Static Context Header Compression and Fragmentation ARQ/FEC mode

draft-munoz-schc-over-dts-iot

Rodrigo Muñoz
Universidad de Chile
rmunozlara@ing.uchile.cl

Sandra Céspedes
Concordia University
sandra.cespedes@concordia.ca

Javier Alejandro Fernandez
IMT Atlantique
javier-alejandro.fernandez@imt-atlantique.fr



November 17th, 2026

Internet Engineering Task Force
© 2022 IETF Trust
Production by Meetecho

IETF®

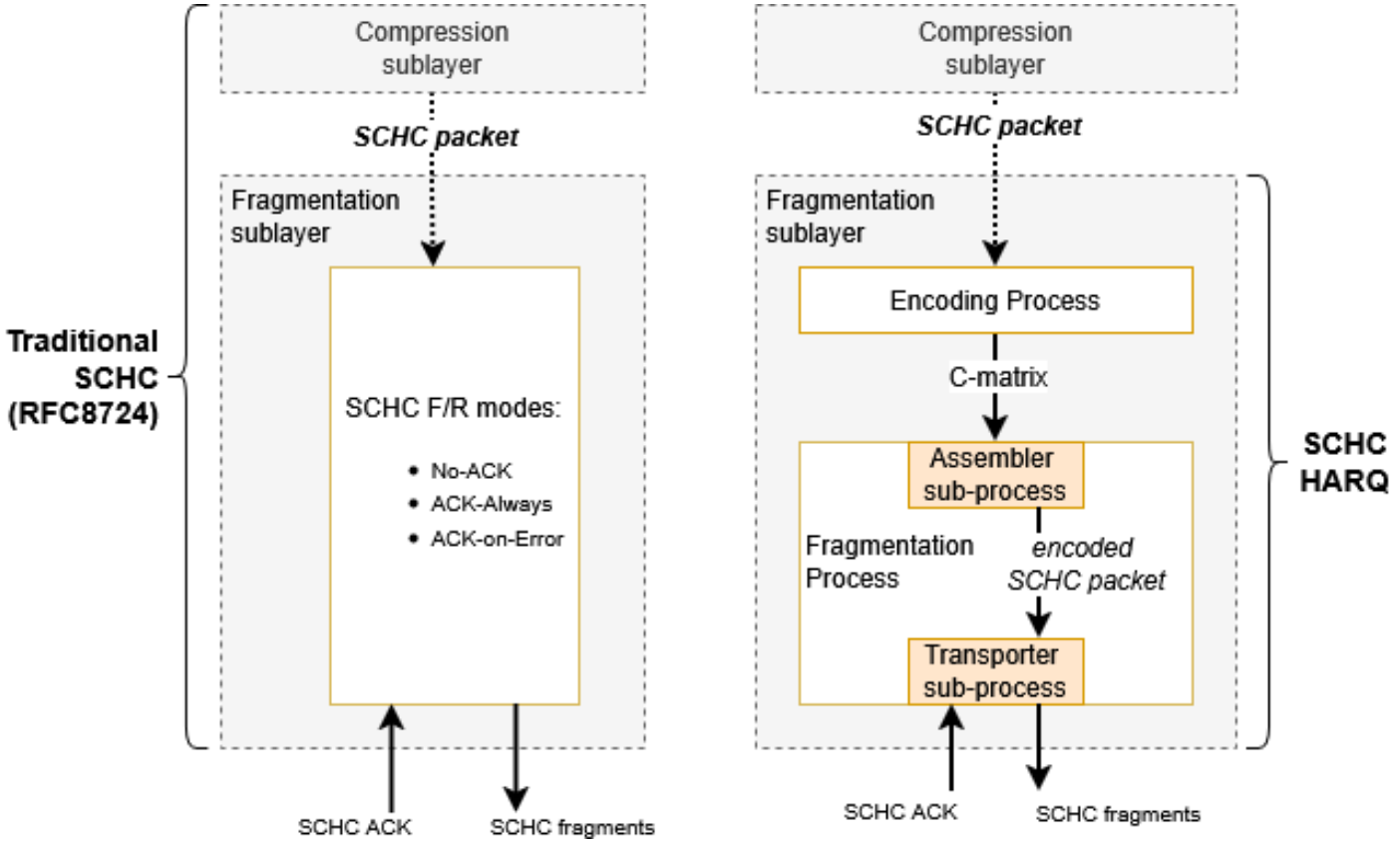
Let us recall the problem…

When SCHC works on links with delays or interruptions (Direct to Satellite IoT, underwater, Zero-Energy, etc.)

