Meeting Reliability and Real-Time Demands in Wireless Industrial Communication

Magnus Jonsson and Kristina Kunert

CERES –
Centre for Research on Embedded Systems
Halmstad University, Sweden

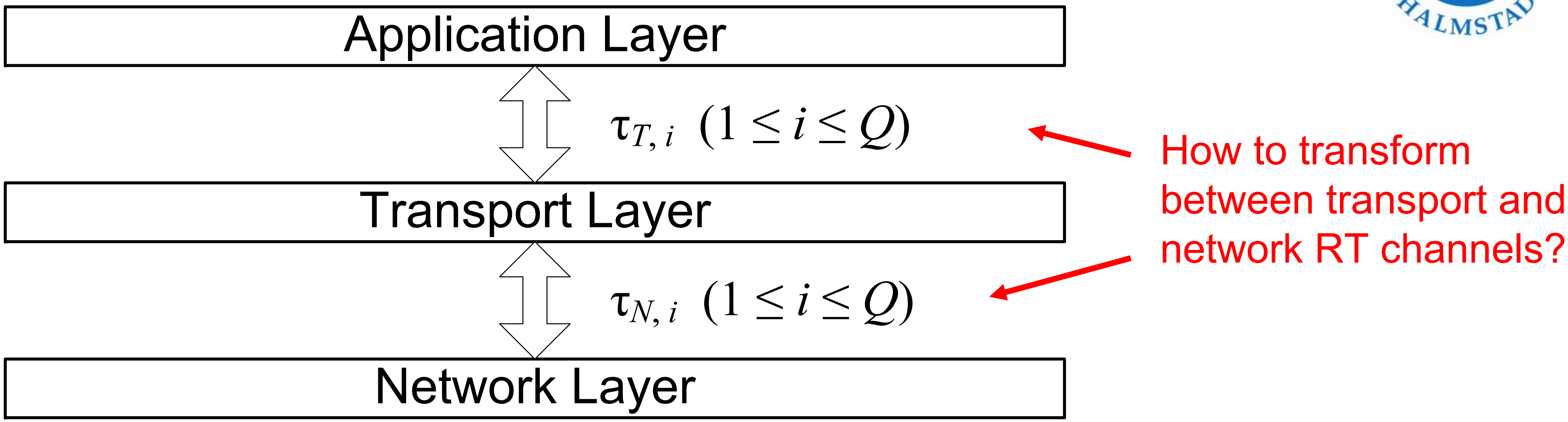
Motivation

- **Reliability in combination with real-time performance**
  - Has almost only been addressed for safety-critical systems, where a lot of redundancy is added
- **Increased reliability without additional hardware**
  - In this way, the reliability of future products with tough **timing constraints** can be improved at **minimal cost**
- **Application examples:**
  - Distributed automation systems
  - Elderly care products and surveillance applications using wireless sensor networks
  - Radio base stations
  - Radar signal processing systems
  - Multimedia communication
- **Wireless real-time communication**
  - Especially important with improved reliability

General Approach

- Develop a framework with
  - Communication methods and protocols
  - Real-time analysis
- Handle retransmissions of erroneous data packets, not violating the timing requirements of other packets

Transport Layer with ARQ Supporting Logical Real-Time Channels



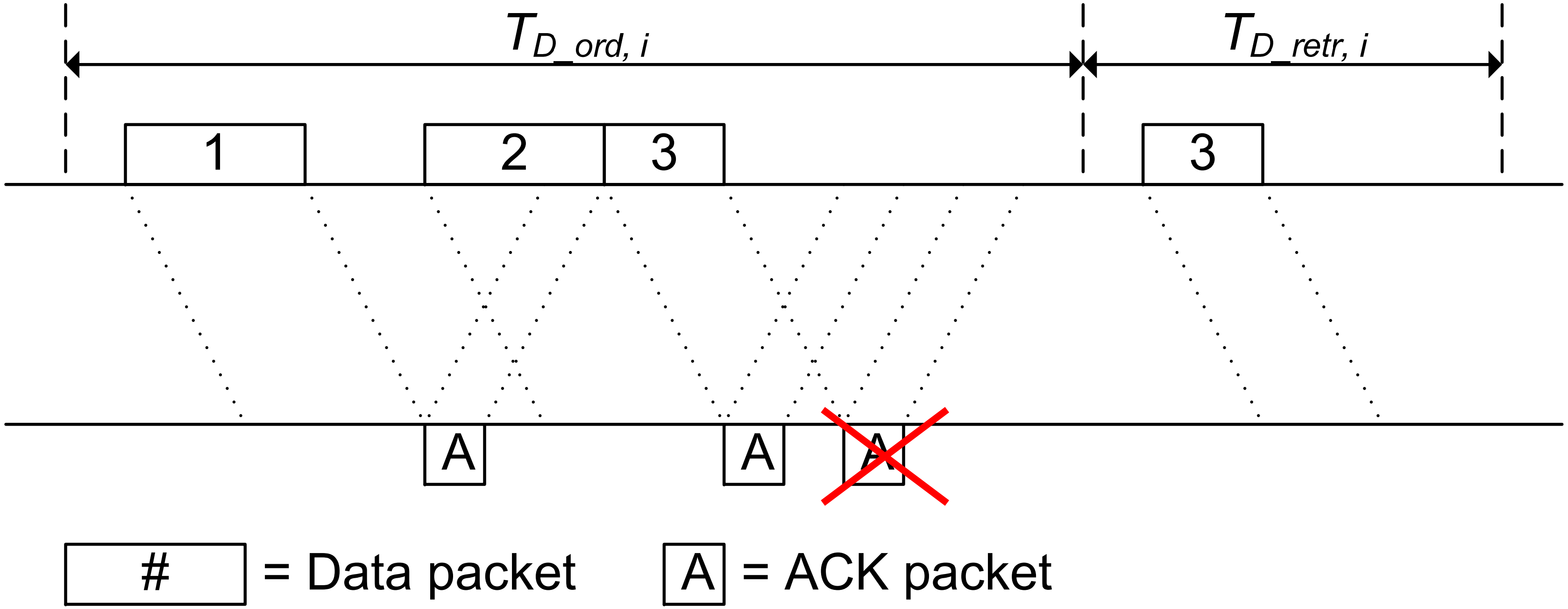
- The transport layer support the application layer with a "reliable" service through the concept of RT channels:

$$\tau_{T,\,i} = \{m_{s,i},\, m_{d,i},\, P_{T,i},\, L_{T,i},\, D_{T,i}\}$$

- Each RT channel has a corresponding network-layer RT channel with guaranteed but unreliable performance

