Gefördert durch:
Bundesministerium für Wirtschaft und Energie
aufgrund eines Beschlusses des Deutschen Bundestages

Friedrich-Alexander-Universität
Technische Fakultät
Lehrstuhl Informatik 7

Protocols, QUIC, and SATCOM

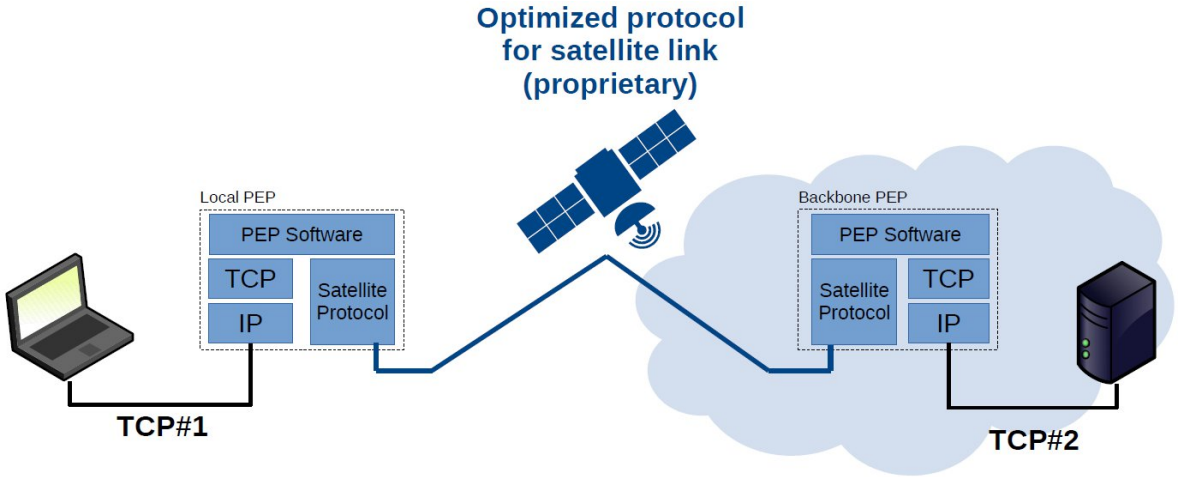
2nd QUIC and Satellite Open Stakeholder Meeting

Thursday 2nd December 2021 (online)

Joerg Deutschmann joerg.deutschmann@fau.de
Sebastian Endres
Kai-Steffen Hielscher
Reinhard German

Motivation

- Performance Enhancing Proxies not applicable in case of encrypted transport layer headers (e.g., VPNs or QUIC)
  https://www.cs7.tf.fau.eu/research/quality-of-service/qos-research-projects/sat-internet-performance

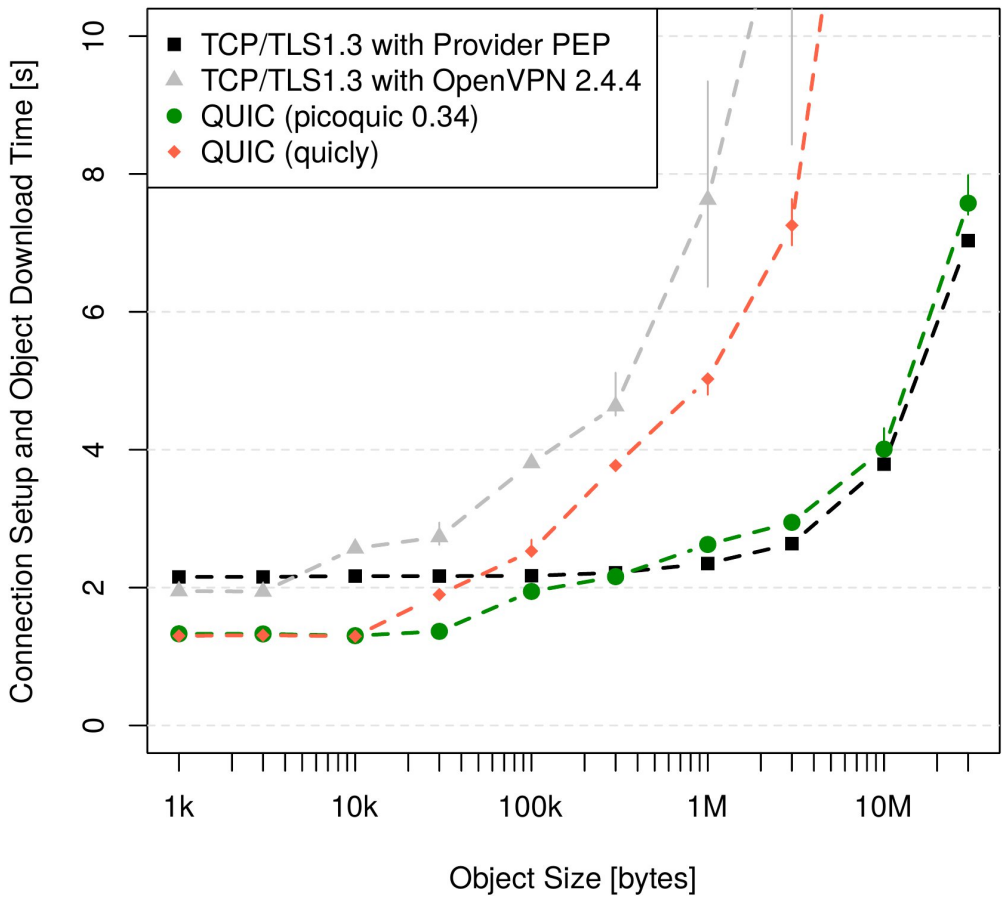


- 5G NTN SA WG2 Meeting S2-2105611
  Key Issue: Discussion on connected protocols in case of GEO SAT long delays
  https://www.3gpp.org/ftp/tsg_sa/WG2_Arch/TSGS2_146E_Electronic_2021-08/Docs/S2-2105611.zip

