SCA Based Implementation of STANAG 4285 in a Joint Effort under the NATO RTO/IST Panel

Sarvpreet Singh, Marc Adrat, Stefan Couturier, Markus Antweiler

FGAN/FKIE, Germany
singh@fgan.de

Martin Phisel, Steve Bernier

CRC, Canada
phisel@crc.ca

SDR'08, Technical Conference and Product Exposition
Washington, D.C., October 26th – 30th, 2008

Outline

- NATO Group
- Motivation
- Tools and Waveform
- SCA Implementations (Different Granularity)
- Profiling Results
- Conclusion

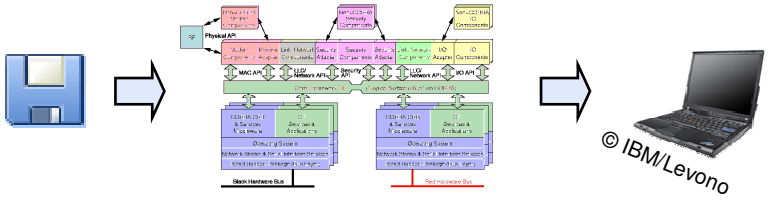
NATO RTO/IST-080 RTG-038 on SDR

- NATO RTO/IST RTG on SDR (estd. 2007)
  - Research and Technology Organisation (RTO)
  - Informations Systems Technology (IST)
  - Regular Task Group (RTG)
- Main Objectives
  - Share knowledge and experience on SCA/SDR Development
  - Share waveforms and SCA based waveform components
  - Report results to others groups like NC3B SDRUG or SDR Forum

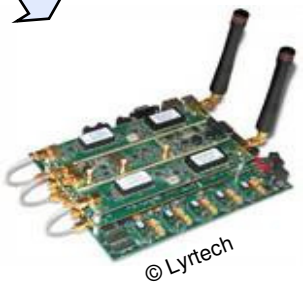
Motivation

- Target Workflow

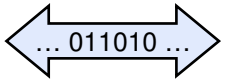
1. SCA-based implementation of STANAG 4285 waveform

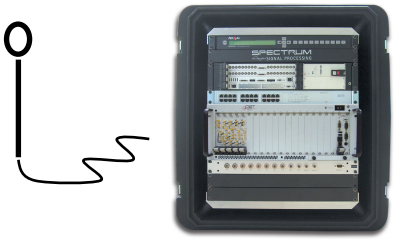


2. demonstrate portability onto national SDR platforms



3. demonstrate interoperability between the different implementations



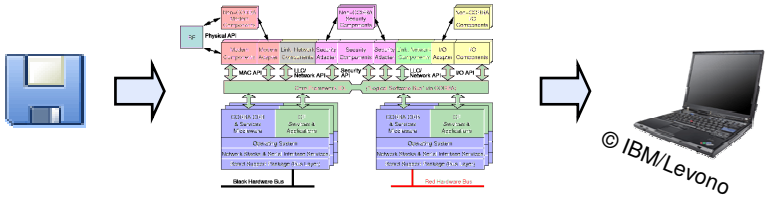


© Spectrum Signal Processing

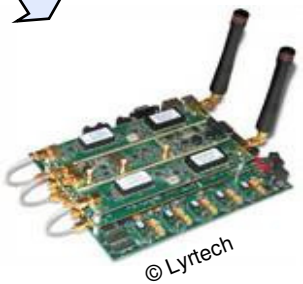
Motivation

- Target Workflow

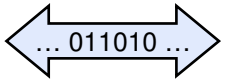
1. SCA-based implementation of STANAG 4285 waveform

