QIC 891 Topics in Quantum Safe Cryptography

Module 1: *Post-Quantum Cryptography*

Lecture 1

Lecturer: Fang Song *May 10, 2016*

Today. Part I (on slides): Setting the scene for post-quantum crypto: classical cryptographic schemes that are secure against quantum attacks. Part II (here): candidate problems proposed in lattices, coding theory and multivariate quadratic equations that people use to construct post-quantum cryptosystems.

1 Lattice-based

Two prominent problems in lattice cryptogrphy are *Short Integer Solution* (SIS) and *learning with errors* (LWE). They can be defined purely by linear algebra (matrices) without referring to lattices. We will defer the discuss about lattices and the connection between SIS & LWE and (actual) lattice problems in the last lecture.

Short Integer Solution [Ajt96, MR07] ($\mathsf{SIS}_{n,q,\beta,m}$). Let $\mathbb{Z}_q$ be the additive group modulo a large integer q.

- **Given**: $A = (a_1, \ldots, a_m) \in \mathbb{Z}_q^{n\times m}$, $a_i \in \mathbb{Z}_q^n$.
- **Goal**: Find $x \in \mathbb{Z}_q^m$ with $\|x\| \le \beta$ s.t. $f_A(x) := Ax \pmod q = 0$.

Observations.

- f_A is *surjective* under typical setting: $m \ge n\log q$ for $m = \mathrm{poly}(n)$, $q > \beta \cdot \mathrm{poly}(n)$ and $x \in \{0 \pm 1\}^m$ (hence $\beta \sim \sqrt{m}$).
- f_A is (approximately) homomorphic: $f_A(x_1 + x_2) = f_A(x_1) + f_A(x_2)$. This is true only when the inputs remain small norm.
- We can also consider solving the inhomogeneous system $(A, b) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m$. It is essentially equivalent to the homogeneous version for typical parameters.

Assumption 1. Let $A \in_R \mathbb{Z}_q^{n\times m}$ be uniformly at random, then $\mathsf{SIS}_{n,q,\beta,n}$ is hard to solve for poly-time algorithms (classical & quantum). Likewise let $A \in_R \mathbb{Z}_q^{n\times m}$ and $x \leftarrow U$ for uniform distribution U on $\{x \in \mathbb{Z}_q^m : \|x\| \le \beta\}$), then $f_A(x)$ is hard to invert.

Learning With Errors [Reg09] ($\mathsf{LWE}_{n,q,\chi,m}$). Let χ be some error distribution on $\mathbb{Z}_q$.

- **Given**: (A, b), where $A = (a_1, \ldots a_m)^T \in \mathbb{Z}_q^{m\times n}$, $a_i \in \mathbb{Z}_q^n$ and
$$b = g_A(s, e) := As + e \pmod q \in \mathbb{Z}_q^n, \text{ with } s \in \mathbb{Z}_q^n, e \leftarrow \chi^m.$$
- **Goal**: Find s.

Observations.

- g_A is *injective* under typical setting: $n\log q + m\log\mathscr{E} \le m\log q$, where $\mathscr{E} \sim \sqrt{n}$ is the bound on errors drawn from χ.
- g_A is (approximately) homomorphic in the following sense: $g_{A_1}(s, e_1) + g_{A_2}(s, e_2) = g_{A_1+A_2}(s, e_1 + e_2)$.

Assumption 2. Let $A \in_R \mathbb{Z}_q^{n\times m}$, $s \in_R \mathbb{Z}_q^n$ and $e \leftarrow \chi^m$ for some χ (e.g. rounded Gaussian $p(z) \propto e^{-\pi|z|^2/r^2}$ with $r \geq \sqrt{n}$), then $\mathsf{LWE}_{n,q,\chi,m}$ is hard to solve for poly-time algorithms (classical & quantum), i.e. $g_A(s,e)$ is hard to invert. This implies that g_A is also a pseudorandom generator (via a Search to Decision reduction) in the sense that $(A, b := g_A(s,e)) \approx_c U(\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q)$ ($\approx_c$ means "computationally hard to distinguish for any poly-time algorithms").