Outline

- Part 1
  - Simple black-box measurements
  - Varying object sizes
  - Real satellite operator, TCP with and without OpenVPN, quicly and picoquic
- Part 2
  - Simple black-box measurements
  - Different HTTP versions
  - Different real satellite operators, with and without Wireguard VPN
- Part 3
  - QUIC interop runner with geostationary satellite links
  - With and without loss, real satellite operators
  - Time-offset plots for detailed analysis

Part 1
Object size vs. connection setup and object download time



- Eutelsat Konnect Zen
  - 50 Mbit/s forward link
  - 5 Mbit/s return link
- TCP/TLS1.3
  - TCP handshake 1 RTT
  - TLS1.3 handshake 1 RTT
- QUIC
  - client/server with default settings
  - Handshake 1 RTT
  - picoquic `a91130d` Jan. 2021
  - quicly `8a1346e` Jan. 2021
- OpenVPN client/server running on hosts

Outline

- Part 1
  - Simple black-box measurements
  - Varying object sizes
  - Real satellite operator, TCP with and without OpenVPN, quicly and picoquic
- Part 2
  - Simple black-box measurements
  - Different HTTP versions
  - Different real satellite operators, with and without Wireguard VPN
- Part 3
  - QUIC interop runner with geostationary satellite links
  - With and without loss, real satellite operators
  - Time-offset plots for detailed analysis

Part 2
Page load time of artificial website using different HTTP versions

- *off-the-shelf* software
  - Google Chrome 94.0.4606.54
  - OpenLiteSpeed 1.7.4 web server HTTP/1.1, HTTP/2, HTTP/3 (QUICv1)
- Artificial website based on httparchive.org statistics
  - 70 objects * 30 kbyte/object = 2.1 Mbyte
- Wireguard VPN client/server running on hosts
- Real satellite operators
  - Eutelsat Konnect Zen (50/5 Mbps)
  - skyDSL2+ L Premium (50/6 Mbps)
  - Novostream/Astra Connect L+ (20/2 Mbps)
  - Bigblu Konnect Bronze (16/3 Mbps)
  - Starlink Beta

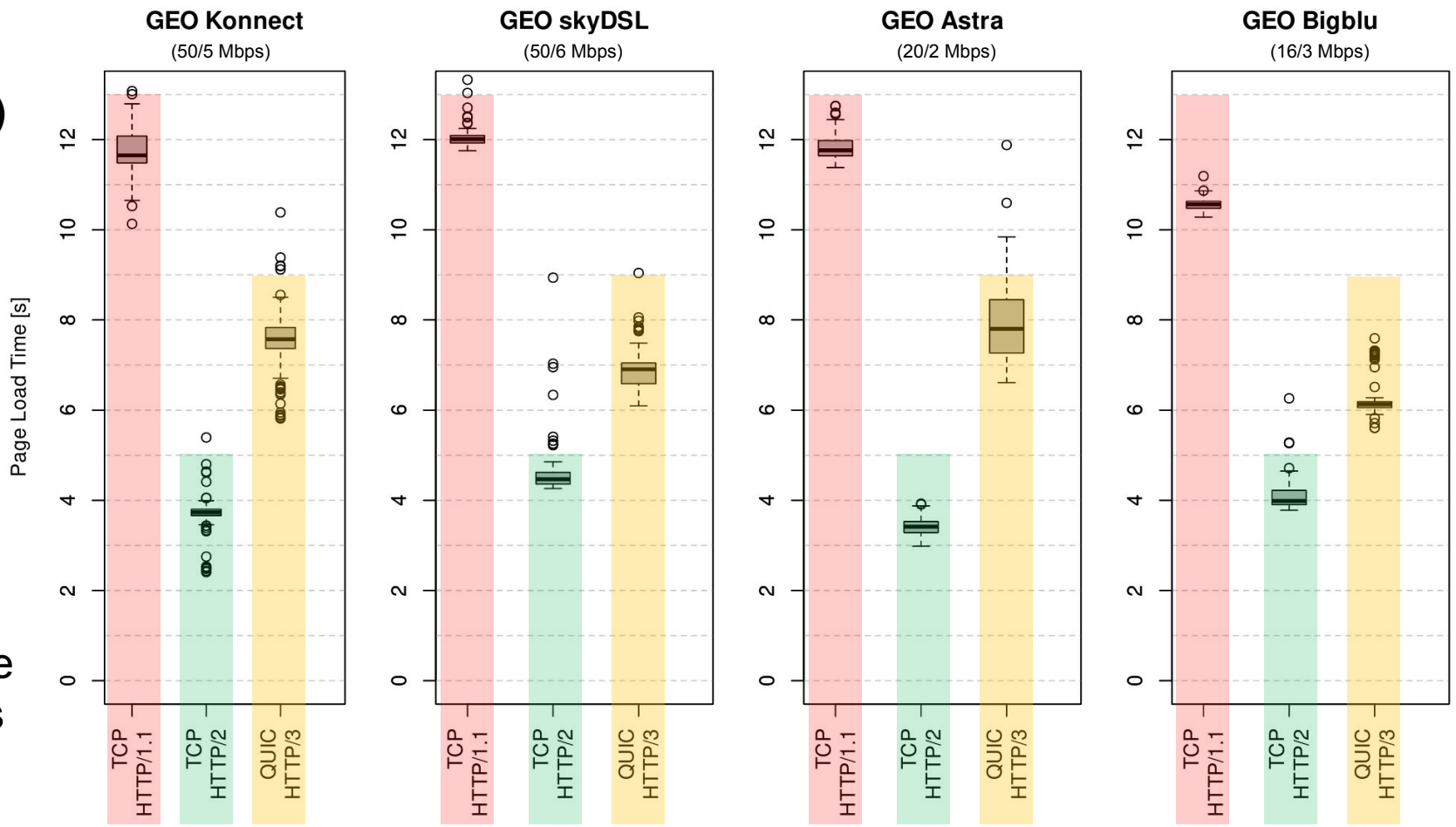
without VPN (with PEP)