If there are long delays	SCHC timers expire	SCHC sessions ended unexpectedly
if there are lost tiles	successive tile retransmissions	increased energy consumption and transfer delay
if there are tiles with errors	inefficient use of transmission time	SCHC sessions ended unexpectedly

The scope of the draft

- The SCHC Packet is **encoded** (+ parity data).
- The **Coded Structure** is Fragmented.
  - a matrix-based encoding geometry ("C-matrix")
  - a streaming encoding geometry ("C-Stream")
- Sender sends ***n*** fragments.
- Receiver only needs ***k*** fragments (k-out-of-n) to **decode** the original SCHC Packet.



Recent Updates

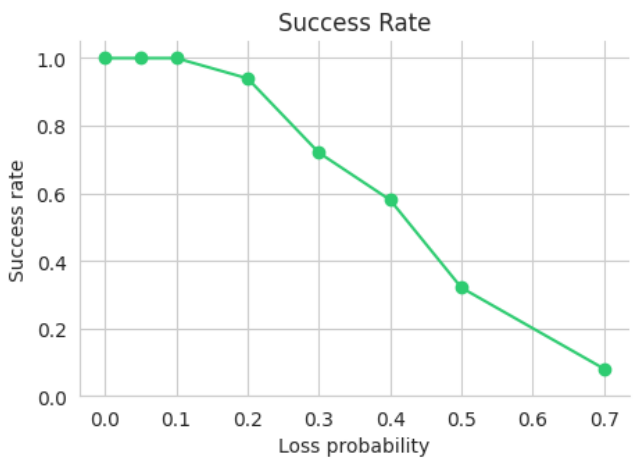
- No major changes.
  - Refined some parameters & wording.
  - Added some constraints.
- *IETF124: WG asked for test/data.*

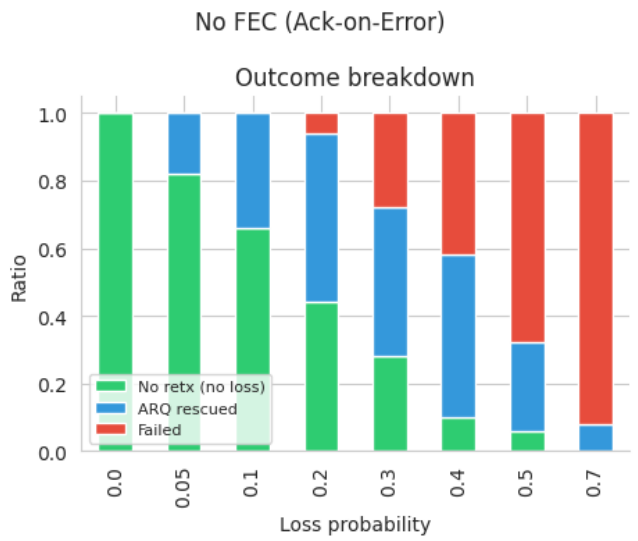
Feature	draft-02	**draft-03**
Signaling	Dynamically sends ***S*** (matrix rows)	Sends ***k*** (source block size, 8-bit field); ***S*** is fixed
Initial ACK Handshake	Mandatory	None (Immediate regular transmission)
Receiver ACK Triggers	4 cases	3 cases
Matrix Tile Placement	Dynamic calculation	Direct formula: row=1, col=f(W,FCN)
C-Stream Constraint	Flexible tile order across windows	Strictly non-decreasing ***W*** (No-Wrap rule/constraint)

