Post-quantum cryptosystems based on coding theory

Paulo S. L. M. Barreto
(SFI Walton Fellow)

Contents

- Motivation
- Essentials of coding theory
- Coding-based PQC
- Current challenges… and solutions

Motivation

- The overwhelming majority of deployed crypto-systems rest on only two security assumptions:
  - Integer Factorization (IFP): RSA, BBS.
  - Discrete Logarithm (DLP): ECC, PBC.
- Shor's quantum algorithm can efficiently solve the IFP and the DLP.

Post-quantum cryptosystems

- Entirely classical systems:
  - plug-in replacements for RSA/ECC.
  - avoid expensive (sometimes non-existing) purely quantum technologies.
- Security assumptions related to NP-complete/NP-hard problems, apparently beyond the capabilities of quantum computers.

Coding-based cryptosystems

- Many cryptographic primitives supported:
  - encryption,
  - digital signatures and identification,
  - identity-based signatures and identification,
  - oblivious transfer...
- Efficiency and simplicity:
  - O(n^2) encryption/decryption.
  - plain arithmetic with matrices and vectors.
- **Drawback: very large keys.**

Linear codes

- A linear $[n, k]$-*code* $\mathcal{C}$ over $\mathbb{K}$ is a k-dimensional vector subspace of $\mathbb{K}^n$.
- A code may be defined by either
  - a *generator* matrix $G \in \mathbb{K}^{k \times n}$, or
  - a *parity-check* matrix $H \in \mathbb{K}^{(n-k) \times n}$,
  - $HG^T = O$,
  - $\mathcal{C} = \{uG \in \mathbb{K}^n \mid u \in \mathbb{K}^k\} = \{v \in \mathbb{K}^n \mid Hv^T = o^T\}$.
- The vector s such that $Hv^T = s^T$ is called the *syndrome* of v.
- Hard problems involving codes?

General decoding (GDP)

- **Input:** positive integers n, k, t; a finite field $\mathbb{F}_q$; a linear $[n, k]$-code $\mathcal{C} \subseteq (\mathbb{F}_q)^n$ defined by a generator matrix $G \in (\mathbb{F}_q)^{k \times n}$; a vector $c \in (\mathbb{F}_q)^n$.
- **Question:** is there a vector $m \in (\mathbb{F}_q)^k$ s.t. $e = c - mG$ has weight $w(e) \leq t$?
- NP-complete!
- **Search:** find such a vector m.

Syndrome decoding (SDP)

- **Input:** positive integers n, k, t; a finite field $\mathbb{F}_q$; a linear $[n, k]$-code $\mathcal{C} \subseteq (\mathbb{F}_q)^n$ defined by a parity-check matrix $H \in (\mathbb{F}_q)^{r \times n}$ with $r = n - k$; a vector $s \in (\mathbb{F}_q)^r$.
- **Question:** is there a vector $e \in (\mathbb{F}_q)^n$ of weight $w(e) \leq t$ s.t. $He^T = s^T$?
- NP-complete!
- **Search:** find such a vector e.

Alternant and Goppa codes

- Let $q = p^d$ for some $d > 0$, and p a prime power.
- An *alternant code* $\mathcal{A}(L, D)$ over $\mathbb{F}_p$ is defined by:
  - a sequence $L \in (\mathbb{F}_q)^n$ of distinct elements with $n \le p$;
  - a sequence $D \in (\mathbb{F}_q)^n$ of nonzero elements;
  - easily decodable ($t/2$ errors) syndromes from $H = \mathrm{T}_p(\mathrm{vdm}_t(L)\ \mathrm{diag}(D))$.
- A *Goppa code* $\Gamma(L, g)$ over $\mathbb{F}_p$ is an alternant code where:
  - $L \in (\mathbb{F}_q)^n$ satisfies $g(L) \neq 0$, and $D = (1/g(L))$ for some monic polynomial $g(x) \in \mathbb{F}_q[x]$ of degree t;
  - good error correction capability (all t design errors) in characteristic 2.

McEliece cryptosystem

- Key generation:
  - Choose a “secure”, uniformly random $[n, k]$ t-error correcting alternant code $\mathcal{A}(L, D)$ over $\mathbb{F}_p$, with $L, D \in (\mathbb{F}_q)^n$.
  - Compute for $\mathcal{A}(L, D)$ a systematic generator matrix $G \in (\mathbb{F}_p)^{k \times n}$.
  - Set $K_{priv} = (L, D)$, $K_{pub} = (G, t)$.