HTTP/1.1
poor

HTTP/2
good

HTTP/3
mediocre

neglectable differences between operators



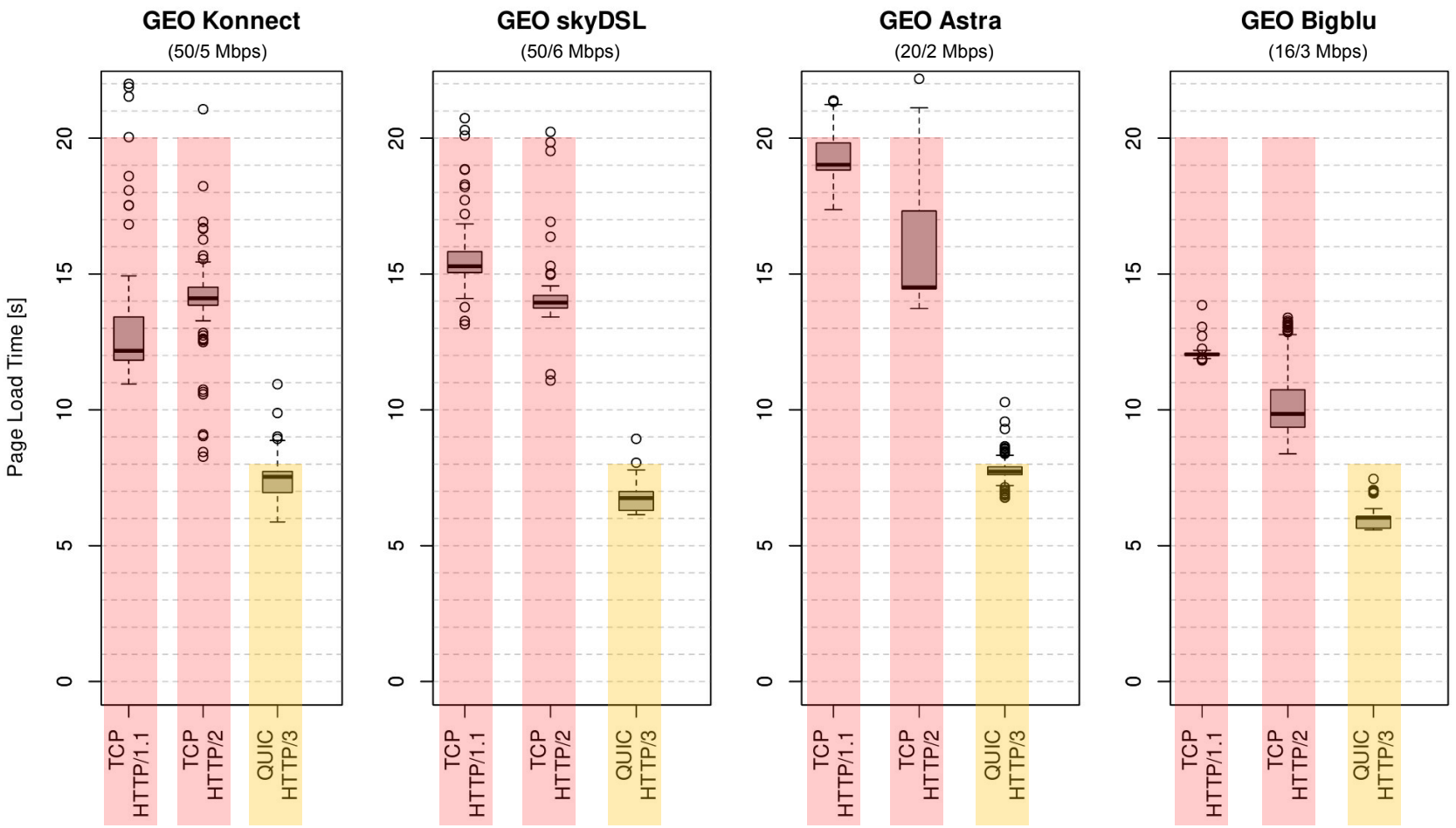
with VPN (without PEP)

