Lecture-5 notes for SSY150: Multimedia and video communications

Transport Compressed Multimedia Data Through Error-Prone Networks

Professor Irene Yu-Hua Gu

Chalmers University of Technology, Sweden

April 23, 2020

Content

1. Motivations
2. Network architecture: OSI reference model and IP model.
3. IP Protocol stacks
4. End-to-end performance optimization: options and adjustable parameters
5. Main QoS parameters
6. About laboratory 3 exercise

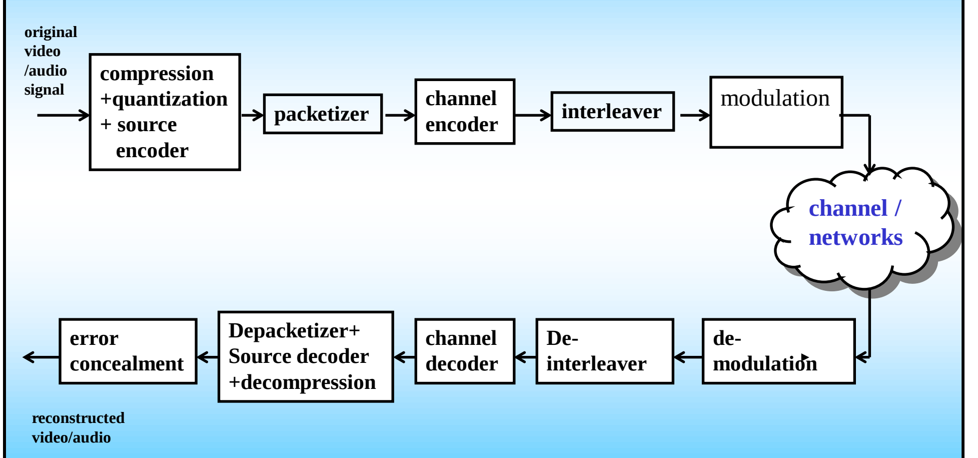
1. Motivations

Applications of multimedia communications:

- Video communications
- Video conferencing, video broadcast, video telephony, video streaming
- IP-phone, audio streaming
- Distance learning
- E-medicine, E-health care

....

Multimedia communications



2. Network architecture: OSI 7-layer reference model and the IP model

OSI's 7-layer communication model vs. the 5 (or 4)-layer TCP/IP model

The OSI model	The TCP/IP model (merge layers 5,6,7)
7. Application layer	Application layer (7+6+5)
6. Presentation layer	
5. Session layer	
4. Transport layer	Transport layer
3. Network layer	Internet layer
2. Data link layer	Data link layer
1. Physical layer	Physical layer (exclude in 4 layer's model)

Functionalities of OSI's layers

1. **Physical-layer:** provides point-to-point, point-to-multipoint **bit** transport services over **wires, optical fibers, or free space**.
   Handles physical aspects, such as physical media, electrical impulse, transmitter power, modulation…
   Hubs are in this layer.

2. **Data link-layer:** provides point-to-point, point-to-multipoint **packet** services (e.g. detect bit errors, re-transmit of lost/error packets), defines the format of data on the network.
   Switches are in this layer.

3. **Network-layer:** is responsible for addressing, routing, and congestion control. It carries packets end to end across subnets (connected by **routers**).
   The path of packets is determined by routing protocols.

Functionalities of OSI's layers (cont'd)

4. **Transport-layer**: performs end-to-end error checking/correction and flow control. Transfers data between communication end points (or, end systems, hosts).
   TCP and UDP protocols are in this layer where the Internet architecture is primarily built on.