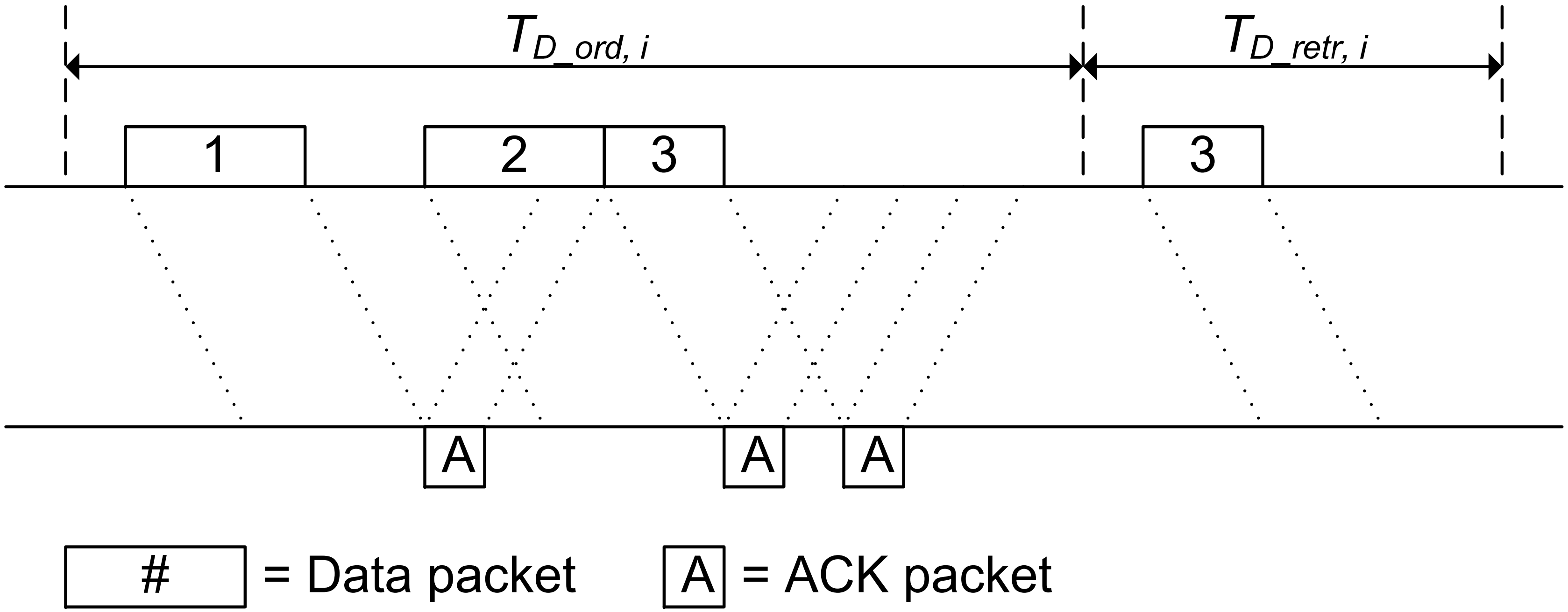
Real-Time ARQ for a Point-to-Point Link with Earliest Deadline First (EDF) Packet Scheduling

ARQ – Splitting the Delay Bound



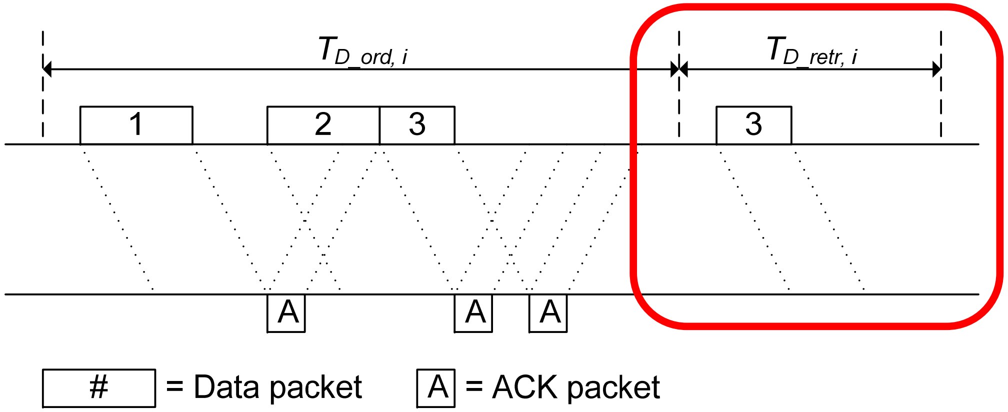
- Q transport layer RT channels: $\tau_{T,i} = \{P_{T,i}, L_{T,i}, D_{T,i}\}$, $1 \le i \le Q$
- Split the delay bound into one for ordinary transmissions and one for possible retransmissions: $D_{T,i} = T_{D\_ord,i} + T_{D\_retr,i}$

