High-Speed Implementation of Privacy Amplification in Quantum Key Distribution

Ririka Takahashi, Yoshimichi Tanizawa, Alexander R. Dixon
Corporate Research and Development Center, Toshiba Corporation
E-mail: ririka.takahashi@toshiba.co.jp

***Abstract*—This paper discusses privacy amplification implementation for a high-speed quantum key distribution system. A coprocessor is used for secure key rate increase. We report the evaluation result of implementations.**

Introduction—Quantum Key Distribution (QKD) provides information theoretic security based on physical principles to users of cryptographic communication. Since the amount of communication data has increased in recent years, the key distribution rate of QKD is an important factor to be taken into consideration. Privacy amplification (PA) is a necessary post-processing step in QKD. However due to its computational complexity it can often have limited throughput, limiting the secure key rate. This paper presents the results of PA implementation using a coprocessor in order to support a high-speed QKD system.

Method—PA generates a secure key of length s by multiplying the input bit of length l with a hash function. As the computational complexity of hash matrix multiplication is large, we used a number theoretical transform (NTT) applied to a Toeplitz matrix. This belongs to the family of universal$_2$ hash functions in order to perform calculations at high speed by reducing the computational complexity to $O(n\log_2 n)$ instead of $O(n^2)$ [1]. In addition, in order to efficiently utilize the coprocessor for matrix multiplication, we apply; 1) SIMD (Single Instruction Multiple Data) for matrix transpose and butterfly computation of NTT, 2) suitable instruction set regarding cache hit ratio, 3) loop unrolling, 4) parallelization by multithread processing of data input and output, and 5) parallelization of matrix multiplication and secure key length estimation.

TABLE 1 SERVER SPECIFICATION

Parameter	Value
OS	RHEL 6.5
CPU	Intel® Xeon® E5-2620v2 (2.1GHz) x2
Memory	128 GB
Coprocessor	Intel® Xeon Phi™ coprocessor 7120A
Compiler	Intel® C++ Compiler 15.0

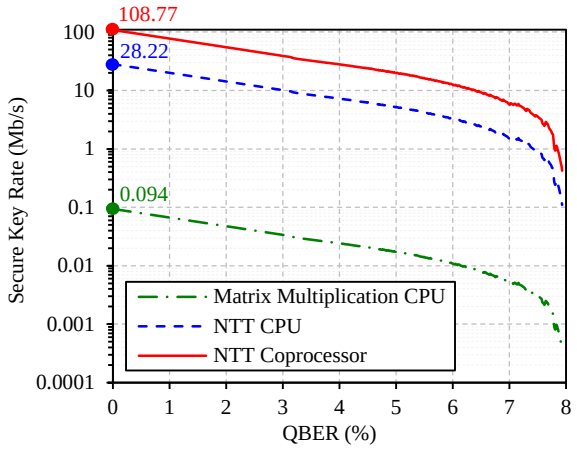

Fig. 1. Secure key rate for implementation of matrix multiplication, NTT, on the CPU and NTT on the coprocessor. Secure key rate is simulated based on [2] by using PA throughput. The number displayed near the vertical axis indicates measured PA throughput.

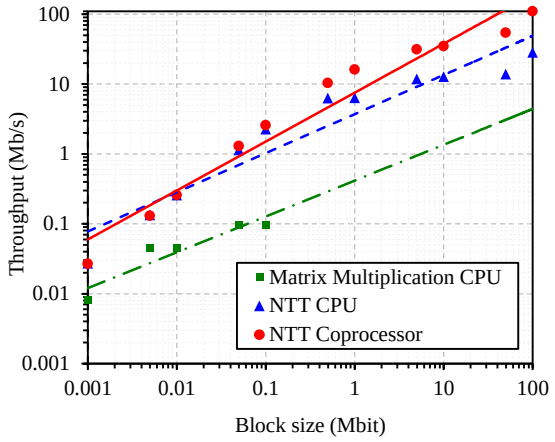

Fig. 2. PA throughput for implementation of matrix multiplication, NTT on the CPU and NTT on the coprocessor as a function of the block size.

Results—Three implementations of PA were evaluated. They are 1) calculating direct matrix multiplication on the CPU, 2) using NTT for computational complexity reduction on the CPU and 3) using NTT on the coprocessor to perform parallel computing. The specifications of the server are shown in Table 1*. Fig. 1 shows the secure key rate as a function of quantum bit error rate (QBER). The secure key rate is the maximum possible rate simulated according to [2] using the PA throughput calculated by the processing time during hashing, data input and output. Looking at the result, the coprocessor achieved a throughput of 108.77 Mb/s, whereas it was 0.094 Mb/s for matrix multiplication, and 28.22 Mb/s for NTT on the CPU. When the throughput is converted into the secure key rate, it remains more than 20 Mb/s even if the QBER is 5%. To the author's knowledge, this is the fastest recorded PA throughput. Fig. 2 shows the PA throughput as a function of the block size. The block size is extended to l=100 Mbit for NTT implementations as a large block size is needed to give a high secure key rate due to finite size effects in the security calculation [2]. The result for the coprocessor shows that the throughput increases in linearly with the block size. As previously mentioned, this can be explained by the efficient use of a number of cores in parallel computing.

Conclusion—The high-speed PA was implemented using a coprocessor and the performance was evaluated. Although the PA is necessary to achieve information theoretic security, it limits the secure key rate of QKD as it suffers from high computational complexity. In this paper, it was shown that the coprocessor PA implementation achieved a throughput of 108.77 Mb/s with 100 Mbit block size and it led to the secure key rate of 20 Mb/s.

REFERENCES

[1] C. H. Bennett *et al.*, *IEEE Trans. Inf. Theory* **41**, 1915-1923 (1995).
[2] M. Lucamarini *et al.*, *Opt. Expr.* **21**, 24550-24565 (2013).

* Intel, Xeon, and Intel Xeon Phi are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries. All other product names (mentioned herein) may be trademarks of their respective companies.