5. **Session-layer**: establishes/manages connection

6. **Presentation-layer**: ASCII Text, Sound (syntax layer)

7. **Application-layer**: services to users and programs, e.g. file transfers, http, email, video streams …

Functionalities of TCP/IP 5 (or 4)-layers

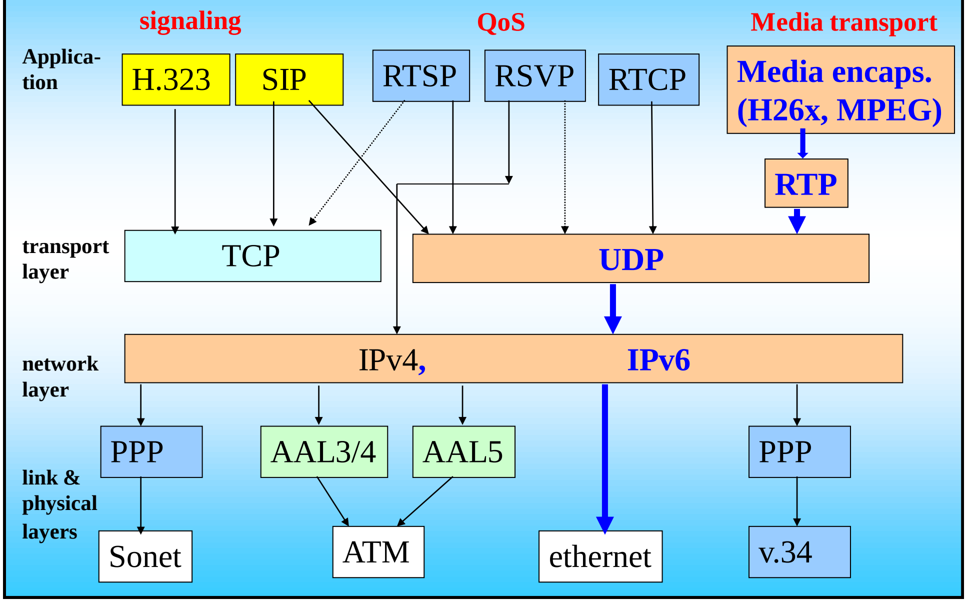
- **Application Layer**: contains the logic needed to support the various user applications. Each application requires a separate module.   Contains high level protocols: FTP, HTTP, SMTP
  (create data unit called message)
- **Transport Layer:** flow control and end-to-end error checking
  (add header to form segment)
- **Internet Layer**: IP provides the routing functions across multiple networks (routers). TCP/IP protocols are in this layer
  (add header to form datagram)
- **Data link layer**: access and link data across a network between two end systems (switches)
  (add header to form frame)
- **Physical Layer**: covers physical interface between PC/ workstation and a transmission medium/network (Hubs)

3. Internet Multimedia Protocol Stacks

Protocols

- **Protocols are used for communications between networks, achieves:**
  - What, how, when, in which sequential order: is the data communicated between networks ?
- **Examples of protocols:**
  - WAN Protocol: TCP/IP
  - LAN Protocol: Media Access Control (MAC); Contention; Token Passing

Internet multimedia protocol stack



Abbreviations of Protocols

SIP: Session initiation protocol (and H.323 for IP telephony)

RTSP: Real time stream protocol (for media on demand)
RTCP: Real time control protocol (for control, management)
RSVP: Resource reservation protocol

TCP: Transmission control protocol
UDP: User datagram protocol
RTP: Real time transport protocol

Transport Control Protocol (TCP)

- For connection-oriented stream of bytes (create a circuit between the sender and the receiver during transmission).

- Error free-transmission: provide window-based (buffer) positive acknowledgement (ACK) with a go-back-N retransmission of lost or error packets (→ no packet loss).

- Having its own congestion control - flow control.

- Reliable sequenced byte stream service, ordered packets and reassembling packets.
- Could introduce unbounded time delay due to persistent re-transmission
- Not suitable for real time video applications

User Datagram Protocol (UDP)

- For connectionless datagram service;
- No re-transmission of lost packets;
- Does not provide sequencing (or ordering) of packets;
- Without its own congestion control mechanism;
- Simple checksum option for verifying errors in arrived packets (currently, only intact packets are forwarded to the application layer);
- Allows variable length in data payload;
- Does not provide reliable service;
- Suitable for real time video applications.

