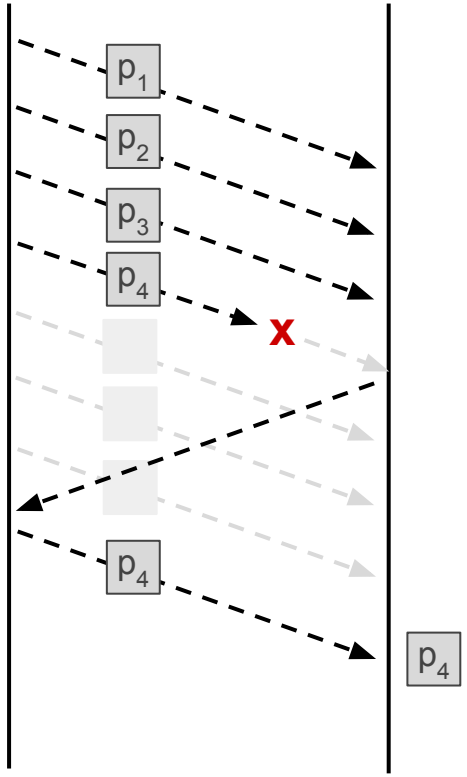
# Steinwurf

## Latency for block and sliding window codes

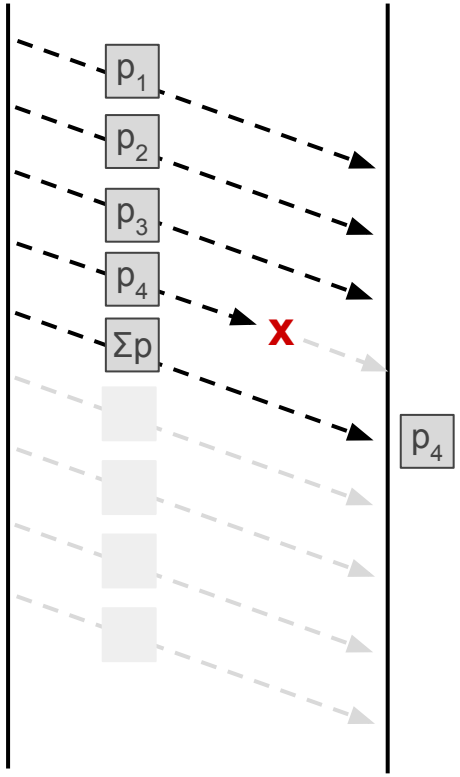
## IETF 110 / NWCRG

# Introduction

One of the key reasons for using ECC/FEC is to minimize latency.



