Radical 2-isogenies and cryptographic hash functions in dimensions 1, 2 and 3

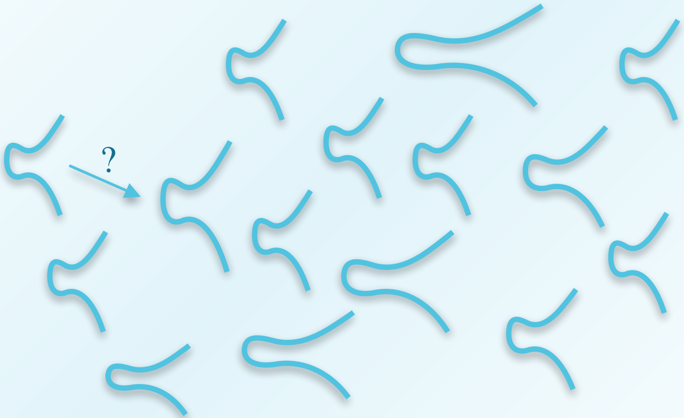
Sabrina Kunzweiler, Luciano Maino, Tomoki Moriya, Christophe Petit,
Giacomo Pope, Damien Robert, **Miha Stopar**, and Yan Bo Ti

privacy + scaling explorations

Isogeny graphs (genus 1)

- The *pure isogeny problem*—finding an explicit isogeny between two elliptic curves.
- Most isogeny-based protocols rely on the pure isogeny problem or on some variants of this problem.

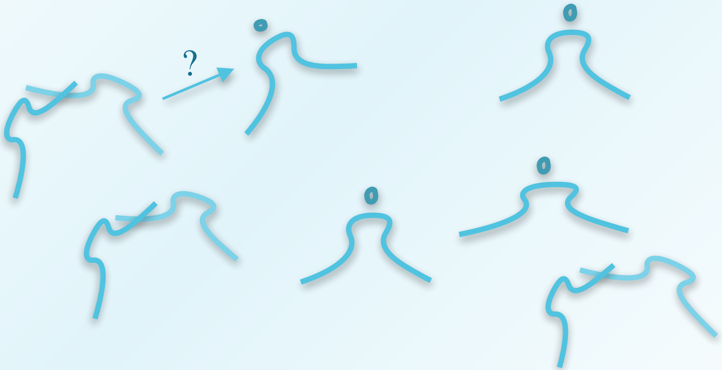
Genus 1 $(y^2 = x^3 + ax + b)$



Isogeny graphs (genus 2)

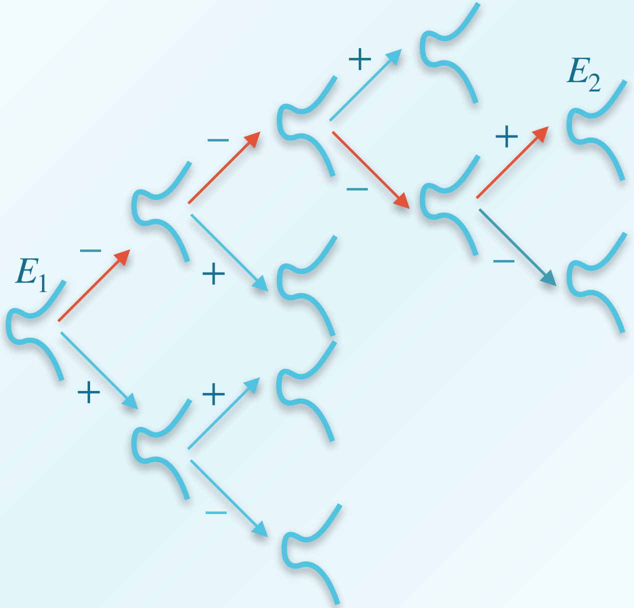
- The isogeny problem can be generalized to higher dimensions (genus 2, genus 3,…).
- Genus 2: abelian surfaces (picture $\Longrightarrow$).
- Genus 3: abelian threefold.