Comparison: UDP vs TCP

	UDP	TCP
Packet loss	Yes	No
Data	Packet	Byte stream
Ordering	No	Always in order
Duplication	Possible	No
Multicast	Yes	No
Acknowledge	No	Yes

Internet protocol (IP): IPv4, IPv6

- uses a set of rules to send/receive messages at Internet address level
- Is a connectionless protocol (i.e. no established connection between end points that are communicating.)
- Responsible for delivery packets (while TCP for assembling packets)
- Is a data-oriented protocol for data communicating across a packet-switched inter-network.
- has checksum for the IP header (not for the data)
- is encapsulated in a data link layer protocol (e.g., Ethernet)

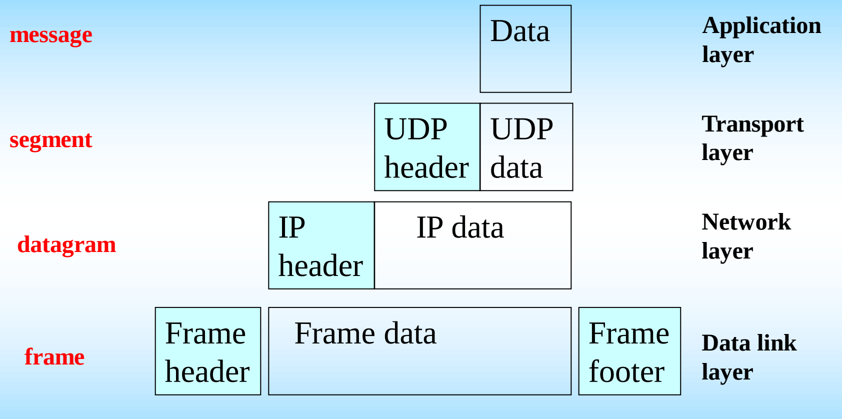
IPv4:

- designed for fixed networks and 'best effort' applications (with low network requirements, .e.g. email, file transfer).
- unreliable service that is subject to packet loss, reordering, duplication unbounded delay. Not appropriate for real-time multimedia services.
- no mobility support

IPv6:

Includes some means of QoS and mobility support (suitable for IP-based wireless); More suitable for multimedia services

Example of data encapsulation



Efficiency: proportional to the header/data (payload) ratio

(from Wikipedia, the free encyclopedia)

Real-time Transport Protocol (RTP)

- Operates in the application layer (usually in conjunction with UDP/IP) for real-time applications such as video and audio.
- Consists of data part and control part (called RTCP, Real-time Transport Control Protocol)
- Provides end-end network transport functions suitable for transmitting real-time data over unicast/multicast networks
- RTP payload formats are defined on a range of codes (e.g. H263, H261, JPEG, MPEG and some audio codes)

Real-time Streaming Protocol (RTSP)

- Operates in the application layer (on top of UDP/IP) for real-time stream applications
- Establishes and controls time-synchronized (one or multiple) media delivery with real-time constraints.
- It does not depend on use of RTP, or a particular media format (e.g. MPEG)

e.g. Application RealPlayer

Example: Protocols for Video Streaming

- Network-layer protocol:
  Internet Protocol (IP)