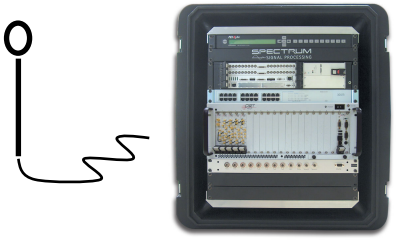


2. demonstrate portability onto national SDR platforms



3. demonstrate interoperability between the different implementations

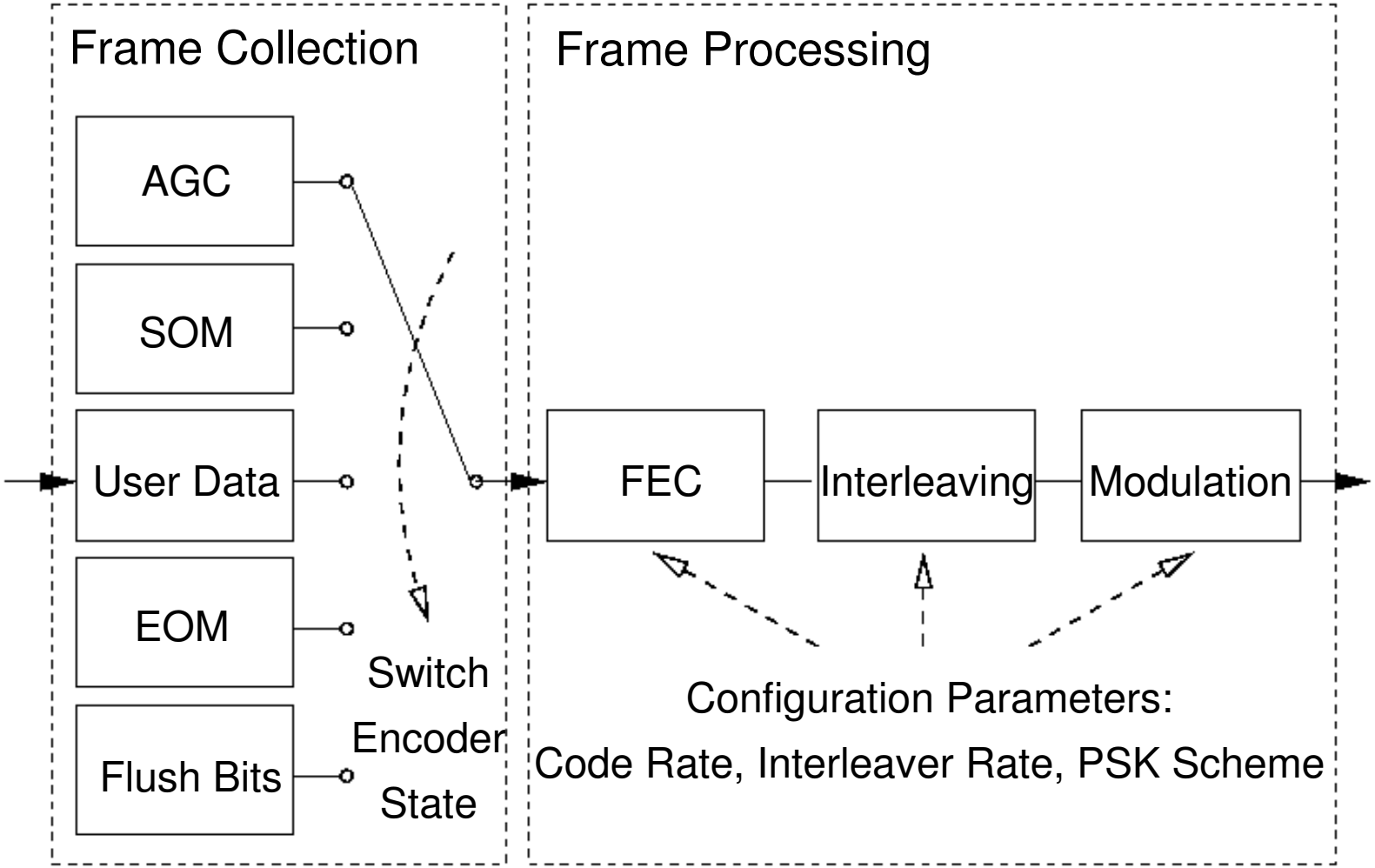




© Spectrum Signal Processing

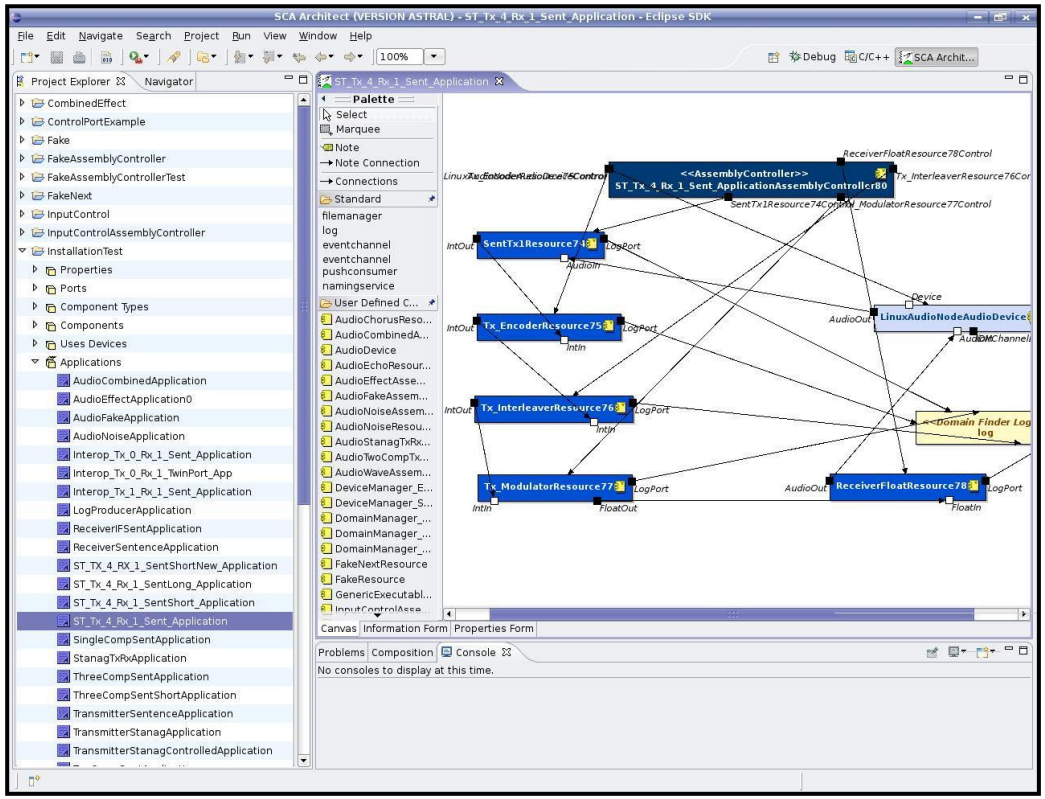
STANAG 4285 – A NATO HF Waveform

- Offers six modes between 75 bits/sec and 2.4 kbits/sec



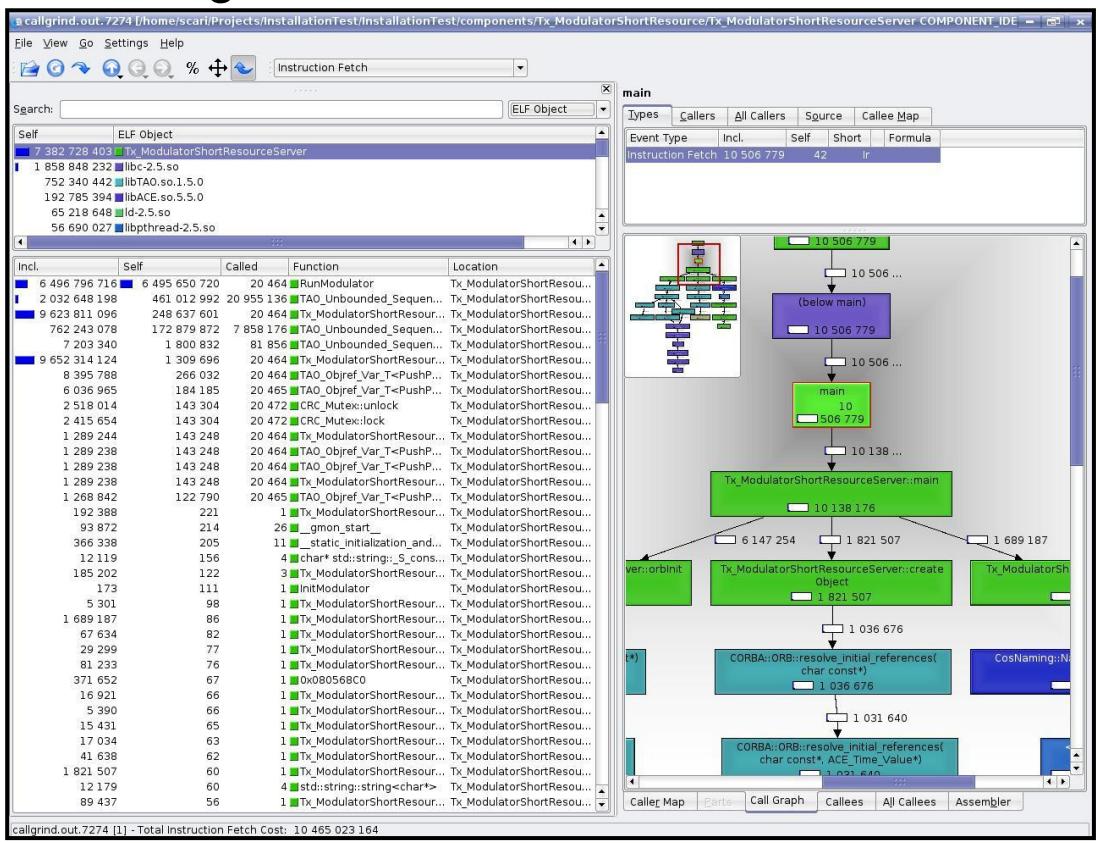
SCA Architect (CRC, Canada)

- An IDE as a plug-in to Eclipse Framework
- SCA Compliant Component Development
- Create Graphical Models of various Elements
- Source Code Generation
- Assemble elements into applications and nodes
- Focus more on self functionality and not on SCA



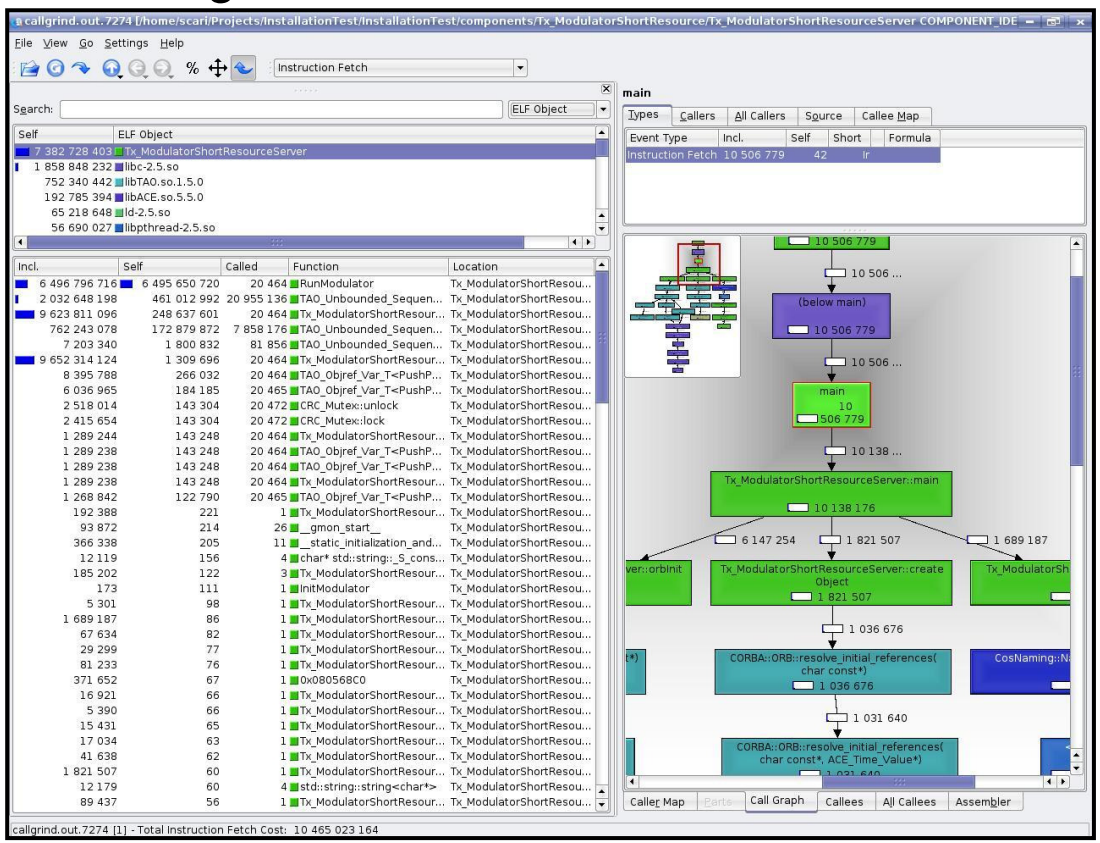
Profiling Tool – Valgrind (Open Source)

- Valgrind – Framework for building dynamic analysis tools
  - Detect Memory Management Bugs
  - Detect Threading Bugs
  - Program Profiling
- Callgrind – Cache Profiler
  - Cache misses
  - Memory References
  - Instructions Executed
- KCachegrind – Visualizer



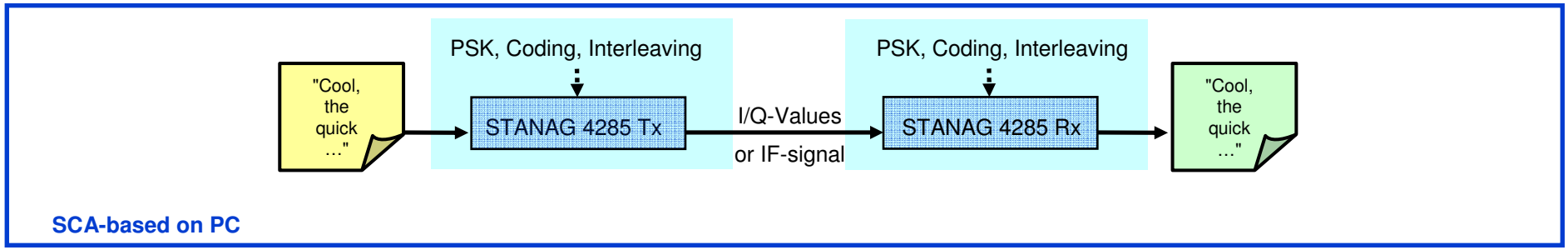
Profiling Tool – Valgrind (Open Source)

