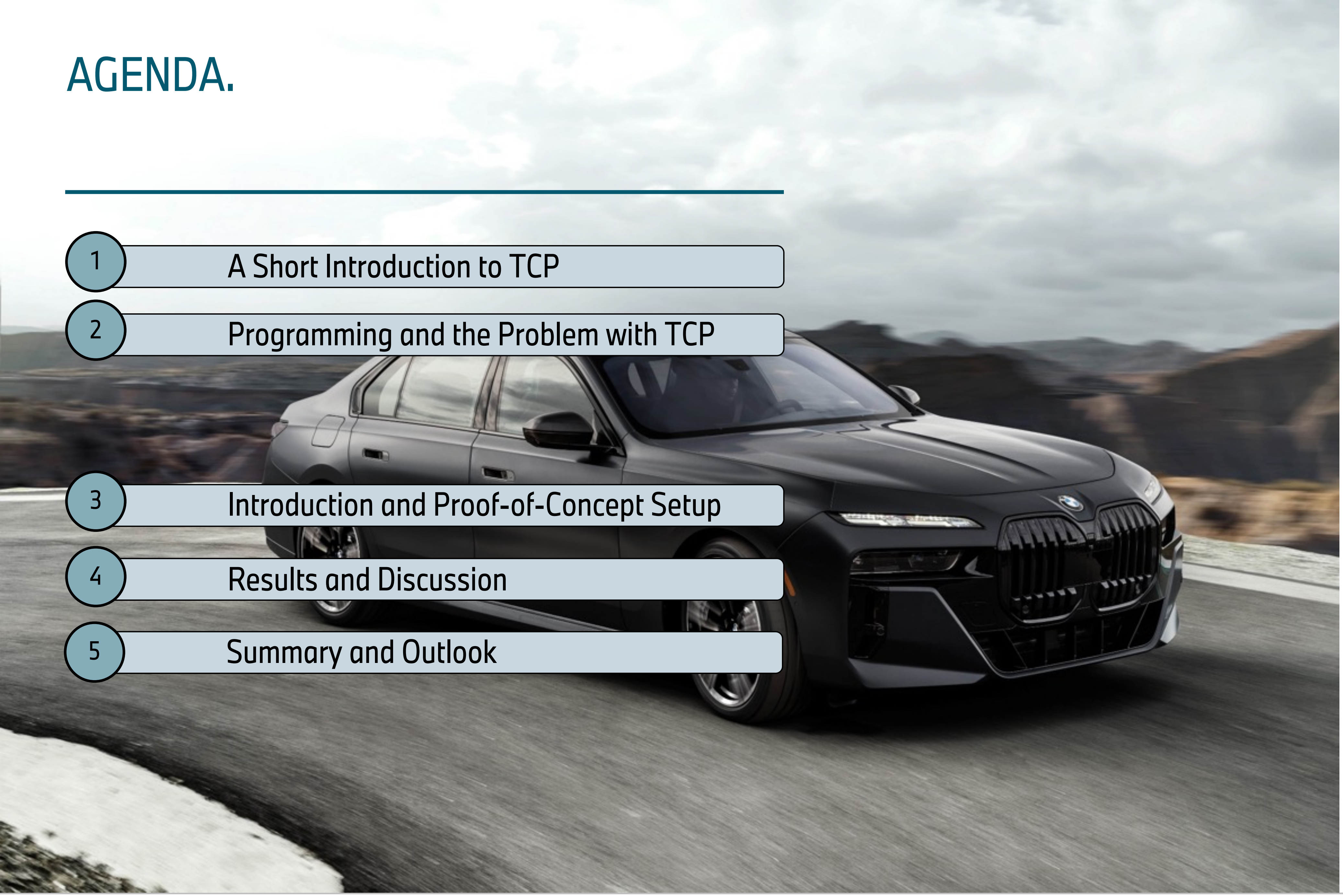
TCP AND AUTOMOTIVE ETHERNET –
HOW TO RESOLVE THE EVERLASTING STRUGGLE.

KARL BUDWEISER (BMW AG)
IAGO ÁLVAREZ, STEFAN LACHNER, LARS VÖLKER (TECHNICA ENGINEERING GMBH)

ETHERNET & IP @ AUTOMOTIVE TECHNOLOGY DAY

AGENDA.



1. A Short Introduction to TCP
2. Programming and the Problem with TCP
3. Introduction and Proof-of-Concept Setup
4. Results and Discussion
5. Summary and Outlook

TCP AND AUTOMOTIVE ETHERNET

A short Introduction to TCP

Ethernet & IP @ Automotive Technology Day

A SHORT INTRODUCTION TO TCP. FUNDAMENTALS OF TCP.

BMW GROUP

Connection oriented (end-to-end) protocol

- One sender, one receiver, bidirectional
- "Three way Handshake" to establish necessary states

Segmentation

- Packetization & reassembly of stream data in tx/rx buffer

Data Streaming

- Reliable, in-order delivery of packets, without duplications or loss
  - → Acknowledgements, sequence IDs, timeouts and clever retries

Pipelining

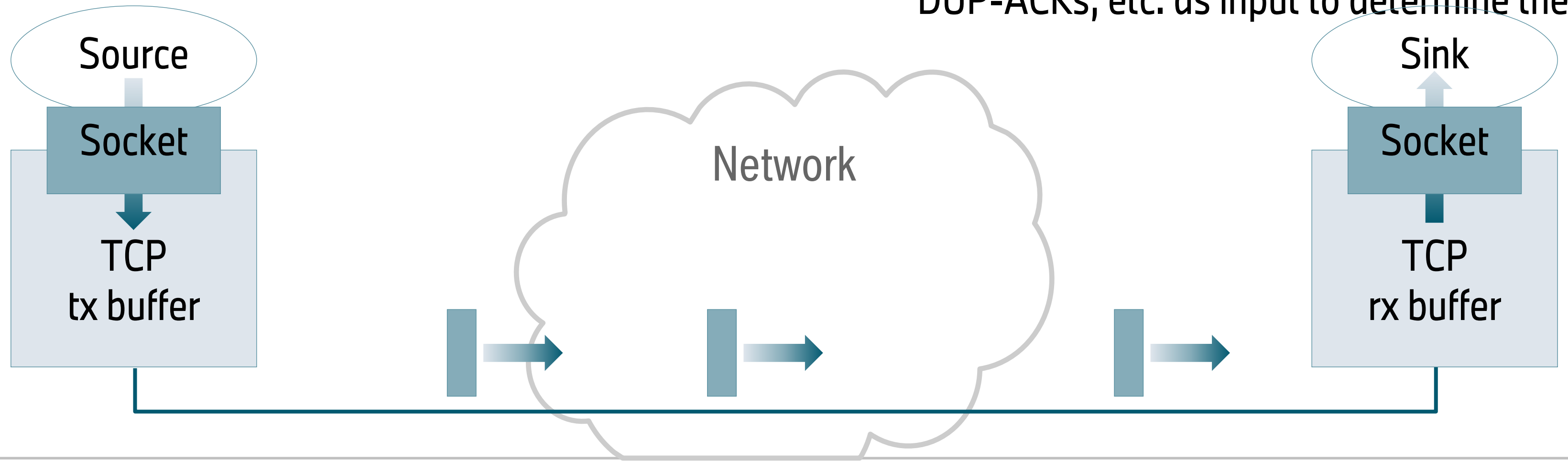
- Permission for parallel packet transmission