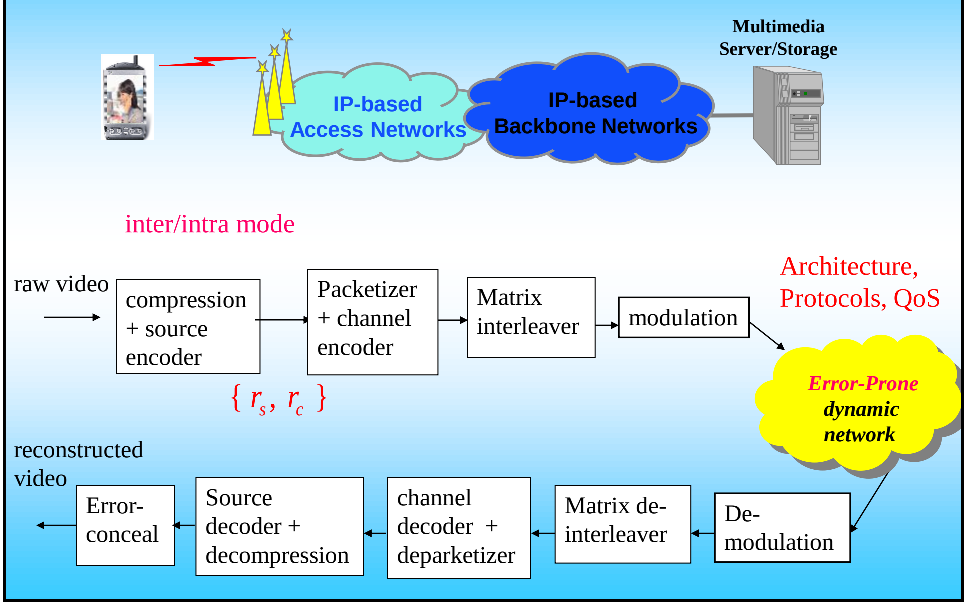
- Transport protocol:
  – Lower layer: UDP & TCP
  – Upper layer: Real-time Transport Protocol (RTP) & Real-Time Control Protocol (RTCP)

- Session control protocol:
  – Real-Time Streaming Protocol (RTSP): RealPlayer
  – Session Initiation Protocol (SIP): Microsoft Windows Media Player; Internet telephony

4. End-to-end performance optimization

Options and adjustable parameters

Compressed Video over Communication Networks



Adjustable parameters for error & rate control

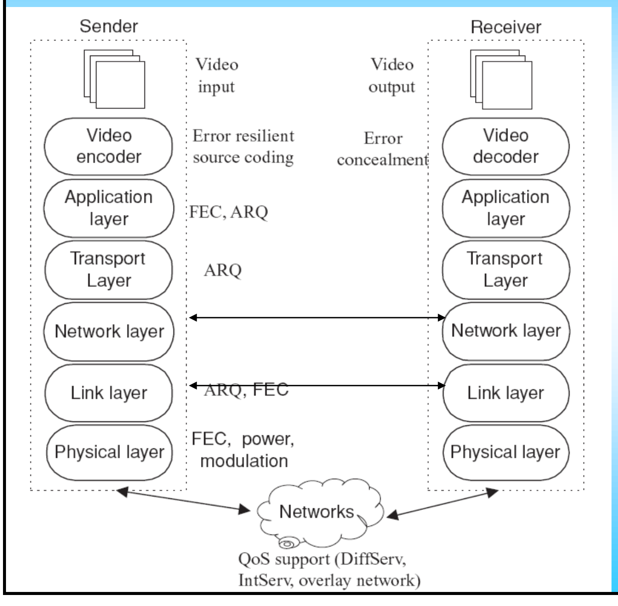


Figure: components that error and rate control can be applied for improving end-end performance

(From Dr. Fan zhai)

Components for Error & Rate Control: Overview

At the SENDER side:

Source encoder:

- Error resilience

  compression methods, VLC, scalability …

- Rate control:

  inter/inter modes, MC scheme,

  intra refreshing rate

At the SENDER side (cont'd):

FEC for inter and intra packets (application/link/physical layer) to combat network/channel errors and control of congestion

- channel coding ***across*** packets (application/transport layer)
- channel coding ***within*** packets (link layer)

ARQ (automatic repeat request): re-transmit lost packets in the physical layer (application/transport/link layer)

Control/adjust transmission power in wireless for prioritized packet transmission (physical layer)

At the RECEIVER side:

- error concealment

In the network:

QoS classes based on congestion control models (e.g. TCP-friendly …)

QoS support for IntServ and DiffServ (integrated services, differentiated services)

Intelligent network management: e.g. Routing

Aim of congestion control: sharing network resources fairly with other users in the transport layer.

