Quantum Cryptanalysis of Group Action

Dania Lazzarini[1], Université Libre de Bruxelles

QCQPA Workshop

University of Manchester
20th -24th July 2026

[1] Funded by a FRIA fellowship of the F.R.S.-FNRS.

The Goal of the Talk

- Recap of the instruments to understand the current scenario on the quantum cryptanalysis
  - group actions (but specialized)
  - hidden shift problem (HSP) and Kuperberg's attack
- Overview on the state-of-the-art on computation of quantum oracles
  - from the on-going joint project with X. Bonnetain, P. Dartois, M. Duparc, S. Jaques, L. Maino, C. Martindale, C. Petit, S. Schaeffler, A. Schrottenloher, A. Sferlazza

The Background

Topics presented during some previous talks:

- useful background in isogenies till the definition of group action
- background on Kuperberg and more
- state of the art on understanding how to optimize the classical-quantum trade-offs for different relevant scenarios

Storyboard

During this talk the central question will be:

How did we end up with qt-Pegasis as the SoTA quantum oracle?

- exploration of the evolution of group action evaluation algorithms
- identification of the requirements for an efficient quantum oracle
- realistic quantum resource estimation

Group Action(s): recall

Recall:

- $\mathcal{E}\ell\ell_p(\mathcal{O}) = \{E/\mathbb{F}_p \text{ supersingular elliptic curve} \mid \mathrm{End}_p(E) \cong \mathcal{O}\}/\sim_{\mathbb{F}_p}$
- $\mathrm{cl}(\mathcal{O}) = \{\text{invertible fractions } \mathcal{O} - \text{ideals}\}/\{\text{principal ideals}\}$

The action:

$$\begin{array}{rrr} \mathrm{cl}(\mathcal{O}) \times & \mathcal{E}\ell\ell_p(\mathcal{O}) \to & \mathcal{E}\ell\ell_p(\mathcal{O}) \\ ([\mathfrak{a}], & E) \mapsto & [\mathfrak{a}] \star E = E/E[\mathfrak{a}] =: E_{\mathfrak{a}} \end{array}$$

The goal: have an efficient algorithm that, given $\mathfrak{a}$, finds $E_{\mathfrak{a}}$

Group Action(s): from Clapoti to qt-Pegasis

Main idea of Clapoti

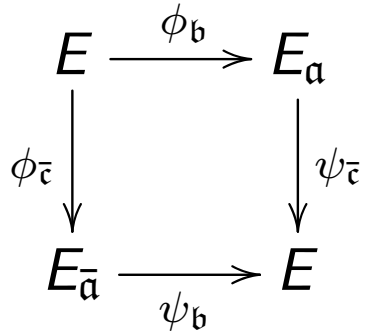
- Let $\mathfrak{a}$ be an ideal
- Find two equivalent ideals $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$ with coprime norms
- Define

$$N := \mathrm{N}(\mathfrak{b}) + \mathrm{N}(\mathfrak{c})$$

credits: Lorenz Panny's slides for Okinawa school

Group Action(s): from Clapoti to qt-Pegasis

Main idea of Clapoti



Kani's construction yields an N-isogeny

$$\Phi : E \times E \longrightarrow E_{\mathfrak{a}} \times E_{\bar{\mathfrak{a}}}$$

whose kernel characterizes the desired group action

The kernel of the isogeny is

$$\ker(\Phi) = \{(\widehat{\phi}_{\mathfrak{b}}(R), \psi_{\bar{\mathfrak{c}}}(R)) : R \in E_{\mathfrak{a}}[N]\}$$

credits: Lorenz Panny's slides for Okinawa school

Group Action(s): from Clapoti to qt-Pegasis

The problem: computing this description appears to require explicit knowledge of $\widehat{\phi}_{\mathfrak{b}}$ and $\psi_{\overline{\mathfrak{c}}}$

Fortunately, alternative description of the kernel:

$$\ker(\Phi) = \{([\mathrm{N}(\mathfrak{b})]R, \gamma(R)) : R \in E[N]\},$$

where $\gamma \in \mathrm{End}(E)$ generates the principal ideal $\mathfrak{b}\overline{\mathfrak{c}}$

