
Error Control for Real-Time

Audio-Visual Services

Georg Carle

Institut EURECOM

Sophia-Antipolis — FRANCE

carle@eurecom.fr

<http://www.eurecom.fr/~carle/>

(Joint work with E. Biersack, J. Nonnenmacher and S. Rössli)

Multicast Error Control for Real-Time Audio-Visual Services

Overview

- Motivation
- Real-time error recovery by ARQ and FEC
- RTMC-Protocol
- Mechanism selection and dimensioning
- Priority-based error control
- Charging
- Conclusions

Multicast Applications

- Existing applications involving audio-visual data streams:
 - Real-time audio and video transmission using tools such as vat, ivs, vic, and nv.
Since there is no retransmission of lost data, the applications are built to handle and conceal (if possible) loss
 - Shared Workspace for collaborative work using a tool such as wb: loss is handled at application level (SRM Protocol):
 - NACKs are transmitted via multicast to all other receiver
 - any receiver who has the missing data does retransmission
 - Dissemination of stored continuous media streams
 - News on demand of audio and video information such as weather info, radio emissions, CDs, educational material, movies

Example Application

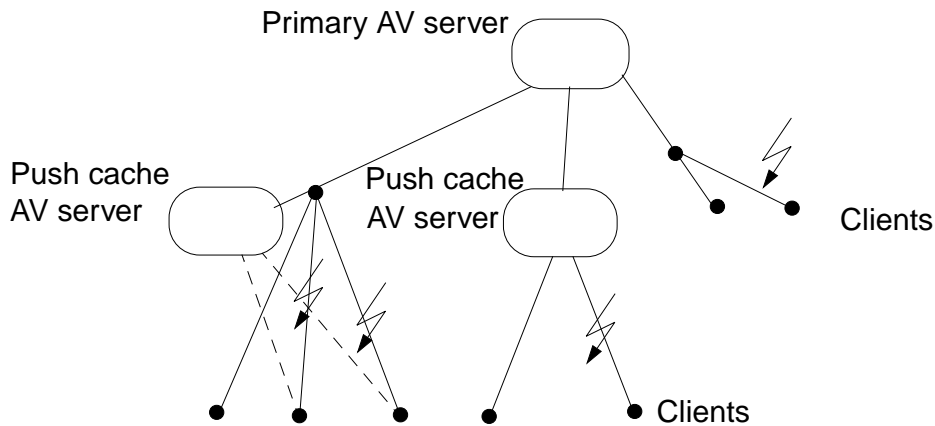
Error Control for Audio-Visual Servers

- Audio-visual server for browsing of video-clips (Fast Forward etc.)
- Goal: use of cheap network services, tolerating high loss rates, and delay violations by network and by server(s)
 - ↳ Powerful error control needed!
- Hierarchical caching scheme attractive for avoidance of bottle-neck at primary AV server
 - Web proxy caches suffer from low hit ratio for large documents
 - Potential solution: server push caching to exploit overall vision of primary server
 - ↳ Requirement for Real-Time Multicast Protocol to distribute data to multiple push cache AV servers

Audio-Visual Servers with ARQ/FEC

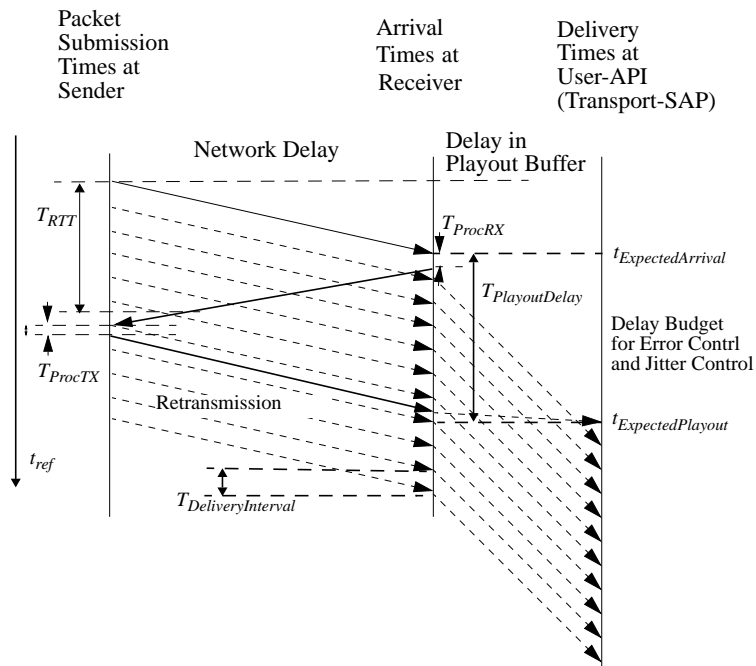
○ Error control needed:

- Between Primary AV server and client;
- Between primary AV server and push cache AV server;
- Between push cache AV server and client.

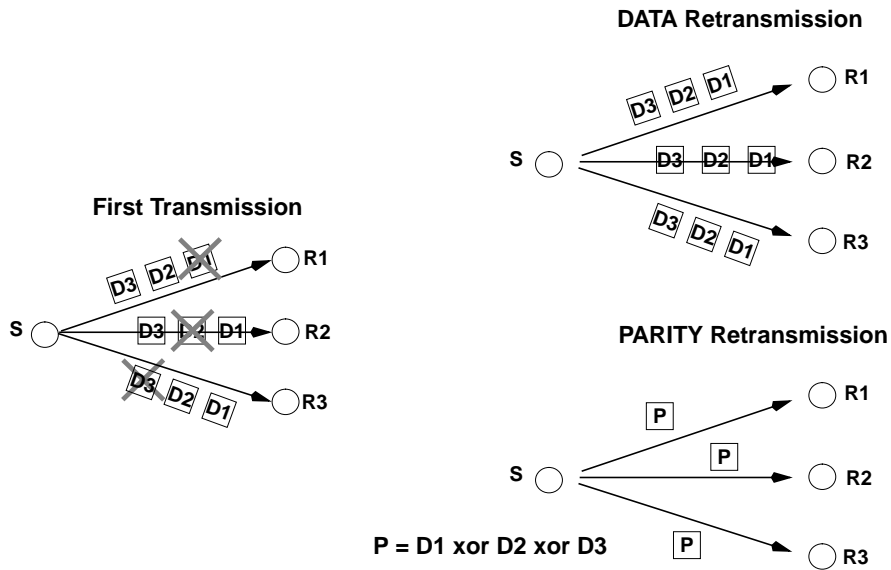


Error Control for AV Services - ARQ

Exploitation of End-to-End Delay Budget



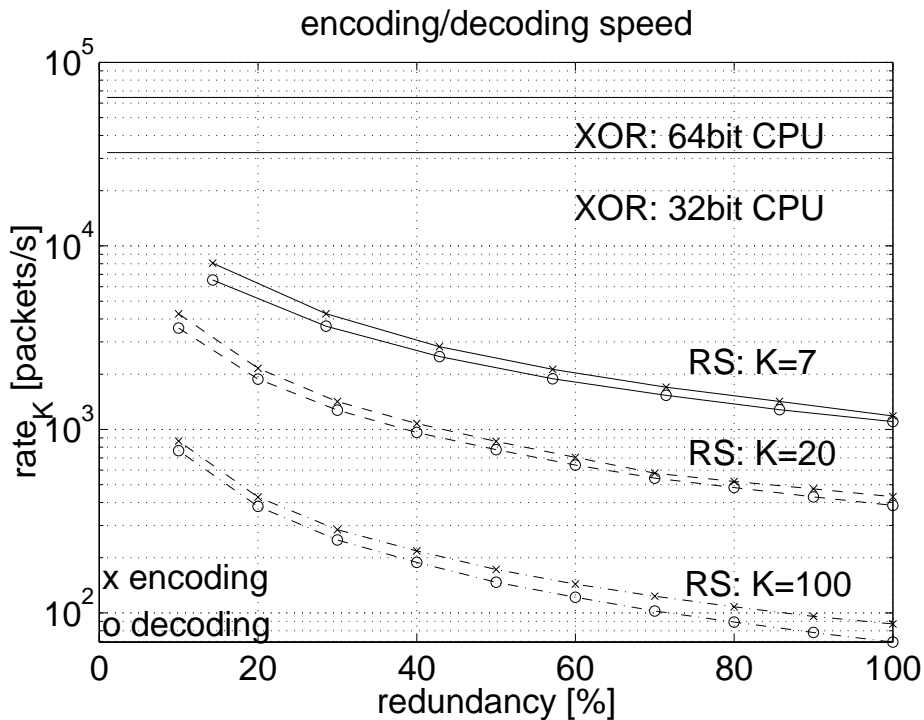
Why FEC for Multicast Error Control?



