ImmediaTV Corporation

Innovative video networking

Innovations today for Broadcasters of tomorrow

Issues in Transporting MPEG Over IP

Ciro A. Noronha, Ph.D.

Innovations today for Broadcasters of tomorrow

Outline

- Objective: *transport MPEG streams over a traditional packet-switched IP network with acceptable quality.*
- Steps:
  - ***Understand the MPEG requirements into the network (given the current state of the art)***
  - ***Understand what it takes for the network to satisfy these requirements, in regards to:***
    - ***Infrastructure***
    - ***Protocol***

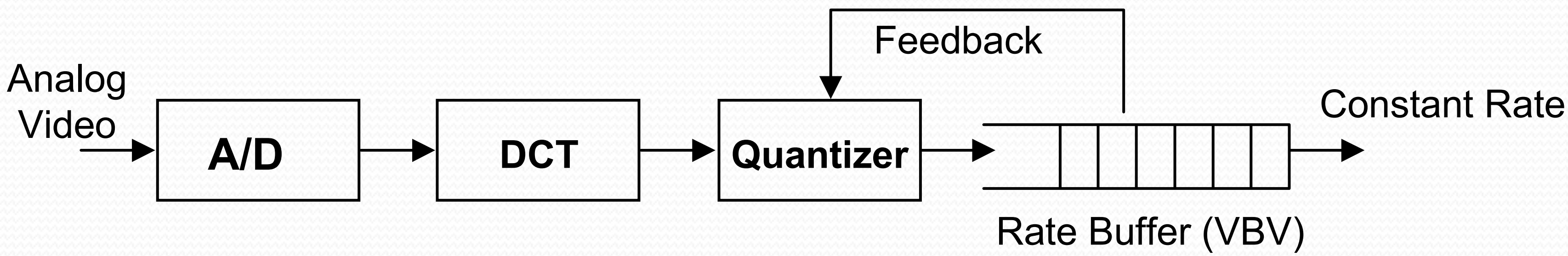
MPEG Requirements Overview

- To play properly, an MPEG stream must be delivered to the decoder in a *timely fashion* with *acceptable errors or losses*.
  - Timely fashion: the stream must be delivered with acceptable delay jitter.
  - Errors or Losses: the packet loss and/or corruption must be kept to an acceptable level.

The Jitter Requirement

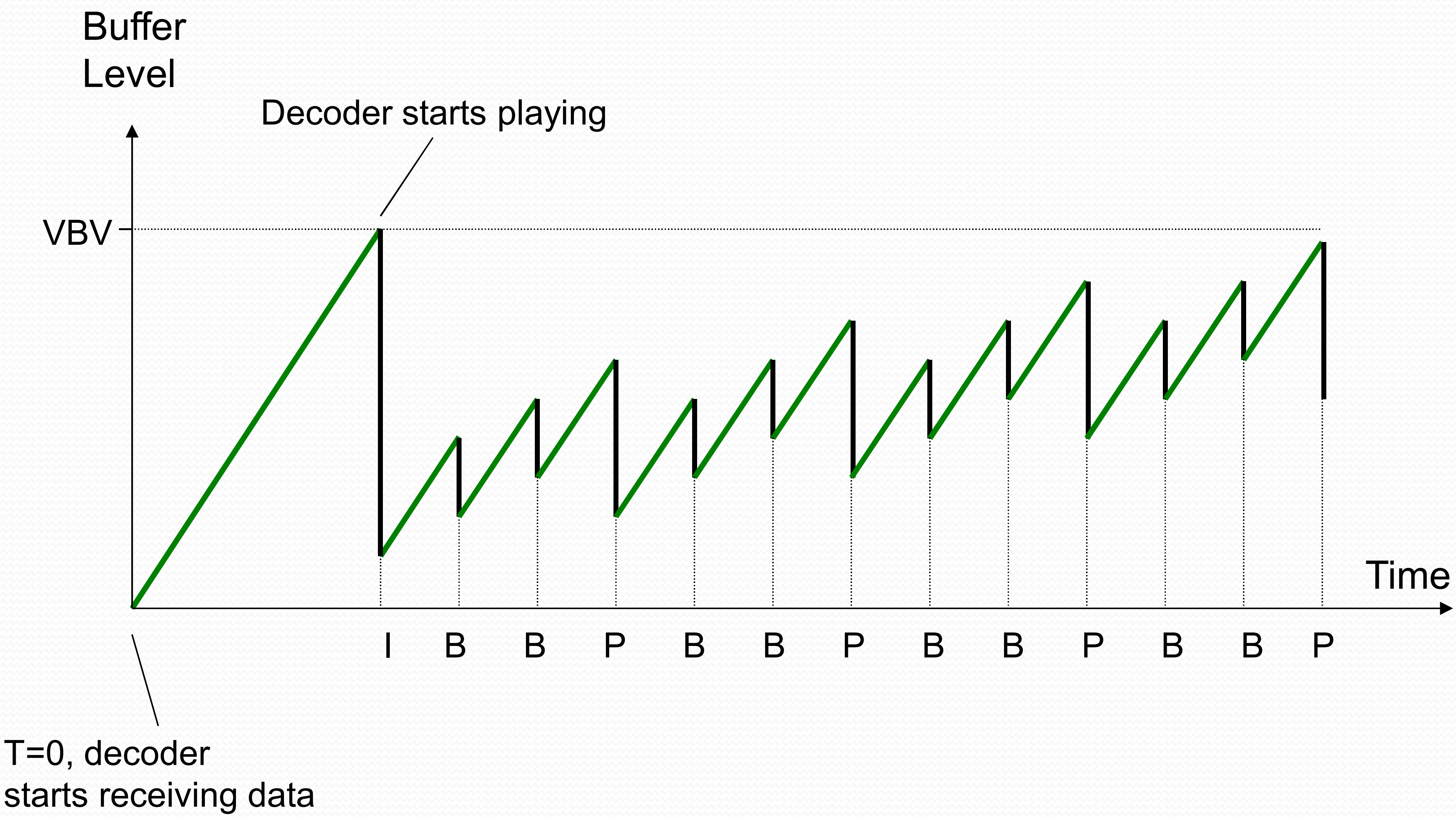
- In the "ideal" case, the MPEG bitstream is delivered to the decoder in a steady fashion
  - Satellite and Digital Cable systems work this way
- In a packet-switched network:
  - The bitstream is divided into packets
  - Packets are carried through the network
  - Successive packets may experience different delays → ***delay jitter***!
- How much delay jitter is acceptable?