HTTP/1.1
poor

HTTP/2
poor

HTTP/3
mediocre

some differences between operators

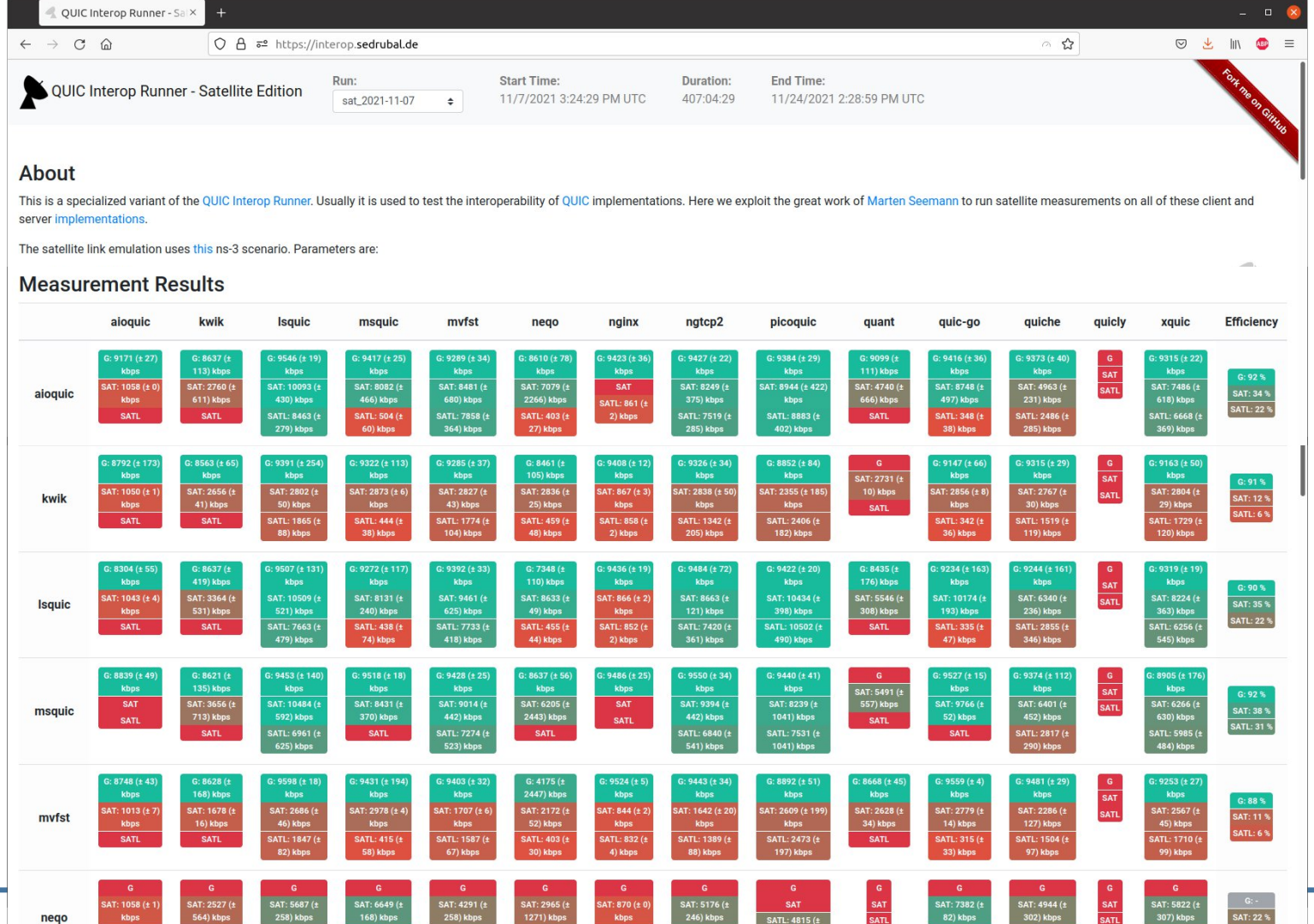


Outline

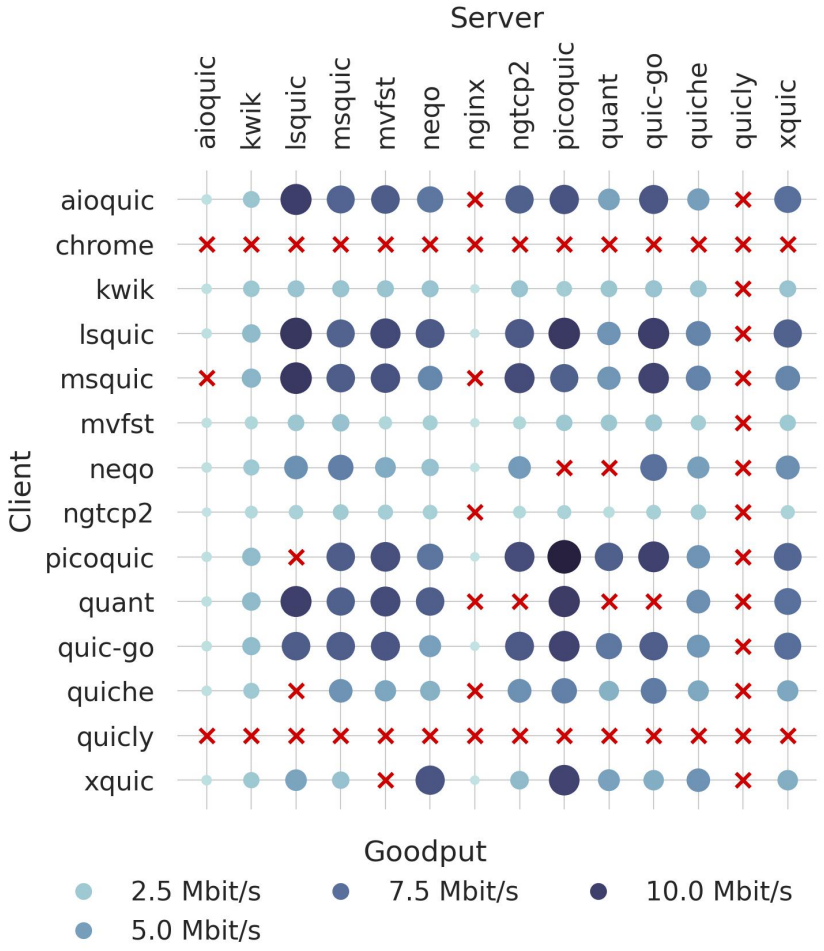
- Part 1
  - Simple black-box measurements
  - Varying object sizes
  - Real satellite operator, TCP with and without OpenVPN, quicly and picoquic
- Part 2
  - Simple black-box measurements
  - Different HTTP versions
  - Different real satellite operators, with and without Wireguard VPN
- Part 3
  - QUIC interop runner with geostationary satellite links
  - With and without loss, real satellite operators
  - Time-offset plots for detailed analysis