- A single parity packet can be used by different receivers to repair the loss of different data packets.

FEC: Reed Solomon Coding vs. XOR Coding

Coding and Decoding Speed



ARQ for AV Applications

- Audio transmission (data rates typically 10..64 kbit/s):
 - Unicast interactive voice for small RTT (Slack ARQ by Dempsey and Liebeherr);
 - Non-interactive voice to multiple recipients with large RTT (STORM by Xu, Yavatkar et al.)
Designated receiver (DR) for local recovery.
- Video transmission (data rates typically 100kbit/s ..10 Mbit/s):
 - Challenge (Internet): rate control
 - Potential solution: layering;
 - Receiver-driven layered multicast (RLM, by McCanne et al.).
 - Retransmission-based loss recovery protocol (LVMR by Li, Paul, Ammar).

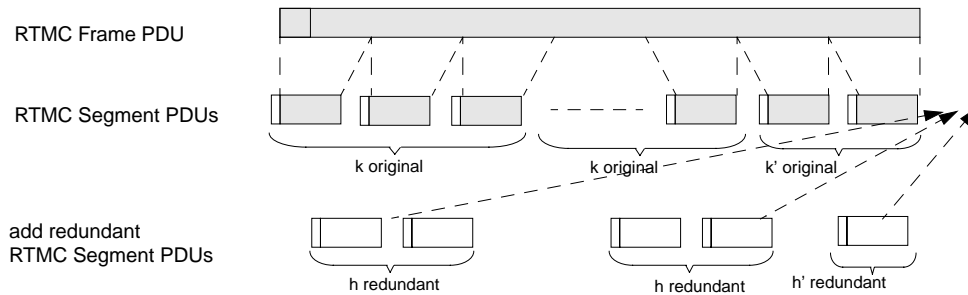
Protocol with ARQ/FEC for audio-visual services

RTMC Protocol Mechanisms

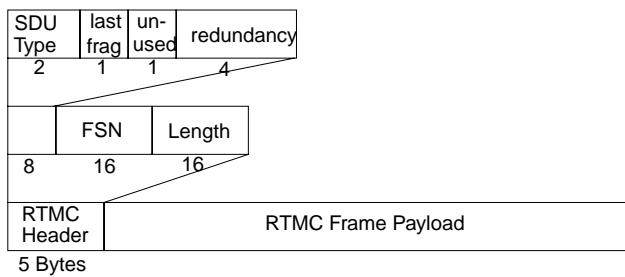
- RTMC: Real-Time MultiCast Protocol with ARQ/FEC for AV services
- Receiver mechanisms:
 - Perform error detection;
 - Attempt error recovery using FEC from first transmission;
 - Send NACK when recovery by FEC failed;
 - Avoid late NACKs based on RTT estimation and SDU relevance interval (MPEG-Frames: relevance within GOP).
- Sender mechanisms:
 - Avoid duplicated retransmission using RTT information;
 - Avoid late retransmission based on maximum playout buffer;
 - Perform reasonable scheduling by rate control for retransmissions.