Flow Control

- Control sending rate to prevent sender from overwhelming receiver
  - → Sliding window protocol that synchronizes to "receive window"

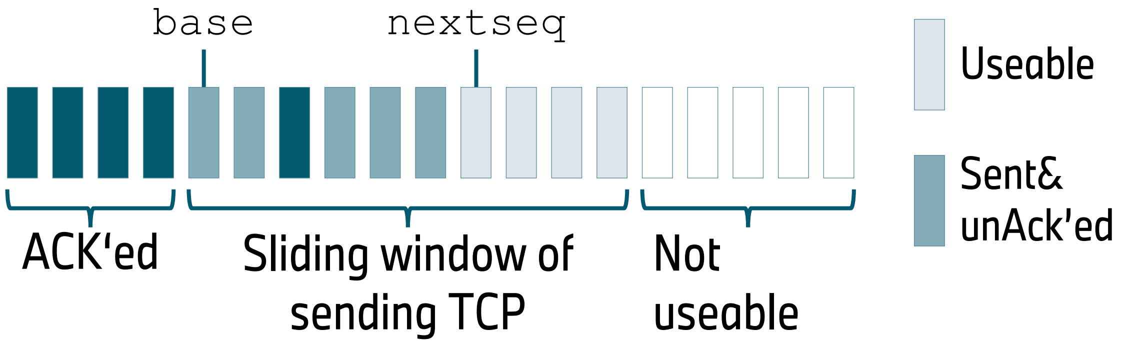
Congestion Control

- Avoid packet loss due to network congestion
  - → Adaptive algorithm that takes timeouts, timing variations, DUP-ACKs, etc. as input to determine the congestion window



A SHORT INTRODUCTION TO TCP.
FLOW CONTROL AND THE TCP SLIDING WINDOW PROTOCOL.

Basic principles of the TCP sliding window protocol:



- **Pipelining:** send window allows more than one "packet in-flight"
- **Flow Control:** continuous synchronization of send window to receive window using the `RcvWindow` field to slow down sender
- ACKs acknowledge all previous packets incl. the last received one according to the **acknowledgment policy**
  - DUP-ACKs hinting towards packet loss are possible
    → 3 DUP-ACKs: **Fast Retransmit** of oldest unACK'ed packet
- Sender has a timer for each unACK'ed packet
  → **RTO-Timeout:** unACK'ed packet is retransmitted

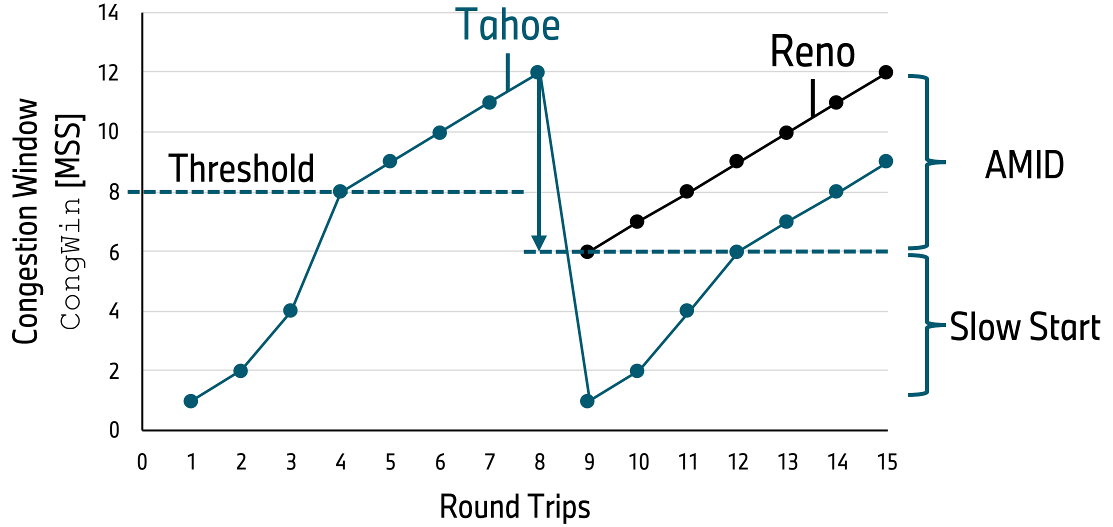
Acknowledgement policy:

Event	Action of receiving TCP
Reception of in-order segment with expected sequence number	Initiate **Delayed ACK** and wait 500ms for other in-order segment(s), otherwise send ACK
Reception of another in-order segment with `seq=nextseq`, while delayed ACK is active	Delayed ACK mechanism is still active
Reception of an out-of-order segment with `seq>nextseq`	Gap in data stream! Send DUP-ACK acknowledging again next expected segment (`nextseq`)
Reception of an in-order segment that fills a gap in the data stream	Send a cumulative ACK for all segments received w/o gap, provided that `seq=nextseq`

A SHORT INTRODUCTION TO TCP.
PRINCIPLES OF TCP CONGESTION CONTROL.

BMW GROUP

TCP Congestion Control: dynamic sending-side **fine-tuning of the send window** based on direct and indirect feedback from both network (timeouts) and receiver (DUP-ACKs) to avoid network overload scenarios.



