

Sampling fixed weight vectors via unranking in HQC

Gustavo Zambonin Ricardo Custódio Daniel Panario

Universidade Federal de Santa Catarina & Carleton University

May 10, 2026

CBCrypto 2026

Fixed-weight vectors in HQC

HQC [2] samples vectors in $V_{n,\omega} = \{\mathbf{v} \in \mathbb{F}_2^n : |\text{supp}(\mathbf{v})| = \omega\}$.

Two samplers in the spec., named `SampleFixedWeightVect` (SFVV) and `SFVV§`.

SFVV_§ is used in key generation, based on rejection sampling. It is uniform, but needs a variable amount of randomness.

SFVV is used elsewhere, based on [4]. It is slightly biased, requiring correction factors to the IND-CCA2 bound.

Goal

Can we have a *single, deterministic, uniform* sampler?

Combinatorial unranking

Definition

A *ranking function* is a bijection $\mathbf{R} : S \rightarrow [0, |S|)$ defining a total order on S . The *unranking function* is the inverse $\mathbf{U} = \mathbf{R}^{-1}$.

Our approach replaces SFVV and SFVV_§ simultaneously.

- ① Squeeze B bytes from the XOF, with $\binom{n}{\omega} \cdot 2^{-8B} < 2^{-2\lambda}$.
- ② Reduce the output r modulo $\binom{n}{\omega}$.
- ③ Evaluate $\mathbf{U}(r)$, which outputs an element of $V_{n,\omega}$.

We must ensure the modulo bias is cryptographically negligible with respect to the security parameter λ .

Two flavors of unranking: $\mathbf{U}_{\text{lex}}$ and $\mathbf{U}_{\text{rb}}$

Any $r \in [0, \binom{n}{k})$ decomposes to $r = \sum_{i=1}^k \binom{c_i}{i}$, with $c_k > \dots > c_1$ [3, p. 360], giving k -subsets of $\{0, \dots, n-1\}$ in lexicographic order.

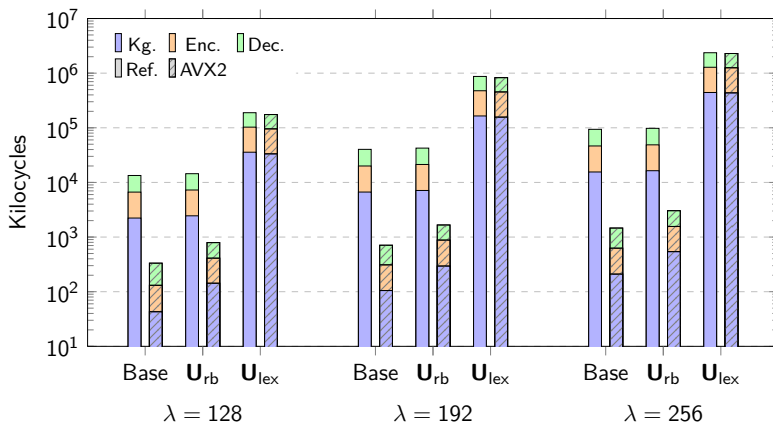
$\mathbf{U}_{\text{lex}}$ finds the unique sequence c_i [1]. We use a variant of Pascal's rule to make $\mathbf{U}_{\text{lex}}$ constant-time, requiring long integer arithmetic.

We can also split $\binom{n}{k} = \sum_j \binom{\lfloor n/2 \rfloor}{j} \binom{\lceil n/2 \rceil}{k-j}$, which defines a “recursive block” order and thus a new unranking algorithm $\mathbf{U}_{\text{rb}}$.

It uses long integers at upper levels of the recursion path. It is faster, but variable-time due to the unbalanced path.

Performance measurements

Average of 100 runs on an AMD Ryzen™ 7 9700X



Randomness consumption is also reduced from 4 to 19%.

What's next

The bottleneck for $\mathbf{U}_{\text{lex}}$ is long integer arithmetic, and AVX2 does not help (yet).

We are working in making $\mathbf{U}_{\text{rb}}$ constant-time, and exploring libraries such as cryptoint and s2n-bignum, and the Jasmin language.

We are also extending this to other cryptosystems with similar sampling procedures, such as BIKE, NTRU and Classic McEliece.

Partially supported by FAPESC (62/2024), ON-RCPN of Brazil, and NSERC of Canada (RGPIN-2024-05341).

Thank you! Any questions?

References

- [1] Thomas M. Cover. “Enumerative source encoding”. In: *IEEE Transactions on Information Theory* 19.1 (Jan. 1973), pp. 73–77. DOI: 10.1109/TIT.1973.1054929.
- [2] Philippe Gaborit et al. *Hamming Quasi-Cyclic (HQC)*. Aug. 2025. URL: https://pqc-hqc.org/doc/hqc_specifications_2025_08_22.pdf.
- [3] Donald E. Knuth. *The Art of Computer Programming, Volume 4A: Combinatorial Algorithms, Part 1*. 1st. Addison-Wesley Professional, 2011. ISBN: 9780201038040.
- [4] Nicolas Sendrier. *Secure Sampling of Constant-Weight Words - Application to BIKE*. Cryptology ePrint Archive, Paper 2021/1631. 2021. URL: <https://eprint.iacr.org/2021/1631>.