Part 3
QUIC interop runner with geostationary satellite links

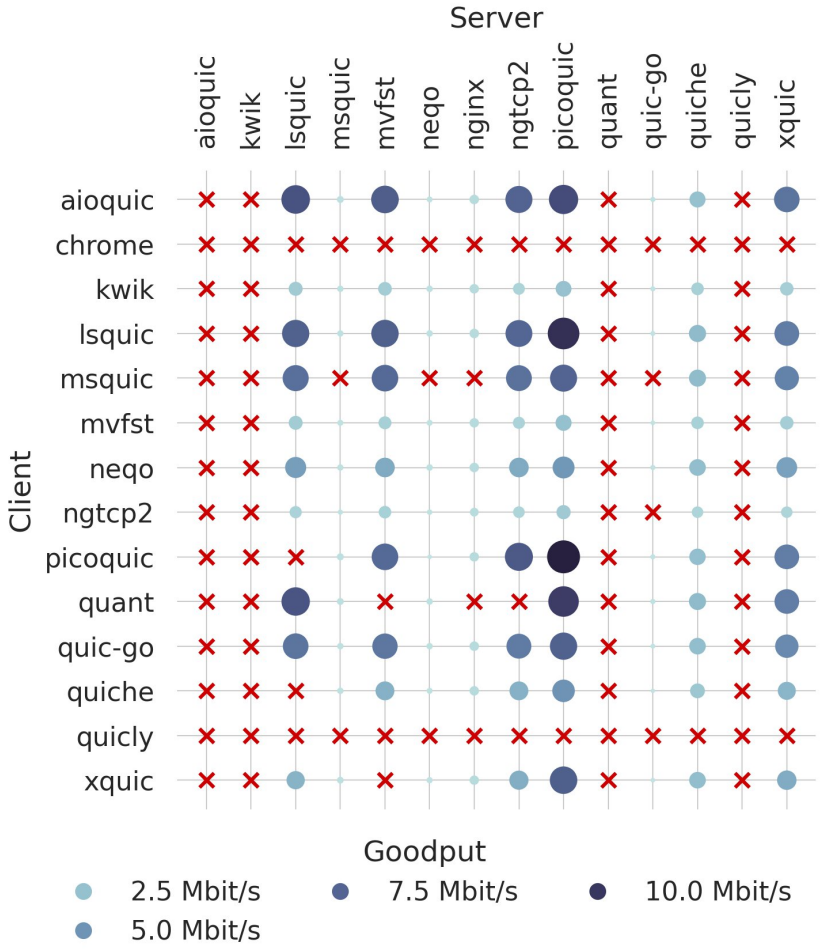
- QUIC interop runner https://interop.seemann.io
  - Interoperability testing of 14 QUIC implementations
  - Implementations available in Docker containers, link emulation with ns-3
  - Two performance-related measurements: goodput, crosstraffic
    - Client requests a 10 Mbyte file from the server
    - Goodput test: good performance for all implementations
- Added satellite related scenarios
  - Master thesis of Sebastian Endres (thanks!)
  - Link emulation with ns-3
    - SAT with 600ms RTT, 20 Mbps forward link, 2 Mbps return link
    - SATLOSS with additional 1% uniform loss rate
  - Running tests over real satellite links
  - Added time-offset and other diagrams generated from pcap traces


QUIC Interop Runner - Satellite Edition
Run:
sat_2021-11-07
Start Time:
11/7/2021 3:24:29 PM UTC
Duration:
407:04:29
End Time:
11/24/2021 2:28:59 PM UTC
Fork me on GitHub
About
This is a specialized variant of the QUIC Interop Runner. Usually it is used to test the interoperability of QUIC implementations. Here we exploit the great work of Marten Seemann to run satellite measurements on all of these client and server implementations.
The satellite link emulation uses this ns-3 scenario. Parameters are:
Measurement Results
aioquic
kwik
lsquic
msquic
mvfst
neqo
nginx
ngtcp2
picoquic
quant
quic-go
quiche
quicly
xquic
Efficiency

SAT (20/2 Mbps, 600ms RTT)



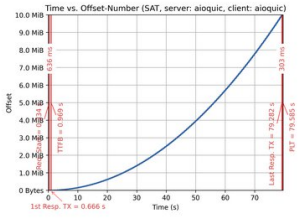
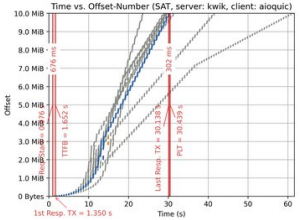
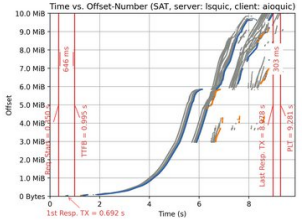
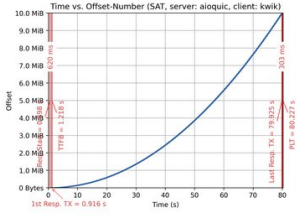
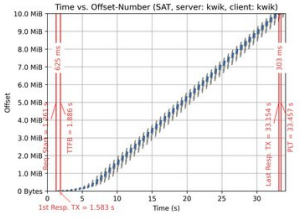
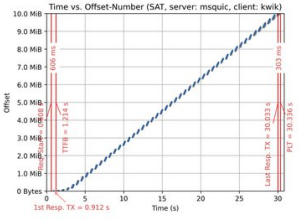
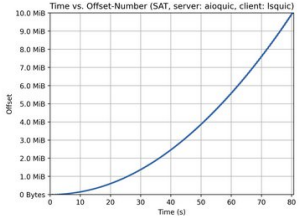
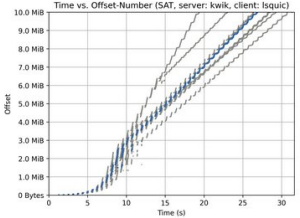
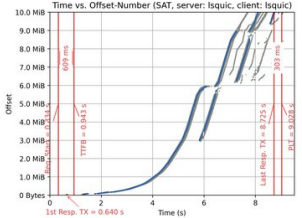
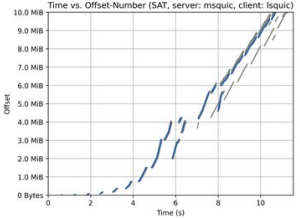
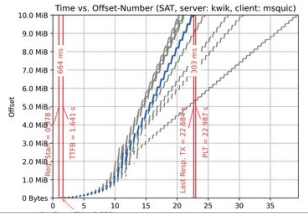
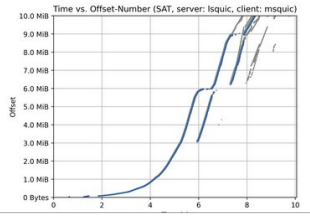
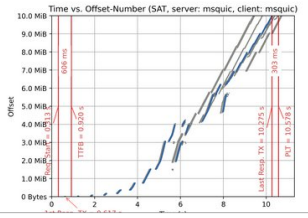
SAT (20/2 Mbps, 600ms RTT, 1% loss)