- **Slow Start:** `CongWin++` for each ACK → exponential increase of congestion window until `Threshold` is reached
- **AIMD:** Additive Increase Multiplicative Decrease approach to be more careful&fair while searching for further bandwidth
  - Additive Increase: Increase `CongWin` by 1MSS per RTT (Round Trip Time) till an incident is detected
  - Multiplicative Decrease: Decrease `CongWin` and set `Threshold = CongWin/2`, when an incident is detected
- **Reaction to incidents:**
  - → 3 DUP-ACKs: **Fast Recovery** possible (see Reno) by setting `CongWin = CongWin/2`, as network is still able to deliver data
  - → Timeout: Set `CongWin = 1` and begin Slow Start

TCP AND AUTOMOTIVE ETHERNET

Programming and the Problem with TCP

Ethernet & IP @ Automotive Technology Day

VEHICLE PROGRAMMING AND THE PROBLEM WITH TCP. COMMONALITIES IN OBD AND OTA PROGRAMMING.

BMW GROUP

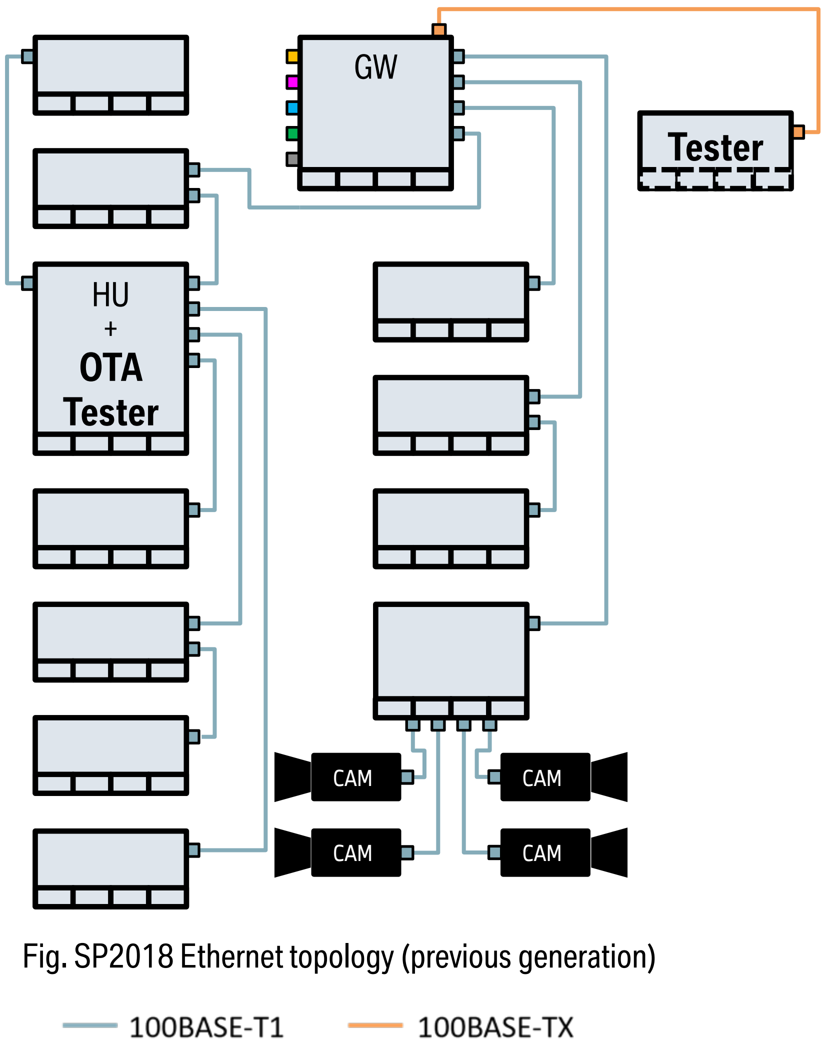


Fig. SP2018 Ethernet topology (previous generation)

Regular OBD programming procedure:

- Activation pins in OBD port of GW ECU are bridged physically:
  - → GW relays `StatusOBD` in SI OBD to signal Ethernet ECUs presence of an (external) tester
  - → Ethernet ECUs adapt to the external tester network by acquiring an (external) IP address
- GW announces IP addresses of Ethernet ECUs to the tester so that the acquired IP addresses can be linked with a VIN
- After changing the "Default Diagnostic Bus" from internal to external the tester can program Ethernet ECUs directly

Basic OTA programming procedure:

- OTA Tester requests form GW to mimic presence of an external tester by updating `StatusOBD`
  - → Regular OBD programming is started

VEHICLE PROGRAMMING AND THE PROBLEM WITH TCP.
TYPICAL LIMITATIONS OF EMBEDDED TCP STACKS.

No.	Time [s]	Delta [s]	Src	Dest	Protocol	Seq	Lenght	Seq.Next	ACK	Expert
9787	06.376114	--	HU	GW	TCP	60726	1460	**62186**	287	Bulk Transfer
9788	06.376659	0.000545	GW	HU	TCP	287	0	287	**62186**	ACK for Bulk Transfer
9789	06.377139	0.000480	HU	GW	TCP	63646	1460	65106	287	[Loss of segment 62186]
9790	06.377761	0.000622	GW	HU	TCP	287	0	287	62186	[Dup ACK 9788#1]
9791	06.419921	0.042160	GW	HU	DoIP	287	20	307	62186	Multiple payload, Delay!
9792	06.420241	0.000320	HU	GW	TCP	65106	0	65106	307	ACK & window probe
9793	06.420778	0.000537	GW	HU	TCP	307	65106	307	62186	[Dup ACK 9788#2]
(...)										
9810	06.581574	--	HU	GW	TCP	62186	1460	63646	387	[TCP Retransmission]
9811	0.6582413	0.000939	GW	HU	TCP	387	0	387	63646	[No Out-of-Order Support]

Findings in trace:

- tcp.analysis.zero_window and small `RcvWindow = 2MSS`
  → Indication of inefficient bulk transfer towards GW
- tcp.analysis.lost_segment shows packet loss
- >1 UDS messages packetized although `NODELAY = ON`

General Root Causes:

- Dependencies between UDS and TCP due to general protocol design issues: `UDS_Timeouts << RTO`
- No 3 DUP-ACK due to small `RcvWindow`, if GW isn't sending
- Retransmission chains due to missing OoO support

VEHICLE PROGRAMMING AND THE PROBLEM WITH TCP. ARCHITECTURAL LESSONS LEARNED.