Genus 2 $\quad y^2 = f(x),\ deg(f) \in \{5,6\}$



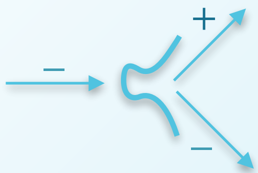
CGL hash function ’08

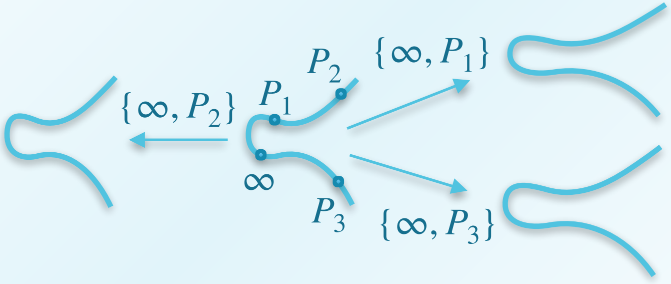
- Random walks in an l-isogeny graph.
- For $l = 2$, each node has exactly 3 outgoing 2-isogenies.
- The hash of the message is the **j-invariant** of the final curve.
- For example, the message $m = 0001$ corresponds to the path $- - - +$: $H(m) = j(E_2)$.



CGL hash function '08

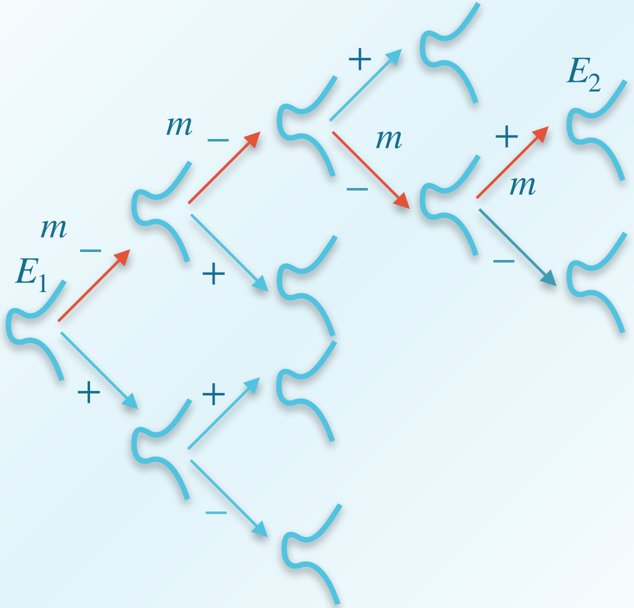
- For $l = 2$ isogeny, one needs to compute 2-torsion point (3 possibilities: $+$, $-$, backtracking).
- To compute 2-torsion point, one needs to do a square root computation in $\mathbb{F}_{p^2}$ with a handful of extra operations.
- $y^2 = (x - \alpha_1)(x - \alpha_2)(x - \alpha_3) \implies E[2] = \{(\alpha_1,0), (\alpha_2,0), (\alpha_3,0), \infty\}$





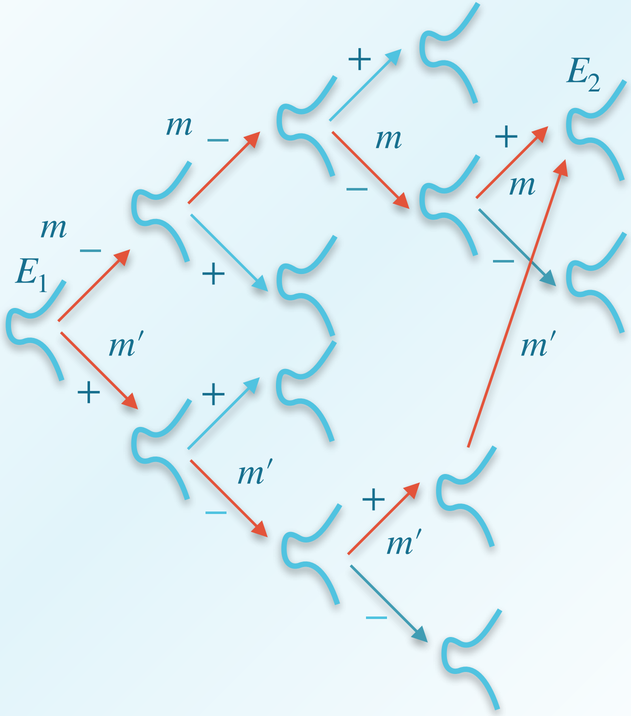
Preimage resistance

- Given $j \in \mathbb{F}_{p^2}$, it is hard to find m with $H(m) = j$.
- It is the same as finding a 2^*-isogeny between two supersingular elliptic curves.