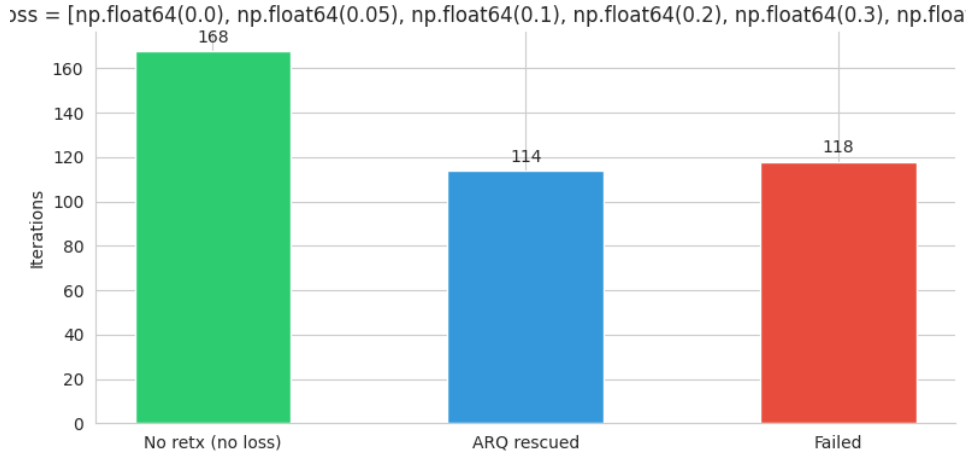
Testbed 1: PoC: **NO FEC** vs **FEC** (XOR)