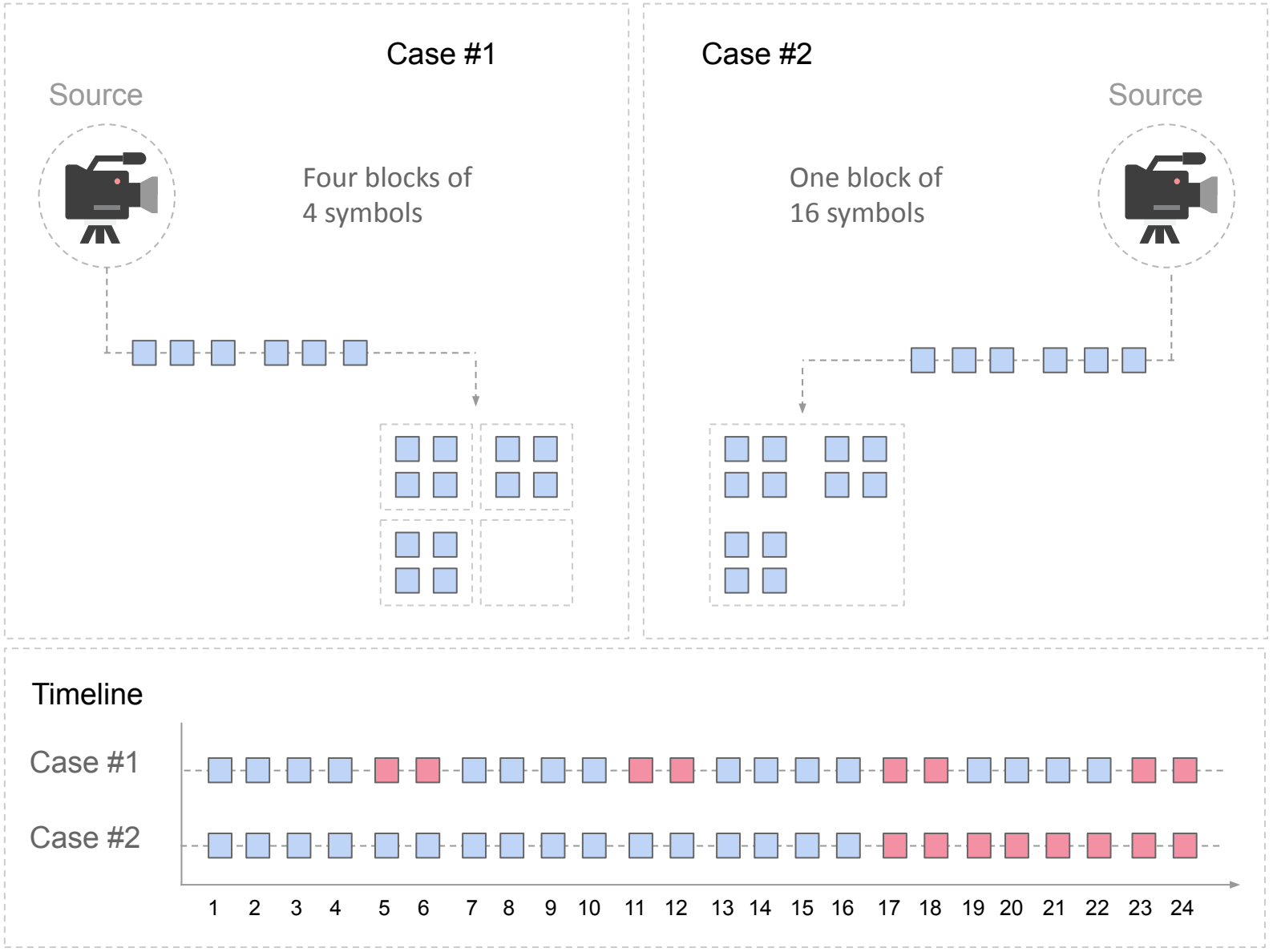

**ARQ Recovery latency:**
1 RTT per retransmission




**ECC/FEC Recovery latency:** Distance to the repair