Guarantees



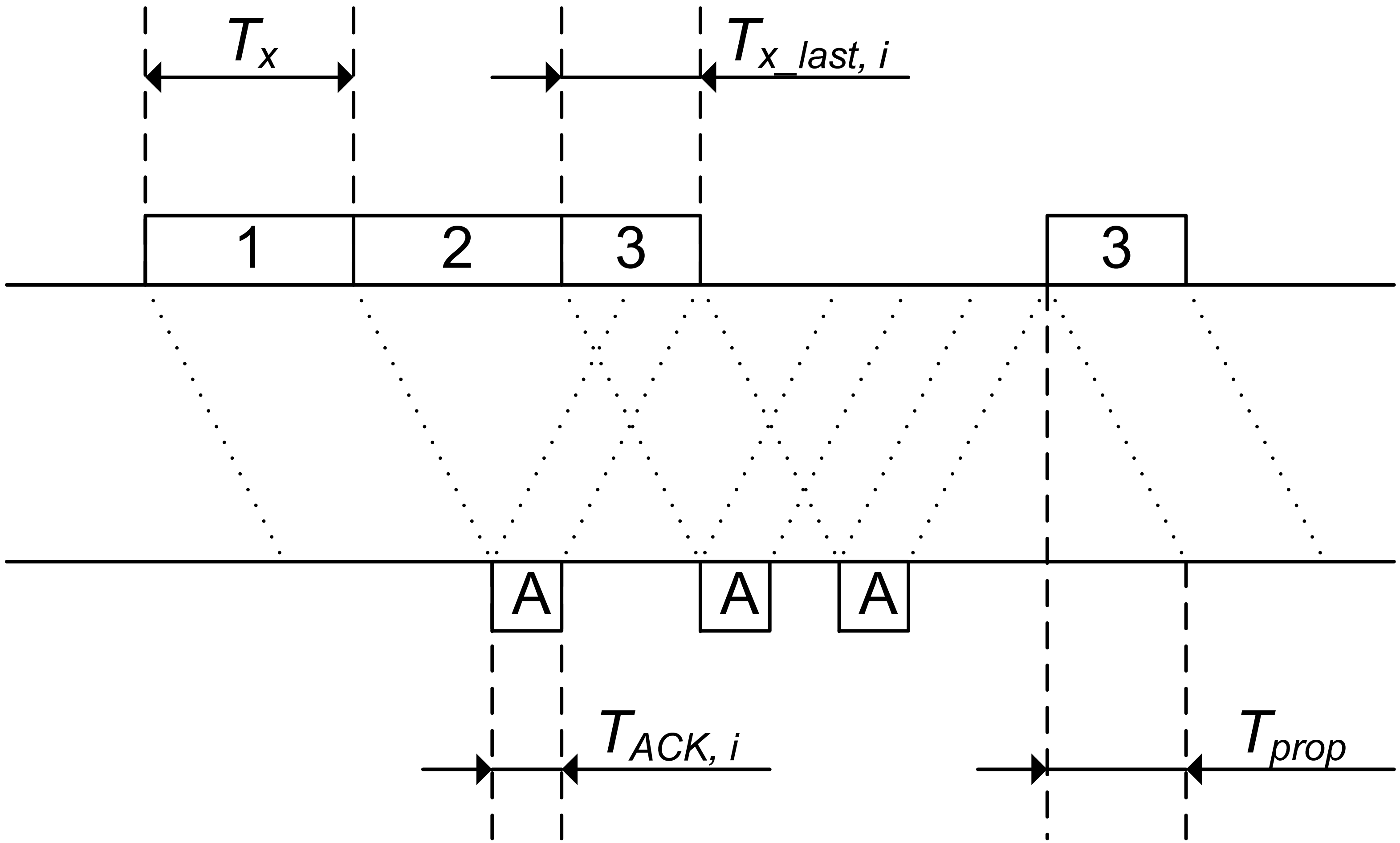
- Guarantee to meet the delay bounds for all ordinary transmissions
- One or several **extra logical RT channels** in the network layer are used **for retransmissions** (shared by all normal RT channels)
- Only allow retransmission if the retransmitted packet will arrive in time

Retransmission RT Channel



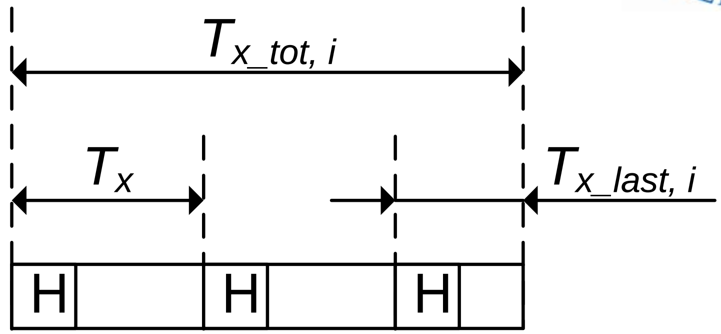
- Dedicated RT channels for retransmissions
  - $\tau_{retr,i} = \{P_{retr,i}, L_{retr,i}, D_{retr,i}\}$
- Guarantee retransmission of one packet every period of $P_{retr,i}$ from any ordinary RT channel
  - $L_{retr,i}$ is set to the maximum sized packet
  - $D_{retr,i}$ is set to D_{retr}, which is a system parameter with which we can set the time allocated for possible retransmissions
- Several retransmission channels can be setup

Timing Analysis – Introduction



Timing Analysis – From Amount of Pure Data to Total Message Transmission Time

$L_{T,i}$ Amount of pure data per message of τ_i
L_{data} Maximum amount of data per packet
L_{header} Header length
R Bit rate of the physical link



Number of packets per message:

$$N_{pack,i} = \left\lceil \frac{L_{T,i}}{L_{data}} \right\rceil$$

Number of maximum sized packets:

$$N_{pack\_max,i} = N_{pack,i} - \left(\left\lceil \frac{L_{T,i}}{L_{data}} \right\rceil - \left\lfloor \frac{L_{T,i}}{L_{data}} \right\rfloor \right)$$

Length of the last packet of a message if being shorter than L_{pack} (otherwise it is zero):

$$L_{last,i} = \left(N_{pack,i} - N_{pack\_max,i} \right) \cdot \left(L_{T,i} - N_{pack\_max,i} L_{data} + L_{header} \right)$$

The transmission time of a full-length packet is:

$$T_x = \frac{L_{pack}}{R}$$

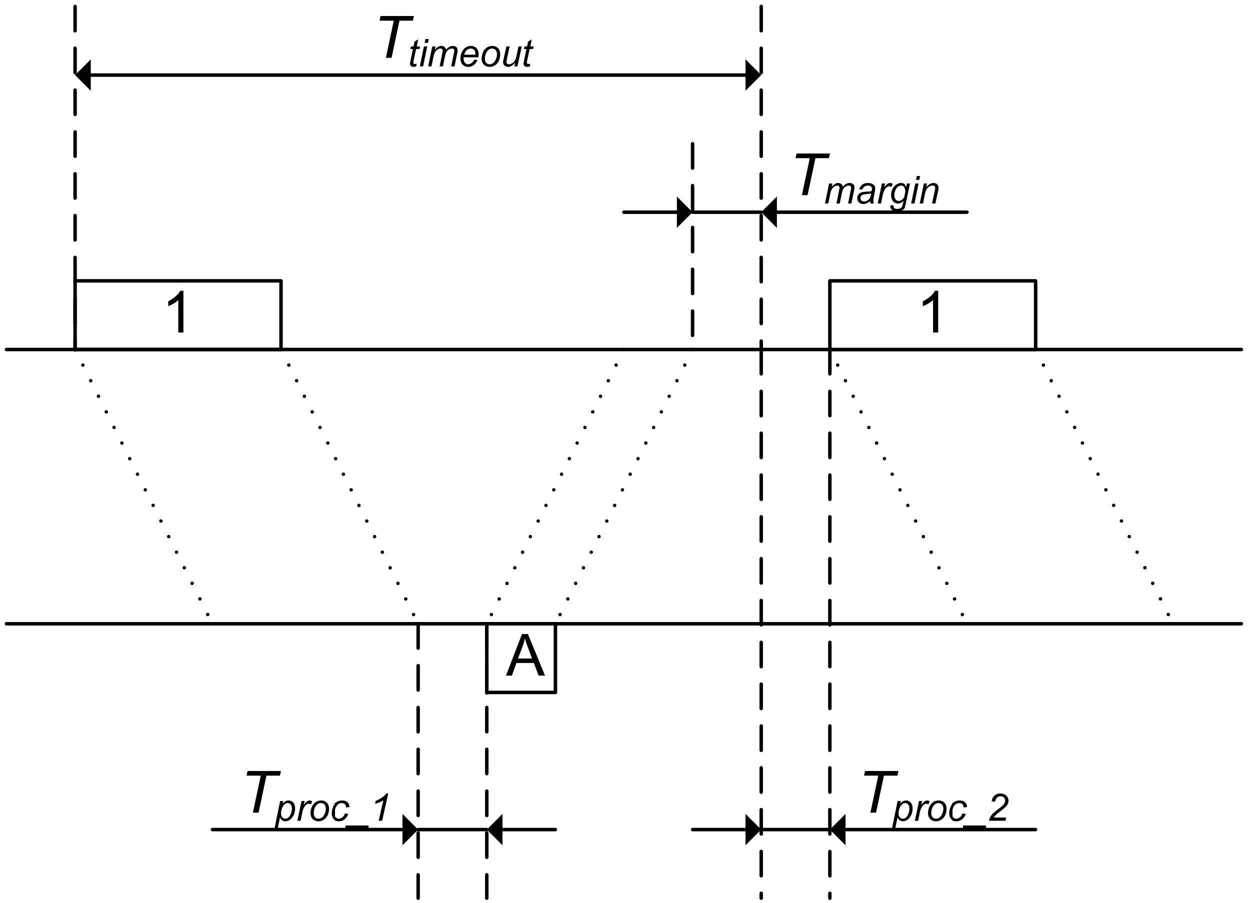
If the last packet is shorter than L_{pack}, the transmission time of this packet is:

$$T_{x\_last,i} = \frac{L_{last,i}}{R}$$

The total transmission time of a message is:

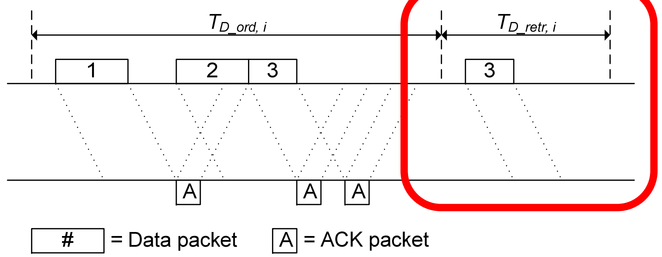
$$T_{x\_tot,i} = \frac{N_{pack\_max,i} L_{pack} + L_{last,i}}{R}$$

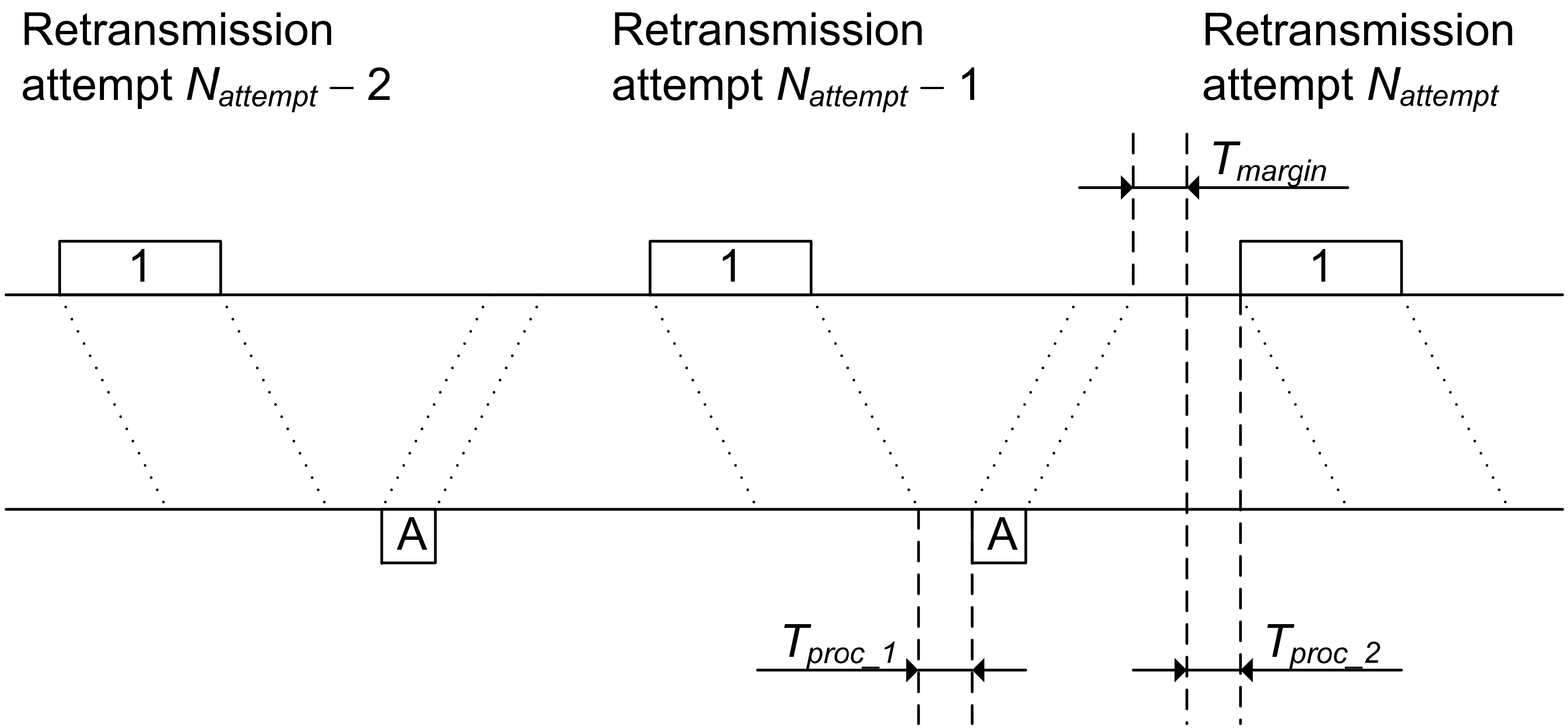
Timing Analysis (cont.)