- Valgrind – Framework for building dynamic analysis tools
  - Detect Memory Management Bugs
  - Detect Threading Bugs
  - Program Profiling
- Callgrind – Cache Profiler
  - Cache misses
  - Memory References
  - Instructions Executed
- KCachegrind – Visualizer



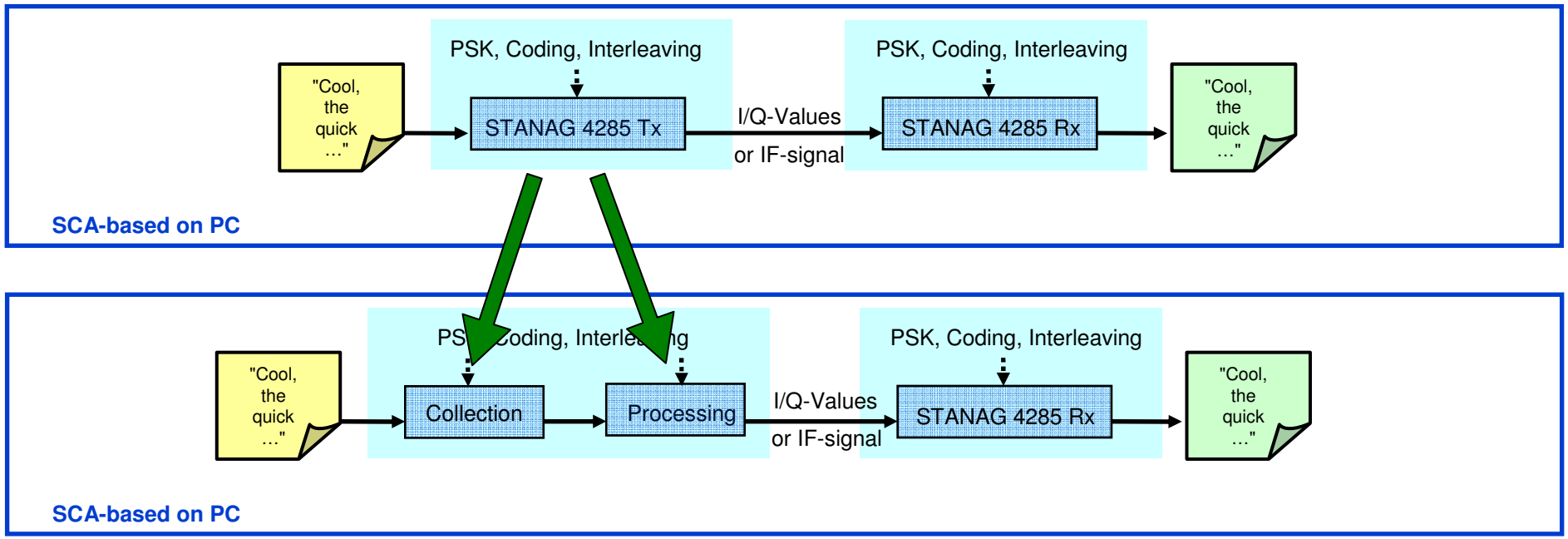
SCA based Implementation of STANAG 4285

- SCA based implementations of different TX granularity level



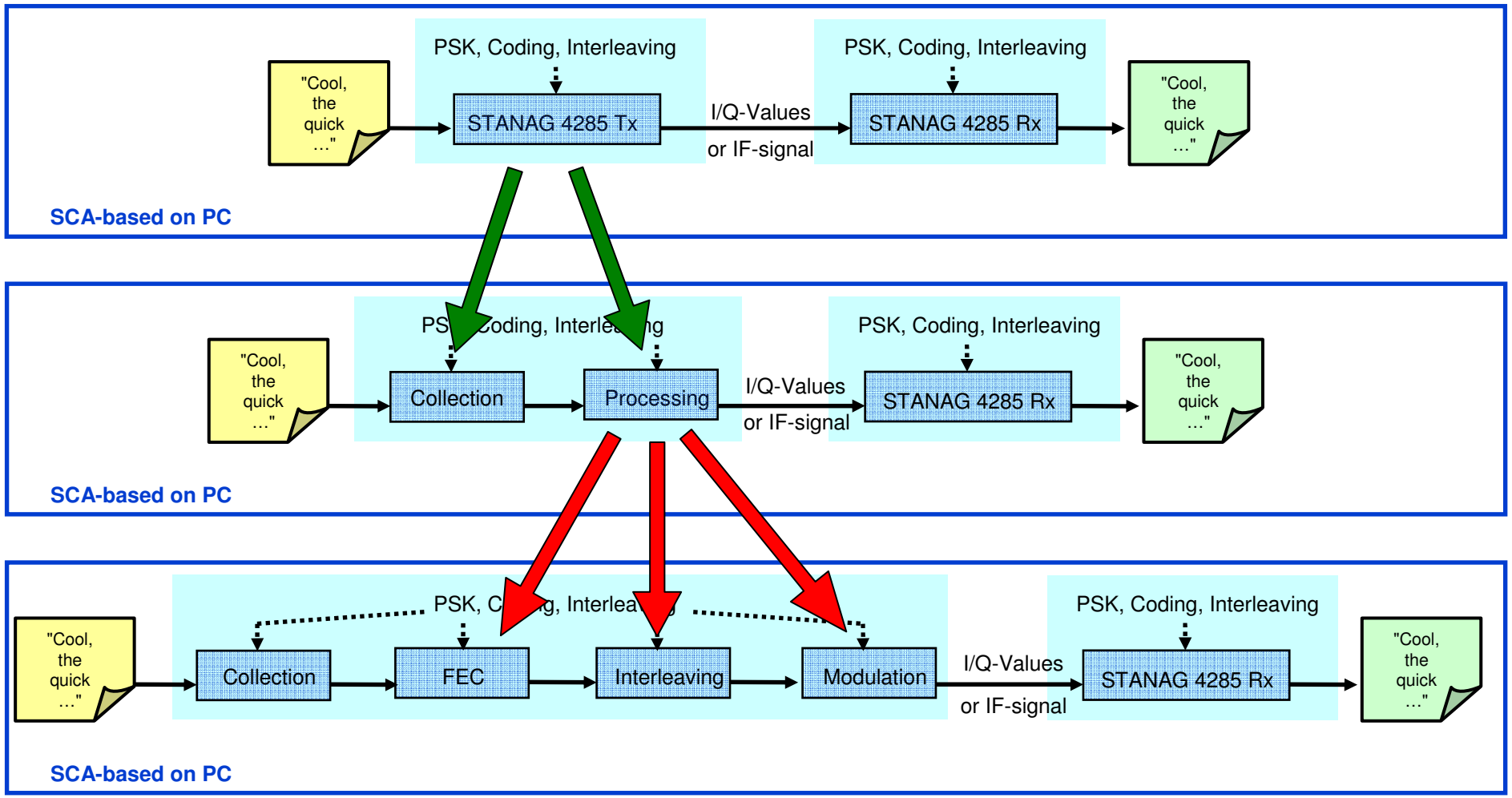
SCA based Implementation of STANAG 4285