McEliece cryptosystem

- Encryption of a plaintext $m \in (\mathbb{F}_p)^k$:
  - Choose a uniformly random t-error vector $e \in (\mathbb{F}_p)^n$ and compute $c = mG + e \in (\mathbb{F}_p)^n$ (IND-CCA2 variant via e.g. Fujisaki-Okamoto).
- Decryption of a ciphertext $c \in (\mathbb{F}_p)^n$:
  - Use the trapdoor to obtain the usual alternant parity-check matrix H (or equivalent).
  - Compute the syndrome $s^T \leftarrow Hc^T = He^T$ and decode it to obtain the error vector e.
  - Read m directly from the first k components of $c - e$.

CFS signatures

- System setup:
  - Choose m, t, and $n \approx 2^m$.
  - Choose a hash function $\mathcal{H}$: $\{0, 1\}^* \times \mathbb{N} \to (\mathbb{F}_2)^{n-k}$.
- Key generation:
  - choose a uniformly random $[n, k]$ t-error correcting binary alternant code $\mathcal{A}(L, D)$.
  - compute for it a systematic parity-check matrix H.
  - $K_{private} = (L, D)$; $K_{public} = (H, t)$.
- Observation:
  - Let H_0 be the trapdoor parity-check matrix for $\mathcal{A}(L, D)$, so that $H_0 = MH$ for some nonsingular matrix M.
  - If $s^T = He^T$ for some t-error vector e, then $s_0^T = Ms^T = MHe^T = H_0e^T$ is decodable using the trapdoor.

CFS signatures

- Signing a message m:
  - find $c \in \mathbb{N}$ such that, for $s \leftarrow \mathcal{H}(m, c)$ and $s_0^T \leftarrow Ms^T$, s_0 is decodable with the trapdoor H_0, and decode s_0 into a t-error vector e, i.e. $s_0^T = H_0 e^T$ and hence $s^T = He^T$.
  - the signature is (e, c).
- Verifying a signature (e, c):
  - compute $s^T \leftarrow He^T$.
  - accept iff $w(e) = t$ and $s = \mathcal{H}(m, c)$.

CFS signatures

- Density of decodable syndromes: $1/t!$
- Signature length (permutation ranking) is $\approx \lg(n^t/t!) + \lg(t!) = t \lg n$.
- Public key is huge: mtn bits.
- Recommendation for security level $\approx 2^{80}$:
  - original: $m = 16$, $t = 9$, $n = 2^{16}$, signature length = 144 bits, key size = 1152 KiB.
  - updated: $m = 15$, $t = 12$, $n = 2^{15}$, signature length = 180 bits, key size = 720 KiB.

Reducing the key size

- Replace a generic code by a permuted and shortened [W 2006] subfield subcode of a quasi-cyclic [BCGO 2009] or quasi-dyadic [MB 2009] code.

- O(n) instead of O(n^2) space.

- O(n lg n) instead of O(n^2) time.

Cauchy matrices

- A matrix $H \in \mathbb{K}^{t \times n}$ over a field $\mathbb{K}$ is called a *Cauchy matrix* iff $H_{ij} = 1/(z_i - L_j)$ for disjoint sequences $z \in \mathbb{K}^t$ and $L \in \mathbb{K}^n$ of distinct elements.
- Property: any Goppa code where $g(x)$ is square-free admits a parity-check matrix in Cauchy form [TZ 1975].
- Compact representation, but:
  - code structure is apparent,
  - usual tricks to hide it (permute, scale, puncture, systematize, etc) also destroy the Cauchy structure.

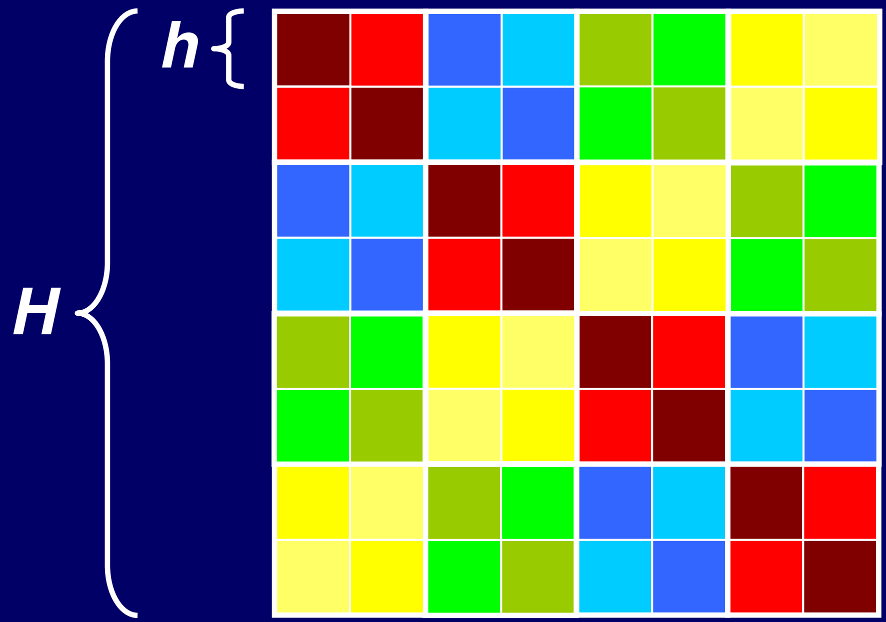
Dyadic matrices

- Let r be a power of 2. A matrix $H \in \mathcal{R}^{r \times r}$ over a ring $\mathcal{R}$ is called *dyadic* iff $H_{ij} = h_{i \oplus j}$ for some vector $h \in \mathcal{R}^r$.
- If A and B are dyadic of order r, then