BMW GROUP

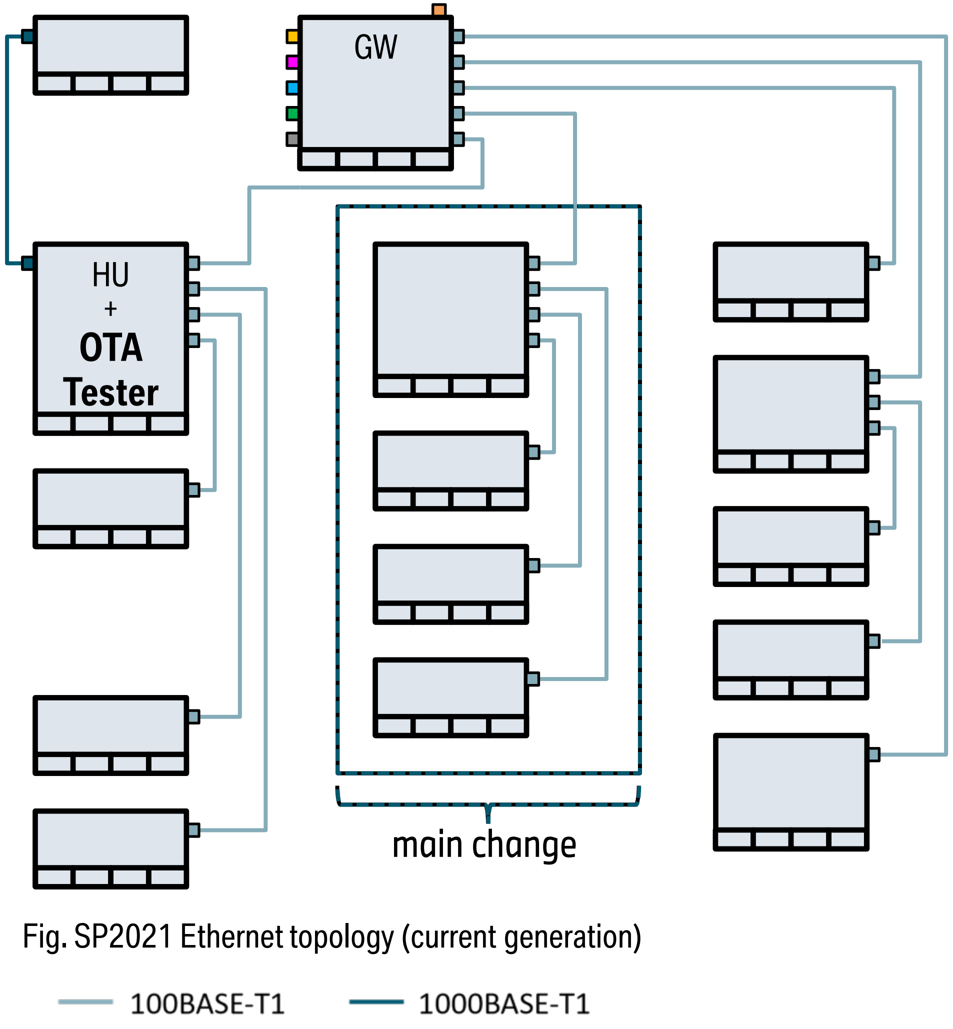


Fig. SP2021 Ethernet topology (current generation)

Architectural differences in OTA and their implications:

- Flash sequence optimized for diagnostic-access via GW
  - Packets are distributed less evenly across ports/links, when partitioning tester elsewhere
- Switches w/o dedicated packet buffer per port/TC may introduce unpredictable side effects on other streams that are a little bursty
- Issues from mixed speed-grades for OTA Tester that did not turn up in previously generation due to less performant HW
  - Less traffic offer from OTA Tester
  - Unknowing implicit "traffic shaping" by receivers due to smaller `RcvWindow` of SP2018 ECUs

Minor deficiencies may have severe consequences in presence of the slightest changes, when TCP is neither fine-tuned nor adapted to automotive constraints!

technica engineering

TCP AND AUTOMOTIVE ETHERNET

Introduction and PoC Setup

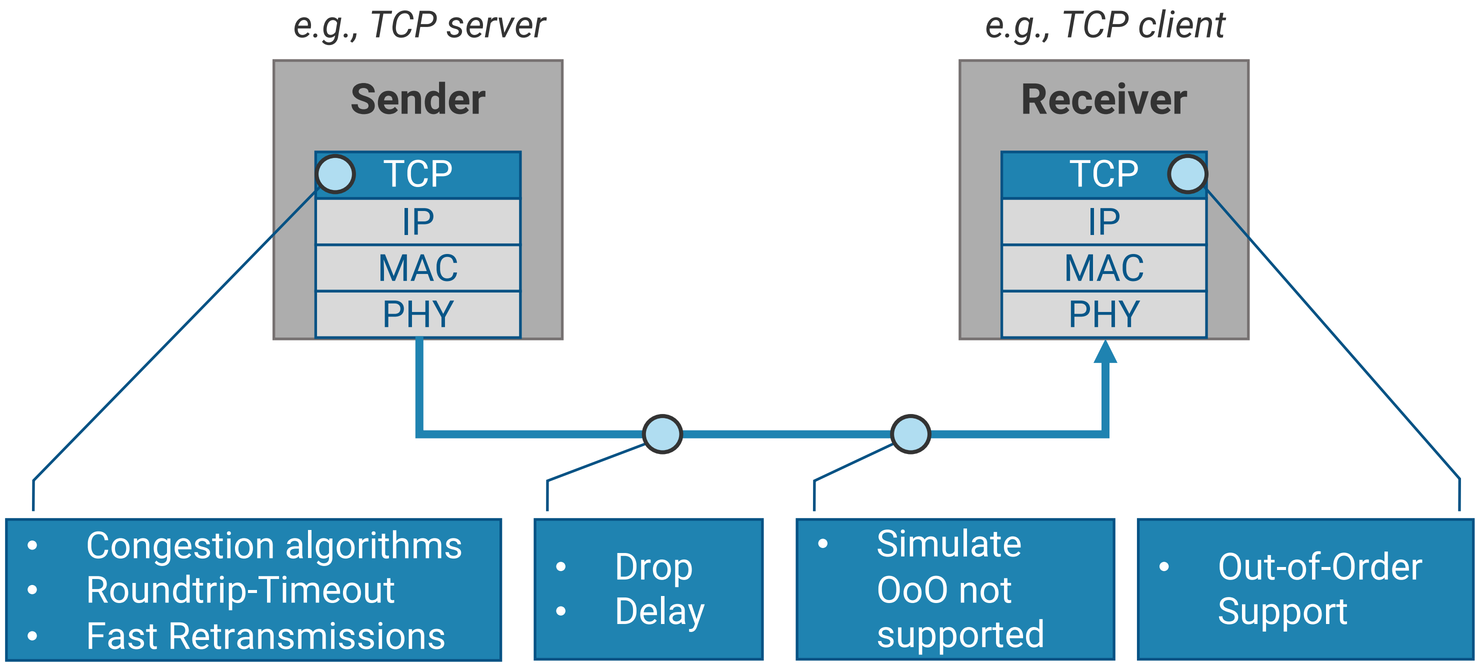
Ethernet & IP @ Automotive Technology Day

