Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Scalable Online TCP Throughput Limitation Analysis

Towards Real-time TCP Throughput Root Cause Monitoring

Introduction

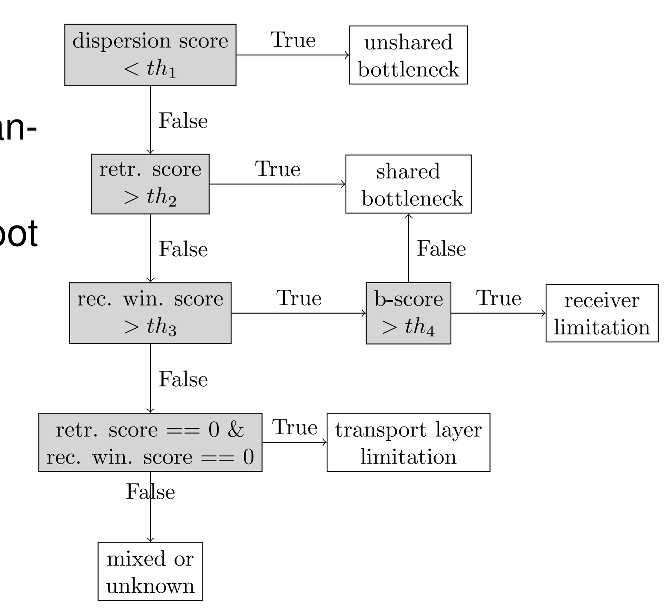
So far:

- Siekkinen et al. [5, 6]: Toolkit for the detection of Application Limited Periods (ALP) and different kinds of network limitations
- **Limitation:** Approach is based on analysis of full traffic captures → only offline analysis

Goal: Develop a tool to perform TCP throughput RCA in real-time

RCA Method:

- Calculate different scores, each indicating another root cause
- Use decision tree to determine actual root cause
- Scores:
  - Dispersion Score
  - Retransmission Score
  - Receiver Window Score
  - Burstiness Score



Design

Design goals:

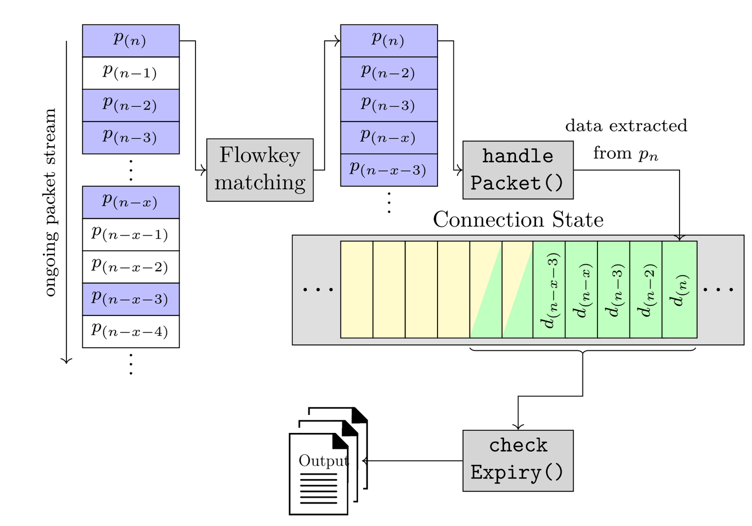
- Scalability and efficient resource consumption
- Modularity

Implementation based on **FlowScope** [1, 2]

- Capable of processing data rates up to 100 Gbit/s and beyond
- Per packet processing and periodically processing functionality

Modules:

- Position Estimation
- ALP Detection
- RTT Estimation
- Capacity Estimation
- RCA Score calculation



Data Set for Evaluation

Generated **data set** for the evaluation of capacity estimation and RCA classification with a TCP measurement framework by Jaeger et al. [4]

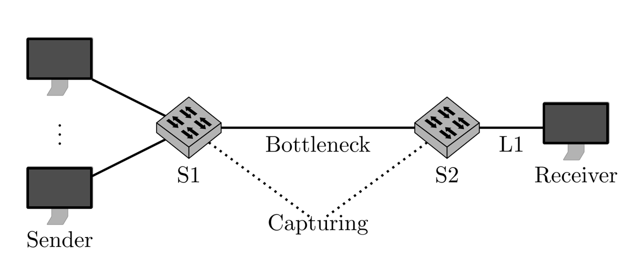
- Labeled data and reproducibility
- Allows to analyze a wide spectrum of test cases

Capacity estimation:

