</td>
<td>5 pkts every 25</td>
<td>20%</td>
<td>263 ms</td>
<td>53 ms</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>10 pkts every 50</td>
<td>20%</td>
<td>526 ms</td>
<td>105 ms</td>
</tr>
<tr>
<td>20</td>
<td>5</td>
<td>20 pkts every 100</td>
<td>20%</td>
<td>1052 ms</td>
<td>211 ms</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10 pkts every 100</td>
<td>10%</td>
<td>1052 ms</td>
<td>211 ms</td>
</tr>
</tbody>
</table>
ARQ

• ARQ stands for:
  – Automatic Repeat reQuest
  – Automatic Repeat Query

• This is the generic name for a number of retransmission strategies in the face of packet loss
  – Standard TCP uses a couple of ARQ variants

• In video transmission, the most useful variant is “Selective Retransmission” (NACK-based)
  – If you don’t hear from me, everything is OK
  – If I miss anything, I let you know and you resend just that
Transmitted packets are saved for possible retransmission.

Packet Lost, Resend (NACK)

If the buffers are big enough, multiple retransmissions of the same packet can be supported.
Comparison of FEC and ARQ

• FEC and ARQ have “decent” latency (typically 1 second or less)
  – May be acceptable for some forms of live contribution
• How do these two protocols compare?
  – Statistical models
  – Testing on a simulated network
  – Measurement data
A little probability and statistics...

• Assume independent loss probability for each transmitted packet (binomial distribution)
• Calculate the rate of packets still lost after correction with statistical analysis
• This allows us to theoretically compare the performance of the various protocols and settings
• Our variables are:

  \[ R = \text{number of requests (ARQ)} \]
  \[ N = \text{number of packets per row (FEC)} \]
  \[ M = \text{number of packets per column (FEC)} \]
Network Simulator

- Windows-based network simulator custom-built for this test
- Random drops, random burst drop size
- Test scenario:
  - End-to-end real-time video
  - Select max burst loss
  - Increase loss percentage until video is “not watchable” (subjective)
Simulator Results

Measured Packet Loss Upper Bound

- ARQ: max 4 retries
- FEC: 20x5 Row and Column
Field Test Data

- **Locations:**
  - Santa Clara, CA
  - Champaign, IL
- **ISP:** Comcast
- **Network Round Trip Time:** 75 ms
- **Number of hops:** 12
- **Target bit rate:** 3 Mb/s
- **Equipment:**
  - 9223 Encoder
  - 9990-DEC Decoder
RTP/SMPTE-2022 Test Data

Parameters: 20x5 matrix, row and column

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Duration</td>
<td>65 hours</td>
</tr>
<tr>
<td>Test Start Date</td>
<td>05/19/17, 3:50PM</td>
</tr>
<tr>
<td>Network Packet Loss</td>
<td>0.0158%</td>
</tr>
<tr>
<td>Corrected Packet Loss</td>
<td>0.0027%</td>
</tr>
<tr>
<td>Correction Ratio</td>
<td>83%</td>
</tr>
<tr>
<td>Bandwidth Overhead</td>
<td>25%</td>
</tr>
<tr>
<td>Network Glitch Interval</td>
<td>1 minute 13 seconds</td>
</tr>
<tr>
<td>Corrected Glitch Interval</td>
<td>7 minutes 12 seconds</td>
</tr>
<tr>
<td>Protocol Latency</td>
<td>702 ms</td>
</tr>
</tbody>
</table>

### RTP/SMPT-2022 Test Data

- **Network Packet Loss**: 0.0158%
- **Corrected Packet Loss**: 0.0027%
- **Correction Ratio**: 83%
- **Bandwidth Overhead**: 25%
- **Network Glitch Interval**: 1 minute 13 seconds
- **Corrected Glitch Interval**: 7 minutes 12 seconds
- **Protocol Latency**: 702 ms
### RTP/ARQ Test Data

Parameters: up to 4 retries allowed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Duration</td>
<td>169 hours</td>
</tr>
<tr>
<td>Test Start Date</td>
<td>05/24/17, 12:30PM</td>
</tr>
<tr>
<td>Network Packet Loss</td>
<td>0.0257%</td>
</tr>
<tr>
<td>Corrected Packet Loss</td>
<td>0.000078%</td>
</tr>
<tr>
<td>Correction Ratio</td>
<td>99.7%</td>
</tr>
<tr>
<td>Bandwidth Overhead</td>
<td>0.027%</td>
</tr>
<tr>
<td>Network Glitch Interval</td>
<td>46 seconds</td>
</tr>
<tr>
<td>Corrected Glitch Interval</td>
<td>4 hours 7 minutes</td>
</tr>
<tr>
<td>Protocol Latency</td>
<td>400 ms</td>
</tr>
</tbody>
</table>
FEC/ARQ Comparison

Scaling:
- Latency
  - ARQ latency is constant
  - FEC latency decreases with increasing bit rate
- Overhead
  - ARQ overhead will increase with packet loss
  - FEC overhead is constant

<table>
<thead>
<tr>
<th>Parameter</th>
<th>2022 FEC</th>
<th>ARQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Packet Loss</td>
<td>0.0158%</td>
<td>0.0257%</td>
</tr>
<tr>
<td>Corrected Packet Loss</td>
<td>0.0027%</td>
<td>0.00078%</td>
</tr>
<tr>
<td>Correction Ratio</td>
<td>83%</td>
<td>99.7%</td>
</tr>
<tr>
<td>Bandwidth Overhead</td>
<td>25%</td>
<td>0.027%</td>
</tr>
<tr>
<td>Network Glitch Interval</td>
<td>1 minute 13 seconds</td>
<td>46 seconds</td>
</tr>
<tr>
<td>Corrected Glitch Interval</td>
<td>7 minutes 12 seconds</td>
<td>4 hours 7 minutes</td>
</tr>
<tr>
<td>Protocol Latency</td>
<td>702 ms</td>
<td>400 ms</td>
</tr>
</tbody>
</table>
ARQ Standardization Status

• The Video Services Forum (VSF) started a group around NAB 2017 to standardize a low-latency video transport protocol over the Internet
• **RIST**: Reliable Internet Stream Transport
• ARQ has been selected as the base protocol
• VSF TR-06-1 was published October 2018
Thank You

Ciro A. Noronha, Ph.D.
Cobalt Digital
ciro.noronha@cobaltdigital.com
+1 650 208-0605