Multimedia networks

KR: Kurose and Ross chapter 7
(KR3: 3rd ed)

Jargon
- Codec: coder/decoder converts audio/video between analog and digital
- VoIP Voice over IP / Internet phone

Additional references

Analog to Digital (S5 4.3)
- Sampling theorem
  - Original signal can be fully recovered if it is sampled at a rate at least twice the bandwidth (or highest frequency)
Analog to Digital

- If sampling rate is too low, reconstruction may be incorrect

Pulse Amplitude Modulation

- Analog pulse height represents signal value at discrete times

Pulse Code Modulation

- Quantise pulses and represent as digital output
- Reconstruction is no longer exact

Audio data

- Telephone speech quality
  - 4kHz
  - 8000 samples/second
    - 128 quantisation levels → 7 bits → 56kbps
    - 256 levels → 8 bits → 64kbps
- CD audio quality
  - 44100 samples/second
  - 16 bits/sample, signed
  - 2 channels
  - 1411.2 kbps

Image data

- TV resolution
  - PAL: 768 X 576 25fps
    - Horizontal resolution actually more like 350 lines
  - Interlaced: vertical resolution reduced to 400
- DVD
  - PAL: 704 X 576 (4:3 or 16:9 non-square)
- HDTV 1080i 25fps interlaced
  - 1920 X 1080 (16:9 square pixels)

Compression

- Lossless
  - Original data can be exactly restored
  - Run-length coding
  - Lempel-Ziv algorithms, LZW
  - Huffman coding
- Audio lossless
  - Linear prediction
  - FLAC
  - Apple’s ALAC
  - Compressed file 40-70% of original size

Lossless Compression

- Images
  - GIF
    - 8 bit image + colour table
    - LZW patent expired 2003-2004
  - PNG
    - Full colour
    - Prediction of pixel colour based on previous pixels
    - LZ77
  - Prediction:
    - 85 83 86 84 88 50 20 14 12 08 07 09 10 09 11
    - 85 -2 +3 -2 +4 -38 -30 -6 -2 -4 -1 +2 +1 -1 +2

Lossy Compression

- Audio
  - Psycho-acoustic
  - MP3
    - Tightly specified decoder
    - Encoders vary depending on what they discard
    - 32-320kbps (128kbps is common)
    - Variable Bit Rate (individual frames of audio can use different bit rates)
**MP3 format**
- Frame = header + data

<table>
<thead>
<tr>
<th>Entry</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Data</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>Bytes</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Meaning</th>
<th>MP3 Sync Word</th>
<th>Version</th>
<th>Layer</th>
<th>Codec Protection</th>
<th>Bit Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Sync Word</td>
<td>1 = MPEG</td>
<td></td>
<td></td>
<td>1 = MP3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(1000 = 160)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Pcm Bit</th>
<th>Pcm Bit</th>
<th>Mode</th>
<th>Audio Extension (used with joint stereo)</th>
<th>Copy</th>
<th>Original Quality</th>
<th>Enhancements</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 = 44.1kHz</td>
<td>0 = Frame in</td>
<td>0 =</td>
<td>(joint stereo)</td>
<td>0 =</td>
<td>0 =</td>
<td>0 =</td>
<td></td>
</tr>
</tbody>
</table>


**Lossy Compression**
- Images
  - JPEG
  - DCT
  - Compression artefacts

- Video
  - MPEG
    - Intraframe compression (similar to JPEG)
    - Interframe compression
      - Copies data from previous frame, possibly with modifications
      - Independent frame has no dependency on prior frames

**Types of multimedia services**
- Streaming stored media
- Streaming live media
- Interactive media
  - VoIP
**QoS challenges** *(KR)*

- End-to-end delay
- Jitter
- Packet resequencing
- Packet loss

**QoS work-arounds**

- End-to-end delay
  - No work-around – delays play out
- Jitter
  - Timestamp packets
  - Buffer with increased delay; drop excessively late packets (see packet loss)
- Packet resequencing
  - Sequence numbers on packets
- Packet loss
  - Forward Error Correction; interpolation of missing data

**Delayed play out**

- Acceptable delay
  - Streaming media can accommodate delays up to several seconds
  - VoIP internet phone
    - 150ms unnoticeable
    - 400ms tolerable
    - >400ms frustrating or unintelligible conversation

**Delayed play out**

- Fixed delay

![Diagram showing delayed play out with missed packet](After KR fig 7.6)
Adaptive play out delay (KR)