packet

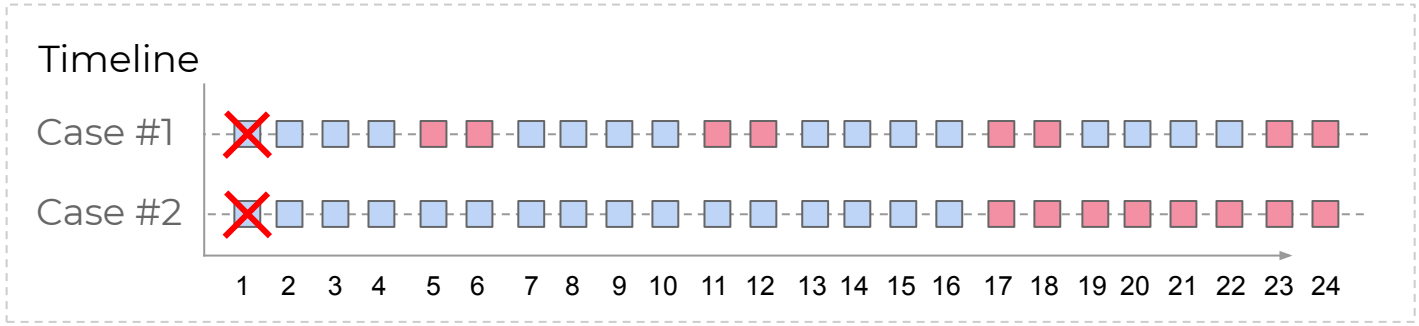
STEINWURF

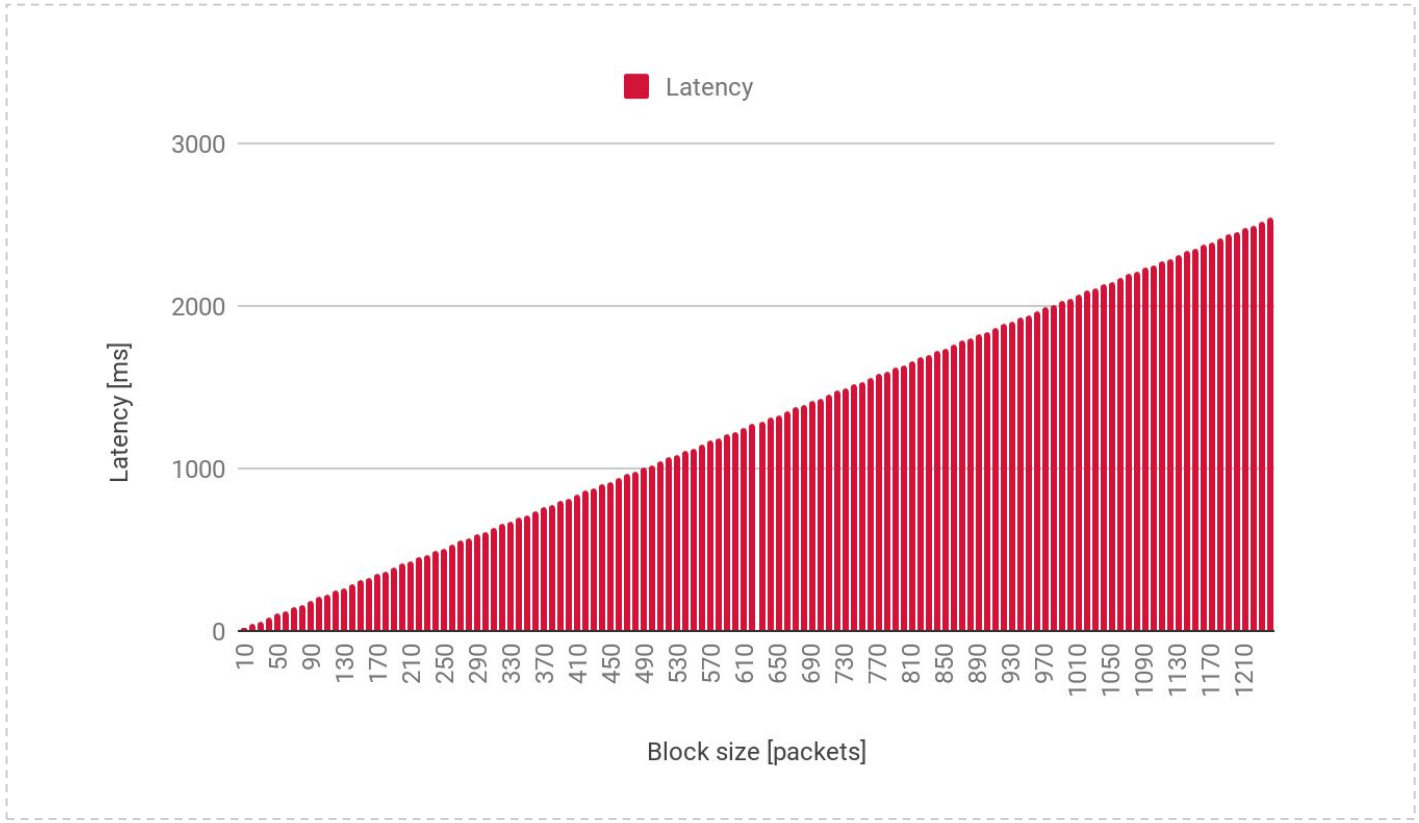
# Block ECC/FEC




STEINWURF

# Block ECC/FEC