- Retransmission timer: $T_{timeout} = T_{D\_ord,i} - T_{proc\_2}$
- All packets of a message timeout at the same time

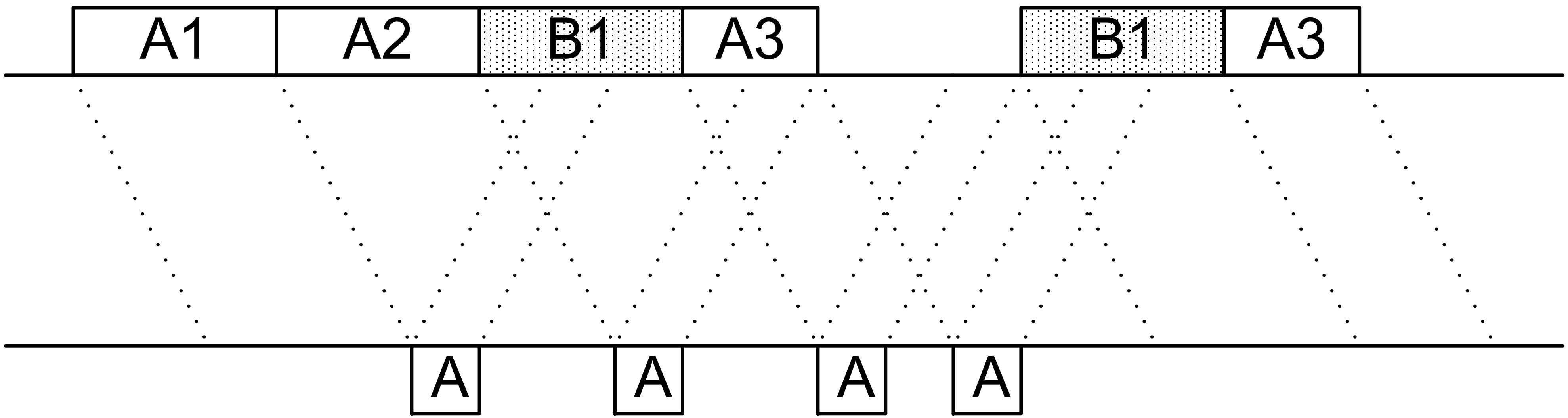
Supporting Several Retransmission Attempts





- $N_{attempt}$ retransmission attempts are supported
- The last retransmission attempt has less delay components since it is not acknowledged

Real-Time Scheduling Analysis



- The EDF queuing delay must be analyzed

Real-Time Scheduling Analysis (cont.)

- The pure scheduling (queuing) deadline need to be extracted by subtracting other delays:

$$T_{d\_ord,i} = T_{D\_ord,i} - 2 \cdot T_{prop} - T_{proc1} - T_{proc2} - T_{margin} - 3 \cdot T_x$$

  - One T_x is the worst-case blocking time due to non-preemptive transmission of lower-priority (longer deadline) packet
  - One T_x is the blocking delay for piggyback acknowledge
  - One T_x is the piggyback acknowledge transmission time

Real-Time Scheduling Analysis for Retransmission RT Channels

- The pure scheduling (queuing) deadline need to be extracted by subtracting other delays:

$$T_{d\_retr,i} = \frac{D_{retr} - T_{prop} - T_x - (N_{attempt} - 1) \cdot T_{retr\_const}}{N_{attempt}}$$

$$T_{retr\_const} = 2 \cdot T_{prop} + T_{proc1} + T_{proc2} + T_{margin} + 3 \cdot T_x$$

- As mentioned, the last retransmission attempt has less delay components since it is not acknowledged

Real-Time Scheduling Analysis

- Utilization check where *U* must be less than 1:

$$U = \sum_{i=1}^{Q} \left(\frac{T_{x\_tot,i}}{P_{T,i}} \right) + \sum_{i=1}^{M} \left(\frac{T_{x\_retr,i}}{P_{retr,i}} \right)$$

M = Number of retransmission RT channels

- Delay bound check where $h(t) \leq t \quad \forall t$:

$$h(t) = \sum_{\substack{i \in [1,Q], \\ T_{d\_ord,i} \leq t}} \left(1 + \left\lfloor \frac{t - T_{d\_ord,i}}{P_{T,i}} \right\rfloor \right) \cdot T_{x\_tot,i} + \sum_{\substack{i \in [1,M], \\ T_{d\_retr,i} \leq t}} \left(1 + \left\lfloor \frac{t - T_{d\_retr,i}}{P_{retr,i}} \right\rfloor \right) \cdot T_{x\_retr,i}$$