Collision resistance

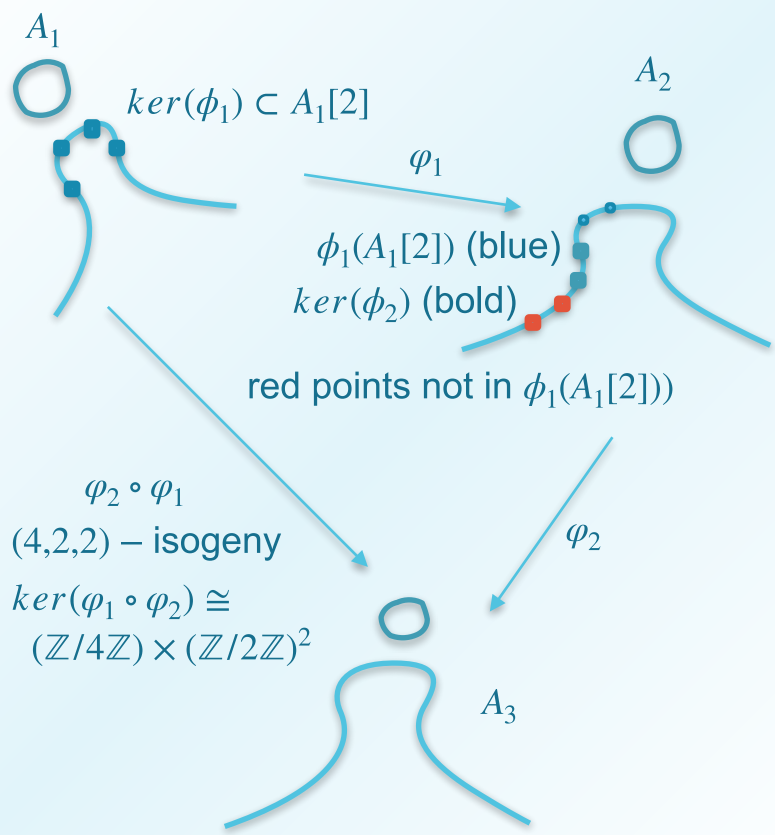
- It is hard to find two messages $m \neq m'$ with $H(m) \neq H(m')$.
- The same as finding an endomorphism of degree 2^* of E_1 (path of m and the "inverse" of path of m' give a cycle).
- Finding a non-scalar endomorphism of a random supersingular elliptic curve is a common hardness assumption in isogeny-based cryptography.



Genus > 1 hash functions

- Takashima hash function ’18: supersingular graph.
- Flynn-Ti ’19: an attack on Takashima hash. Problems: supersingular graph, bad extensions.
- Bad extension: $ker(\varphi_2) \cap \phi_1(A_1[2])$ non-trivial (picture $\Longrightarrow$).
- Note: $A_1[2] \cong (\mathbb{Z}/2\mathbb{Z})^4$, $A_2[2] \cong (\mathbb{Z}/2\mathbb{Z})^4$.

Note: points are actually on the Jacobian of the curve, not on the curve itself!

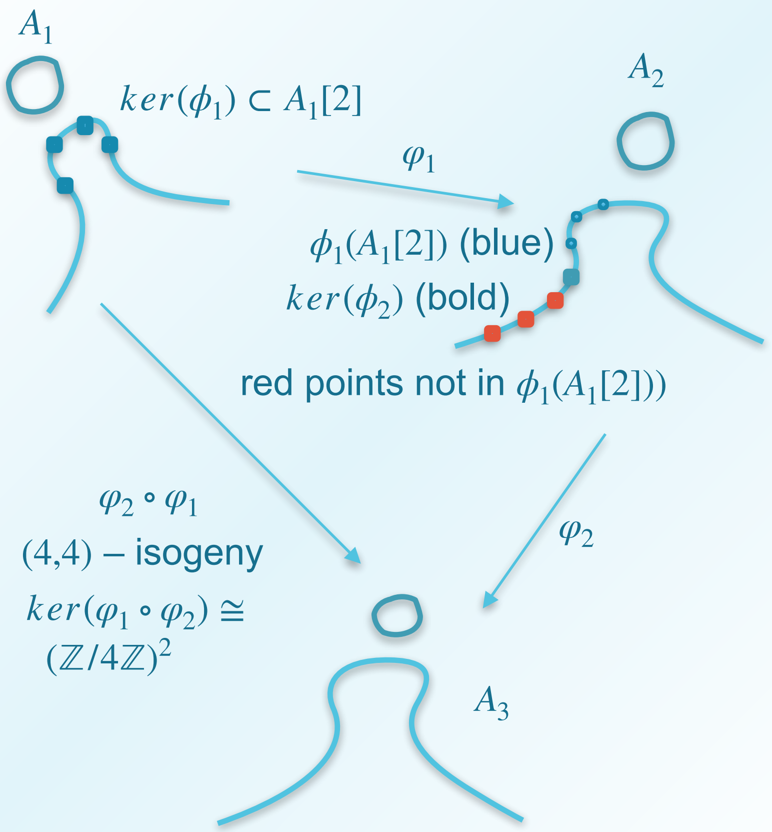


Genus > 1 hash functions

- Castryck-Decru-Smith hash function ’20: superspecial graph, using only good extensions.

