From: 'Francesco Antognazza' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Monday, April 8, 2024 3:52 AM
To: pqc-forum
Subject: [pqc-forum] Official comment: 4th round HQC specification - polynomial sampling

Dear NIST PQC Forum,

In the past year, we designed an RTL hardware accelerator for the HQC KEM scheme, exploiting its potential for parallelism.
The results will be published at the HOST 2024 conference in May.

Since the 5th PQC Standardization Conference takes place before the said conference, we decided to report a straightforward yet effective optimization of the HQC algorithm we discovered and applied in our work.

The proposed optimization enhances hardware and software performance without leading to security losses, and only impacts the currently distributed KATs.
It lies in sampling the sparse polynomials ("x", "y", "r_1", "r_2", "e") in the order given by the data dependencies of algorithms instead of what is currently specified in the official public documentation.

In particular, the proposed optimized generation order of the polynomials mentioned above is:

- sampling "y" before "x" during the key generation
- consequently, sampling only "y" during the decapsulation ("x" is not used)
- sampling "r_2" and "e" before "r_1" during the encapsulation

Those changes to the specification allowed us to provide an HW design exhibiting a performance gain ranging from 13% (decapsulation of hqc-256) to 38% (keygen of hqc-128) in terms of latency w.r.t. an implementation strictly observing the HQC specification.

SW implementations could also benefit from the optimizations, both in terms of latency (e.g., "x" is not sampled during the decapsulation) and runtime memory usage (opens to potential optimizations after careful variable liveness analysis: e.g., during the keygen "y" is produced and used immediately, before being overwritten by "x").

The SystemVerilog source code of the implementation will be available at https://doi.org/10.5281/zenodo.10135379

Best regards,

Francesco Antognazza,
on behalf of Alessandro Barenghi, Gerardo Pelosi, and myself
--