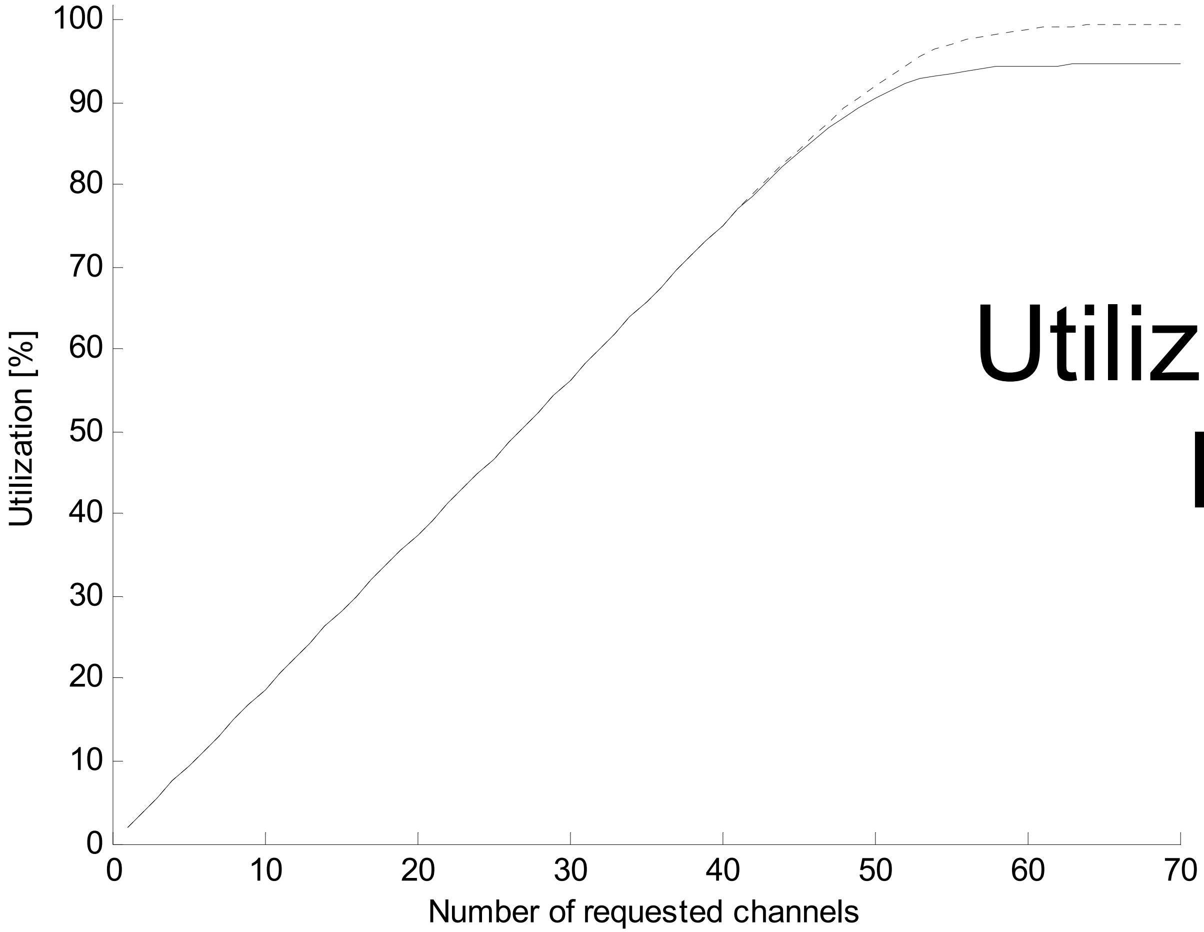
Simulation Implementations

- Admission control simulation
  - Generate traffic flows (logical real-time channels) one at a time
  - Perform real-time analysis to see whether the requirements for all real-time channels can be met
- Packet-level simulation
  - Simulate how messages/packets are sent over time and how some are “lost” (contain bit errors)
  - Traffic is generated using the admission control simulation

Traffic Parameters

Traffic class	*P*	*D*	*L*
1	2 ms	2 ms	4 000 bits
2	4 ms	4 ms	4 000 bits
3	8 ms	8 ms	4 000 bits
4	1 6 ms	1 6 ms	4 000 bits

- 50 Mbit/s, L_{pack} = 1 000 bits, $T_{prop} = 1\ \mu s$ (≈200 m)
- T_{proc1}, T_{proc2} and T_{margin} are assumed to be negligible
- BER is varied between 10^{-6} and 10^{-4}
- Experienced BER above possible forward error correction