- Good extension: $ker(\varphi_2) \cap \phi_1(A_2[2]) = \{0\}$ (picture $\Longrightarrow$).

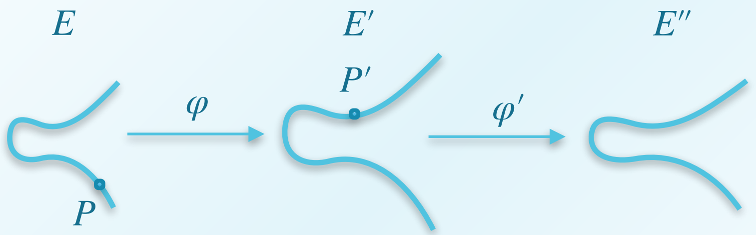
Note: points are actually on the Jacobian of the curve, not on the curve itself!



Genus > 1 hash functions

- Castryck-Decru hash function '21: *multiradical isogenies*.
- Radical isogenies: instead of generating a random N-torsion point on E', a point P' is constructed algebraically (picture $\Longrightarrow$).
- Multiradical isogenies: generalization to genus $g > 1$.

$\varphi : E \to E' = E/ < P >$



$\varphi' : E' \to E'' = E/ < P' >$

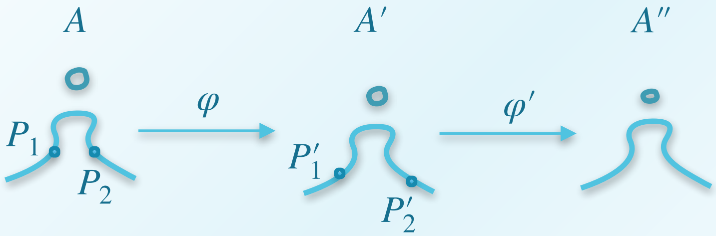
$\varphi' \circ \varphi$ cyclic of degree N^2

Note: points are actually on the Jacobian of the curve, not on the curve itself!

Multiradical isogenies

- Input is a tuple $(A, P_1, \ldots, P_g)$.
- Output is a tuple $(A', P'_1, \ldots, P'_g)$
- $(P'_1, \ldots, P'_g)$ generate the kernel of an N-isogeny φ'.
- $\varphi' \circ \varphi$ is a good extension (type $(N^2, \ldots, N^2)$).

$\varphi : A \to A' = A / < P_1, P_2 >$



$\varphi' : A' \to A'' = A' / < P'_1, P'_2 >$

$\varphi' \circ \varphi$ is of degree N^2

Theta model

- A theta model is an alternative coordinate system for describing abelian varieties.
- Over $\mathbb{C}$, every principally polarized abelian variety (ppav) of dimension g can be described analytically as a complex torus $A = \mathbb{C}^g/\Lambda$.
- Given lattice Λ, you can define special functions on $\mathbb{C}^g$ called theta functions.
- Instead of working with curve equations, one works with theta constants.

Genus 1 $\quad (y^2 = x^3 + ax + b)$

$$\theta(z, \tau) = \sum_{n \in \mathbb{Z}^g} \exp\left(\pi i\, n^\top \tau n + 2\pi i\, n^\top z\right)$$

Level-2 theta structure

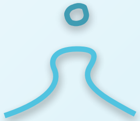
- Ppav equipped with a level-2 theta structure θ_A (informal and incomplete intuition: choice of labelling of the 2-torsion points).
- Coordinate functions of θ_A:
  $\theta_A(P) = (\theta_i(P))_{i \in (\mathbb{Z}/2\mathbb{Z})^g}$
- It allows to identify A with a point
  $a = \theta_A(0) = (a_{0\ldots0}, \ldots, a_{1\ldots1}) \in \mathbb{P}^{2^g-1}$
  (known as *level-2 theta null point* of A).