$$C = \begin{bmatrix} A & B \\ B & A \end{bmatrix}$$

is dyadic of order $2r$.

Dyadic matrices



$$H_{ij} = h_{i \oplus j}$$

Dyadic matrices

- Dyadic matrices form a subring of $\mathcal{R}^{r \times r}$ (commutative if $\mathcal{R}$ is commutative).
- Compact representation: $O(r)$ rather than $O(r^2)$ space.
- Efficient arithmetic: multiplication in time $O(r \lg r)$ time via fast Walsh-Hadamard transform, inversion in time $O(r)$ in characteristic 2.
- **Idea:** find a dyadic Cauchy matrix.

Dyadic codes

- **Theorem:** a dyadic Cauchy matrix is only possible over *binary* fields, and any suitable $h \in (\mathbb{F}_q)^n$ satisfies

$$\frac{1}{h_{i \oplus j}} = \frac{1}{h_i} + \frac{1}{h_j} + \frac{1}{h_0}$$

with $z_i = 1/h_i + \omega$, $L_j = 1/h_j - 1/h_0 + \omega$ for arbitrary ω, and $H_{ij} = h_{i \oplus j} = 1/(z_i - L_j)$.

Constructing dyadic codes

- Choose distinct h_0 and h_i with $i = 2^u$ for $0 \leq u < \lceil \lg n \rceil$ uniformly at random from $\mathbb{F}_q$, then set

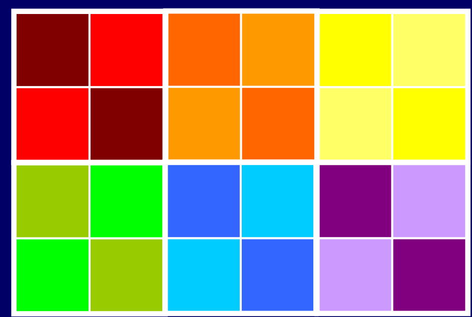
$$h_{i+j} \leftarrow \frac{1}{\frac{1}{h_i} + \frac{1}{h_j} + \frac{1}{h_0}}$$

for $0 < j < i$ (so that $i + j = i \oplus j$).

- Complexity: $O(n)$.

Quasi-dyadic codes

- Structure hiding:
  - choose a long code over $\mathbb{F}_q$,
  - blockwise shorten the code,
  - permute dyadic block columns,
  - dyadic-permute (and $\mathbb{F}_p$-scale) individual blocks,
  - take a $\mathbb{F}_p$ subfield subcode of the result.
- Quasi-dyadic matrices: $(\mathbb{F}_p^{t\times t})^{d\times \ell}$.



Compact keys

- Quasi-dyadic codes over $\mathbb{F}_{2^8}$ from trapdoor codes over $\mathbb{F}_{2^{16}}$, with $t \times t$ dyadic submatrices:

level	*n*	*k*	*t*	size	generic	shrink	RSA	NTRU
2^{80}	512	256	128	4096 bits	57 KiB	112	1024 bits	–
2^{112}	640	384	128	6144 bits	128 KiB	170	2048 bits	4411–7249 bits
2^{128}	768	512	128	8192 bits	188 KiB	188	3072 bits	4939–8371 bits
2^{192}	1280	768	256	12288 bits	511 KiB	340	7680 bits	7447–11957 bits
2^{256}	1536	1024	256	16384 bits	937 KiB	468	15360 bits	11957–16489 bits

Efficient processing

- Preliminary timings against RSA (times in ms):

level	RSA keygen	QD keygen	RSA encrypt	QD encrypt	RSA decrypt	QD decrypt
2^{80}	563	17.2	0.431	0.817	15.61	3.685
2^{112}	1971	18.7	1.548	1.233	110.34	4.463
2^{128}	4998	20.5	3.467	1.575	349.91	5.261
2^{192}	628183	47.6	22.320	4.695	5094.10	17.783
2^{256}	–	54.8	–	6.353	–	21.182

- How about security?

Quasi-dyadic GDP/SDP

- Solve the GDP or the SDP for quasi-dyadic codes.
- **Theorem:** the QD-GDP and the QD-SDP are NP-complete.
- Caveat:
  - only constitutes trapdoor one-way functions!
  - average-case complexity?
  - structural attacks?