PROOF OF CONCEPT

INTRODUCTION AND POC SETUP

A PoC setup (under Linux) has been built to cover the following needs:

- **Tune the following TCP parameters:**
  - Roundtrip-Timeout
  - Out-of-Order support (OoO)
  - Congestion algorithms
  - Fast Retransmission support
- **Impairment simulation:**
  - Dropped segments
  - Delayed segments
  - OoO not supported
- **Evaluate effects on**
  - Round trip time / Latency
  - Goodput



TCP AND AUTOMOTIVE ETHERNET

Results and Discussion

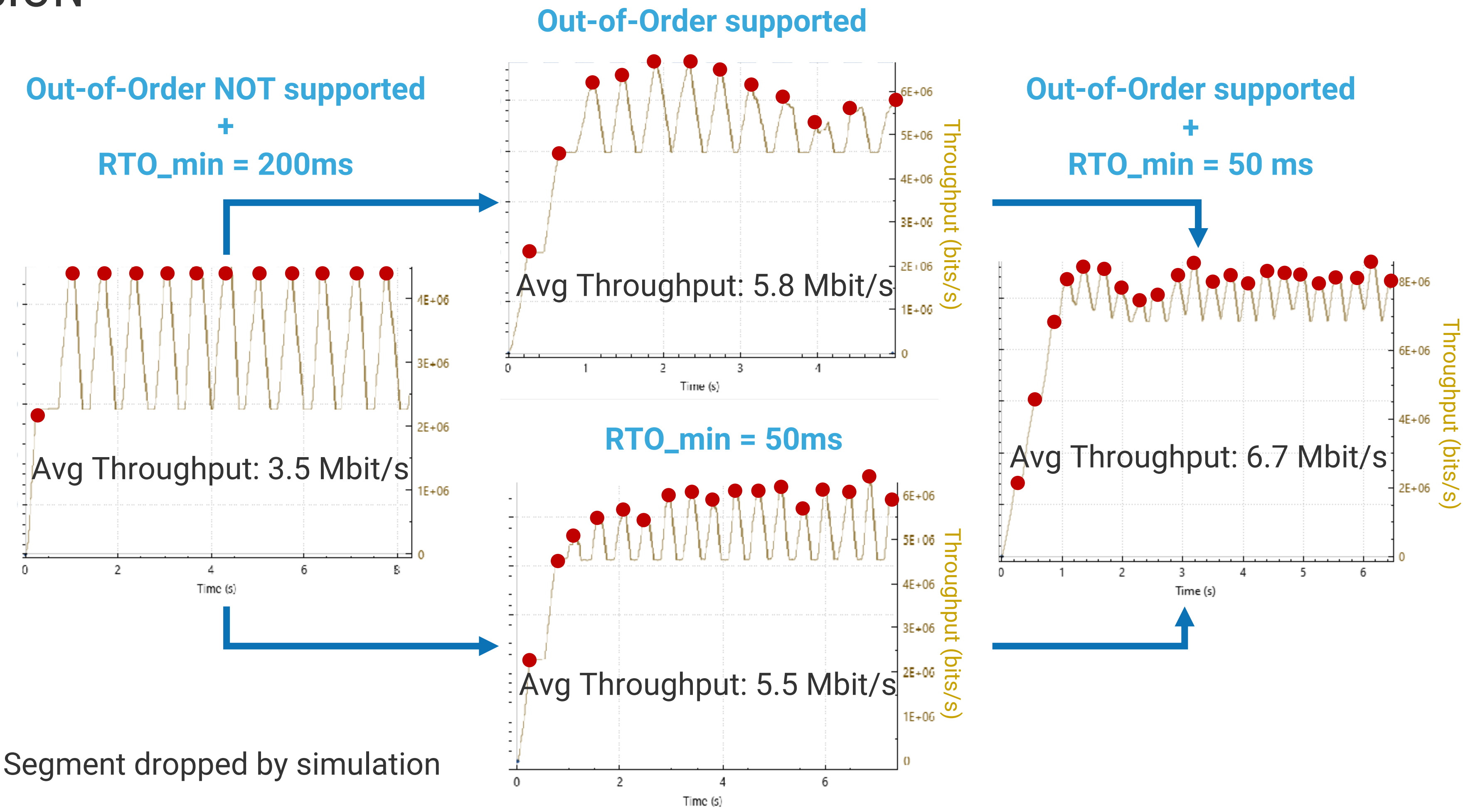
Ethernet & IP @ Automotive Technology Day

BIG IMPACT PROBLEMS IN EMBEDDED STACKS

RESULTS AND DISCUSSION

Issues

- Retransmission chains due to missing OoO support.
- Poor recovery speed / performance after loss.