Challenges

- **Quality of Service (QoS):** in terms of bandwidth, packet loss probability, delay distribution, throughput.
  There is a mismatch between the video applications and communication networks!
- Channel noise, especially in wireless channel due to fading, multi-path, shadowing effect, resulting in higher bit error rates and low throughput for wireless video communications. Current standards discard (at the link layer) a packet with unrecoverable bit errors. Packet losses due to erasure channels.
- Error control: Limit the effect of channel/network errors
- Congestion control: share network resources fairly with other users at the transport layer, routing;
- Throughput: Limited by the network resources

Further details:
improving end-to-end performance

Source Coding

- Selecting coding/compression methods:
  DCT, DWT, …

- Select inter/intra mode:
  control coding rate and prevent error propagation

- Parameters within intra mode:
  quantization step size (→ different errors),
  variable length codes (VLC), intra refreshing rate

- Parameters within inter mode:
  MC schemes, coding of MC errors

Joint Source and Channel Coding

Why joint source-channel coding?

Shannon's Separation Principle breaks down

- Finite block length in coding
- Lack of complete knowledge of channel statistics

Shannon theory for communication holds, IF:

a) Infinite block length → infinite delay

b) Known channel statistics

Joint source and channel compression/coding that optimizes the end-to-end packet video performance

Packetization and Packet Protection

- **UEP** (unequal error protection) **of packets**:
  different rates for packets of MVs, DCT coefficients

- **Packetization schemes:**
  - Source packetizer:
    ensure packets being encoded independently,
    size of packets: tradeoff between efficiency in header/payload ratio and packet loss
  - Re-packetizer in the intermediate layer:
    e.g. interleaving, FEC
  - Packetizer in the transport/network layer:
    avoid fragment/split (max. size ~1500 bytes)

Channel Coding Methods

- Channel coding may be applied to
  Inter-packets: usually for packet loss errors
  Intra-packet: usually for bit errors
  (especially for wireless channels)

- Channel coding methods for video data packets
  - RS (Reed-Solomon) codes
  - RCPC codes (rate-compatible punctured convolutional codes)
  - Turbo codes
  ....

Reduce Dominant Errors

Different networks have different types of dominant errors.
Strategy: reducing dominant errors in the network.

For Internet/wired networks:
bursty packet loss (or, erasure channels)

For wireless networks:
high bit errors, in addition to packet loss
(due to narrow bandwidths, multi-path fading and shadow effects)

Prevent Packet Losses

Packet losses

a) occur when network buffer overflows (congestion)
b) discard when packets not arrive in time (excessive delay)
c) due to packet truncation
d) due to unrecoverable bit errors in a packet

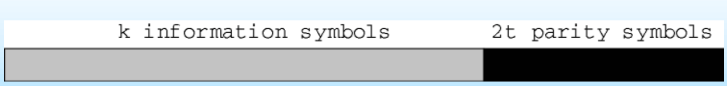
Ways to combat the problem

- UEP in source encoders (intra/inter)
- FEC in application/transport layer (e.g. RS codes)
- Interleaving
- Cross-layer/joint source-channel coding (end-end optim.)
- Re-transmission of packets if applications allow (ARQ)

FEC: RS Codes + Block Interleaving

Reed-Solomon codes

- **Block-based codes suitable for packet video (systematic encoding)**



- **Good correction capabilities against bursty errors**

Reed-Solomon Codes

BCH codes in a Galois field: GF(2^m)

RS(n,k): codeword length = 2^m -1 (m: bits per symbol)

message length = k symbols

parity check = $n - k$ symbols

maximum corrected error = $\lfloor (\text{n-k})/2 \rfloor$ symbols

Block Interleaving

Problems:

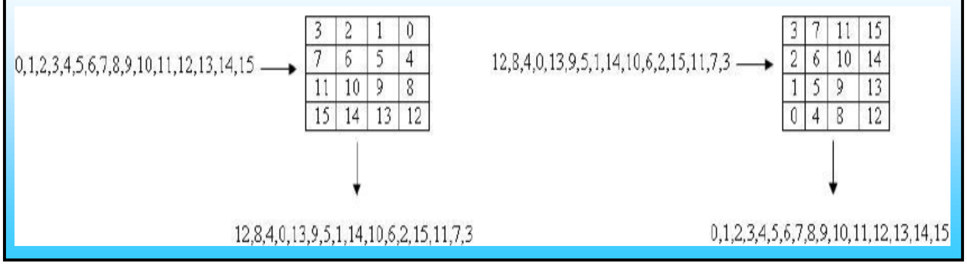
Bursts of symbol errors, or loss of an entire packet

Solution:

Use interleaver to spread errors.

input: codewords in rows

output: columns as to the channel



Error Concealment (at Receiver Side)

Observing: image frames are highly correlated in spatial, temporal, spatial-temporal domain.

Errors from a lost packet can be concealed by using:

- Spatial correlation
- Temporal correlation
- Joint spatio-temporal

5. Main QoS parameters

Packet Delay, Round Trip Time Delay, Latency, Delay jitter, Throughput

Packet delay is caused by:

- Physical distance in transport information between 2 comm. nodes
- Encoder /decoder processing time, encoder/decoder buffer delay
- Queuing delay in the network caused by network congestion.

Round trip time delay (RTT): the time interval between sending a packet and receiving the acknowledgement of the packet.

End-to-end delay (latency): is the average delay time between the input (in the sender) and output (in the receiver).

Delay jitter (variation): is mainly caused by network congestion, multiple paths

Throughput: is the *effective* output rate over the network

(excluding re-transmitted packets and the lost packets)

Main QoS Parameters

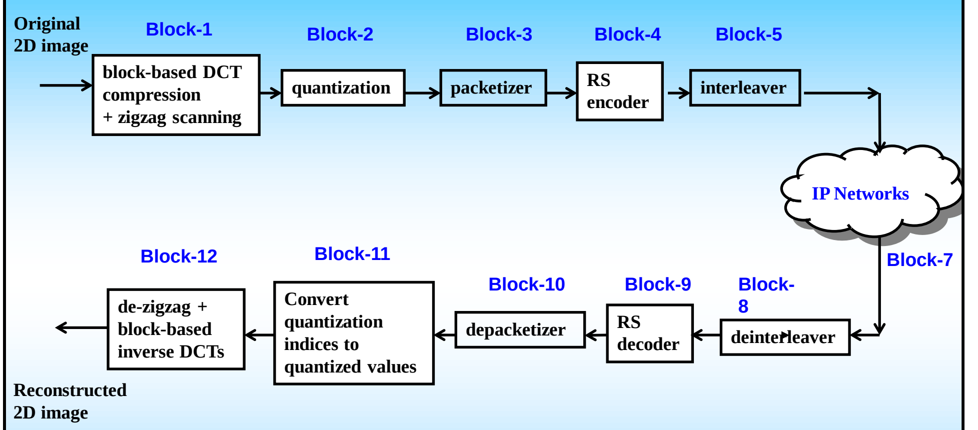
- Throughput
- Packet loss / truncation
- Round trip delay time (RTT)
- End-to-end delay (latency), and delay jitter

Remark

QoS parameters in lower network layers:

not always directly associated with the QoS requirements in application layer !

6. About Laboratory Project 3



Lab.3: Pipeline of an end-to-end image communication system over IP networks, to be built and tested (using Matlab communication toolbox)

Lab.3: Multimedia Communications over IP Networks

1. Building a simple multimedia communication system by Matlab programs (using functions in Matlab communication toolbox).

   include: 2D image compression, zigzag scanning, scalar quantization, packetization, RS codec, matrix interleaving, and their inverse processes.

2. Conduct two sets of tests:
   * Erasure network model: exam the impact of packet losses (in the network layer) to the received image.
   * AWGN channel noise model: exam the impact of bit errors (in the physical layer) to the received image.