**Distance to repair is minimized with small blocks...**

**Repair rate is the same 33%**

**5 Mbit/s stream 1280 byte packets: > 500 block size equals 1 s of latency**

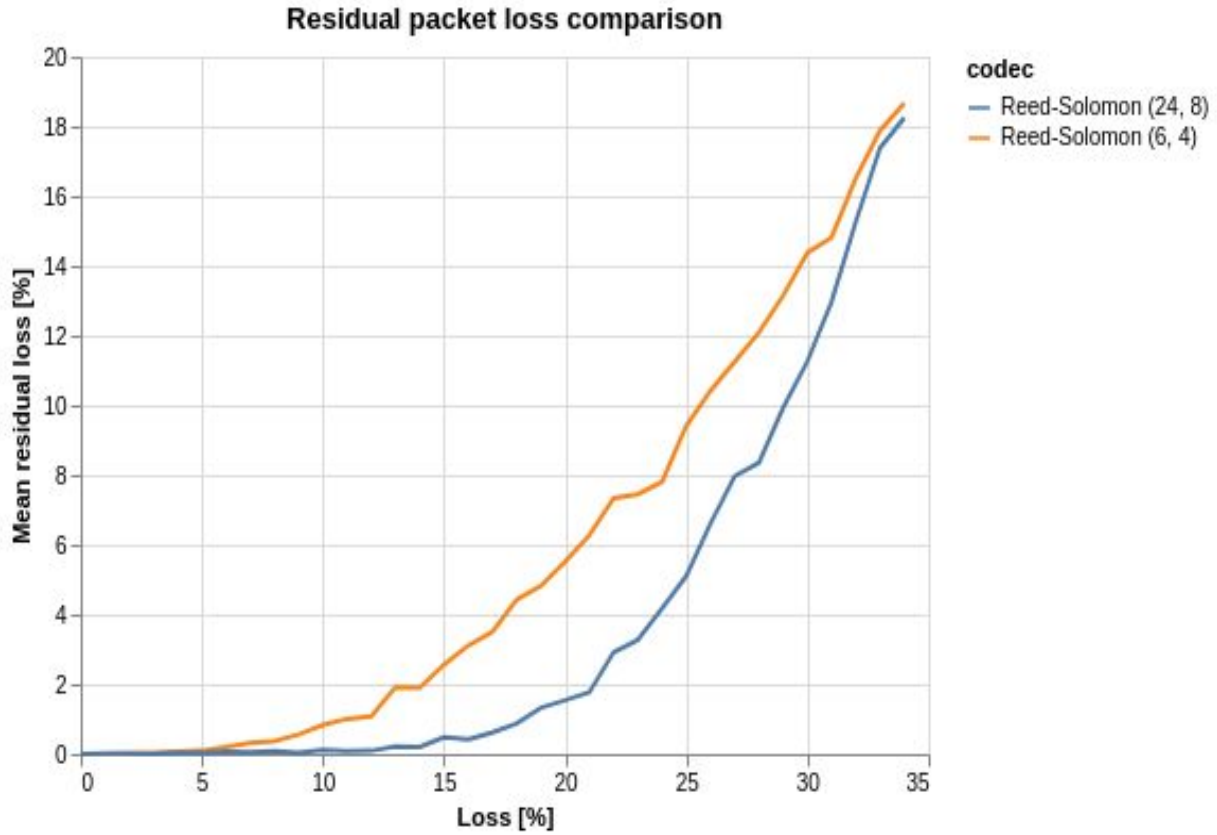
STEINWURF

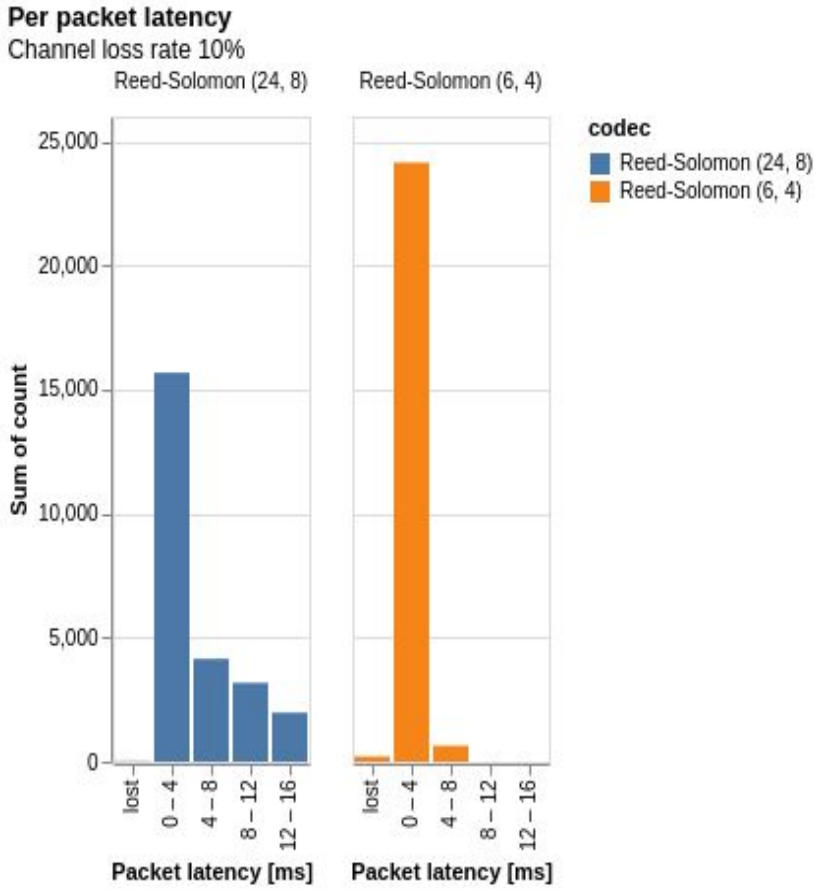
# Comparison

Reed-Solomon (6,4) vs. (24, 16)