- SCA based implementations of different TX granularity level

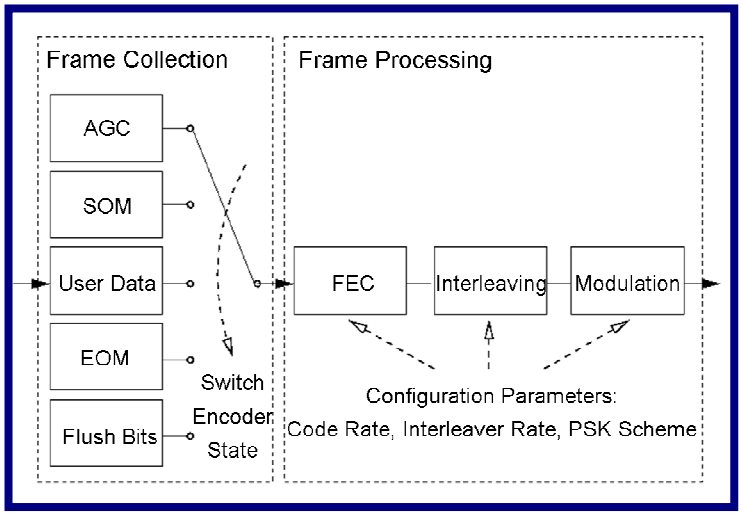


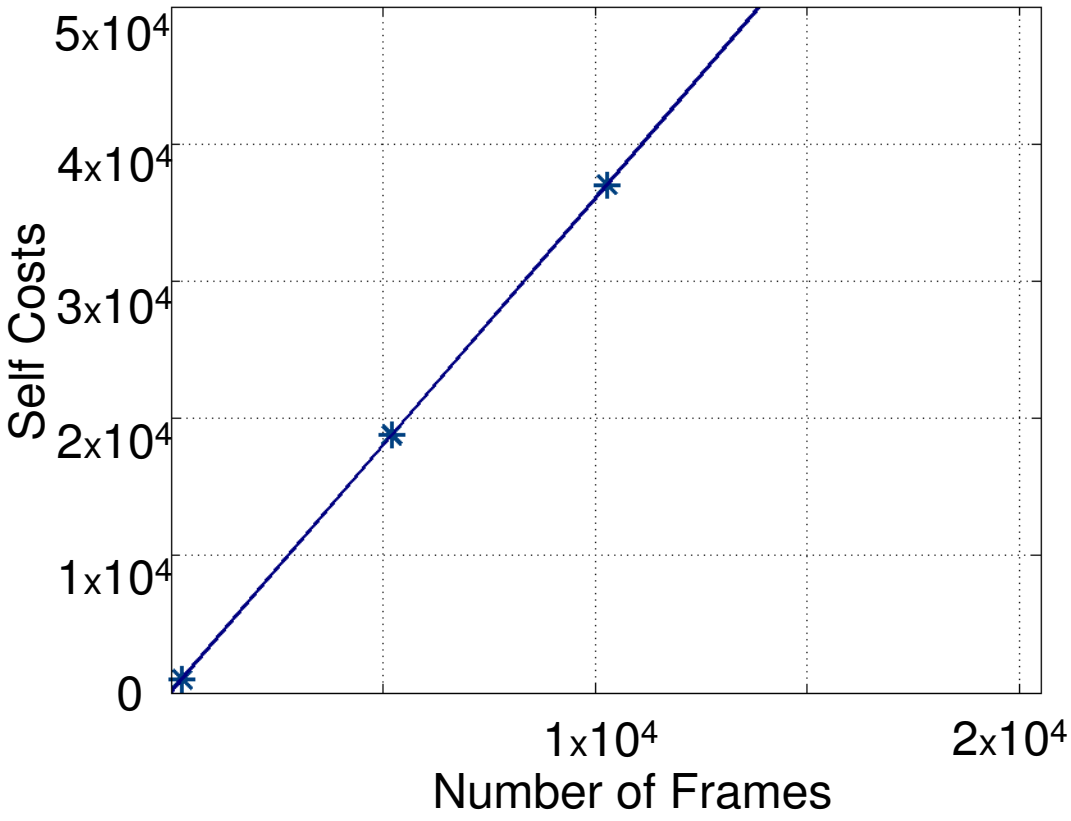
SCA based Implementation of STANAG 4285

- SCA based implementations of different TX granularity level



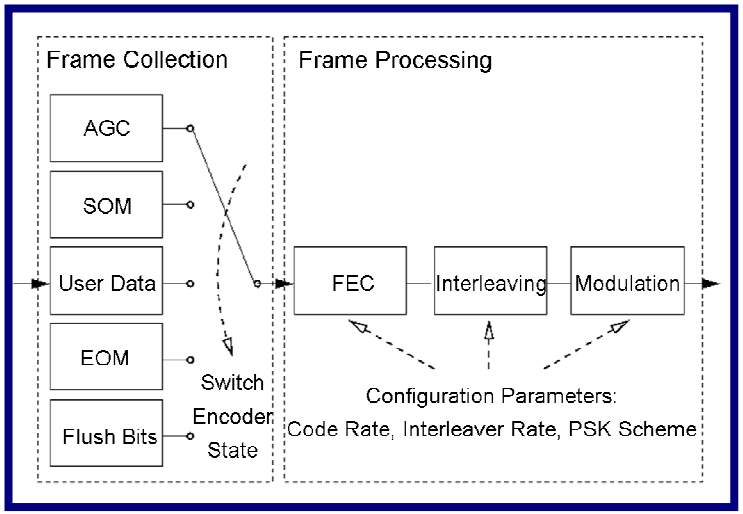
Profiling Results for 1 Component TX

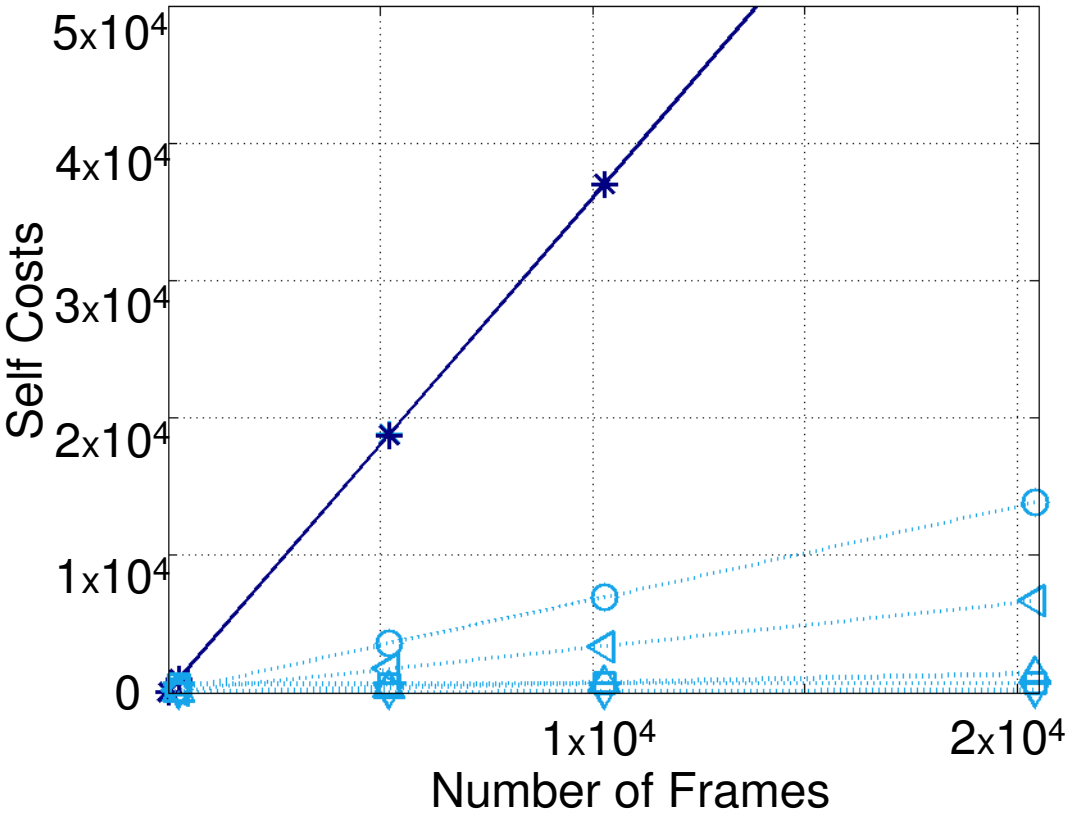




✳	**Exe**	◁	**TAO**
△	**ACE**	○	**C/C++**
▽	**SCA**	□	**ld.so**

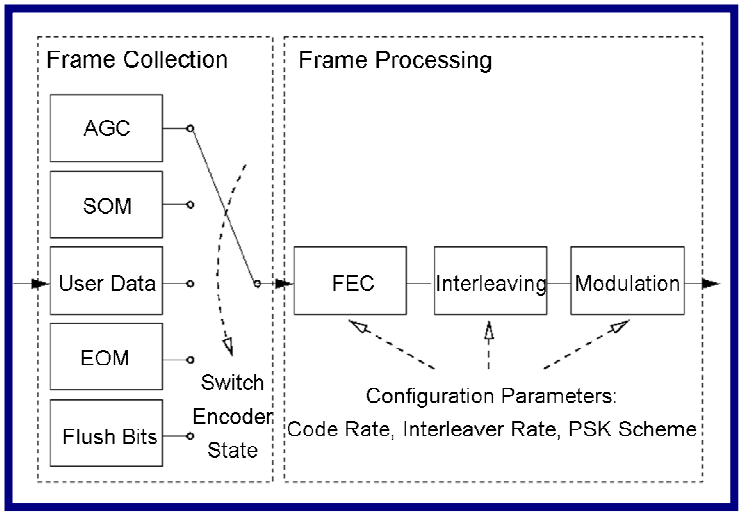