github.com/**alex-fddz/arq-fec-poc**

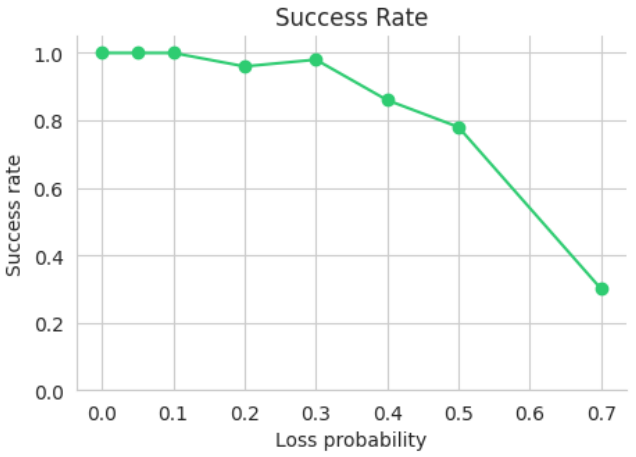
NO FEC

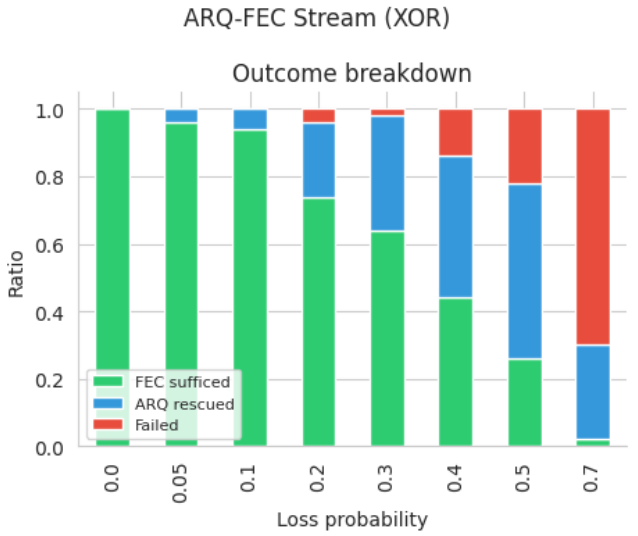


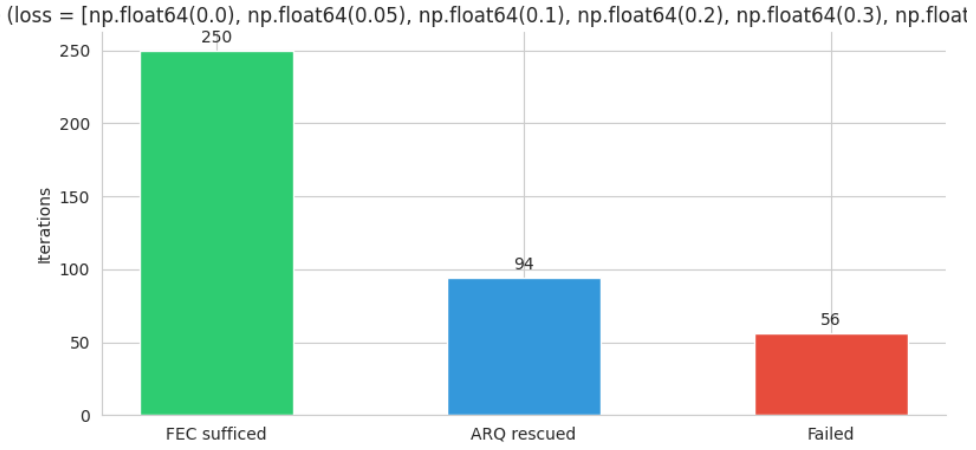