RTMC PDU Format

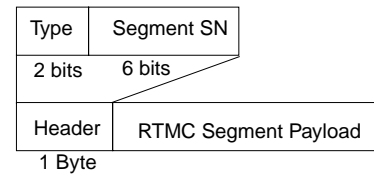
○ Frame and Segment PDUs



RTMC Frame PDU



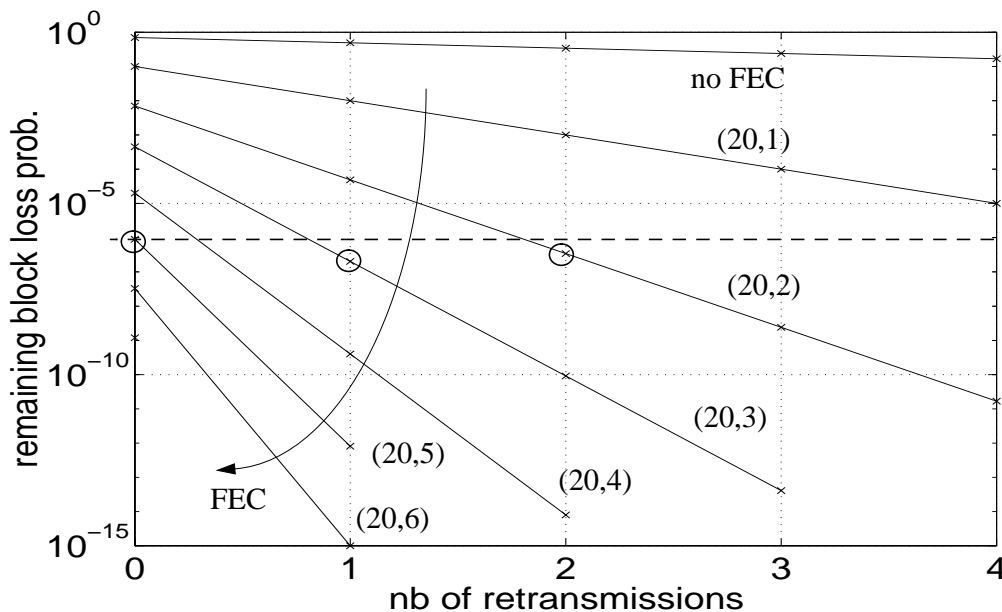
RTMC Segment PDU



Choice and dimensioning of protocol mechanism

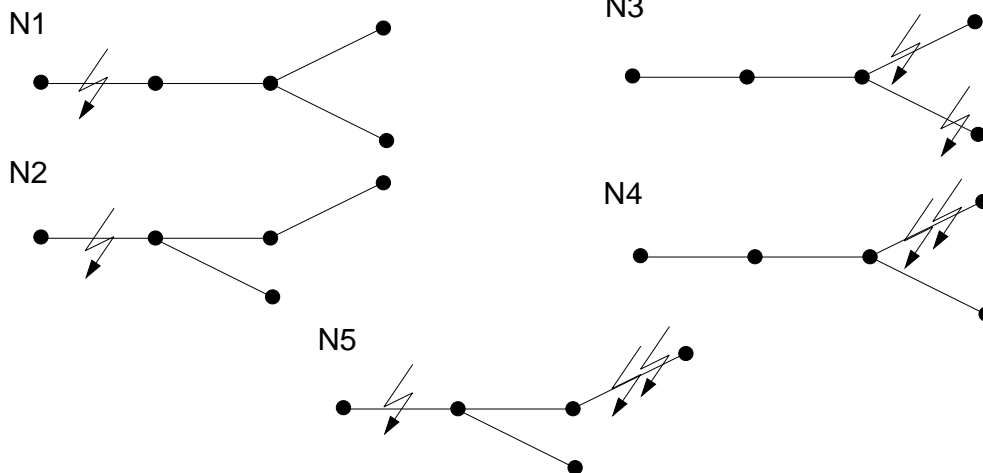
Dimensioning of ARQ/FEC

○ Example: 120 Segment-block, $p = 0.01$, (20,h)-Blocks



Influence of topology: Selected Scenarios for Modeling Heterogeneity

- Loss: on shared links / on individual links;
- Loss: homogeneous/heterogeneous probability;
- RTT: homogeneous/heterogeneous.



Error Control for AV Services

Selected Scenario N2

○ Network Scenario 2:

- losses on shared link
- heterogeneous RTTs.

+ Examples:

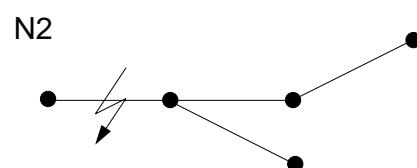
- Different distances to receivers
- Large queueing delays for certain receivers.

+ Problems:

- NACKs for common losses arrive within large time interval;
- FEC has no significant impact on scaling for large groups;
- Local recovery not appropriate.