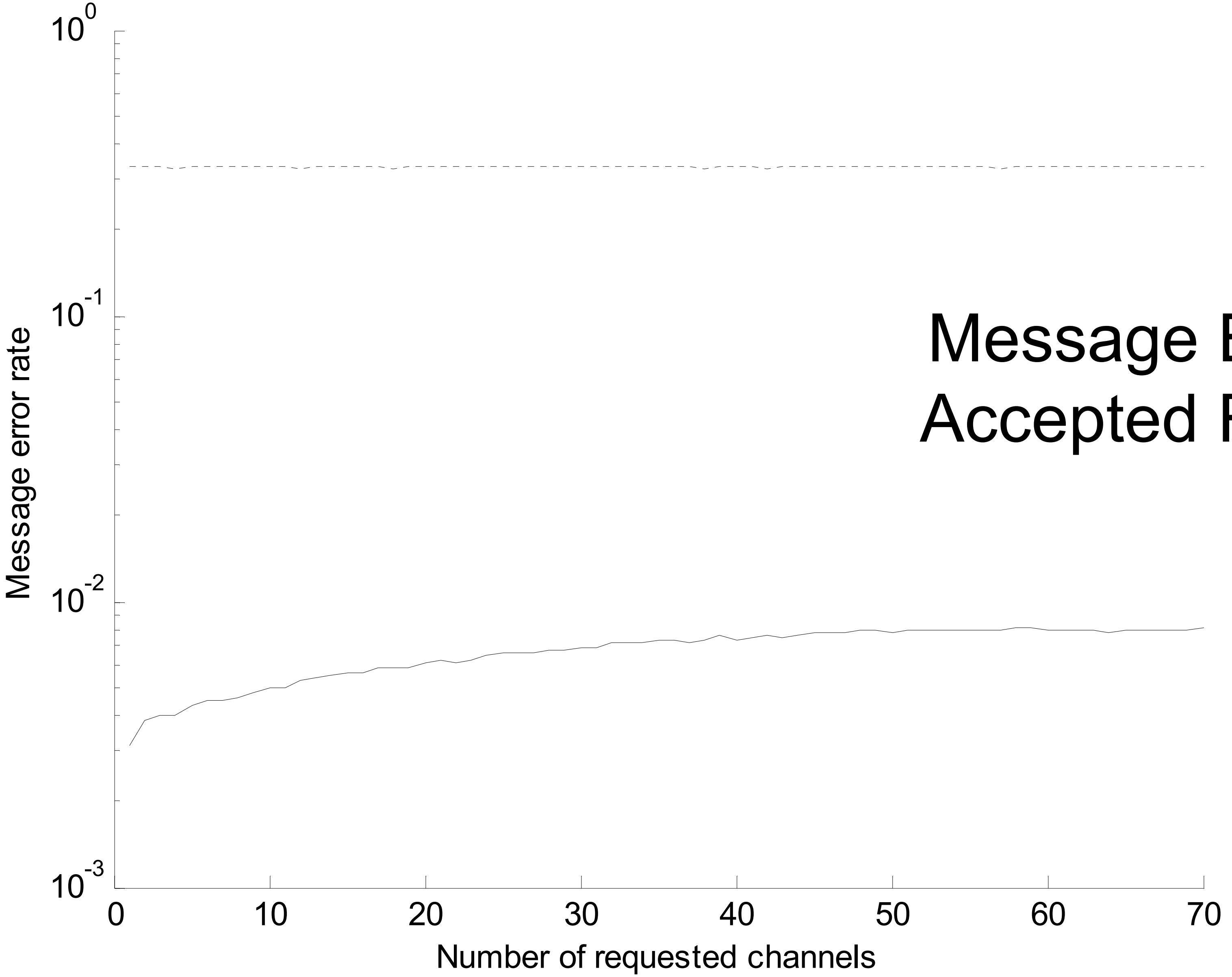


Utilization of Accepted RT Channels

- Dashed line: without ARQ Solid line: with ARQ
- Conclusion: ARQ only requires small overhead

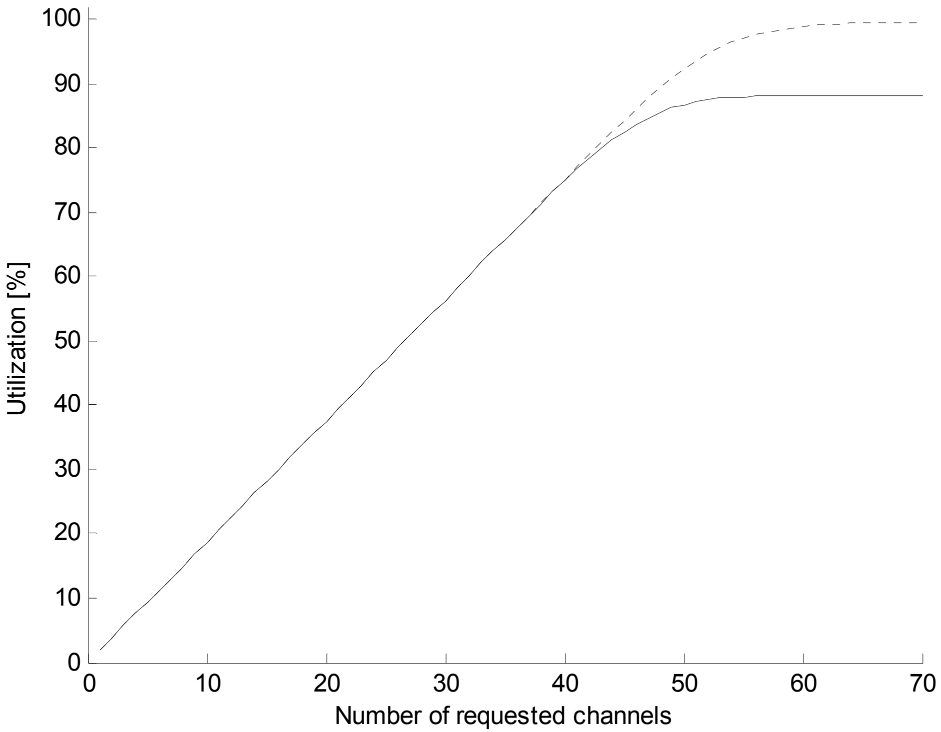
$N_{attempt}$ = 2 retransmission attempts. M = 4 retransmission channels, each with the parameters $P_{retr,i}$ = 2 000 µs, $D_{retr,i}$ = 617 µs, and $L_{retr,i}$ = 1 000 bits. The bit error rate was set to BER = 10^{-4}

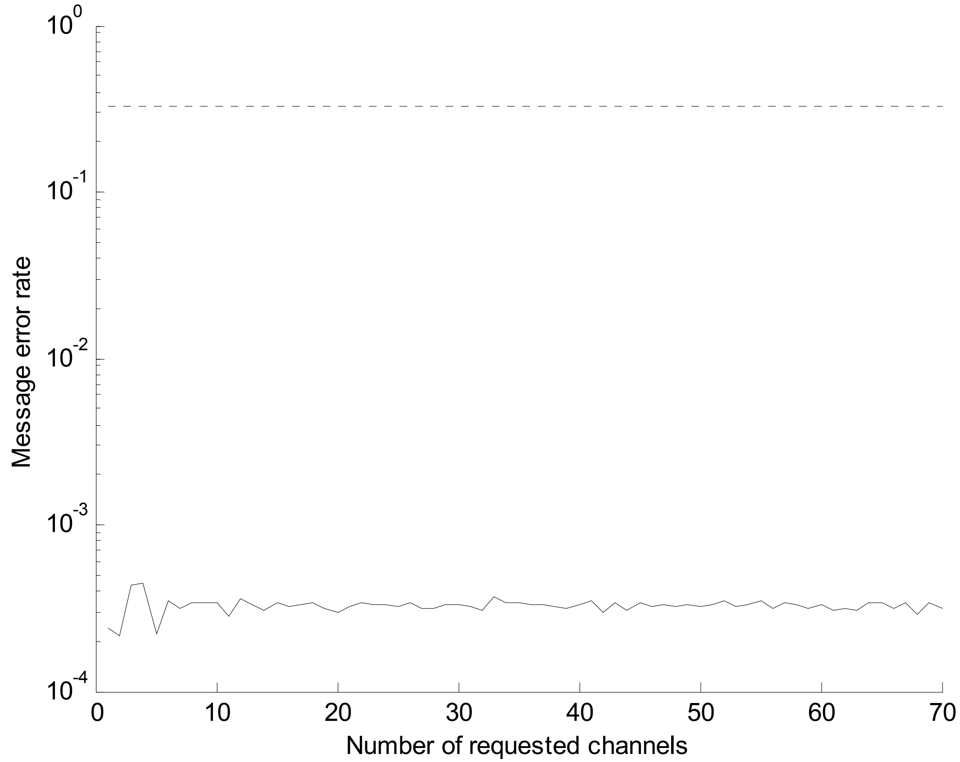
Message Error Rate of Accepted RT Channels