This leads to a polynomial-time algorithm for evaluating the group action, even for arbitrary input ideals

credits: Lorenz Panny's slides for Okinawa school

Group Action(s): from Clapoti to qt-Pegasis

Moving to higher dimensions provides a significantly more efficient framework:

$$\text{Clapoti (2D)} \quad \longrightarrow \quad \textbf{Pegasis} \text{ (4D)}$$

The central problem becomes solving the norm equation

$$2^e = u \cdot N(\mathfrak{b}) + v \cdot N(\mathfrak{c}),$$

where u and v are sums of two squares

- lots of existence and efficiency restrictions that leave only a narrow range of feasible parameters

credits: Lorenz Panny's slides for Okinawa school

Group Action(s): from Clapoti to qt-Pegasis

Then, **Qlapoti**

It provides a direct method for solving

$$\mathrm{N}(\mathfrak{b}) + \mathrm{N}(\mathfrak{c}) = 2^e,$$

without introducing the auxiliary variables appearing in Pegasis

$\longrightarrow$ **qt-Pegasis**

It combines the simplicity of Qlapoti with the higher-dimensional framework of Pegasis, in a substantially simpler and faster algorithm, making it the current state of the art for quantum evaluation of isogeny group actions

credits: Lorenz Panny's slides for Okinawa school

qt-Pegasis: high-level description

- works over fields of characteristic $p = f2^e - 1$
- given an ideal $\mathfrak{a} \subset \mathbb{Z}\left[\frac{1+\sqrt{-p}}{2}\right]$ and an elliptic curve E oriented by $\mathbb{Z}\left[\frac{1+\sqrt{-p}}{2}\right]$, the goal is to compute $\varphi_{\mathfrak{a}} : E \to E_{\mathfrak{a}}$ corresponding to $\mathfrak{a}$

The steps:

- compute four equivalent ideals $\mathfrak{a}_1, \mathfrak{a}_2, \mathfrak{a}_3, \mathfrak{a}_4$ such that

$$\mathsf{n}(\mathfrak{a}_1) + \mathsf{n}(\mathfrak{a}_2) + \mathsf{n}(\mathfrak{a}_3) + \mathsf{n}(\mathfrak{a}_4) = 2^e,$$

  where $N_{1,2} = \mathsf{n}(\mathfrak{a}_1) + \mathsf{n}(\mathfrak{a}_2)$ and $N_{3,4} = \mathsf{n}(\mathfrak{a}_3) + \mathsf{n}(\mathfrak{a}_4)$ are both odd
- ...

qt-Pegasis: high-level description

The steps:

- ...
- define the following two-dimensional isogenies:

$$\Phi_1 := \begin{pmatrix} \varphi_{\mathfrak{a}_1} & \varphi_{\mathfrak{a}_2} \\ -\varphi_{\bar{\mathfrak{a}}_2} & \varphi_{\bar{\mathfrak{a}}_1} \end{pmatrix} : E \times E \to E_{\mathfrak{a}} \times E_{\bar{\mathfrak{a}}},$$

$$\Phi_2 := \begin{pmatrix} \varphi_{\bar{\mathfrak{a}}_3} & -\varphi_{\mathfrak{a}_4} \\ \varphi_{\mathfrak{a}_4} & -\varphi_{\mathfrak{a}_3} \end{pmatrix} : E_{\mathfrak{a}} \times E_{\bar{\mathfrak{a}}} \to E \times E.$$

- ...

qt-Pegasis: high-level description

The steps:

- ...
- let $\Psi_1 : E \times E \to \mathcal{A}$ be the pullback of Φ_2 under Φ_1 and $\Psi_2 : \mathcal{A} \to E \times E$ the pushforward of Φ_1 under Ψ_1, so they fit within the following commutative diagram:

$$\begin{array}{ccc} E \times E & \xrightarrow{\Phi_1} & E_{\mathfrak{a}} \times E_{\bar{\mathfrak{a}}} \\ \downarrow{\scriptstyle \Psi_1} & & \downarrow{\scriptstyle \Phi_2} \\ \mathcal{A} & \xrightarrow{\Psi_2} & E \times E \end{array}$$