WITH FEC

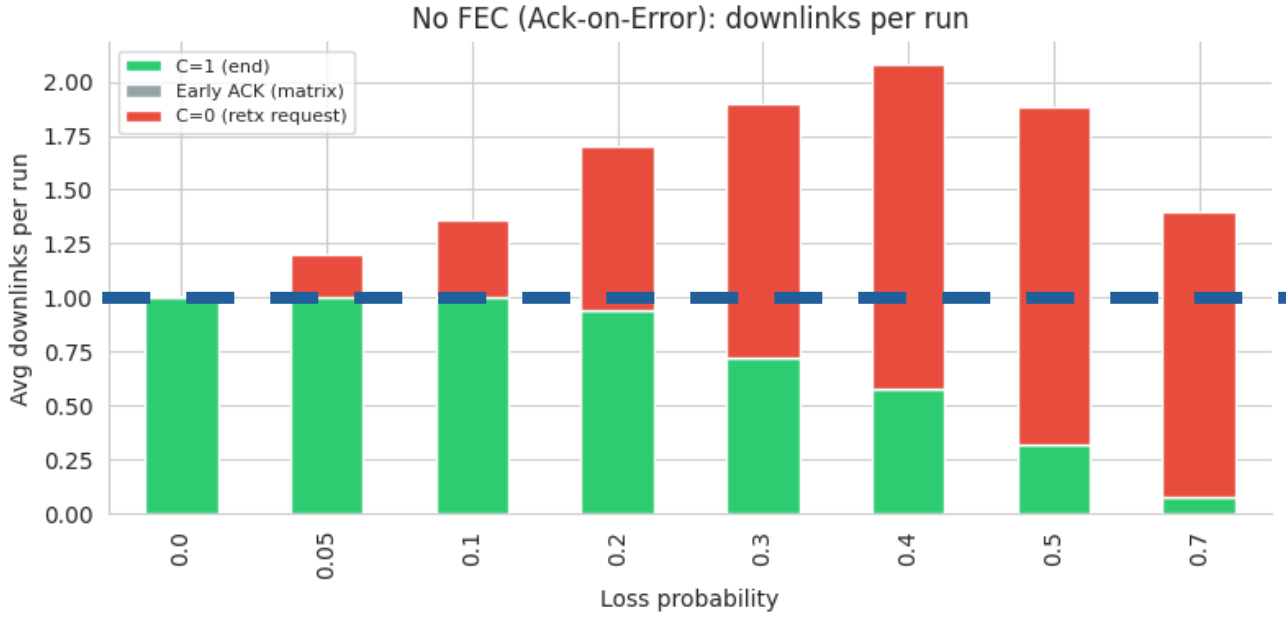


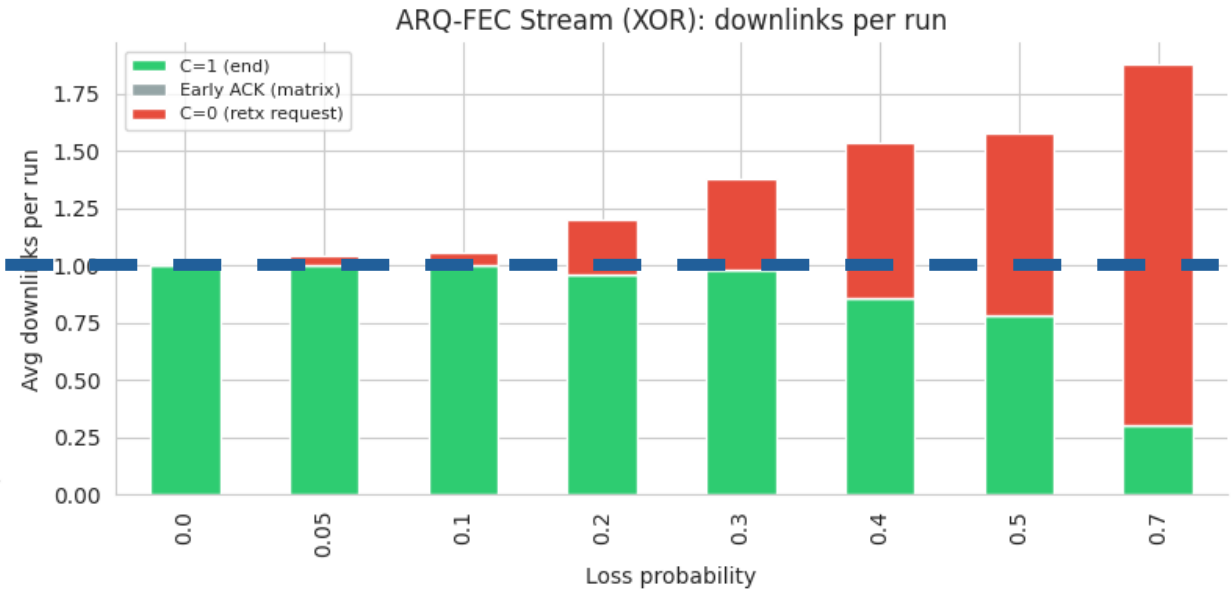




Testbed 1: PoC: **NO FEC** vs **FEC** (XOR)