Simplified view of the encoding process

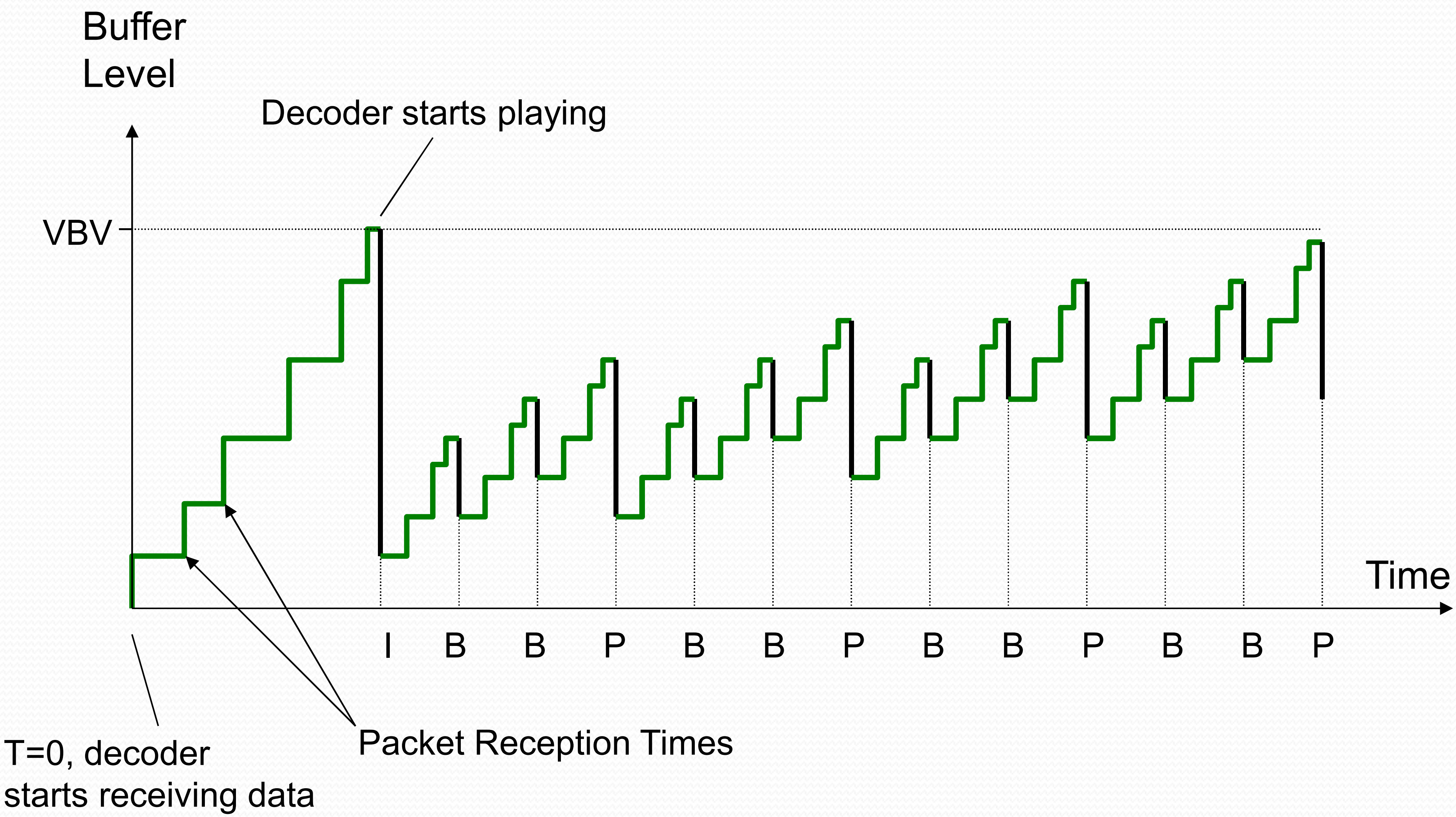


- After compression, the bitstream is deposited in the rate buffer
- The occupancy of the rate buffer controls the quantization; rate buffer never overflows or underflows
- The decoder must buffer the equivalent of one VBV prior to start playing
- Default MPEG-2 VBV size: 224 kbytes.

Decoder Playback Process (constant rate)



Decoder Playback Process (packet delivery)



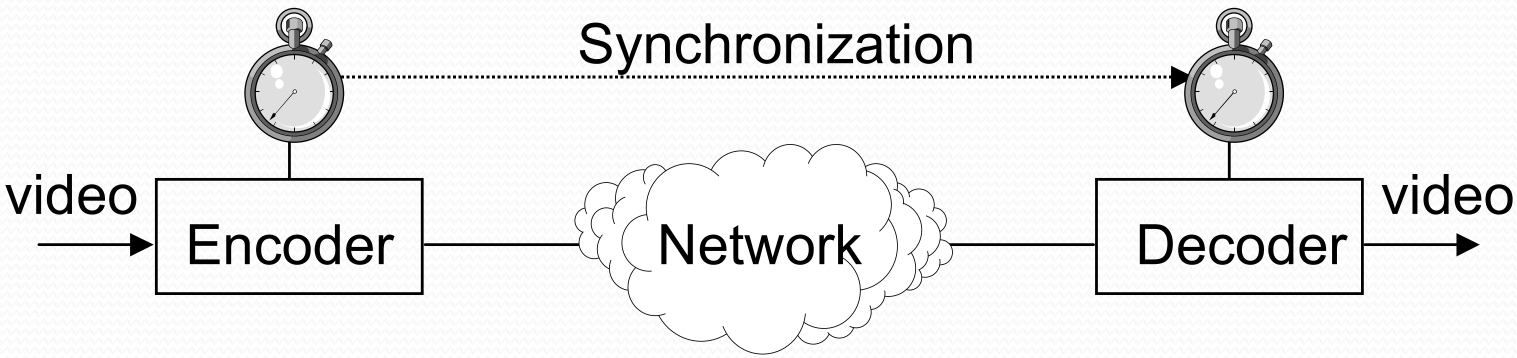
Notes on Packet Delivery

- It really does not matter how the data arrives between frames, as long as the right amount is received.
- Quick calculation:
  - 30 frames/sec → 33 ms between frames
  - MPEG-2 stream at 3.5 Mb/s
  - Network needs to deliver 14,583 bytes every 33ms → about 11 IP packets (Ethernet size)
- No problem on a local switched network; may be an issue on a multi-hop network.

Solution: just more buffering

- Add 1 Mbyte buffer to the decoder
  - At 3.5 Mb/s, it can tolerate latencies of up to 2.3 seconds!
- All current IP-based set-top boxes have large buffers.
- *Conclusion: Delay Jitter is NOT an issue for modern networks and set-top boxes!*

End-to-End Synchronization



- The encoder generates data based on the input video timing
- The decoder plays back the stream based on its own clock, which is nominally the same
- If clocks are not locked, the decoder will eventually overflow or underflow
- Delay jitter makes locking the clocks harder

Approaches to synchronization

- Decoder drops or repeats frames as needed
  - Slowdown or speed-up of the audio clock
  - Worst case computation:
    - MPEG requires 30 PPM precision on the clocks.
    - Worst-case combination: takes about 9 minutes for the clocks to drift the equivalent of one video frame.
    - Frequency of corrective action depends on buffering
- Build a PLL in the decoder
  - Delay jitter will make it harder to design the loop filter, but it is still quite doable.