**Small blocks are good for latency but bad for loss**

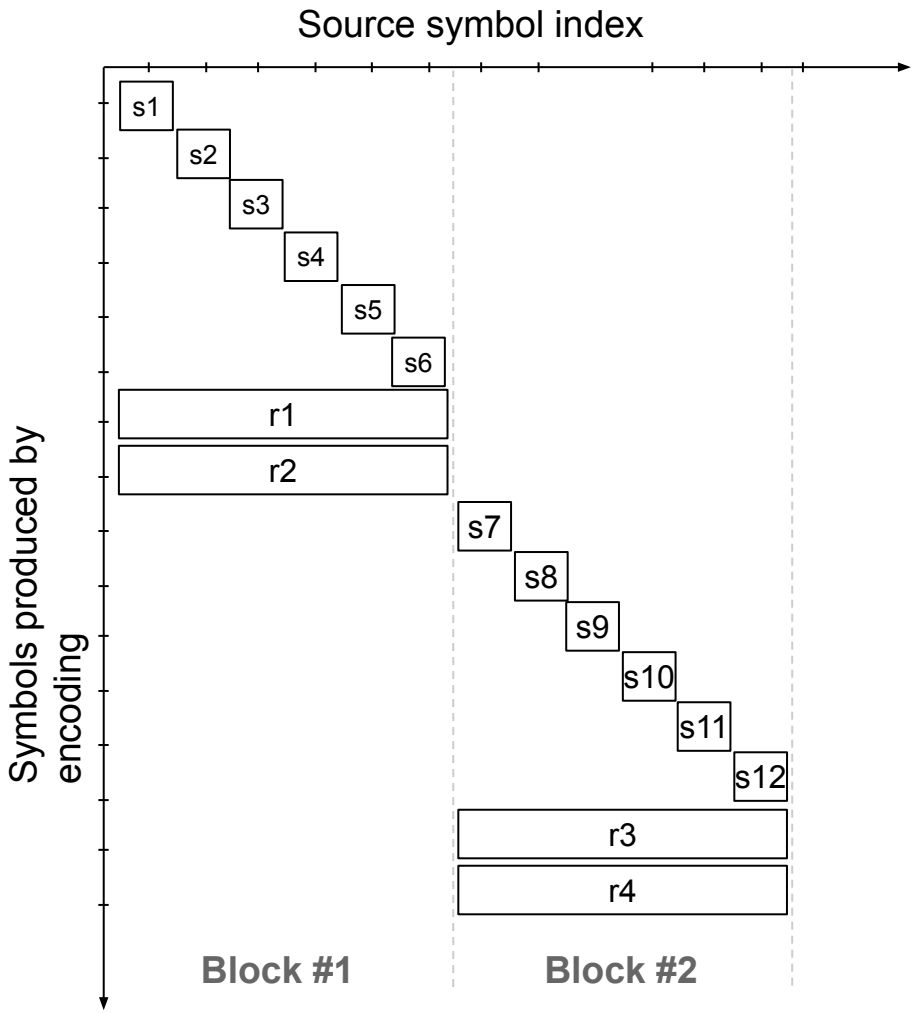
**Large blocks are good for loss but bad for latency**