- $t_i = \text{timestamp of packet } i$
- $r_i = \text{time packet } i \text{ received}$
- So delay of packet $i$ is $r_i - t_i$
- Estimate
  - Mean end-to-end delay
    \[ d_i = (1 - u) d_{i-1} + u (r_i - t_i) \]
  - Variability of end-to-end delay
    \[ v_i = (1 - u) v_{i-1} + u |r_i - t_i - d_i| \]

Recovering from packet loss

- Note: Excessive delay is treated as loss
- Forward error correction
- Interleaving
- Repair

Adaptive play out delay

- At the start of each talk spurt
  - Calculate playout delay for talk spurt as
    \[ q_i = d_i + Kv_i \]
  - Playout time for packet $j$ is then $t_j + q_i$
- Start of talk spurt:
  - Time difference between packets exceeds chunk (e.g. 20ms)
  - Sequence numbers show no missing packet

FEC

- Aim: To provide sufficient data to correct packet loss without retransmission
  - Redundant information (e.g. parity block every $n$ blocks)
    - Increases data rate by $(n+1)/n$
    - Loss may require $n-1$ packets delay to recover

\[ \text{Includes } P \]
\[ \text{Loss} \]
\[ \text{Recovered} \]
FEC
- Low-resolution version of block n-1 in block n (KR3 fig 7.7)
  - Smaller increase in play out delay
  - Occasional low-quality packet not noticeable

Interleaving
- Interleaving
  - Each packet contains small pieces from a number of intervals (KR3 fig 7.8)

Repair
- Repair
  - Packet repetition
  - Interpolation

RTP Real-time transport protocol
- C4 Fig 29.2
- VER: Currently 2
- SEQUENCE NUM: random start for each session
- PTYPE: Payload type (KR3 tables 7.1, 7.2)
- X: Optional extension headers present.
- P: Indicates if zero padding follows data (used w encryption)
- M: Marker (e.g. start of video frame)
- TIMESTAMP: Random start, regular increment, tick rate depends on application
- SSRC: Source of stream (ID chosen at random, protocol allows for resolving conflicts)
- CC: Number of CSRC (for mixing)
RTP Translator

- Processes RTP stream
- e.g.
  - Change format of packet data
  - Reduce data rate (lower quality video/audio)
  - Tunnel RTP packets through a firewall for separate multicasting inside
  - Receive RTP multicast packets and unicast them to individual receivers

RTP Mixing

- Sources unicast to mixer
- Mixer combines streams and multicasts
  - E.g. mix audio streams
  - SSID of combined stream is mixer
  - CSID’s record source SSIDs

RTP in UDP

- RTP is not usually encapsulated into IP
  - KR3 fig 7.10, 7.11
- RTP is encapsulated in UDP
  - UDP provides port number
  - Multiple applications can use RTP on one host
  - RTP uses even UDP port number
    - Session port, not reserved port
    - Application must notify other end
  - RTCP uses the next (odd) UDP port

RTCP

- QoS and congestion control
  - RTCP allows session members to feedback quality of data distribution – multicast to all.
- Identification of users
  - Association of video and audio streams together
- RTCP report frequency is adjusted depending upon number of participants to limit RTCP traffic to 5% of total session traffic.
RTCP sender report  \((\text{S(CNIPT)} \text{ pp374-378})\)
- SSRC of sender
- NTP timestamp (64 bits)
- Corresponding RTP timestamp
- Count of packets sent in stream
- Count of octets sent in stream
- Reports on receiving from other senders
  - Same as for receiver report…

RTCP receiver report
- \((\text{S(CNIPT)} \text{ fig 10.11})\)
- SSRC of sender of report
- Report blocks for each source
  - SSRC of source
  - Fraction and count of packets lost
  - Extended highest sequence number received
  - Interarrival jitter
  - Abbreviated NTP time of last SR received from sender
  - Delay since receiving last SR from sender

Interarrival jitter  \((\text{S(CNIPT)})\)
- \(S(i) = \text{Timestamp from RTP packet } i\)
- \(R(i) = \text{Arrival time of packet } i\)
- \(D(i) = R(i) - R(i-1) - (S(i) - S(i-1))\)
  - Change in spacing between packets from source to destination
- \(J(i) = (1-u) J(i-1) + u |D(i)|\)
  - \(u = 1/16\)
- Estimated average interarrival jitter

RTCP source description
- One or more chunks
- SSRC or CSRC of contributing source
- Descriptive items  \((\text{S(CNIPT)} \text{ table 10.4})\)
  - CNAME Canonical name – unique in RTP session
  - NAME Real user name
  - EMAIL, PHONE, LOC (geo location)
  - TOOL (application generating stream)
  - NOTE, PRIV (extension)
RTCP packet types
- Sender report
- Receiver report
- Source description
- Goodbye (source no longer active)
- Application-defined extension