Profiling Results for 1 Component TX

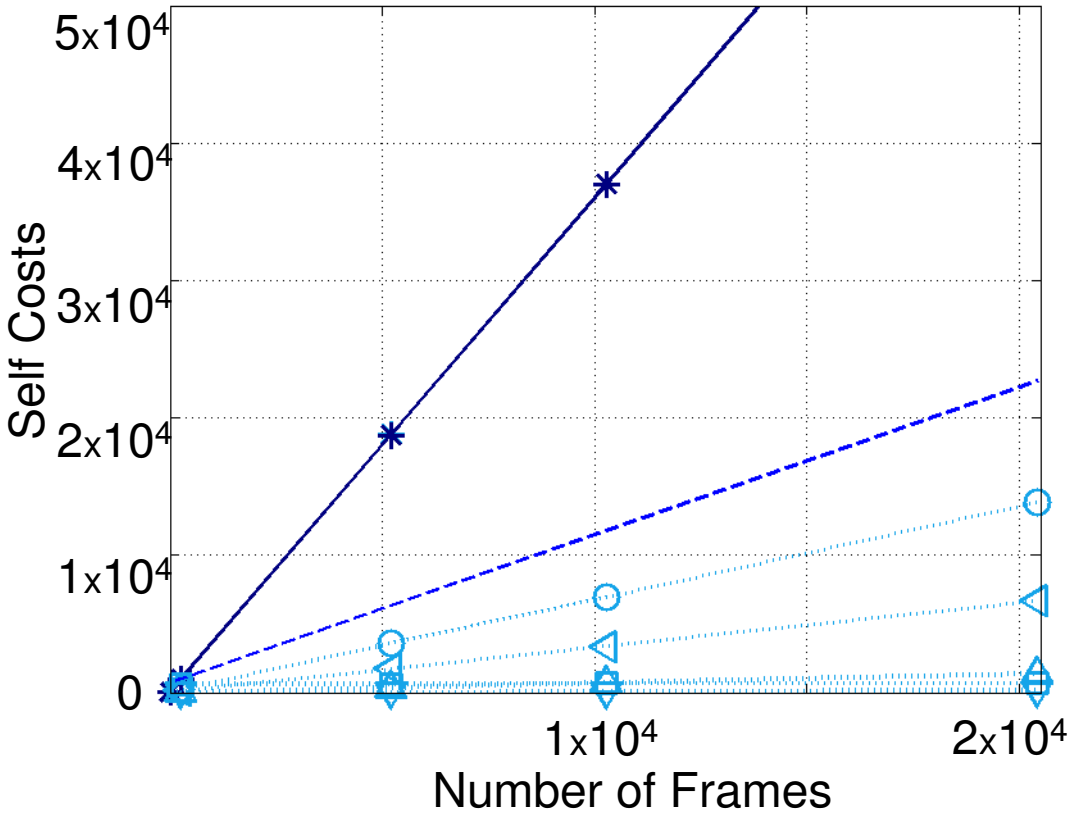


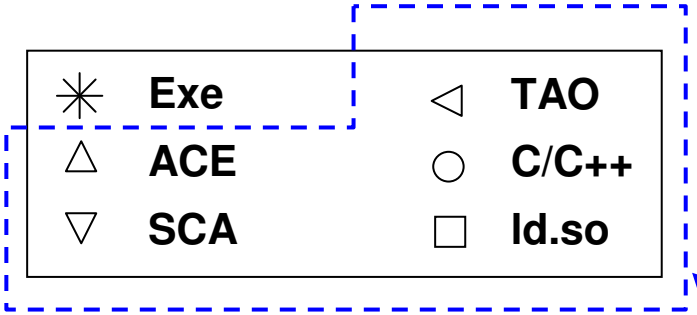


✳	**Exe**	◁	**TAO**
△	**ACE**	○	**C/C++**
▽	**SCA**	□	**ld.so**

Profiling Results for 1 Component TX

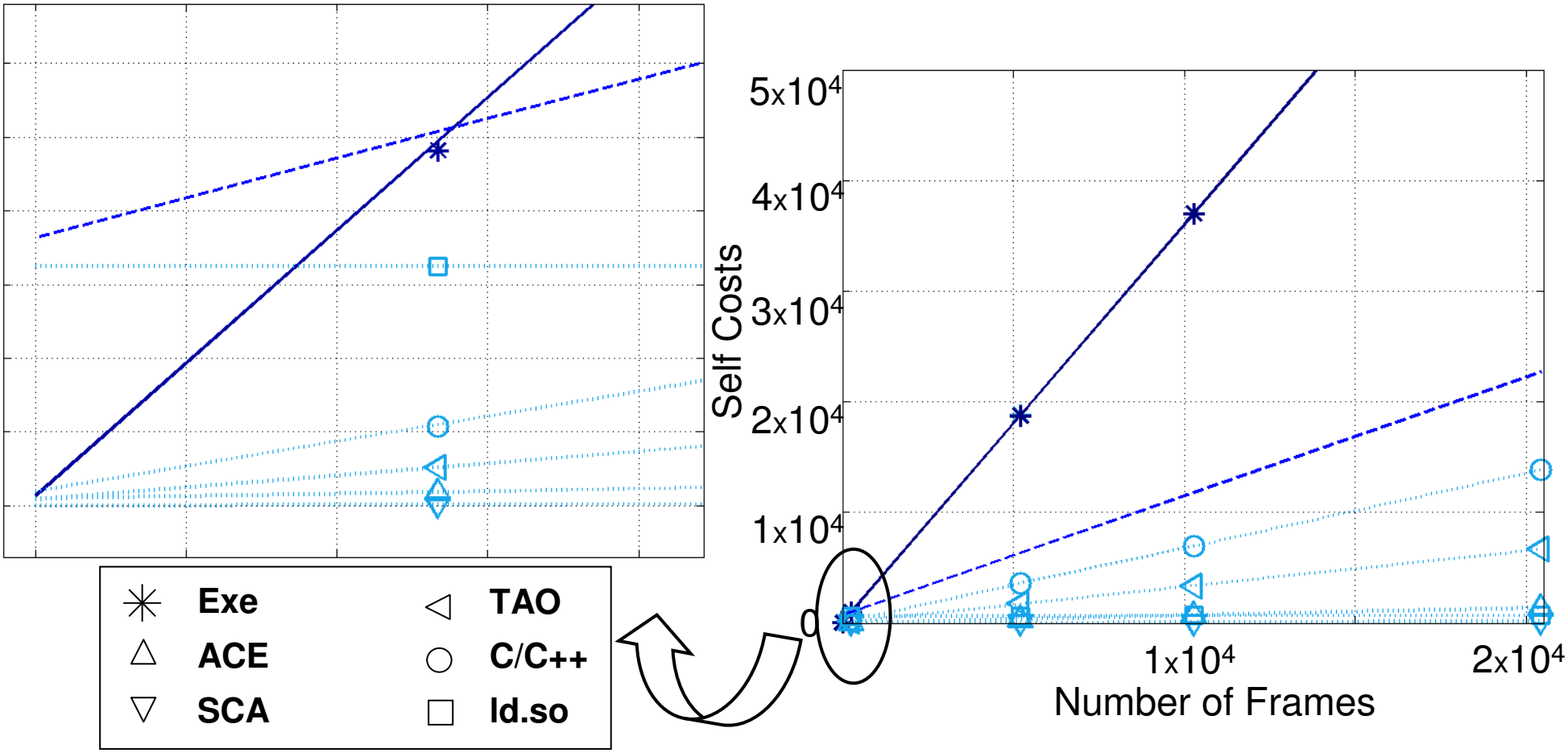




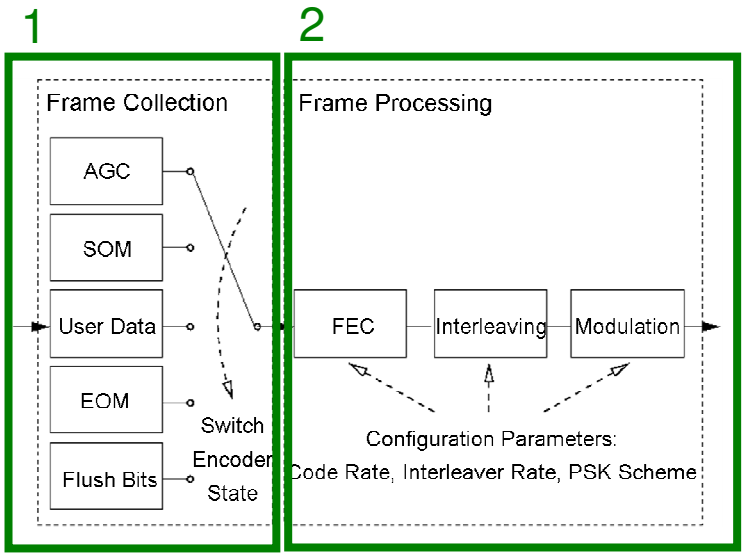


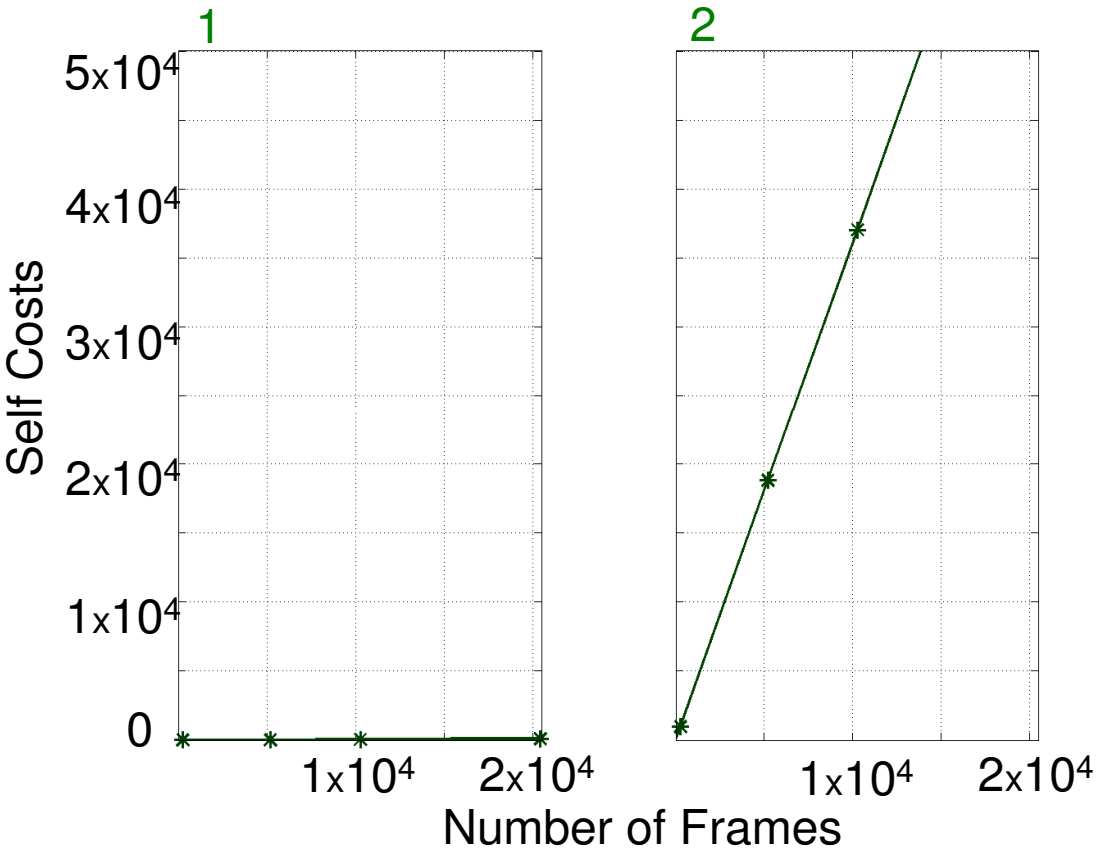
Overhead

Profiling Results for 1 Component TX



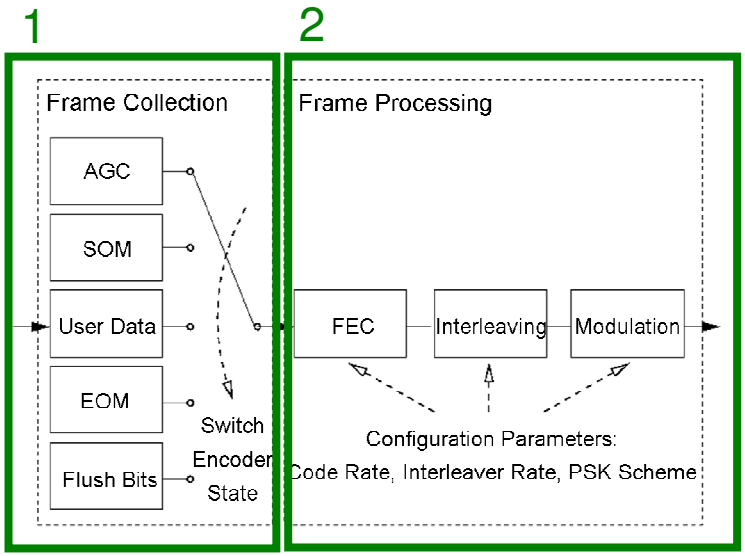
Profiling Results for 2 Component TX

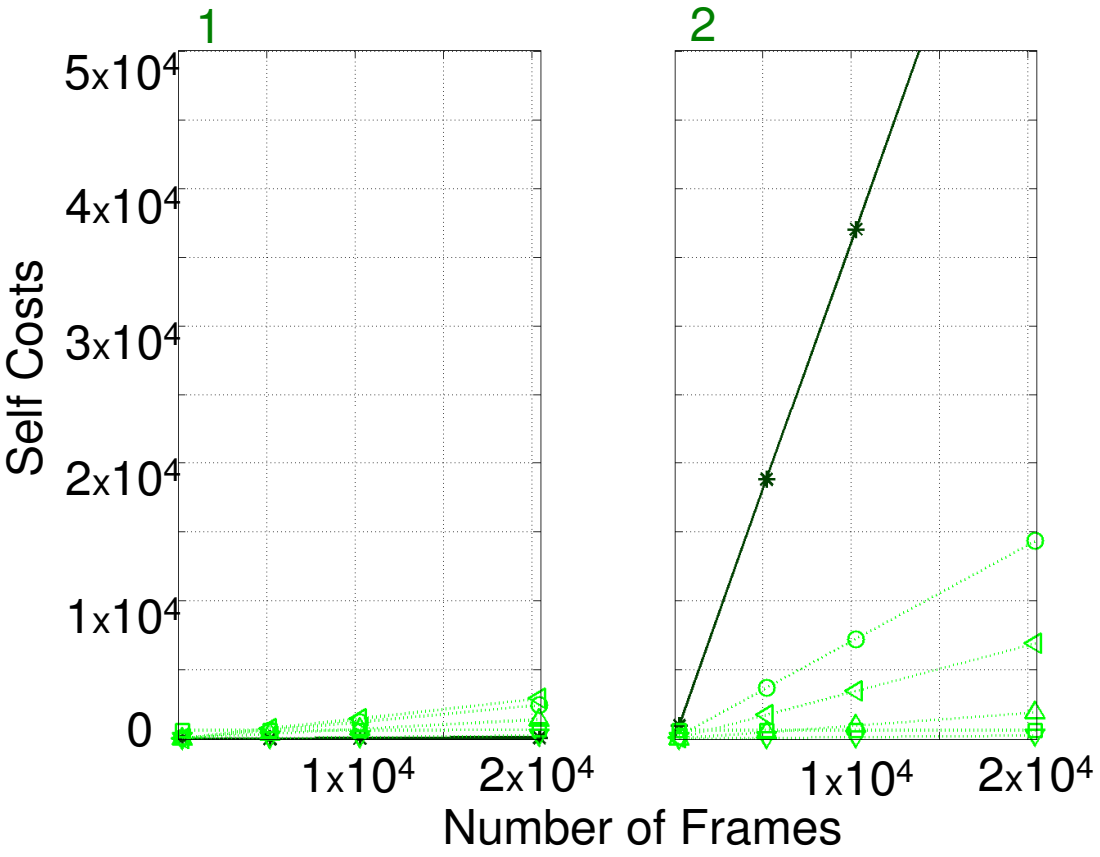