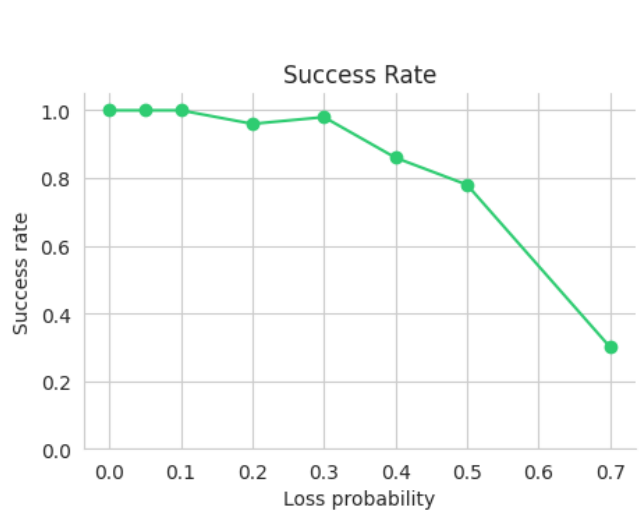
github.com/**alex-fddz/arq-fec-poc**

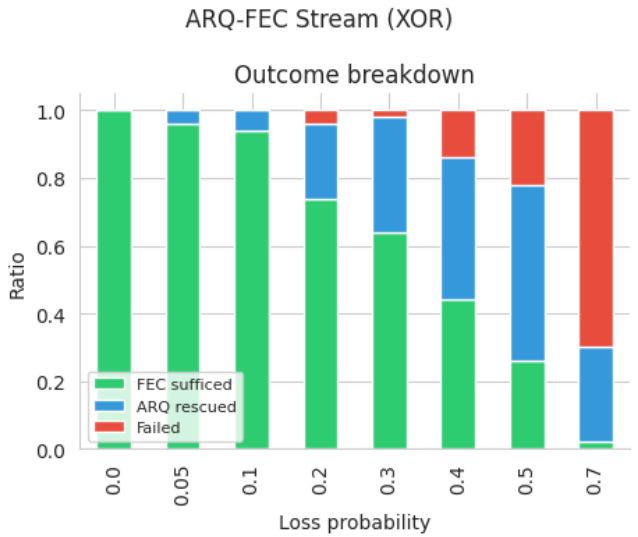


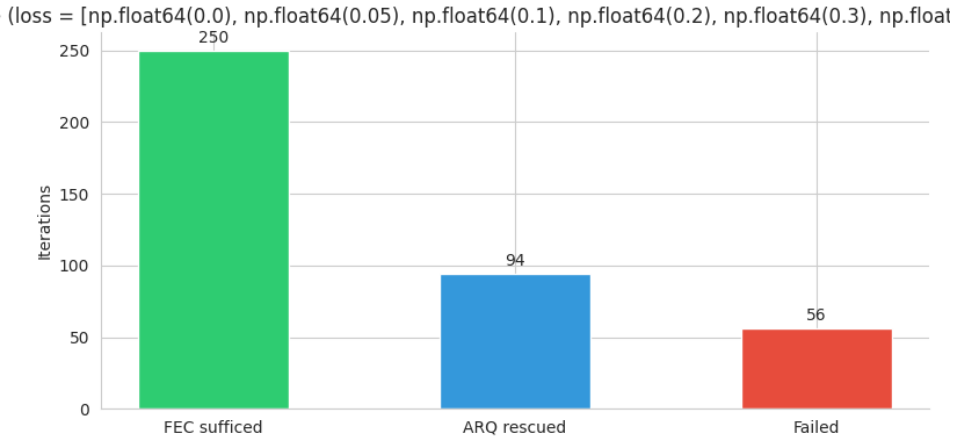


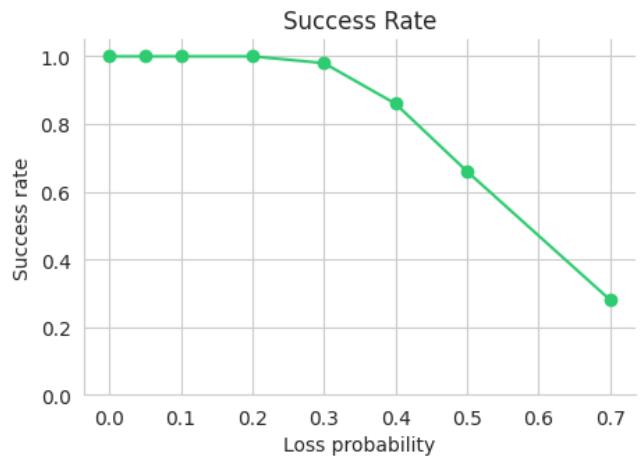
Testbed 1: PoC: Stream (XOR) vs Matrix (RS)