VoIP: Internet Phone
- Establish connection to other user
- Send/receive audio/video data
- Manage QoS impacts
  - Described above

SIP
- Session Initiation Protocol
- Set up, modify, terminate real-time sessions
- Single media or multimedia
- Teleconferencing
- Example: KR3 fig 7.13

SIP features (S(CNIPT) 4.3)
- User location
  - User can access services from other locations (e.g. telephone)
- User availability
  - Is called party willing to engage in communication?
- User capabilities
  - Media to use and parameters
SIP features

- Session setup
  - Point-to-point, multi-party, with agreed parameters
- Session management
  - Transfer and termination of sessions
  - Modify session parameters

SIP interaction

- Client/server
  - Client: sends SIP requests and receives SIP responses
    - User agent clients
    - Proxies
  - Server: receives SIP requests and sends SIP responses
    - User agent service
    - Proxies, Redirect servers, Registrars

SIP network elements (Fig 4.8)

- User agent (SIP end station)
  - User agent client (UAC): issues SIP requests
  - User agent server (UAS): receives SIP requests and issues responses to accept, reject or redirect
- Redirect server
  - Responds to calling device telling it to contact the alternate URI

SIP network elements

- Proxy server
  - Routes requests closer to destination
  - May enforce policy
  - Interprets request and may rewrite part of it before forwarding
- Registrar
  - Accepts REGISTER requests, storing SIP address and corresponding IP address into location service
SIP network elements

- Location service
  - SIP redirect or proxy server queries location service for information about location of the callee
  - Maintains database of SIP address mapped to IP address

SIP Example

- S(CNIPT) Fig 4.9
  - Alice tries to phone Bob who is not signed in
  - Alice contacts her proxy server (1-2)
  - Alice’s proxy server locates Bob’s proxy server and contacts it (3-6)
  - Bob’s proxy server queries its location service (not a SIP protocol operation) (7-8)
  - Bob is not available (9-12)

SIP Example

- S(CNIPT) fig 4.10, 4.11
  - Alice requests notification when Bob comes on line (Extension - not yet part of SIP)

SIP Example

- S(CNIPT) fig 4.12
  - Alice sets up a call to Bob
    - Proxy server has cached DNS data from before (1-4)
    - Location is available (5-6)
    - INVITE is forwarded (7-10)
    - Bob answers (11-13)
    - Direct end-to-end ACK (14)
    - Call proceeds using RTP
SIP protocol

- Requests
  - REGISTER: Notify my current IP address and URL to receive calls
  - INVITE: Establish media session
  - ACK: Confirm reliable message exchange
  - CANCEL: Terminate pending request
  - BYE: Terminate session between two users in a conference
  - OPTIONS: Request information about callee capabilities

SIP protocol

- HTTP syntax
  - E.g. (S(CNIPT) p144)
    INVITE sip:bob@biloxi.com SIP/2.0
    Via: SIP/2.0/UDP 12.26.17.91:5060
    Max-Forwards: 70
    To: Bob <sip:bob@biloxi.com>
    From: Alice <sip:alice@atlanta.com>;tag=1928301774
    Call-ID: s84b4c76e66710@12.26.17.91
    CSeq: 314159 INVITE
    Contact: <sip:alice@atlanta.com>
    Content-type: application/sdp
    Content-length: 142

SDP session description protocol

- ASCII text protocol, sent as content of INVITE
- Describes content of session
  - Follow the high-level structure of HTTP code groups
  - Reuse the same response codes with the same meanings (200, 404, etc)
  - Media streams with types similar to MIME
  - Destination addresses
  - Sending and receiving UDP ports
  - Payload types
  - Start and stop times (broadcast, e.g. TV)
  - Originator of broadcast streams
- E.g. (KR3 p606)
  c=IN IP4 167.180.112.24
  M=audio 38060 RTP/AVP 0
VoIP using SIP

- H5 fig 8.19

SIP and RTP protocol design

- SIP
- ASCII
- HTTP based
- Regular use
  - Low bandwidth impact
- Flexible
  - Negotiations
  - Naming
- Extensible
- RTP, RTCP
- Binary fields

H.323

- Minimal audio codec: G.711 56kbps or 64kbps PCM
- Video optional. Minimal video codec: QCIF H.261 (174 X 144 pixels)
- RAS: resource access service (interactions with gatekeeper)
- Q.931 signalling (end-to-end)
- H.245 control (negotiate capabilities and open logical channels)
- RTP, RTCP: transmission of actual audio/video
- H5 fig 8.23