✳	**Exe**	◁	**TAO**
△	**ACE**	○	**C/C++**
▽	**SCA**	□	**ld.so**

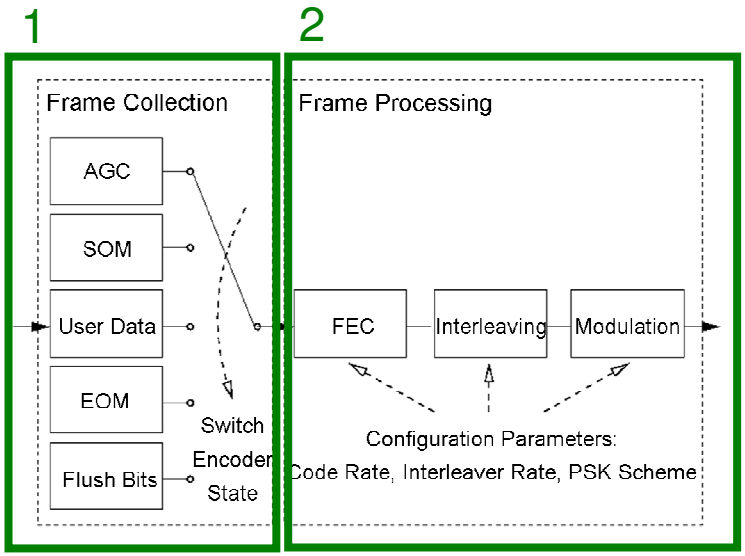
Profiling Results for 2 Component TX

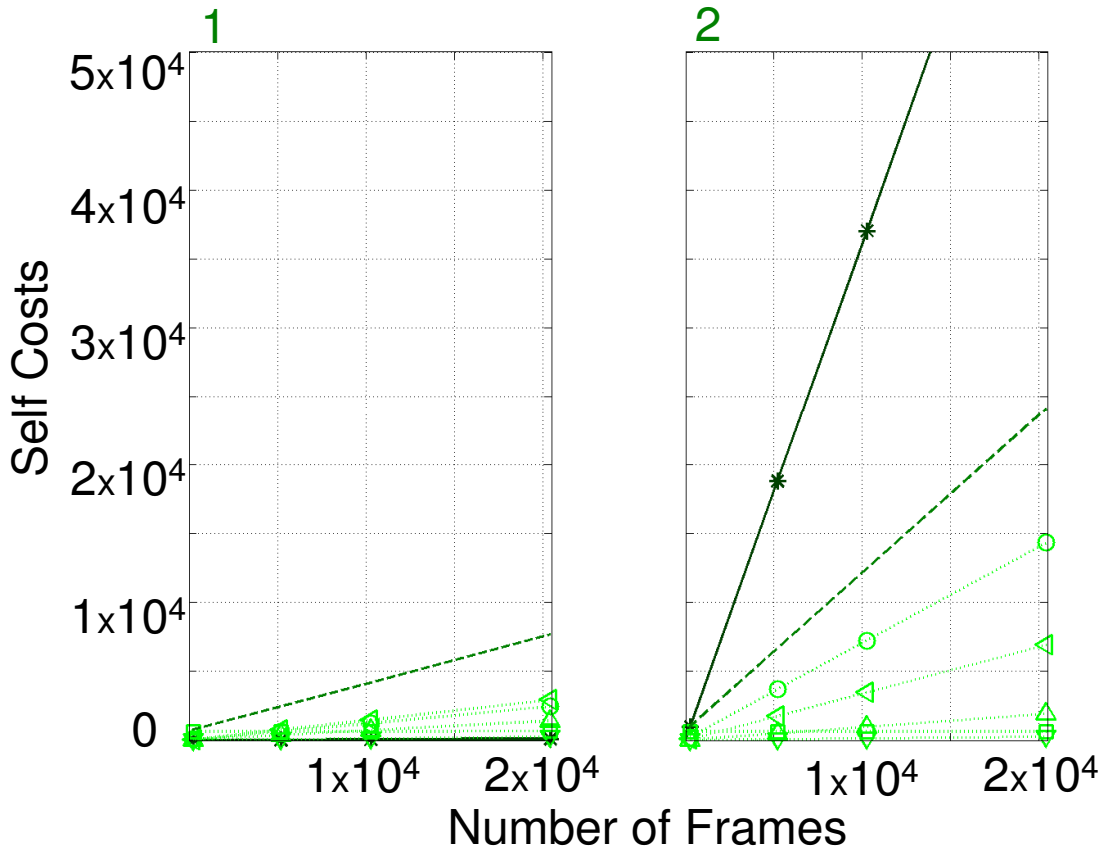




✳	**Exe**	◁	**TAO**
△	**ACE**	○	**C/C++**
▽	**SCA**	□	**ld.so**

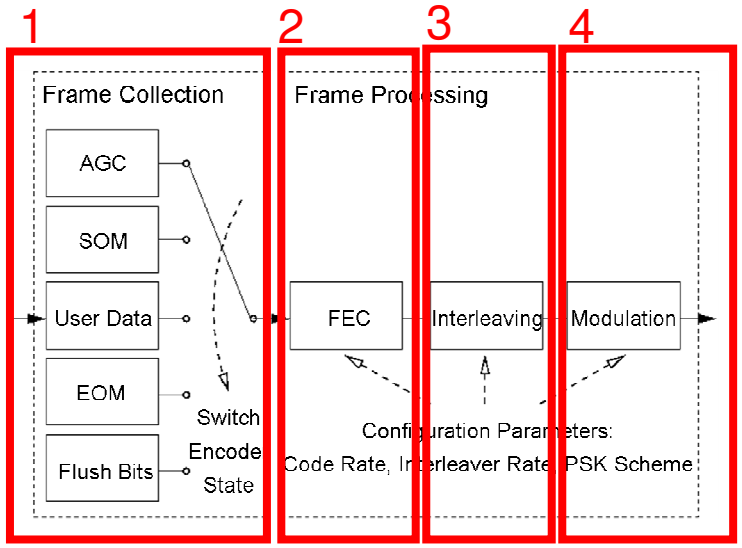
Profiling Results for 2 Component TX

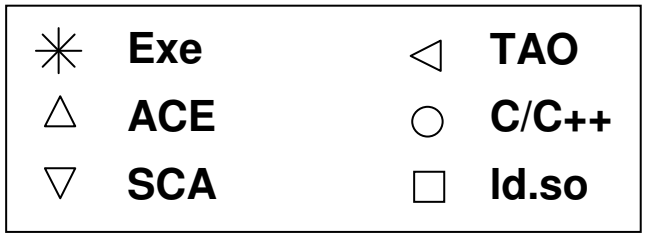


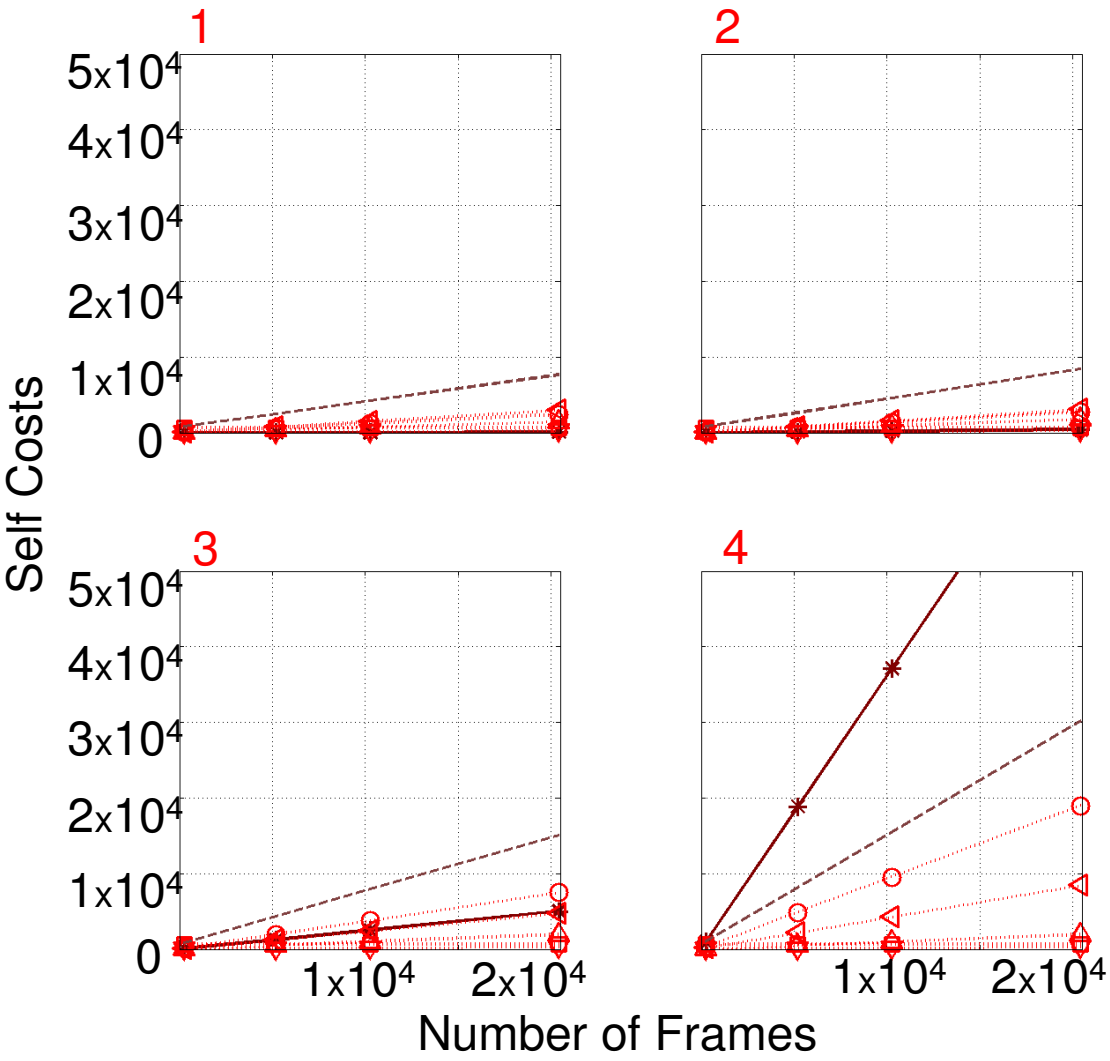


✳	**Exe**	◁	**TAO**
△	**ACE**	○	**C/C++**
▽	**SCA**	□	**ld.so**