Packet Loss and Corruption

- Due to compression, data loss and/or corruption has very significant effects:
  - Glitch due to error in I-frame lasts 0.5 sec
  - Glitch due to error in P-frame averages 0.25 sec
  - Glitch due to error in B-frame lasts 33 ms
- Due to CRCs and checksums at various levels, it is virtually impossible for data corruption to go undetected
  - Corrupted packets are discarded by the stack or hardware
  - Data corruption = packet loss
- Packet loss and corruption needs to be kept at an "acceptable" level
  - What is an acceptable level?

Acceptable Packet Loss

- **Assume a 3.5 Mb/s stream, 1400-byte packet payload**

$$\text{Glitch Interval} = \frac{8 \times \text{Packet Size}}{\text{Bit Rate} \times \text{Packet Loss}}$$

Loosing one packet in	Results in a glitch every
10 million	8 hours 53 minutes
1 million	53 minutes 20 seconds
100 thousand	5 minutes 20 seconds
10 thousand	32 seconds
1 thousand	3.2 seconds

- Acceptable level is subjective, but a good number is around 1 packet per million for this data rate.

Packet Re-Ordering

- MPEG bitstreams do not have re-ordering capabilities
- IP networks may deliver packets out of order
  - This may happen when there are multiple paths between the source and the destination
- Protocol support and re-ordering buffers are required to provide in-order delivery

Network Requirements

- No requirements on delay jitter
- Acceptable packet loss
- In-order delivery
- Point-to-multipoint (one-to-many) delivery capability
  - Application requirement
- Guaranteeing acceptable packet loss is primarily a network infrastructure issue
- Guaranteeing in-order delivery is primarily a transport protocol issue

Causes of Packet Loss

- Bit errors in the transmission medium
  - With current technology, these are almost non-existent; fiber links typically have a BER of 10^{-13} or better.
- Congestion in the network
  - Primary cause of packet loss
  - Links get overloaded, queues overflow, and otherwise "good" packets are dropped for lack of buffer space
- Solution:
  - Bandwidth management/resource reservation

Approaches to Bandwidth Management

- *Traffic Engineering:*
  - The network is designed in such a way that there is always bandwidth for the MPEG stream.
  - Example: dedicated network.
  - Only applicable to some very specific situations.
- *Static Provisioning:*
  - The MPEG traffic is "marked" in some way as to identify it to the network
    - Examples: source/destination address, port, DS codepoint
  - The "marked" traffic is given higher priority wherever the bandwidth is constrained
  - Useful when the stream are static and do not change often; changes require manual reconfiguration of the network

Bandwidth Management (cont.)

- *Resource ReserVation Protocol (RSVP)*
  - Defined by RFC 2205
  - The sender, receiver(s) and all intervening routers must support it
  - Sender defines the stream requirements, receivers make the reservation; works dynamically
  - Problem: keeps lots of state, does not scale well
- *Multi-Protocol Label Switching (MPLS)*
  - Defined by RFC 3031
  - Reservations can also be made on demand
  - The senders and receivers do not necessarily need to support it; the ingress routers can classify and label the traffic, and the egress routers can remove the labels.
  - Designed to scale

Recovering from *occasional* packet loss

- *IF* enough bandwidth is allocated, some techniques can be used to minimize the effect of packet loss:
  - Error concealment: do nothing at the network side, let the MPEG decoder try to conceal the error.
    - Most decoders do this to a certain extent
  - Retransmission: very efficient, but requires a return path; may not scale well with number of receivers.
  - Forward Error Correction: inject redundancy in the stream, rebuild lost packets from received packets
    - Scales well with number of receivers
    - Less efficient than retransmission for small number of receivers

Satisfying Protocol Requirements

- Raw MPEG over UDP:
  - Does not satisfy any of the requirements (no re-ordering capability, no FEC, no way to identify retransmissions).
  - Still usable, if the functions above are not needed.
- RTP:
  - Support for re-ordering, de-jittering and payload type identification
  - Support for compatible FEC
  - No explicit support for retransmission on MPEG (defined only for H.261), but easy to add

Conclusions

- To support MPEG, the IP network infrastructure primarily needs to guarantee *BANDWIDTH*.
- How the bandwidth is managed (statically, dynamically) is a function of the application.
- Techniques such as FEC and retransmission can be employed to correct an *occasional* packet loss, but are no substitute for bandwidth.
- IP Multicast support must be designed into the network as required by the application.
- MPEG should preferably be transported over RTP; raw MPEG over UDP can be used depending on the network.