Remark 1. For efficiency reason, there are also Ring-based SIS and LWE problems, whose hardness relate to computational problems in structured lattices called *ideal* lattices [Mic07,LM06,PR06,LPR13,SSTX09]. Read more about lattice cryptography in Peikert's recent (amazing) survey [Pei15].

2 Code-based

Error correcting codes are ubiquitous in digital communications. They provide a mechanism to encode message to resist (random) errors that occur via communication channels. Here we introduce binary linear codes. A binary linear code denoted $\mathscr{C} : [n,k,d]$ is a subspace of $\mathbb{F}_2^n$.

- n: codeword length
- k: message length, usually referred to as the dimension (the rank really)
- d: minimum distance, i.e., $d := \min_{x,y\in\mathscr{C}} \|x-y\|$. $\|z\|$ represents the Hamming weight of z which is the number of 1's in z.

[Exercise: show that distance d code can (not necessarily efficiently) correct up to $\lfloor \frac{d-1}{2} \rfloor$ errors. Hint: a "packing" argument.]

There are two ways of describing a linear code:

- **Generating matrix** $G \in \mathbb{F}_2^{n\times k}$:

$$\mathscr{C} := \left\{ c \in \mathbb{F}_2^n : c = Gw \text{ for some } w \in \mathbb{F}_2^k \right\}.$$

- **Parity check matrix** $H \in \mathbb{F}_2^{(n-k)\times n}$:

$$\mathscr{C} := \left\{ c \in \mathbb{F}_2^n : Hc = 0 \right\}.$$

Remarks.

- Given a perturbed codeword $c' = c + e$ by error e, observe that $Hc' = H(c+e) = He$. $s := He$ is usually called the error *syndrome*.
- Generating matrix and the parity check matrix of a linear code (n,k,d) can be transformed in a *systematic* form:

$$G = \left(\frac{Q_{(n-k)\times k}}{1_k} \right), \quad H = \left(1_{n-k} | -Q_{(n-k)\times k} \right).$$

[Verify: HG =0]

We are now ready to define two major problems in coding theory as the foundation for code-based crypto (Read [DES06] for an introduction on code-based crypto.). Syntactically, they look similar to SIS & LWE. Let $\mathscr{C} \subseteq \mathbb{F}_2^n$ be an $[n,k,d]$ binary linear code. All operations are in $\mathbb{F}_2$.
Syndrome Decoding ($\mathsf{SD}_{n,k,\beta}$). (Underlying Niederreiter PKE [Nie86])

- **Given**: (parity check matrix) $H \in \mathbb{F}_2^{(n-k)\times n}$ and (syndrome) $s \in \mathbb{F}_2^{n-k}$.
- **Goal**: Find $e \in \mathbb{F}_2^n$ with $\|e\| = \beta$ s.t. $f_H(x) := Hx = s$.

Observations.

- f_H injective under typical setting: $\binom{n}{\beta} \le 2^{n-k}$.

Assumption 3. let $H_0 \in \mathbb{F}_2^{(n-k)\times n}$ be the parity check matrix for some code $\mathscr{C}$ for which syndrome decoding is efficient (e.g., binary *Goppa* code), and $P \in_R S_n$ be a random permutation matrix. Then $g_H(\cdot)$ is hard to invert where $H := H_0 P$.

Codeword Decoding ($\mathsf{CD}_{n,k,\beta}$). (underlying McEliece PKE [McE78])

- **Given**: (generating matrix) $G \in \mathbb{F}_2^{n\times k}$ and (codeword possibly with error) $z \in \mathbb{F}_2^n$.
- **Goal**: Find $w \in \mathbb{F}_2^k$ s.t. $g_G(w) := Gw + e = z$ for some "small" error e with $\|e\| = \beta$.

Observations.

- Coding theory convention usually uses row vectors (e.g. wG^T).
- g_G injective under typical setting: $2^k \times \binom{n}{\beta} \le 2^n$.

Assumption 4. let $G_0 \in \mathbb{F}_2^{n\times k}$ be the generating matrix for some code $\mathscr{C}$ for which codeword decoding is efficient (e.g., binary *Goppa* code), $P \in \mathbb{F}_2^{n\times n}$ be the matrix of a random permutation $\pi \leftarrow S_n$ and $S \in_R \mathbb{F}_2^{k\times k}$ be a random invertible matrix. Then $g_G(\cdot)$ is hard to invert where $G := PG_0S$.