Performance Gain

- With a bit higher bandwidth penalty, the message error rate can be reduced by several orders of magnitude



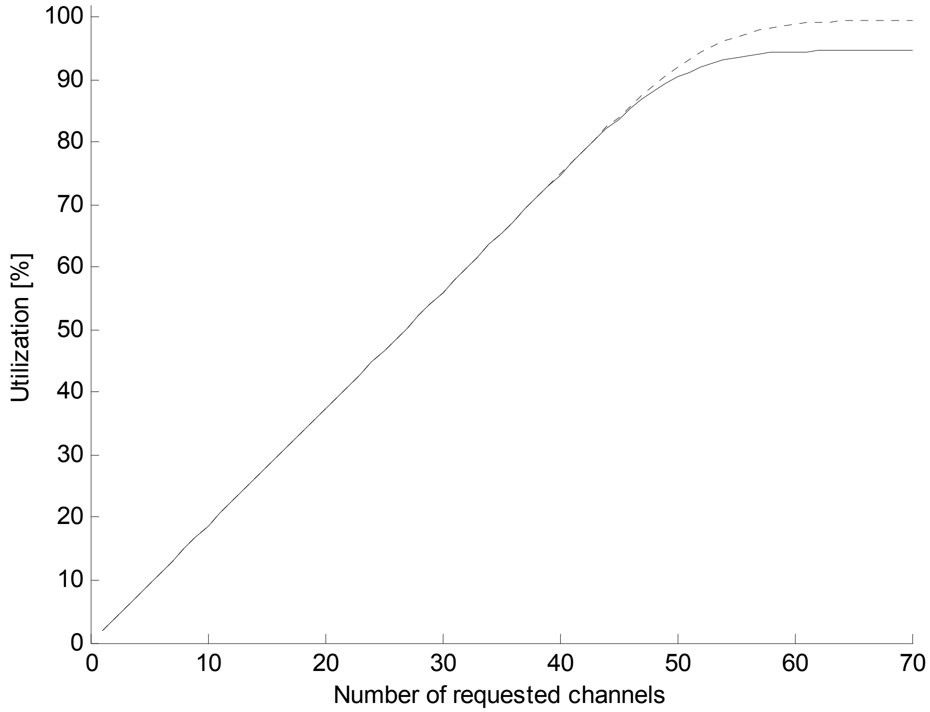


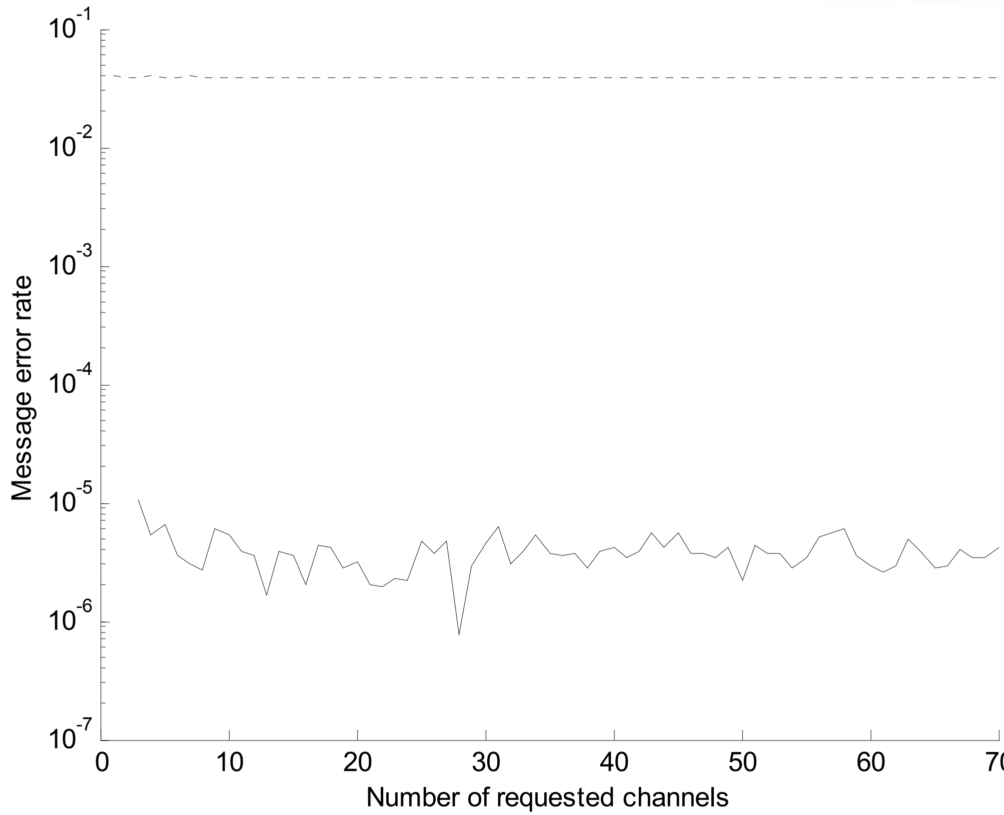
$N_{attempt}$ = 3 retransmission attempts. M = 10 retransmission channels, each with the parameters $P_{retr,i}$ = 2 000 μs, $D_{retr,i}$ = 910 μs, and $L_{retr,i}$ = 1 000 bits. The bit error rate was set to BER = 10^{-4}

CERES
CENTRE FOR RESEARCH ON EMBEDDED SYSTEMS

I SAMARBETE MED KK·stiftelsen

Case with Lower BER





$N_{attempt}$ = 2 retransmission attempts. M = 4 retransmission channels, each with the parameters $P_{retr,i}$ = 2 000 μs, $D_{retr,i}$ = 617 μs, and $L_{retr,i}$ = 1 000 bits. The bit error rate was set to BER = 10^{-5}

Summary

- A reduction of the message error rate by several orders of magnitude is possible with a reasonable utilization penalty
- All real-time requirements of ordinary transmissions are guaranteed to be met
- More equations needed, e.g. for run-time implementation, can be found in our papers together with details needed for different specific situations
  - SIES 2008
  - ETFA 2008
  - IEEE Transactions on Industrial Informatics
  - Book chapter in *Factory Automation*, IN-TECH