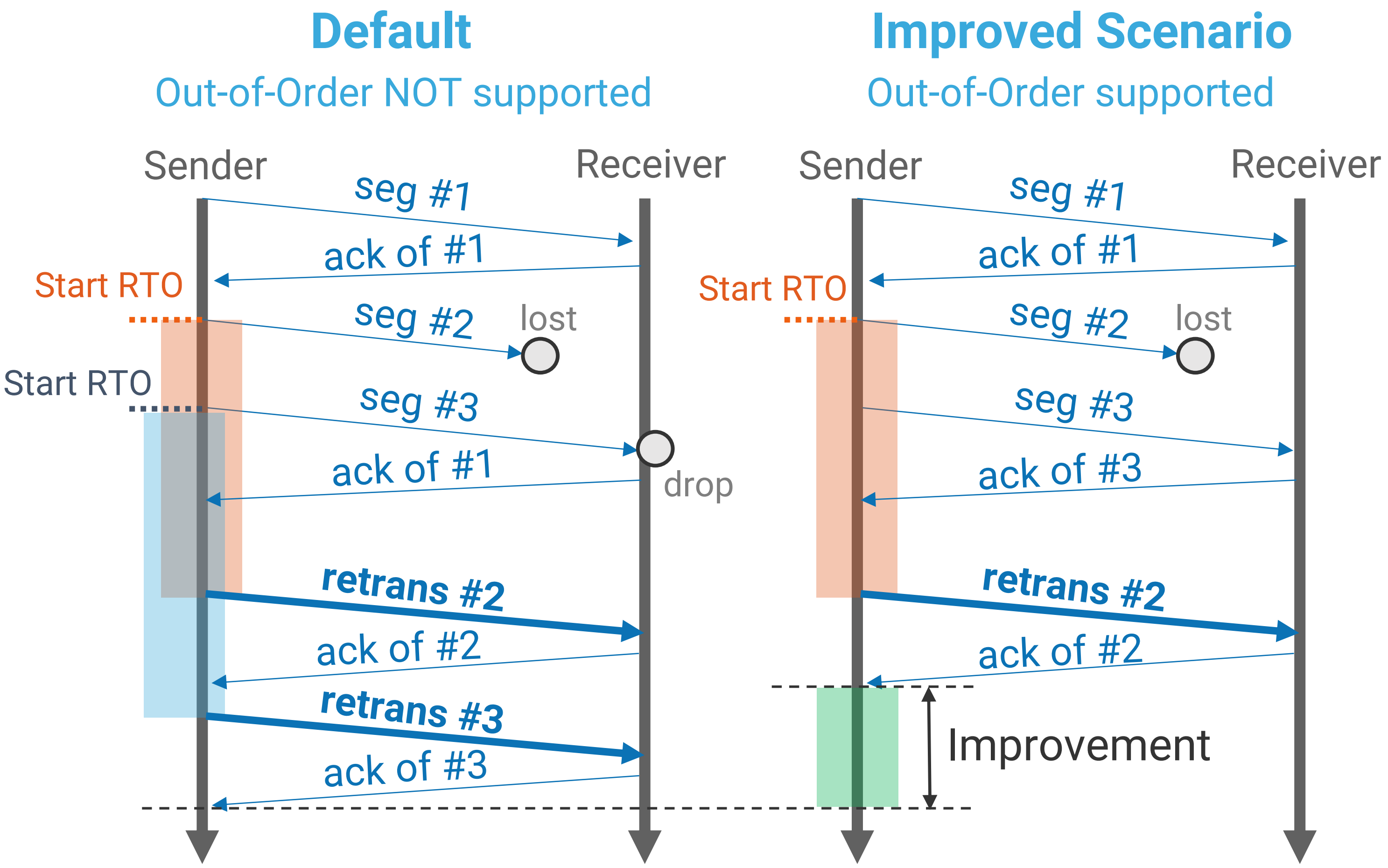
● Segment dropped by simulation

OUT-OF-ORDER SUPPORT

RESULTS AND DISCUSSION

Key Points

- The improved scenario enables the receiver to acknowledge the received OoO segments after a loss.
- Sender retransmits **only** what has been lost.
- OoO support prevents from loss chain effects, which are highly critical for transmission recovery.
- Selective Acknowledgment at the receiver provides additional advantages, if OoO is supported.
- Improvement depends on size of bytes in transit.



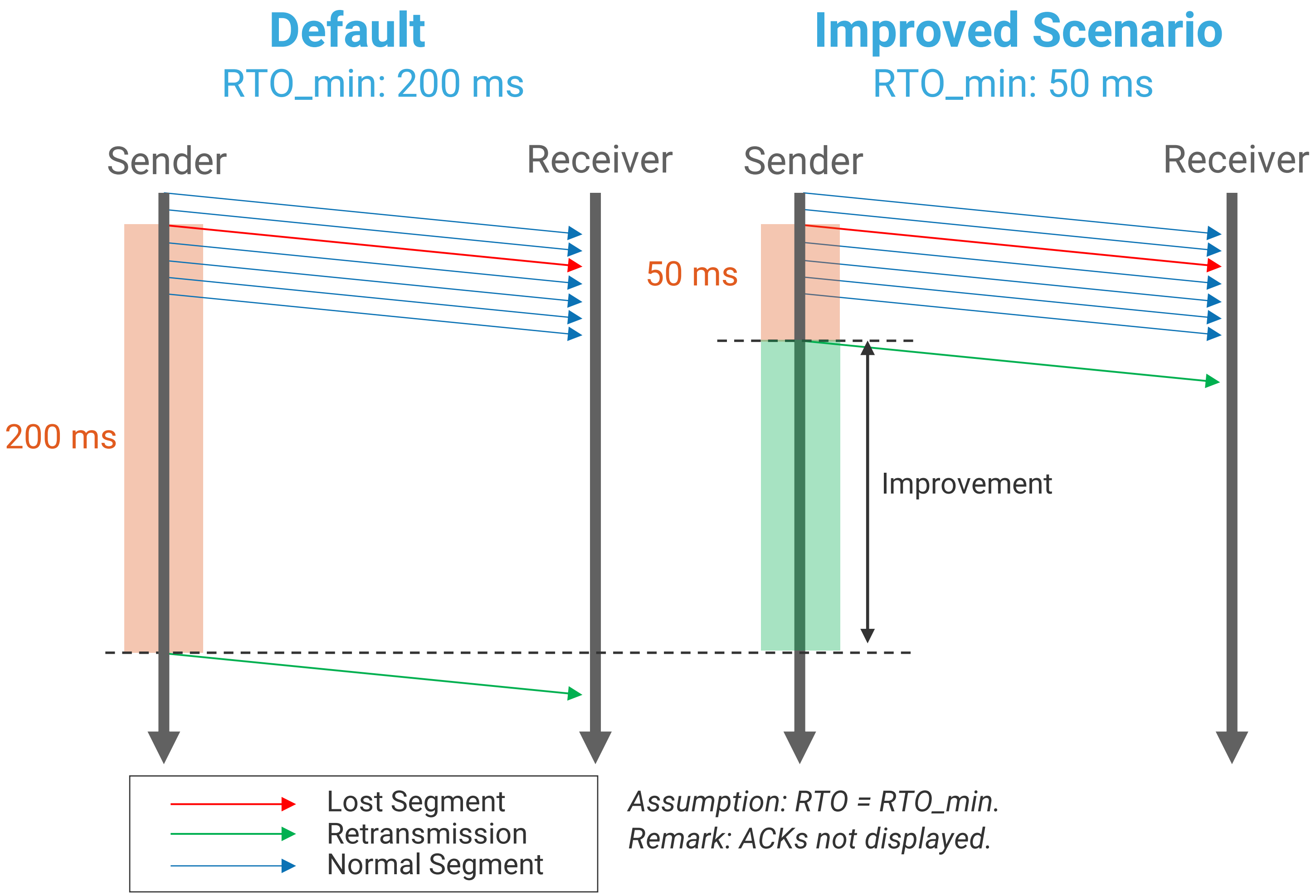
Remark: ack of #x refers to acknowledging segment x, not sequence number x.