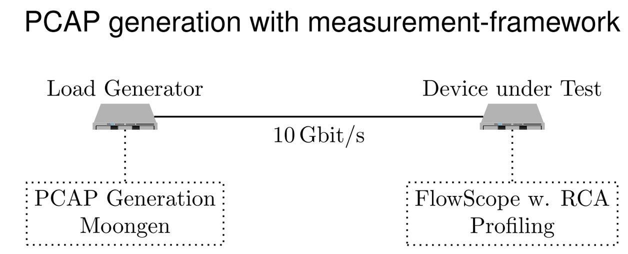
Test Case	Varied Parameters	Values
Cong. Con. Algo.	TCP Algorithm	Reno, Cubic, BBR
Packet loss	Loss on the bottleneck	0% - 25%
Capacity	Bottleneck bandwidth	5 Mbit/s - 100 Mbit/s
Concurrent flows	Number of concurrent flows	1-25

RCA estimation:

Test Case	C.C. Algo.	RTT in ms	Loss in %
Unshared BN	Reno, Cubic, BBR	10 - 300	0, 0.001, 0.01
Shared BN	Reno, Cubic, BBR	10 - 300	0, 0.001, 0.01
Receiver Win.	Reno, Cubic, BBR	50 - 300	0, 0.001, 0.01
Congestion Win.	Reno, Cubic, BBR	150 - 400	0, 0.001, 0.01



PCAP generation with measurement-framework



Test setup in the testbed

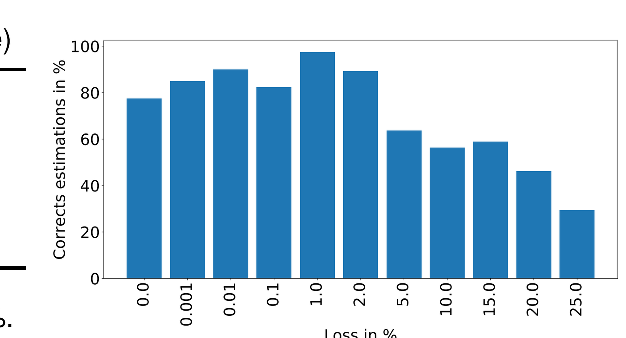
Capacity Estimation Module

Implemented a **passive capacity estimation** module based on former research by En-Najjary et al. [3]

- Works with packet pair dispersion derived from inter-arrival times (IAT)
- Supports estimation near the client and near the server
- More suitable method is selected based on position, to ensure that analyzed packets passed the capacity bottleneck

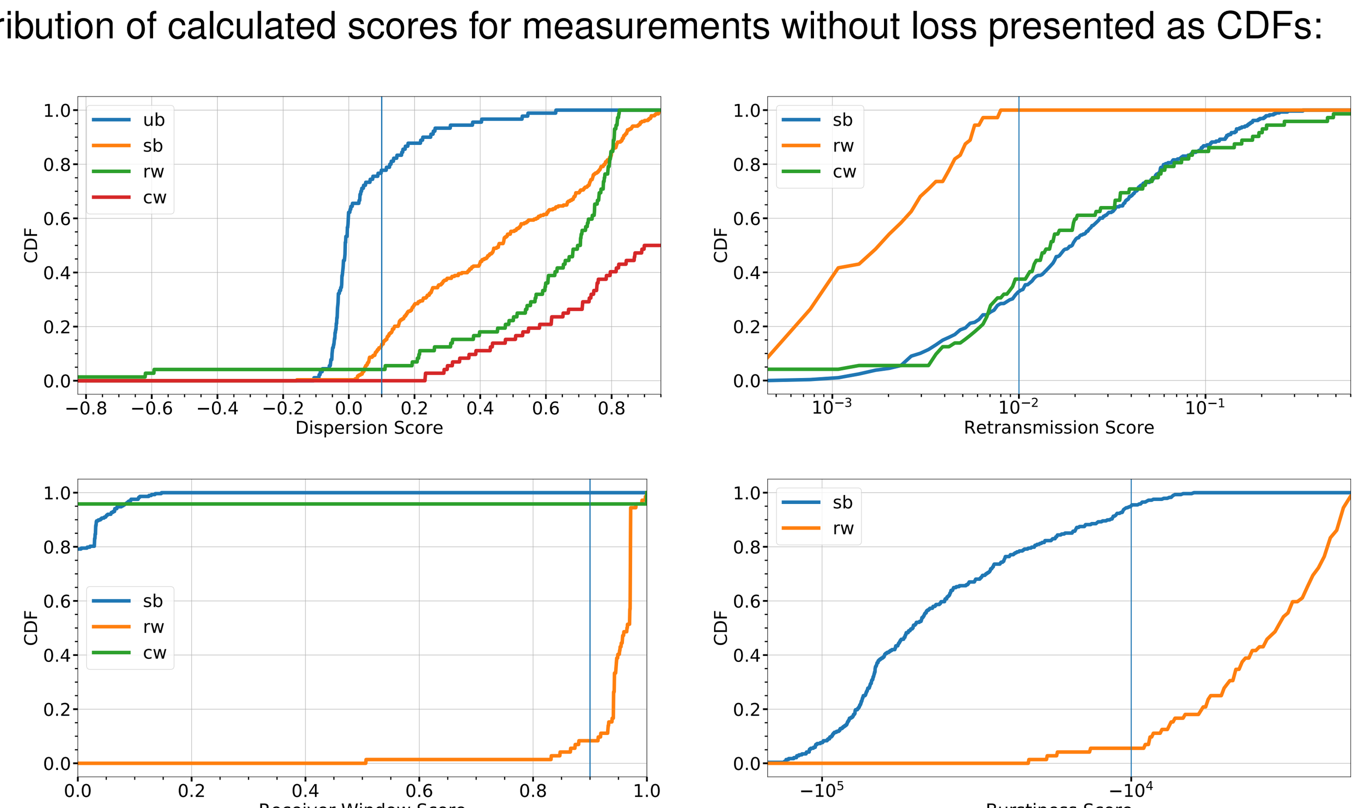
Test Case	Bottleneck	Acc. (client side)	Acc. (server side)
Flow Duration	10mbit	99.0%	100%
TCP Algorithm	10mbit	96.7%	93.3%
Loss	10mbit	78.1%	77.2%
Capacity	5-100mbit	70.0%	81.4%
Concurrent Flows	10mbit	98.4%	97.6%