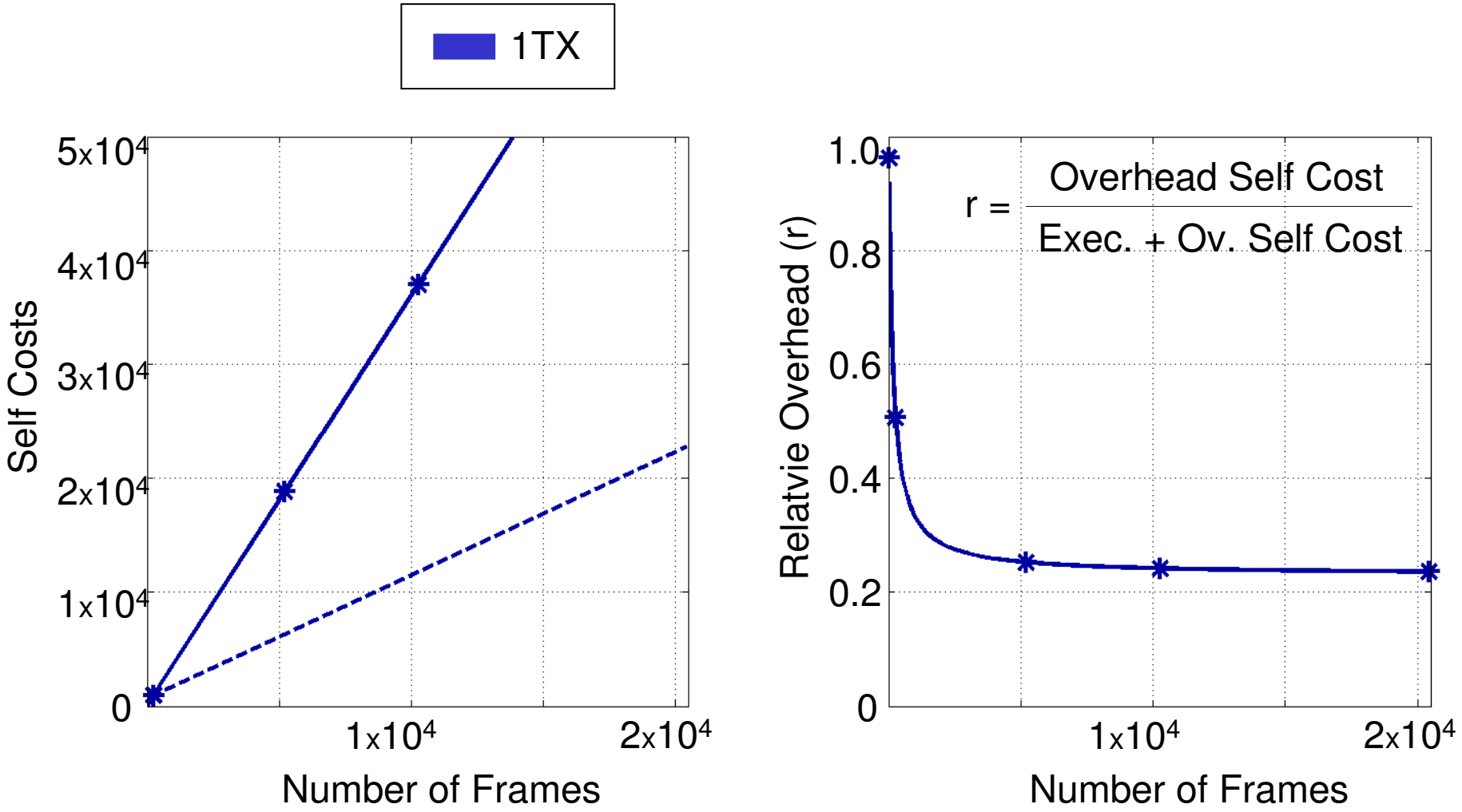
Profiling Results for 4 Component TX



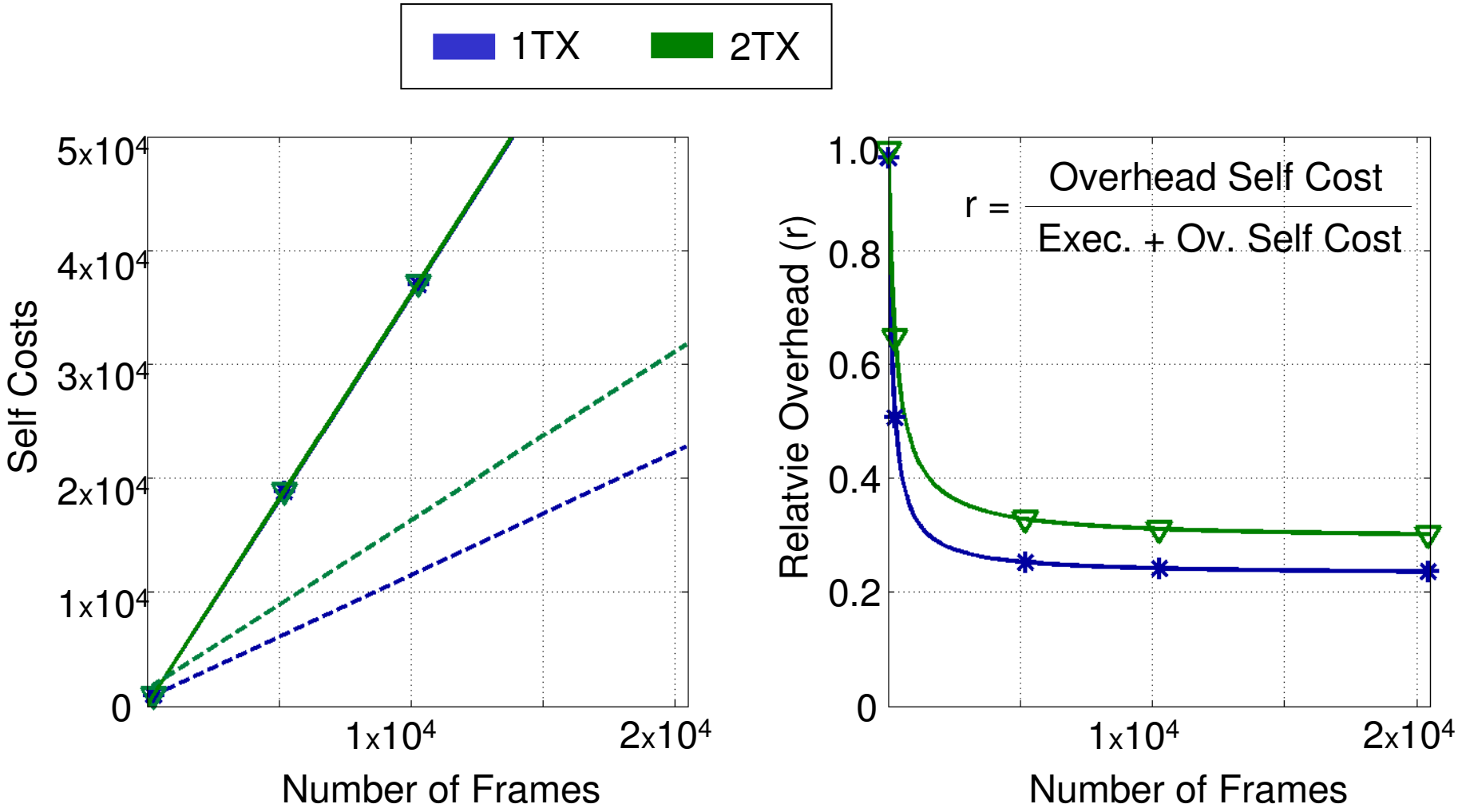




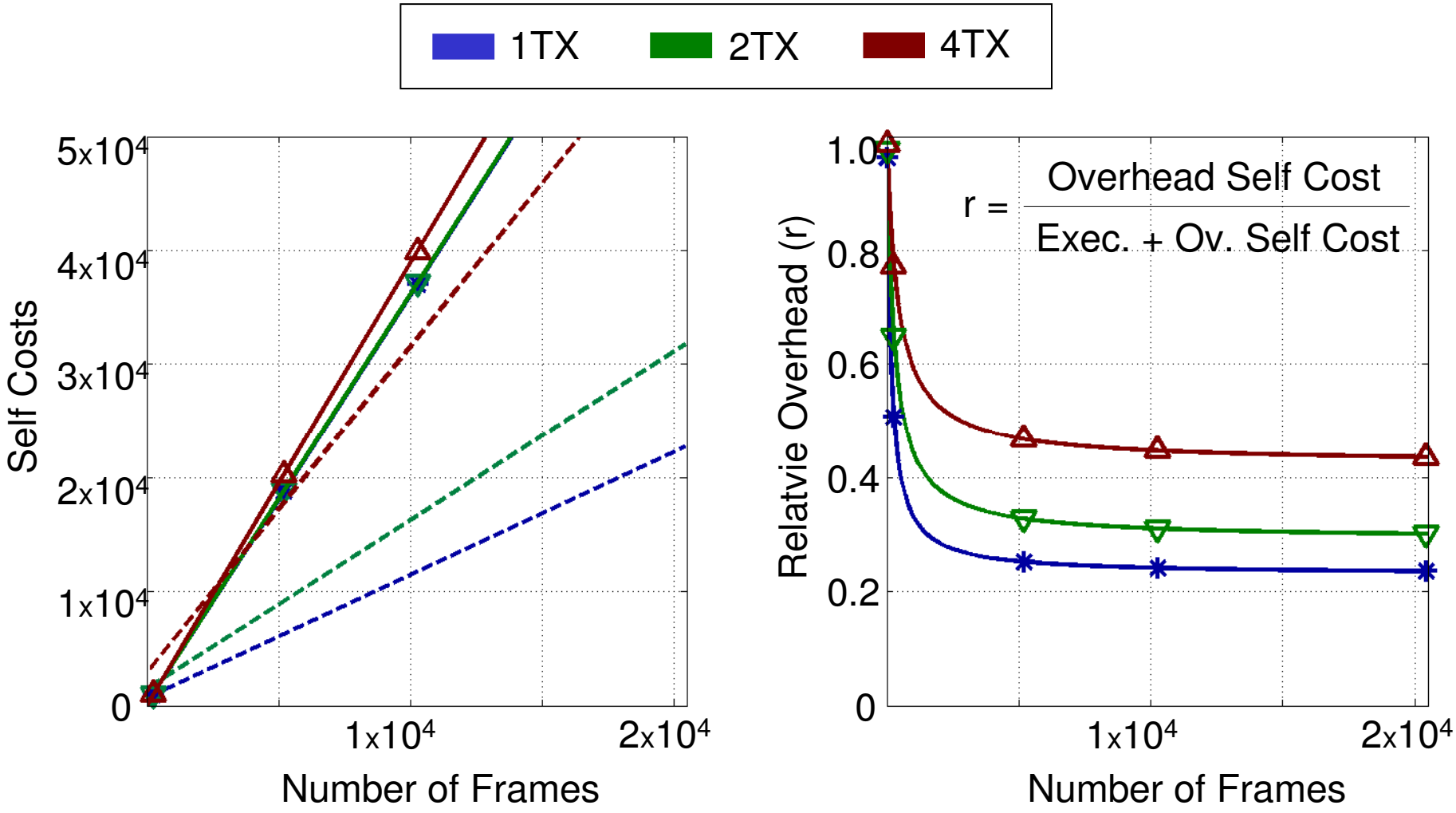
Comparison of Profiling Results



Comparison of Profiling Results



Comparison of Profiling Results



Conclusions

- SCA based implementation of STANAG 4285 as a NATO effort
  - Three implementations of different granularity level available
- Profiling results
  - Division creates Overheads
  - Self costs of both, overhead and executable, increase linearly with the amount of processed data
  - Self costs can contain Fixed Costs and Variable Costs
  - Fixed Costs are a burden if less user data is processed
- Recommendations
  - Perform considerable Signal Processing in each SCA resource
  - Avoid large number of resource divisions

Questions?

Many Thanks for Your Attention ...

Thanks to

- Telefunken Racoms for providing the C code for STANAG 4285
- all the RTO group members for their valuable comments