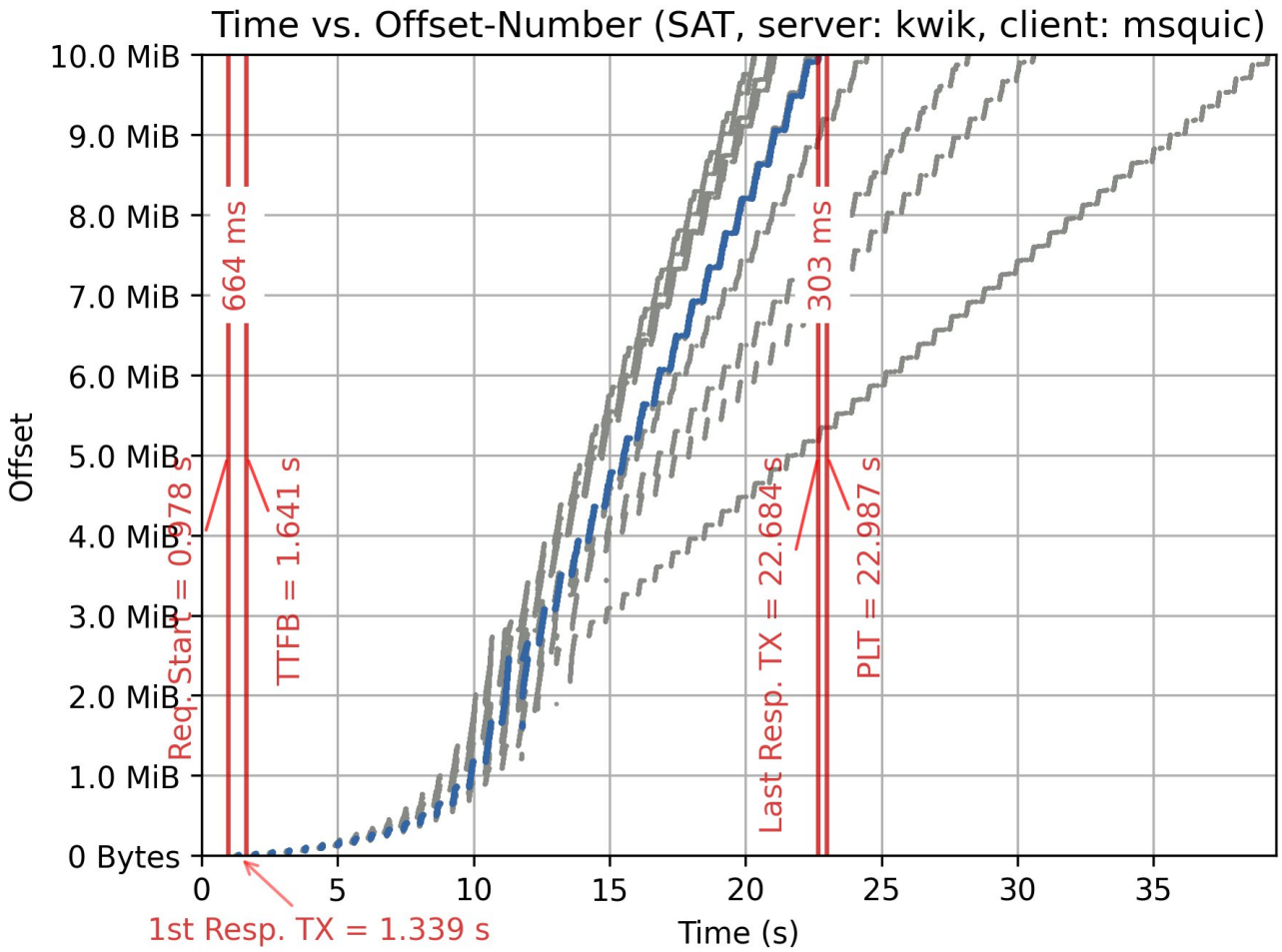


QUIC Interop Runner - Sa

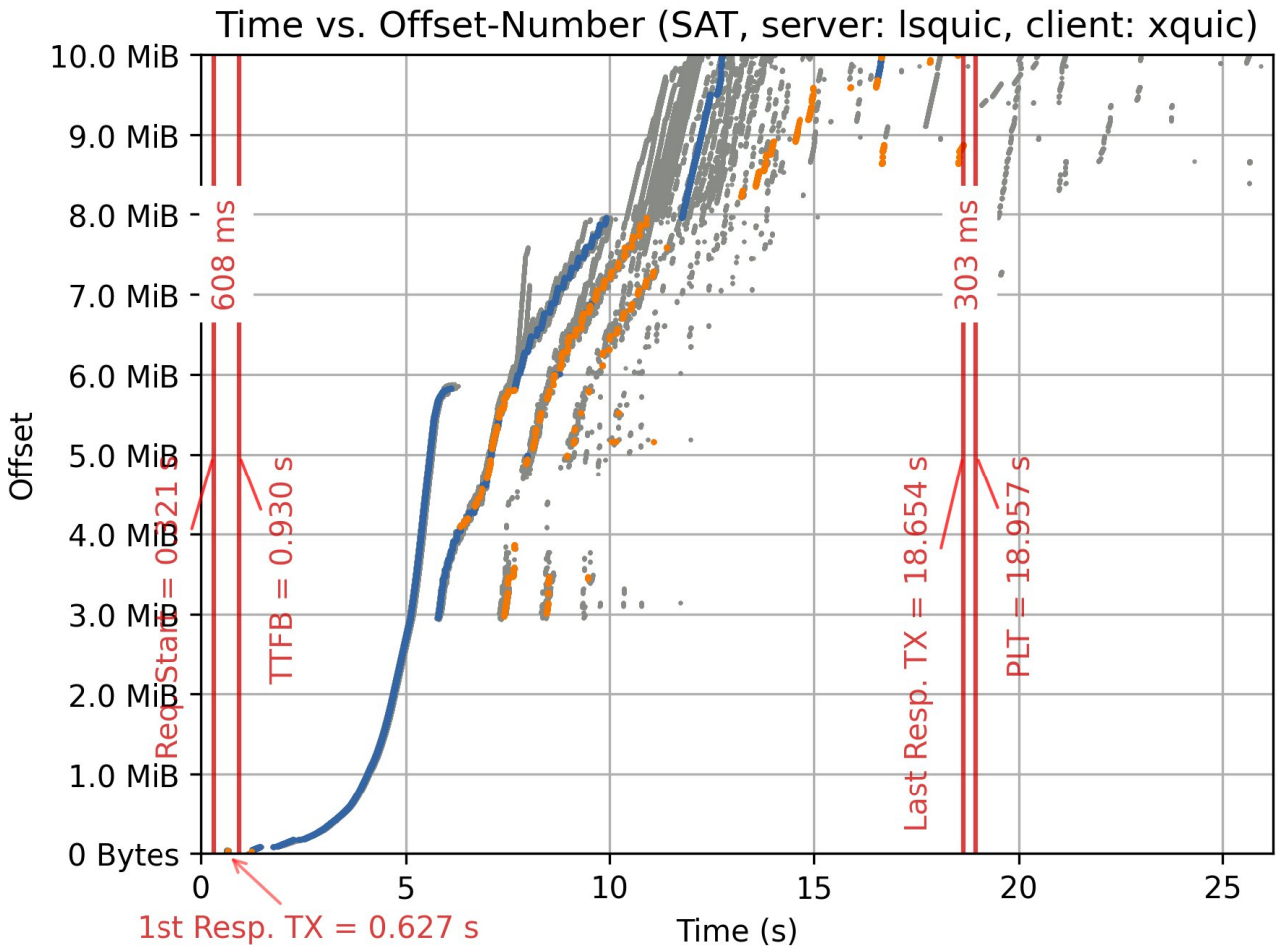
https://interop.sedrubal.de