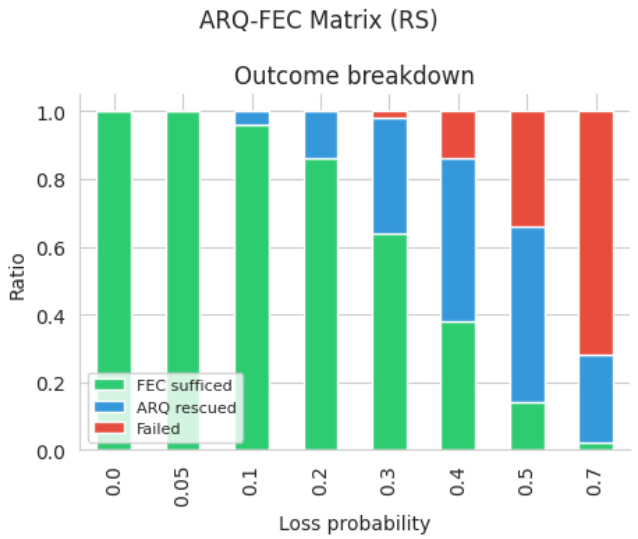
github.com/**alex-fddz/arq-fec-poc**

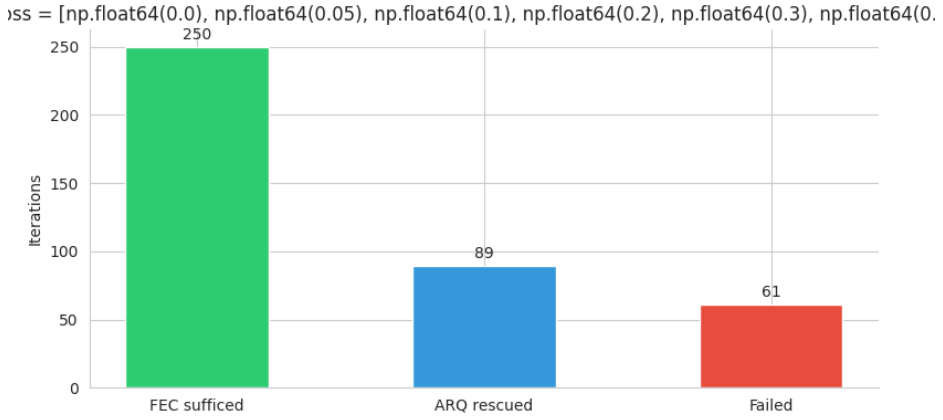






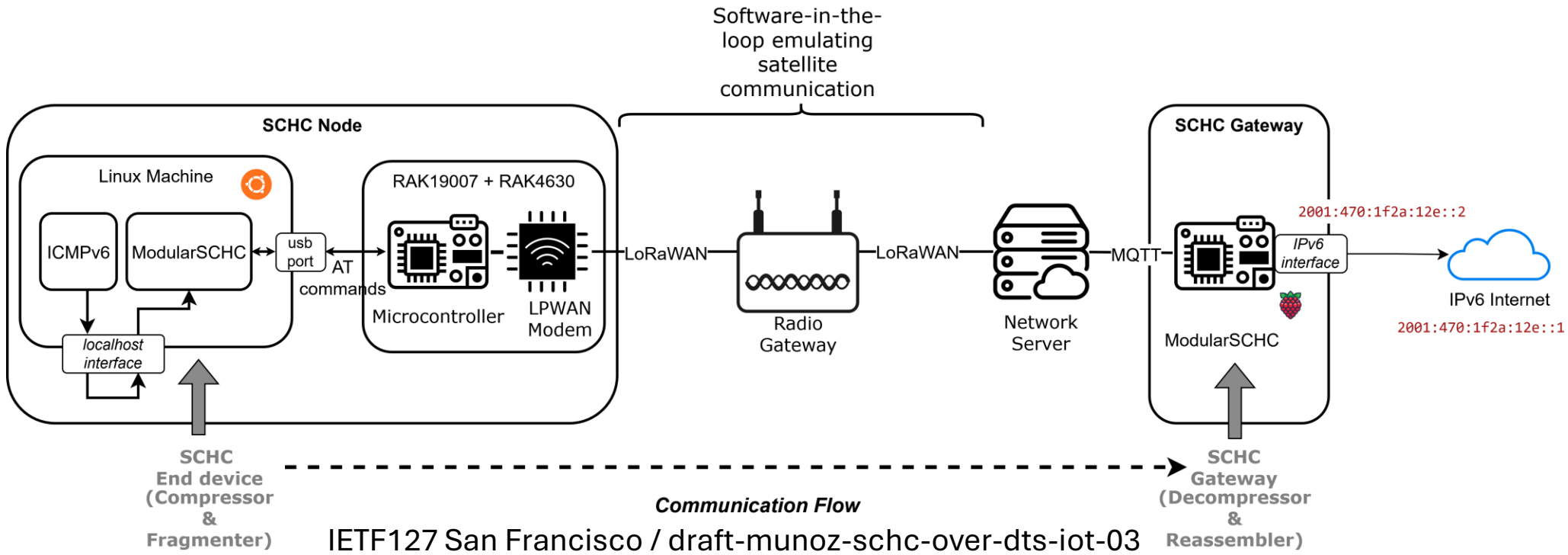




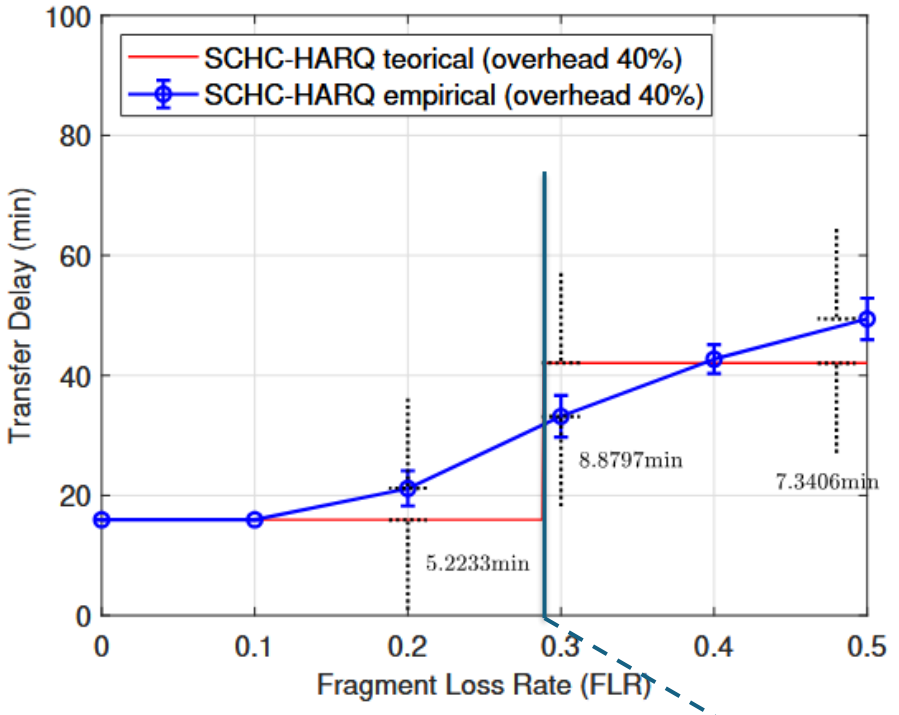


Testbed 2: On Real Hardware

- ICMPv6 msg from end device to node in Internet.
- ICMPv6 length: 7681 bits
- LoRaWAN DR5 (227 bytes) and DR3 (38 bytes)



Testbed 2: On Real Hardware



(a) Transfer Delay for DR3 (overhead 0.4)

Parameter	Value
Terrestrial network Parameters	
Access technology	LoRaWAN
Frequency band	AU915-928 [79]
LoRaWAN data rates	DR5, DR3 [79]
LoRaWAN MTU	227, 38 (Bytes)[1] [79]
SCHC Generic Parameters	
Packet size (L)	7,608 bits
Tile Size (S)	10 Bytes (RFC 9011 [7])
SCHC-HARQ Parameters	
Overhead	0.4
Number of bits in a symbol (m)	8 bits
Number of symbols in a Source Block (k)	$\lfloor \frac{L}{S \cdot 8} \rfloor = 95$
Number of parity symbols (r)	$\lceil k \cdot overhead \rceil$
Number of symbols in an Encoded Block (n)	$k + r$
$p_{\max}$	50% [74, 75]
V_ℓ	Swarm constellation forecast (see table C.1)
G_ℓ	Swarm constellation forecast (see table C.1)

$$\frac{overhead}{1 + overhead} = \frac{0.4}{1 + 0.4} = 0.2857$$

Conclusions

- **More robust comms.**
  - At 30-50 % loss, FEC raises delivery from 72 / 58 / 32 % to 98 / 86 / 78 %.