RETRANSMISSION TIMEOUT (RTO)

RESULTS AND DISCUSSION

Key Points

- RTO defines how long the sender waits for a segment to be ACKed before triggering a retransmission. RTO_min is the minimum RTO used by the stack.
- RTO_min > RTT (but similar range).
- In Linux, the standard RTO_min is 200 ms. This is too large for small networks with low latencies like in the automotive use-case.
- Lowering the RTO_min to a value of ~50 ms has been shown to be reasonable.



Assumption: RTO = RTO_min.
Remark: ACKs not displayed.

SELECTIVE ACKNOWLEDGMENT (SACK)

RESULTS AND DISCUSSION

Experimental Results

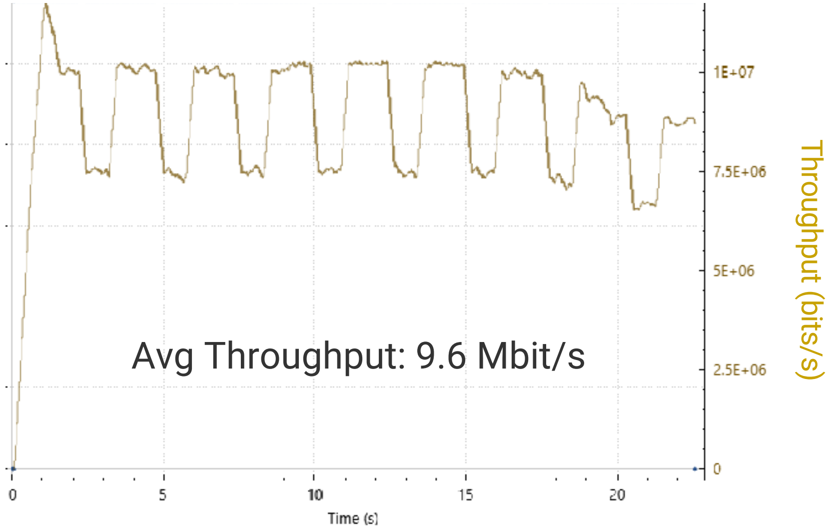
Latency

More than one “hole / gap” can be retransmitted within one RTT, thus improving latency.

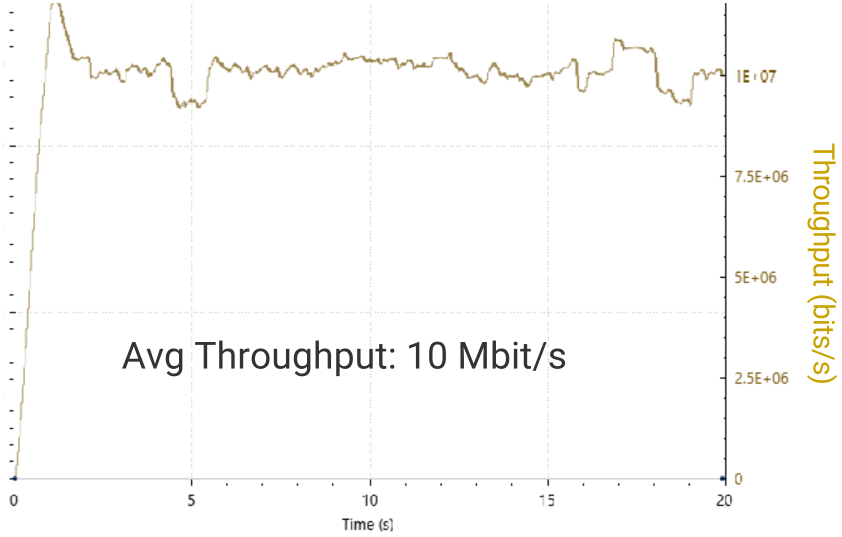
Throughput

SACK features have proven to speed up recovery and smooth out data throughput after losses.

Out-of-Order supported + SACK OFF



Out-of-Order supported + SACK ON

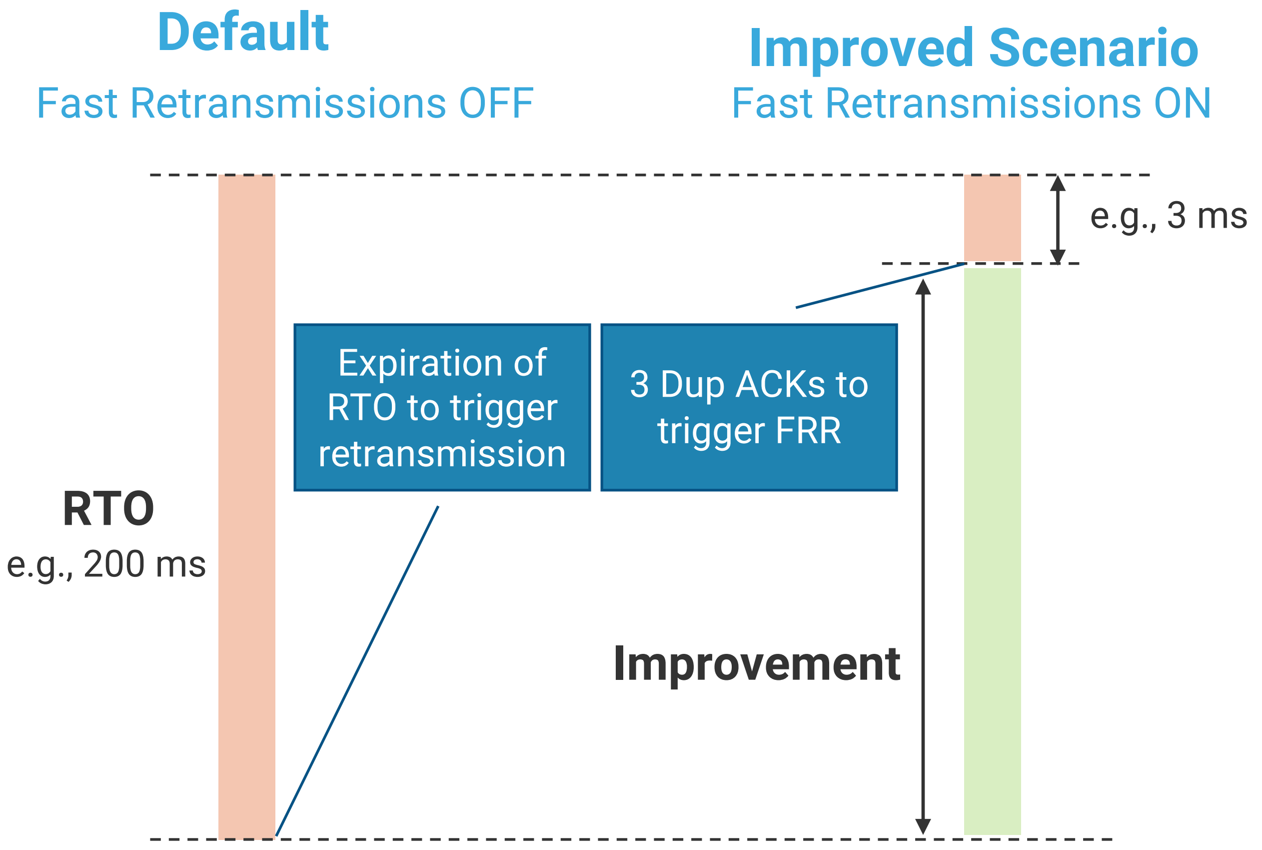


FAST RETRANSMISSIONS AND RECOVERY (FRR) (1)