QD-CFS signatures

- The maximum length of regular QD codes is $n = 2^{m-1}$ even without puncturing.
- Difficulty to get $n \approx 2^m$: the full sequences z and L (length n) are no longer disjoint $\Rightarrow$ $1/(z_i - L_j)$ undefined.
- Binary QD codes: density of decodable syndromes $\approx 1/(2^t\ t!)$, a factor 2^t worse than irreducible codes – but better than $1/(2t)!$, and up to a factor t shorter.

QD-CFS signatures

- Yet only a single block of t rows and a subset of the columns are needed to define a shortened QD code!
- Solution: modify the dyadic construction to allow for $2^{m-1} < n < 2^m$ by admiting undefined entries when they are unused.
- Binary QD codes with minimal puncturing: density of decodable syndromes $\approx 1/(c\ t!)$ for $n \approx 2^m/c^{1/t}$.

QD-CFS signatures

- Suggestion for security level ≈ 2^{80}: $m = 15$, $t = 12$, $n = 2^{15}$, signature length = 180 bits, key size = 180 KiB (vs. 720 KiB for a generic, irreducible Goppa code).
- Structural security: work in progress. 
  - ... but puncturing seems very effective in thwarting such attacks.

Summary

- Coding-based cryptography is a purely classical, post-quantum alternative to quantum cryptography.
- Several pros over traditional systems (quantum immunity, efficient operations), main con already solved (shorter keys).
- New functionalities still a challenge (key agreement, IBE, formal security, dyadic lattices) ⇒ good research opportunity ☺

Questions?

Thank You!

Appendix

McEliece cryptosystem

- “Hey, wait, I know McEliece, and this does not look quite like it!”

- Observations:
  - A *secret*, *random* L is equivalent to a *public*, *fixed* L coupled to a *secret*, *random* permutation matrix $P \in (\mathbb{F}_p)^{k \times k}$, with $\mathcal{A}(LP, DP)$ as the effective code.
  - If G_0 is a generator for $\mathcal{A}(L, D)$ when L is public and fixed, and S is the matrix that puts G_0P in systematic form, then $G = SG_0P$ is a systematic generator of $\mathcal{A}(LP, DP)$, as desired.

McEliece-Fujisaki-Okamoto: Setup

- Random oracle (message authentication code) $\mathcal{H}$: $(\mathbb{F}_p)^k \times \{0, 1\}^* \to \mathbb{Z}/s\mathbb{Z}$, with s = (n choose t) $(p - 1)^t$.
- Unranking function $\mathcal{U}$: $\mathbb{Z}/s\mathbb{Z} \to (\mathbb{F}_p)^n$.
- Ideal symmetric cipher $\mathcal{E}$: $(\mathbb{F}_p)^k \times \{0, 1\}^* \to \{0, 1\}^*$.
- Alternant decoding algorithm $\mathcal{D}$: $(\mathbb{F}_q)^n \times (\mathbb{F}_q)^n \times (\mathbb{F}_p)^n \to (\mathbb{F}_p)^k \times (\mathbb{F}_p)^n$.

McEliece-Fujisaki-Okamoto: Encryption

- Input:
  - uniformly random symmetric key $r \in (\mathbb{F}_p)^k$;
  - message $m \in \{0, 1\}^*$.
- Output:
  - McEliece-FO ciphertext $c \in (\mathbb{F}_p)^n \times \{0, 1\}^*$.
- Algorithm:
  - $h \leftarrow \mathcal{H}(r, m)$
  - $e \leftarrow \mathcal{U}(h)$
  - $w \leftarrow rG + e$
  - $d \leftarrow \mathcal{E}(r, m)$
  - $c \leftarrow (w, d)$

McEliece-Fujisaki-Okamoto: Decryption

- Input:
  - McEliece-FO ciphertext $c = (w, d)$.
- Output:
  - message $m \in \{0, 1\}^*$, or rejection.
- Algorithm:
  - $(r, e) \leftarrow \mathcal{D}(L, D, w)$
  - $m \leftarrow \mathcal{E}^{-1}(r, d)$
  - $h \leftarrow \mathcal{H}(r, m)$
  - $v \leftarrow \mathcal{U}(h)$
  - accept $m \Leftrightarrow v = e$ and $w = rG + e$

CFS signatures

- The number of possible hash values is $2^{n-k} = 2^{mt} \approx n^t$ and the number of syndromes decodable to codewords of weight t is

$$\binom{n}{t} \approx \frac{n^t}{t!}$$

- $\therefore$ The probability of finding a codeword of weight t is $\approx 1/t!$, and the expected value of hash queries is $\approx t!$ assuming all t design errors can be corrected (only true for binary Goppa codes!).