3 Multivariate-Polynomial-based

Multivariate Quadratic Polynomial Equations ($\mathsf{MQ}_{n,k}$). All operations are in some finite field $\mathbb{F}$.

- **Given**: $(p_i, y_i)_{i=1}^k$ where
$$p_i = \alpha_i + \sum_{j,\ell} \lambda_{ij\ell} x_j x_\ell$$
are qudratic polynomial in variables $x_1, \ldots, x_n$ and $\alpha_i \in \mathbb{F}$.
- **Goal**: Find $(x_1, \ldots x_n) \in \mathbb{F}^n$ s.t. $f_P(x_1, \ldots, x_n) := (\ldots, p_i(x_1, \ldots, x_n), \ldots) = (\ldots, y_i, \ldots)$.

Assumption 5. let P_0 be a collection of quadratic polynomials which are easy to solve. Let S and T be random affine transformations $\mathbb{F}^n \to \mathbb{F}^n$. Then $f_P(\cdot)$ is hard to invert where $P := TP_0S$.

$$\{x_i\} \to \underbrace{S \to P_0 \to T}_{P} \to \{y_i\}$$

Remarks.

- The main work in MQ-based crypto is to find "good" center polynomials P_0. Examples include: Oil-Vineger, Unbalanced Oil-Vineger, Hidden Field Equations (HFE) etc.
- Modifiers (+,-,v, ...) making P_0 more secure. +: add eqns; -: discard eqns; s: pick *sparse* polynomials.
- Wolf & Preneel [WP05] gave a nice survey on the taxonomy of MQ-based crypto.

A cautionary remark

Notice that in code-based crypto and MQ-based crypto, as illustrated in Assumptions 3, 4, 5, people start from an easy instance and try to obfuscate it by some randomization trick in hope of producing a hard instance. This might lead to unsafe constructions, and extreme caution should be taken (though it seems difficult in theory to determine what'd be a safe randomization strategy). This is in contrast to lattice crypto, as we have seen in SIS and LWE, where the problem instance is randomly generated. The hardness is then guaranteed by the (assumed) worst-case hardness of lattice problems due to the surprising *worst-case* to *average-case* reduction.

[Exercise. Read about how RSA instances (e.g. p, q, N, e, d) are generated. Compare with the post-quantum proposals.]

References

[Ajt96] Miklós Ajtai. Generating hard instances of lattice problems. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 99–108. ACM, 1996.

[DES06] R. Overbeck D. Engelbert and A. Schmidt. A summary of McEliece-type cryptosystems and their security. Cryptology ePrint Archive, Report 2006/162, 2006. `https://eprint.iacr.org/2006/162`.

[LM06] Vadim Lyubashevsky and Daniele Micciancio. Generalized compact knapsacks are collision resistant. In *Automata, Languages and Programming*, pages 144–155. Springer, 2006.

[LPR13] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, 60(6):43, 2013. Preliminary version in Eurocrypt 2010.

[McE78] RJ McEliece. A public-key cryptosystem based on algebraic coding theory. *The Deep Space Network Progress Report*, 42(44):114–116, 1978.

[Mic07] Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. *Computational Complexity*, 16(4):365–411, 2007. Preliminary version in FOCS 2002.

[MR07] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007. Preliminary version in FOCS 2004.

[Nie86] Harald Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problems of Control and Information Theory*, 15:19–34, 1986. Problemy Upravlenija i Teorii Informacii 15, 159–166.

[Pei15] Chris Peikert. A decade of lattice cryptography. Cryptology ePrint Archive, Report 2015/939, 2015.

[PR06] Chris Peikert and Alon Rosen. Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In *Theory of Cryptography*, pages 145–166. Springer, 2006.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009.

[SSTX09] Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In *Advances in Cryptology–ASIACRYPT 2009*, pages 617–635. Springer, 2009.

[WP05] Christopher Wolf and Bart Preneel. Taxonomy of public key schemes based on the problem of multivariate quadratic equations. Cryptology ePrint Archive, Report 2005/077, 2005. `http://eprint.iacr.org/2005/077`.