RESULTS AND DISCUSSION

Key Points

- Upon receipt of a third Dup ACK, the sender assumes a segment loss and immediately retransmits it.
- Also, unsent segments within the window may be sent immediately after fast retransmissions.
- To trigger FRR, window size must be large enough: applications need to keep sending data. Typical issue with diagnostics and flash update.
- Selective Acknowledgment (SACK) can lead to FRR since every ACK is basically a Dup ACK.



FAST RETRANSMISSIONS AND RECOVERY (FRR) (2)

RESULTS AND DISCUSSION

- Due to a local network, no reordering is expected.
- Therefore, three Dup ACKs are not necessary to determine reordering.
- FRR have experimentally been triggered upon a second and even a first Dup ACK (faster recovery).

FRR after 3rd Dup ACK

Time [s]	Src	Dest	Seq	Lenght	Expert
-	sender	receiver	5178049	1448	Bulk Transfer
-	receiver	sender	1	0	ACK 5180945
REF	sender	receiver	5182393	1448	[Loss seg. 5180945]
0.0014266	receiver	sender	1	0	[Dup ACK 5180945 #1]
0.0012798	sender	receiver	5183841	1448	Bulk Transfer
0.0014155	receiver	sender	1	0	[Dup ACK 5180945 #2]
0.0024309	sender	receiver	5185289	1448	Bulk Transfer
0.0025945	receiver	sender	1	0	[Dup ACK 5180945 #3]
0.0036629	sender	receiver	518095	1448	[TCP Fast Retrans.]

FRR after 1st Dup ACK

Time [s]	Src	Dest	Seq	Lenght	Expert
-	sender	receiver	285257	1448	Bulk Transfer
REF	sender	receiver	288153	1448	[Loss seg. 286705]
0.0001592	receiver	sender	1	0	ACK 286705
0.0012001	sender	receiver	289601	1448	Bulk Transfer
0.0013388	receiver	sender	1	0	[Dup ACK 286705 #1]
0.0023545	sender	receiver	286705	1448	[TCP Fast Retrans.]

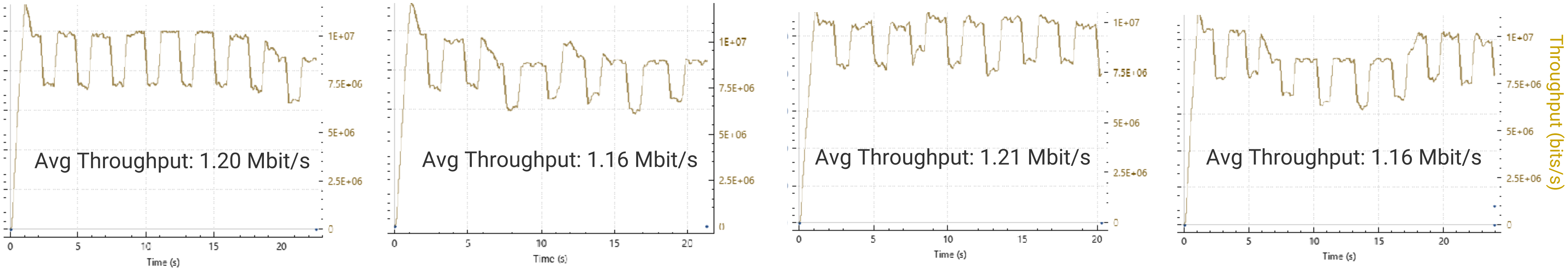
Significantly improved recovery time

LOW IMPACT PARAMETERS IN EMBEDDED STACKS

RESULTS AND DISCUSSION

Impact of Congestion Control?

Reno | Cubic | BIC | BBR



- Impact of different congestion control algorithms was minor (selected examples see above).
- None of these algorithms can cope with retransmission chains.

OVERALL RECOMMENDATION

RESULTS AND DISCUSSION

- RTO shall be tuned to 3-5x main function cycle of the TCP/IP stack: 50-70 ms.
- Out-of-Order shall be supported by TCP receiver to avoid unnecessary retransmissions.
- SACK can provide additional improvement.
- Fast Retransmissions shall be supported to enhance recovery time after segment loss.
  → Larger application messages and TCP window size. Otherwise, 3 Dup ACKs might never be seen.

Limitation

If drop occurs at the end of the message (seg #6) only RTO_min tuning helps to detect loss.

- Congestion control shall be optimized for automotive network properties
  → No reordering expected, therefore no three Dup ACKs necessary to trigger fast retransmissions.

TCP AND AUTOMOTIVE ETHERNET

Summary and Outlook

Ethernet & IP @ Automotive Technology Day

SUMMARY & OUTLOOK
CONCLUSION

- Embedded stacks without OoO support result in loss chains.

- Optimization with most performance impact:
  - **RTO_min tuning** improves TCP reaction time for retransmissions.
  - **Out-of-Order support** stops loss chains.

- Outlook:
  - Advanced traffic shaping can prevent loss due to switch congestion.
  - Custom congestion control algorithm for automotive could improve reaction time.
  - Investigation of application keepalives (e.g., SOME/IP magic cookie) after application data.

BMW
GROUP

Karl Budweiser

Network Architect
Karl.Budweiser@bmw.de
+49-151-601-58073

BMW AG

80788 Munich
Germany

https://www.linkedin.com/in/karl-budweiser-9b899a156/

technica
engineering

Iago Álvarez

System Engineer
Iago.Alvarez@technica-engineering.de

Technica Engineering GmbH
Leopoldstraße 236
80807 Munich
Germany

www.linkedin.com/in/iago-alvarez