- ...

qt-Pegasis: high-level description

The steps:

- ...
- construct the following four-dimensional isogeny:

$$\Phi := \begin{pmatrix} \Phi_1 & \widetilde{\Phi}_2 \\ -\Psi_1 & \widetilde{\Psi}_2 \end{pmatrix} : E \times E \times E \times E \to E_{\mathfrak{a}} \times E_{\bar{\mathfrak{a}}} \times \mathcal{A}$$

- in order to recover the codomain $E_{\mathfrak{a}}$
  - find a good description of the codomain $E_{\mathfrak{a}} \times E_{\bar{\mathfrak{a}}} \times \mathcal{A}$
  - from it, find $E_{\mathfrak{a}}$

Why qt-Pegasis?

Why is qt-Pegasis currently the best candidate for a quantum oracle?

It provides

- computation organized into quantum-friendly subroutines
- less resistant to reversible circuits
- an explicit algorithm whose resource requirements can be analysed

Quantum Cryptanalysis (so far)

Main application of the project: estimation of security parameters for cryptographic schemes based on group action

- for fixing parameter sizes, important that the complexity of attacks is not over-estimated
- slightly under-estimating the attack's complexity leads to a choice of larger than needed parameter (it doesn't create security issues, even if it may cost some performance)

$$\Downarrow$$

we seek a lower bound on the complexity of a quantum circuit executing Kuperberg's attack based on the qt-Pegasis algorithm

Quantum Cryptanalysis

Recall that qt-Pegasis algorithm consists in 2 parts:

1. norm equation solving step
2. isogeny computation step

Note: an intermediate conversion step is needed in order to use the output of the norm equation for the isogeny computation

The second step seems to dominate circuit size

$+$

the first step is more complicated to formalize as a quantum circuit

$\Downarrow$

only seek a rough lower bound on the circuit size of the norm equation step

Quantum Cryptanalysis

How does it work? Which are the instruments we need?

the computational problem $f(x)$ ✓

$\downarrow$

the quantum oracle(s) $|x, y\rangle \mapsto |x,\ y \oplus f(x)\rangle$

$\downarrow$

quantum attack

$\downarrow$

attack cost = (# oracle queries) $\times$ (oracle implementation cost)
$+$ processing cost

Quantum Oracles: norm equation solving step

Algorithm 2 qt-Pegasis: solving the norm equation

Input: An invertible ideal $\mathfrak{a} = (N, \alpha) \subset R = \mathbb{Z}\left[\frac{1+\sqrt{-p}}{2}\right]$, a number $e \in \mathbb{Z}$.

Output: $\mathfrak{a}_1, \mathfrak{a}_2, \mathfrak{a}_3, \mathfrak{a}_4 \sim \mathfrak{a}$ such that $\mathrm{n}(\mathfrak{a}_1) + \mathrm{n}(\mathfrak{a}_2) + \mathrm{n}(\mathfrak{a}_3) + \mathrm{n}(\mathfrak{a}_4) = 2^e\,\mathrm{n}(\mathfrak{a})$.

Let $\mathfrak{a}$ be the smallest ideal in $[\mathfrak{a}]$
Write $\mathfrak{a}$ as (N, α) where $N = \mathrm{n}(\mathfrak{a})$
Let $r = \mathrm{n}(\alpha)/N$
Let $k_{\max} = \min\{\sqrt{2^e/r}, \sqrt{N}\}$
Let $k = k_{\max}$
Compute v as $v \equiv (2^e - r(1+k^2))\,\mathrm{tr}(\alpha)^{-1} \pmod N$
Let $t_1 = r(2k-1)\,\mathrm{tr}(\alpha)^{-1} \pmod N$
Let $t_2 = -2r\,\mathrm{tr}(\alpha)^{-1} \pmod N$
while true **do**
  if $v < k\sqrt{2^e/N - k^2/4}$ **then**
    Let $(b_3, b_4) = (v - \lfloor v/k \rceil k, \lfloor v/k \rceil)$
    Solve the following equation for b_1, b_2 by using Cornacchia's algorithm