90%

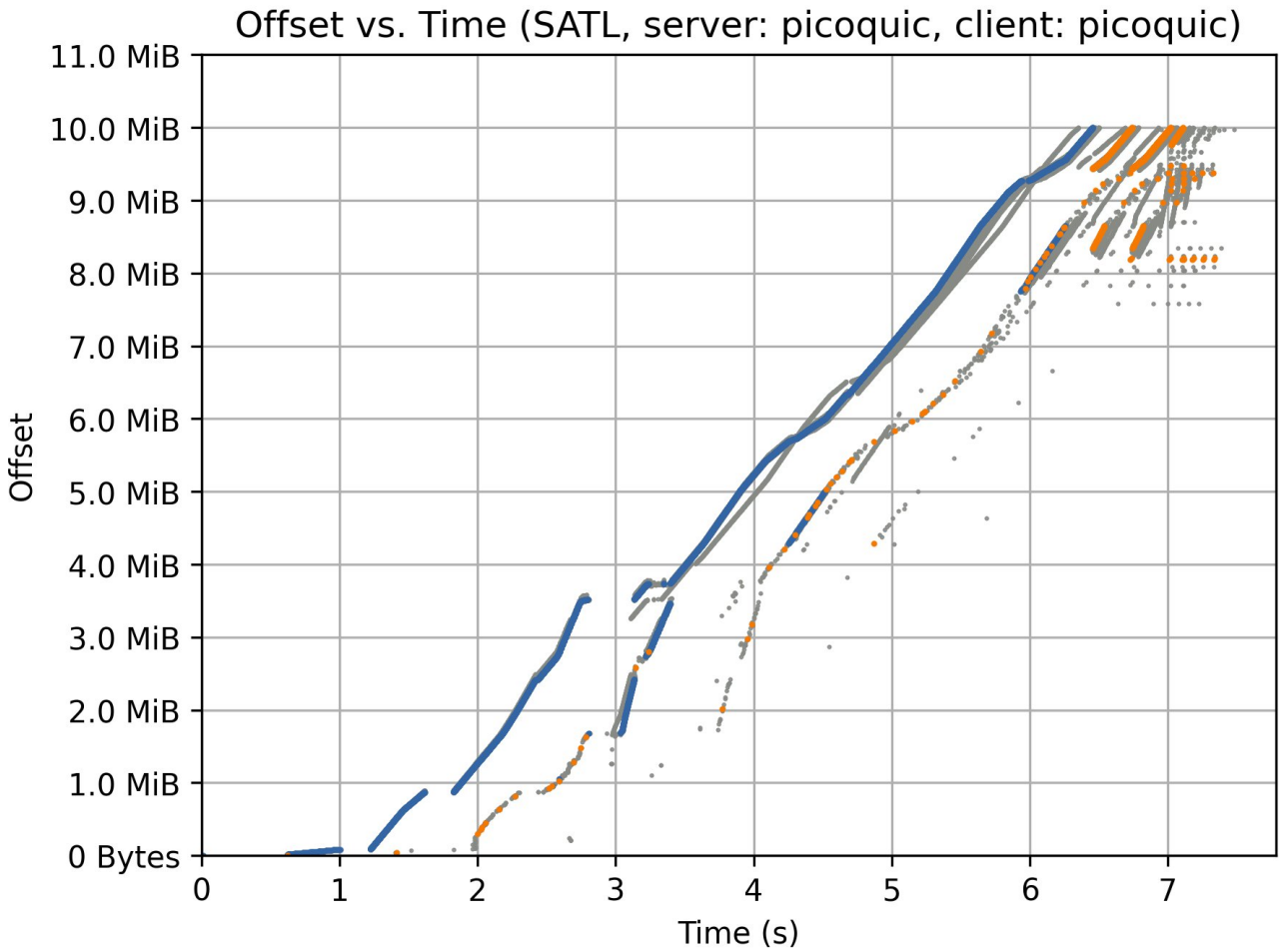
	aioquic	kwik	lsquic	msquic
aioquic				⊘
kwik			⊘	
lsquic				
msquic	Test Failed			