- **Less downlinks.**
  - At 10% loss, 95% of transfers complete without reTx (vs 66%).
- **2 configurations** to choose from:
  - **C-Stream**: simple & lightweight.
  - **C-Matrix**: more complex but more flexible.

- Overhead = ~40%
  - 6-7 UL instead of 4-5 (depends).

- An industrial partner has expressed interest in this work.
  - Potential collab & applicability to real-world deployment.

References

- [1] R. Munoz-Lara, S. Céspedes and M. Diaz, "**SCHC over DtS-IoT: Performance of SCHC Confirmation Modes in Satellite IoT**," *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*, Ottawa, ON, Canada, 2024
- [2] R. Munoz-Lara, M. Díaz and S. Céspedes, "**Evaluating SCHC Under Intermittent Connectivity: Insights From Satellite IoT**," in *IEEE Internet of Things Journal*, vol. 13, no. 14, pp. 30674-30686, 15 July15, 2026, doi: 10.1109/JIOT.2026.3689441.
- [3] GitHub - RodrigoMunozLara/ModularSCHC: **Implementation of the SCHC standard with support for multiple protocols in the layer below SCHC.** --- github.com. https://github.com/RodrigoMunozLara/ModularSCHC, [Accessed 10-11-2026]
- [4] GitHub - alex-fddz/arq-fec-poc: **ARQ-FEC PoC.** — github.com. https://github.com/alex-fddz/arq-fec-poc [Accessed 13-11-2026]



Thank you

Questions?

Adoption?