Measured accuracy for client and receiver side measurements.
Estimate is classified as 'accurate' if the relative error is less than 5%.



Effectiveness

Distribution of calculated scores for measurements without loss presented as CDFs:



- Score modules return expected values
- Trade-off for threshold values: more false-positives vs. less correct estimates
- Insufficient data sets for test cases limited by the transport layer, due to retransmissions during TCP slow start.

Performance Considerations

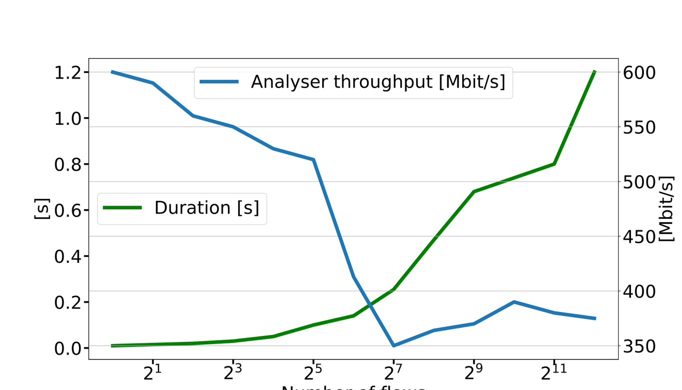
Analyzed **runtime of the expiry check** for each module

- Capacity estimation is very expensive, big potential for improvement

Potential **performance limitations** of our tool:

- **Memory:** Fix connection state size of 156 KB → 160 GB memory required for 1 million concurrent flows
- **CPU:** Measured throughput with one single analyzer thread (without capacity estimation)
  - Significant throughput decline when aggregated connection state size converges LLC size
  - LLC approximately filled with 64 flows

Module	Runtime
Connection identification	21 ns
ALP detection	6 ns
Position estimation	2 ns
RTT estimation	9 ns
Capacity estimation	15167 ns
Dispersion score calculation	94 ns
Retransmission score calculation	13 ns
Receiver window score calculation	2536 ns
Burstiness score calculation	21 ns



[1] P. Emmerich, M. Pudelko, S. Gallenmüller, and G. Carle. Flowscope: Efficient packet capture and storage in 100 gbit/s networks. In *2017 IFIP Networking Conference (IFIP Networking) and Workshops*, pages 1–9, June 2017.
[2] P. Emmerich, M. Pudelko, Q. Scheitle, and G. Carle. Efficient dynamic flow tracking for packet analyzers. pages 1–6, 10 2018.
[3] T. En-Najjary and G. Urvoy-Keller. Pprate: A passive capacity estimation tool. In *2006 4th IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services*, pages 82–89, April 2006.
[4] B. Jaeger, D. Scholz, D. Raumer, F. Geyer, and G. Carle. Reproducible Measurements of TCP BBR Congestion Control. *Computer Communications*, May 2019.
[5] M. Siekkinen, G. Urvoy-Keller, and E. W. Biersack. On the interaction between internet applications and tcp. In *Proceedings of the 20th International Teletraffic Conference on Managing Traffic Performance in Converged Networks*, ITC20'07, pages 962–973, Berlin, Heidelberg, 2007. Springer-Verlag.
[6] M. Siekkinen, G. Urvoy-Keller, E. W. Biersack, and D. Collange. A root cause analysis toolkit for tcp. *Comput. Netw.*, 52(9):1846–1858, June 2008.

Simon Bauer, Kilian Holzinger, Benedikt Jaeger, Paul Emmerich and Georg Carle
{ bauer | holzingk | jaeger | emmericp | carle }@net.in.tum.de