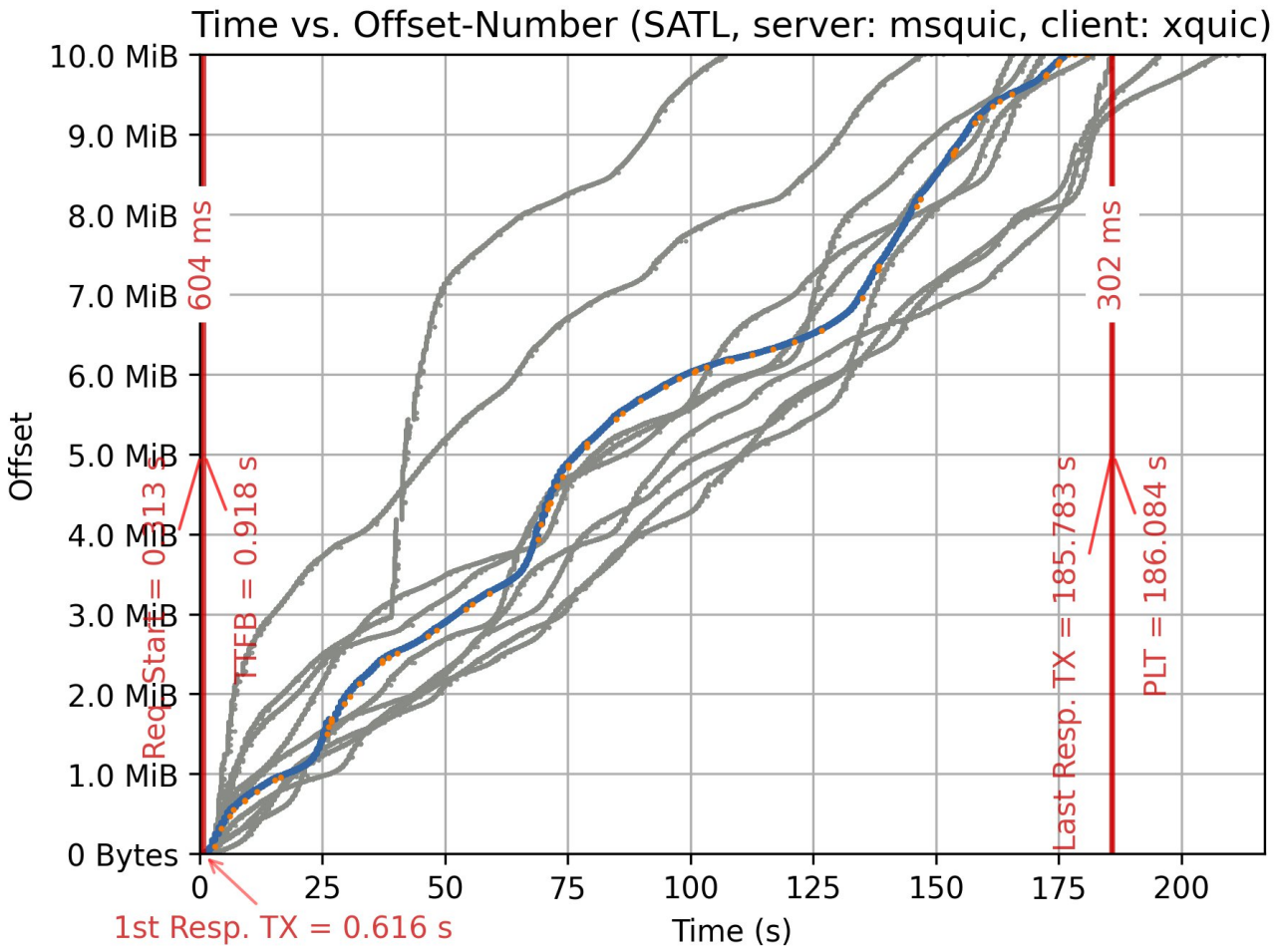
- satellite interop runner
  - 600ms RTT
  - 20/2 Mbit/s
  - no loss
- slow start-up
- varying outcomes



- satellite interop runner
  - 600ms RTT
  - 20/2 Mbit/s
  - no loss
- many retransmissions (orange points)
- non-sequential offset numbers? (parallel lines)



- satellite interop runner
  - 600ms RTT
  - 20/2 Mbit/s
  - 1% loss
- good performance (unlike other implementations in the SATLOSS scenario)



- satellite interop runner
  - 600ms RTT
  - 20/2 Mbit/s
  - 1% loss
- very poor performance

Part 3
QUIC interop runner with geostationary satellite links

- Preliminary results
  - Broad testing of many QUIC implementations
  - Work in progress
  - Some combinations fail due to timeouts or other reasons
  - Performance depends on client and server implementation
    - QUIC implementations are work in progress, too
    - maybe not all implementations strive for high-performance bulk data transfer and/or may only be used as proof of concept?
  - Time-offset plots
    - not always available due to faulty pcap traces
    - provide some insights into the behavior of the implementations
  - Will announce updates on EToSat mailing list

Summary

- Part 1
  - Simple black-box measurements
  - Varying object sizes
  - Real satellite operator, TCP with and without OpenVPN, quicly and picoquic
- Part 2
  - Simple black-box measurements
  - Different HTTP versions
  - Different real satellite operators, with and without Wireguard VPN
- Part 3
  - QUIC interop runner with geostationary satellite links
  - With and without loss, real satellite operators
  - Time-offset plots for detailed analysis