$$b_1^2 + b_2^2 = \frac{2^e - (1+k^2)r - \mathrm{tr}(\alpha)(b_3 + b_4 k)}{N} - (b_3^2 + b_4^2)$$

    if it has solution **then**
      Let $\alpha_1 = b_1 N,\ \alpha_2 = b_2 N,\ \alpha_3 = b_3 N + \alpha,\ \alpha_4 = b_4 N + k\alpha$
      return $\mathfrak{a}_i = \mathfrak{a}\overline{\alpha_i}/N$ for $i = 1, \ldots, 4$
  Update k as $k - 1$
  Update v as $v + t_1 \pmod N$
  Update t_1 as $t_1 + t_2 \pmod N$

Approximately:

- 1-8 $\rightarrow$ precomputation $\rightarrow |T\rangle$
- 9-11 + some singular rejections $\rightarrow$ first Boolean test f_1
- Cornacchia $\rightarrow$ second Boolean test f_2

so the quantum search (for k) is based on two oracles O_{f_1}, O_{f_2}, defined upon two Boolean functions $f_1, f_2 : [0; 2^\iota - 1] \mapsto \{0, 1\}$

$$\begin{cases} O_{f_1} : |T\rangle\,|i\rangle \rightarrow (-1)^{f_1(i)}\,|T\rangle\,|i\rangle \\ O_{f_2} : |T\rangle\,|i\rangle \rightarrow (-1)^{f_2(i)}\,|T\rangle\,|i\rangle \end{cases}$$

Quantum Oracles: isogeny computation step

- computation of the basis of the dimension-4 isogeny
  $\rightarrow \mathcal{B}$ compatible with $\ker(F)$
- construction of a theta structure compatible with $\mathcal{B}$
  - straightforward and can be classically computed
    $\rightarrow \theta^{E^4}$ depending on an invertible matrix $\mathbf{M}_2$ dependent on the qt-Pegasis solution and can be expressed as $\sum_{i=1}^{4} v_i \mathbf{M_2}_i$ with $v_i \in \{0, 1\}$ given by some indicator function
  - globally, apart from v_i, all the precomputations can be done classically
- splitting
  - retrieve a theta null point over the abelian variety $E_{\mathfrak{a}} \times E_{\overline{\mathfrak{a}}} \times A$
  - transform our final theta structure into a product theta structure, i.e. a theta structure of the form $\theta^{E_{\mathfrak{a}}} \otimes \theta^{E_{\overline{\mathfrak{a}}}} \otimes \theta^{A}$ $\rightarrow$ recover $E_{\mathfrak{a}}$

Kuperberg's Attack

Hidden Shift Problem

Let G be a group, acting on a set S, and $g_0, g_1 : G \to S$ functions for which $\exists s \in G$ such that $g_1(x) = g_0(s \cdot x)$. Find s.

In terms of isogenies group action:

- $G = \mathrm{cl}(\mathcal{O}) \curvearrowright S = \mathcal{E}\ell\ell_p(\mathcal{O})$
- base curve E_0, public curve $E_1 = s \star E_0$, $s \in \mathrm{cl}(\mathcal{O})$
- $g_0([\mathfrak{a}]) = [\mathfrak{a}] \star E_0$,
  $g_1([\mathfrak{a}]) = [\mathfrak{a}] \star E_1 = [\mathfrak{a}] \star (s \star E_0) = g_0([\mathfrak{a}]s)$

Kuperberg's Attack

From Hidden Shift to Hidden Subgroup

- Kuperberg's algorithm does not solve Hidden Shift directly
- the problem is transformed into an instance of the Hidden Subgroup Problem

- define the semidirect product

$$G \rtimes \mathbb{Z}_2$$

where elements are pairs $([a], 0),\ ([a], 1)$, where the second coordinate simply indicates which function is being queried

- define

$$F : G \rtimes \mathbb{Z}_2 \to S$$

by $F([a], 0) = g_0([a]),\ F([a], 1) = g_1([a])$

- ...