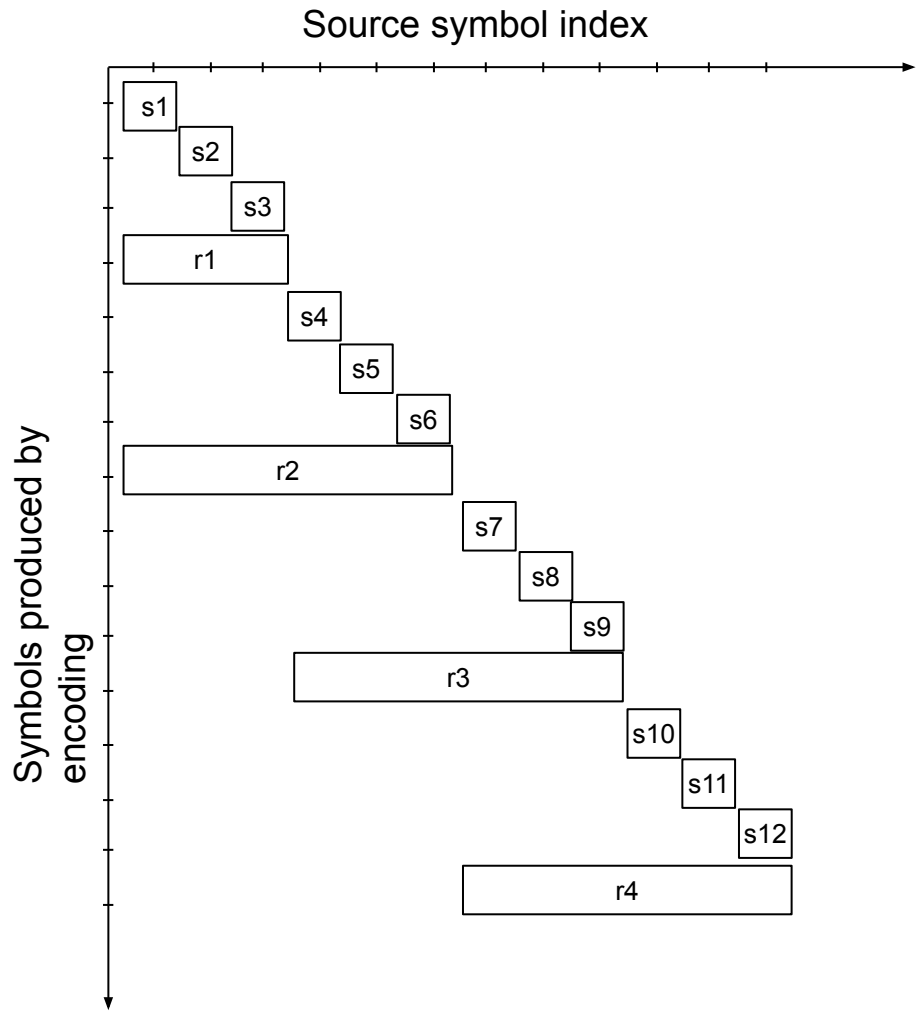


# Sliding window codes

Minimizing distance to repair but offering better packet loss protection

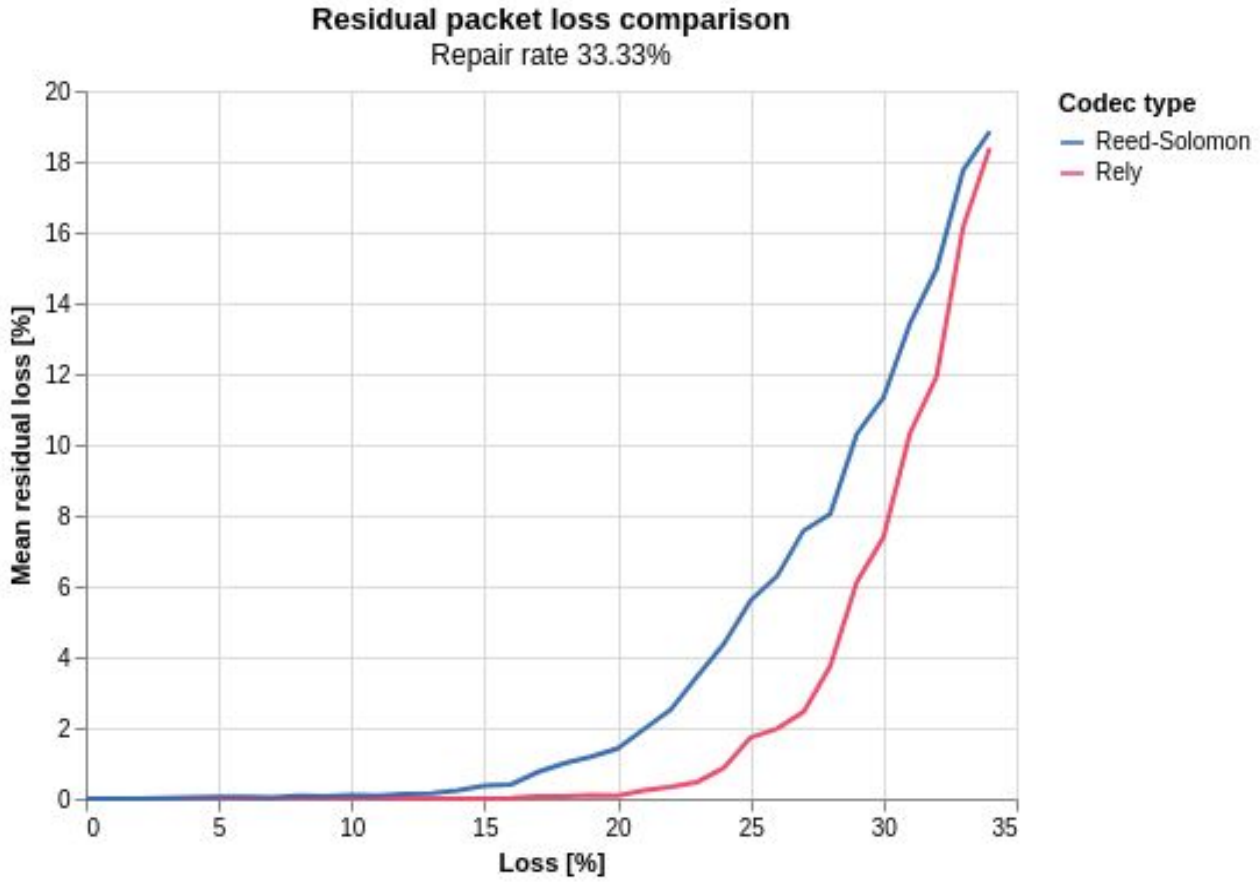


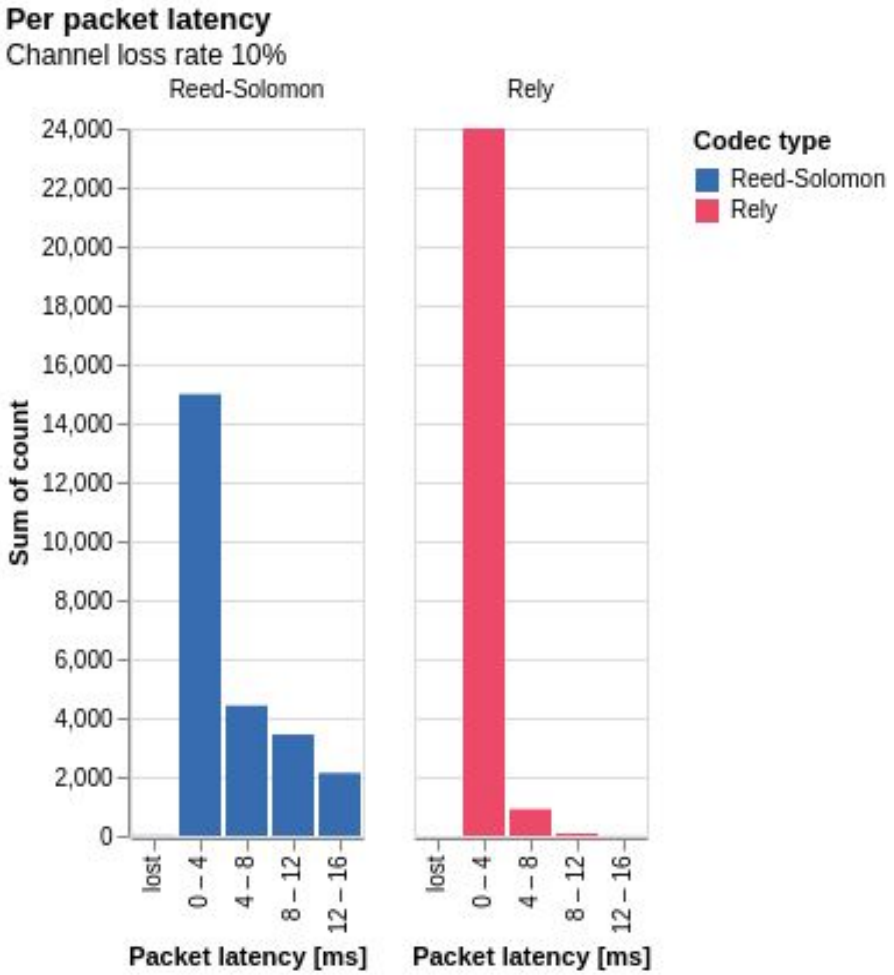



# Comparison

Rely (6,4) vs. Reed-Solomon (24, 16)

**Using sliding window we get good loss recovery and latency properties!**



# Conclusions

- Consider sliding window codes when latency matters
- Sliding window codes worst case is block coding

- More about sliding window coding:
  - https://rely.steinwurf.com/docs/latest/

- Get in touch
  - Morten V. Pedersen (morten@steinwurf.com)
  - www.steinwurf.com


STEINWURF