Performance of Parallelization of the Open Source AIT QKD Software R10 for QKD Post Processing

Oliver Maurhart, Christoph Pacher, Manuel Warum
AIT Austrian Institute of Technology, Donau-City-Strasse 1, 1220 Vienna, Austria

This presentation discusses the performance of parallelization attempts of the recent changes and additions of the Open Source AIT QKD Software “R10” for QKD post processing. This software assembles a great overhaul of work derived from the trusted repeater SECOQC effort and results. It provides a concise set of building blocks to integrate arbitrary sifting, error estimation, error correction and privacy amplification up to full network integration. Accompanying the basic QKD post processing is the Quantum Point-to-Point Protocol (Q3P) node which enforces Information Theoretically Secure (ITS) network peer to peer communication for classical applications.

In order to achieve this goal at the heart of the QKD design lies the “QKD Module” realized as a UNIX process which reads in key material and performs various tasks on this data stream ranging from simple BB84 protocol implementation to LDPC algorithms and beyond. As all QKD modules share the same input/output interface provided by the AIT QKD library written in C++11, integration of a nouveau protocols or algorithms is simple. Furthermore this attempt enables different parallelization techniques by forking and joining the stream of key material. Modules, e.g. error correction, can run concurrently to boost overall key reconciliation throughput. This idea can be applied to a whole series of modules forming several “QKD Pipelines” pushing keys to a single key database.

The design of the AIT QKD R10 software has been done with extensibility, flexibility and robustness in mind. Each module in the pipeline can be easily substituted with implementations of recent developed algorithms by independent teams without touching the rest of the pipeline. Therefore Quantum Bit-Commitment, Quantum-Coin-Flip protocols and Quantum Obliviuous Transfer can be integrated assembling the pipeline with off-the-shelf modules and new created ones.

The AIT QKD “R10” QKD software is bundled with management tools, partly GUI oriented, based on Distributed Bus (DBus) message exchange technology to simply script QKD modules or AIT QKD based applications with Python or even Bash. Utilities and tools as well as a boilerplate setup for QKD module coding projects can be used to create new QKD post processing modules or QKD based user applications. Development stages of new projects utilizing the AIT QKD software are depicted in the presentation ranging from debugging support of parallel and distributed concurrently running QKD modules up to configuration, packaging and deployment.

This poster also shows current results, as depicted in figure 1, in parallelization of this Open Source QKD post processing stack along with different parallelization scenarios as well as bottlenecks and drawbacks caused by hardware limitations.

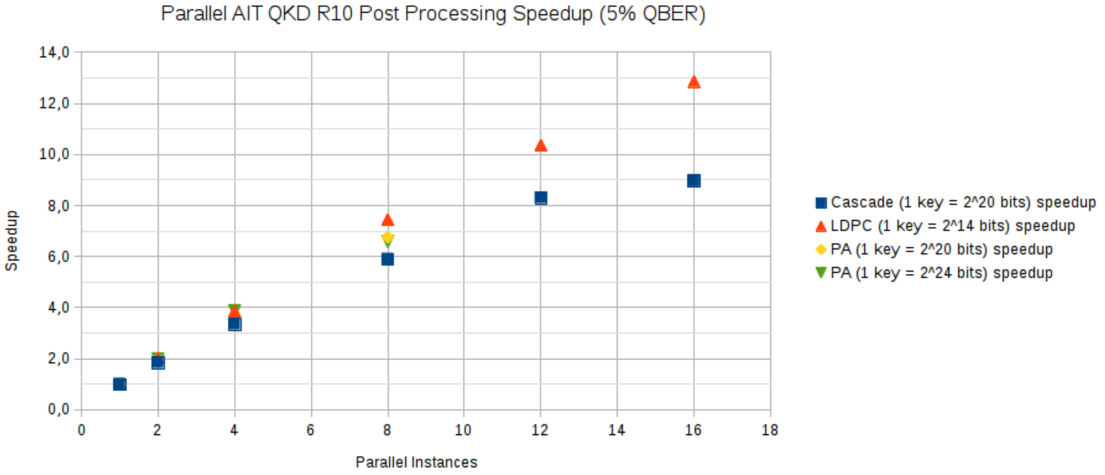

Figure 1: Speedup comparison for parallel instances of some AIT QKD post processing modules