Kuperberg's Attack

From Hidden Shift to Hidden Subgroup

- ...
- using the hidden shift relation,

$$F([\mathfrak{a}], 1) = g_1([\mathfrak{a}]) = g_0([\mathfrak{a}]s) = F([\mathfrak{a}]s, 0),$$

  so F is constant on pairs $([\mathfrak{a}], 0),\ ([\mathfrak{a}]s, 1)$

- specifically, if we consider the subgroup

$$H = \{(\mathrm{id}, 0), (s, 1)\}$$

  its cosets are precisely

$$\{([\mathfrak{a}], 0), ([\mathfrak{a}]s, 1)\},$$

  which are exactly the sets on which F is constant

The Finale

Towards a Complete Resource Estimation

- work in progress
- estimates represent the current state of our analysis and may evolve as the implementation is further refined

Current results include:

- an estimate of the required prime bits size
- a nearly complete resource estimate, pending a refined analysis of the isogeny chain component of the oracle
- the quantum gate count for this parameter choice, optimized over the available r-ary tree configurations
- resource estimates obtained under a number of reasonable assumptions

Numeric Cost Approximation: the SoTA

We use an r-ary tree on a group size n, with r small and even (these are the optimal cases for the sizes we care about)

Then the total cost in quantum gates of the algorithm is

$$\mathcal{C} = 2^Q \cdot K + \frac{1}{T} \cdot 2^C$$

where:

- before factoring in the cost of the oracle, the quantum gate count is 2^Q where $Q = \sqrt{2\log_2(r)n/(r-1)}$
- oracle cost is K
- have to keep $\sim 2^{20}$ qubits stable while performing a classical operation of complexity 2^C where $C = Qr/2$
- 2^C classical bit operations can be performed T times faster than 2^C quantum gates

Numeric Cost Approximation: the SoTA

Assumptions:

- definition of 'easier than breaking AES-128' as fewer than 2^{81} quantum gates
- p has 2048 bites, then n has 1024 bites
- for this p, we have $K \in [2^{48}, 2^{52}]$

Quantum gate count with all the possible optimal choices for r-ary tree, assuming that computing $2^{\mathcal{C}}$ classical bit operations is T times faster than computing $2^{\mathcal{C}}$ quantum gates

$\mathcal{C}$	$T = 1$	$T = 2$	$T = 2^{10}$	$T = 2^{40}$
$r = 2$	$[2^{93}, 2^{97}]$	$[2^{93}, 2^{97}]$	$[2^{93}, 2^{97}]$	$[2^{93}, 2^{97}]$
$r = 4$	$[2^{85}, 2^{89}]$	$[2^{85}, 2^{89}]$	$[2^{85}, 2^{89}]$	$[2^{85}, 2^{89}]$
$r = 6$	2^{98}	2^{97}	2^{88}	$[2^{81}, 2^{85}]$
$r = 8$	2^{119}	2^{118}	2^{109}	$[2^{79}, 2^{82}]$
$r = 10$	2^{137}	2^{136}	2^{127}	2^{97}

Figure: Table with the range of values of $\mathcal{C}$

Numeric Cost Approximation: the SoTA

Thus, a 2048-bites p achieves NIST Level I security only if either:

- the oracle costs at least 2^{51}, in which case $\mathcal{C}$ is instead dominated by $2^Q \cdot K$, or
- the time taken for $2^C = 2^{119}$ classical bit operations, including parallelization, is more than the time taken for 2^{81} quantum gates; i.e. $T \leq 2^{38}$

  Note: this needs to be true in a hypothetical future time at which quantum computers *can* undertake computations costing $2^{78} (\leq 2^Q \cdot K)$ quantum gates

Thank You!

Questions?