+ Solutions:

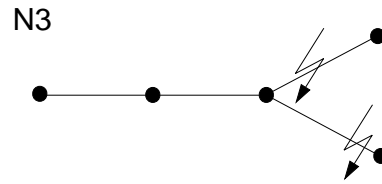
- RTT-aware NACK processing at the transmitter;
- Error detection and NACK processing close to location of error (Group Communication Server).



Selected Scenario N3

○ Network Scenario 3:

- independent losses on individual links
- homogeneous loss probabilities
- homogeneous RTTs.



+ Examples:

- MBONE, with losses mostly occurring in subnets (see measurements by Yajnik, Kurose and Towsley, UMASS)
- Wireless cellular networks, with receivers located in different cells
- Satellite communication with individual losses at downlink.

+ Problem:

- Retransmission of lost PDUs: low efficiency, bad scaling for large groups.

+ Solutions:

- FEC (for first transmission, and for retransmission)
- Local recovery.

Scenario-specific Selection of Mechanisms

○ FEC is of particular benefit in the following scenarios:

- Large groups;
- No feedback;
- Heterogeneous RTTs;
- Limited buffer.

○ ARQ is of particular benefit in the following scenarios:

- Heterogeneous loss;
- Loss in shared links of multicast tree dominates;
- Small groups (Statistic by AT&T: on average < 7 participants in conference);
- Non-interactive applications.

- ARQ by local recovery:

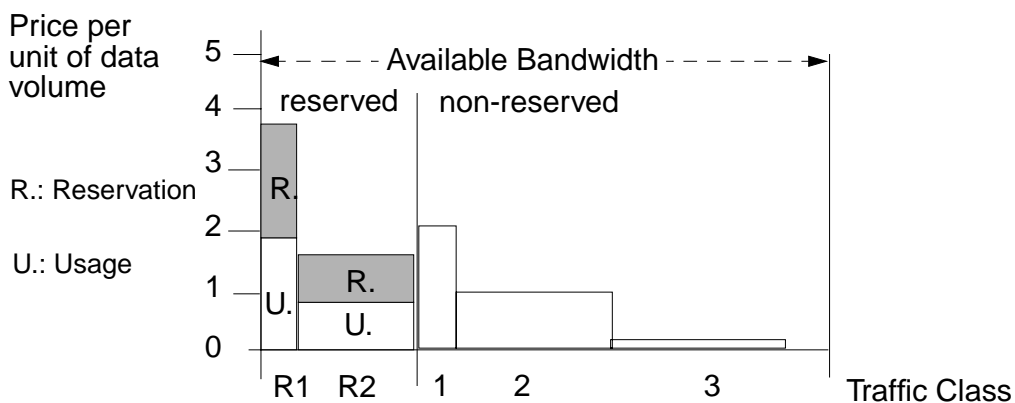
- large groups (good for individual losses, heterogeneous RTT).

Network Support for Priorities

- Multiplexing with priorities for
 - Selective discarding;
 - Selective scheduling.
- Concept: applying prioritized scheduling for recovering from excessively delayed packets:
 - Open-loop error control: Prioritized transmission of *redundancy*;
 - Closed-loop error control: Prioritized NACK-based *retransmission*.
- Prerequisite for recovery from delay errors by priority-based error control:
 - Cooperative applications which use high priority only when needed; or
 - Priority-based charging scheme.

Priority-based Charging

- 2-Priority Scheme:
2nd Class: ordinary best-effort service; 1st Class: low delay.
- 5-Priority Scheme:
3rd Class; 2nd Class, 1st Class, reserved 2nd and 1st Class.



Conclusions

- XOR-based FEC may outperform FEC with Reed-Solomon-Codes;
- ARQ frequently *can* be applied for AV real-time applications;
- ARQ adapted for real-time in combination with FEC is very promising;
- Different network scenarios (reflecting topology and loss / RTT correlation) with several network parameters (loss rate, RTT, ...) and different application scenarios (single/multiple priorities etc...) with several application parameters (delay budget, data rate, ...)
 - ↳ Selection and dimensioning of protocol mechanisms is highly challenging task
- When network elements support different priorities, ARQ and FEC allow for recovery from excessive delays;
- Support of priorities requires appropriate charging scheme.