$g = 1 : a = (a_0, a_1) \in \mathbb{P}^1$

$g = 2 : a = (a_{00}, a_{01}, a_{10}, a_{11}) \in \mathbb{P}^3$

$g = 3 : a = (a_{000}, \ldots, a_{111}) \in \mathbb{P}^7$

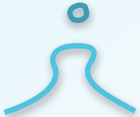
A



$\theta_A(0) = (a_{00}, a_{01}, a_{10}, a_{11})$

identifies A

A'



$\theta'_A(0) = (a'_{00}, a'_{01}, a'_{10}, a'_{11})$

identifies A'

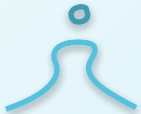
2-isogeny in theta model

- Formula for 2-isogeny $\phi : A \to B$ transforms a theta null point $a \in \mathbb{P}^{2^g-1}$ to a theta null point $b \in \mathbb{P}^{2^g-1}$.
- Dartois-Maino-Pope-Robert '24 formula consists only of:
  - Squaring
  - Scaling
  - Hadamard transform (H)

A

$\phi : A \to B$

B



$\theta_A(0)$ identifies A

$\theta_B(0)$ identifies B

$g = 1:$

$$\begin{pmatrix} p_0 \\ p_1 \end{pmatrix} \overset{\text{Squaring}}{\Longrightarrow} \begin{pmatrix} q_0 = p_0^2 \\ q_1 = p_1^2 \end{pmatrix} \overset{\text{Hadamard}}{\Longrightarrow} \begin{pmatrix} r_0 = q_0 + q_1 \\ r_1 = q_0 - q_1 \end{pmatrix}$$

$$\overset{\text{Scaling}}{\Longrightarrow} \begin{pmatrix} s_0 = r_0 / \tilde{b}_0 \\ s_1 = r_1 / \tilde{b}_1 \end{pmatrix} \overset{\text{Hadamard}}{\Longrightarrow} \begin{pmatrix} t_0 = s_0 + s_1 \\ t_1 = s_0 - s_1 \end{pmatrix}$$

$\begin{pmatrix} \tilde{b}_0 \\ \tilde{b}_1 \end{pmatrix}$ dual theta point of $\theta_B(0) : H(\theta_B(0))$

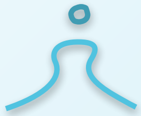
Radical formula

- Idea: $\theta_B(0) = \phi(\theta_A(0))$.
- We have: $\tilde{b}_i^2 = x_i$.
- In general dimension g: $\tilde{b}_i^2 = x_i$ for $i = 0,\ldots,2^g - 1$.

A

$\phi : A \to B$

B



$\theta_A(0)$ identifies A

$\theta_B(0)$ identifies B

$g = 1$:

$$\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \overset{\text{Squaring}}{\Longrightarrow} \begin{pmatrix} a_0^2 \\ a_1^2 \end{pmatrix} \overset{\text{Hadamard}}{\Longrightarrow} \begin{pmatrix} x_0 = a_0^2 + a_1^2 \\ x_1 = a_0^2 - a_1^2 \end{pmatrix}$$

$$\overset{\text{Scaling}}{\Longrightarrow} \begin{pmatrix} \tilde{b}_0 = x_0/\tilde{b}_0 \\ \tilde{b}_1 = x_1/\tilde{b}_1 \end{pmatrix} \overset{\text{Hadamard}}{\Longrightarrow} \begin{pmatrix} b_0 = \tilde{b}_0 + \tilde{b}_1 \\ b_1 = \tilde{b}_0 - \tilde{b}_1 \end{pmatrix}$$

$\begin{pmatrix} \tilde{b}_0 \\ \tilde{b}_1 \end{pmatrix}$ dual theta point of $\theta_B(0) : H(\theta_B(0))$

Radical formula

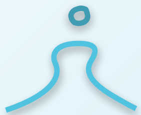
- Idea: $\theta_B(0) = \phi(\theta_A(0))$.
- The operations:
  - Squaring
  - Square roots
  - Hadamard transform (H)

A

$\theta_A(0)$ identifies A

$\phi : A \to B$

B



$\theta_B(0)$ identifies B

$g = 1:$

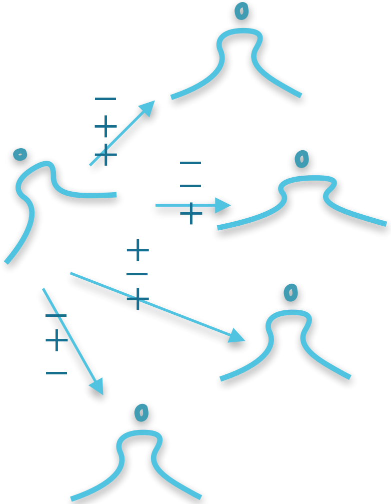
$$\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \overset{\text{Squaring}}{\Longrightarrow} \begin{pmatrix} a_0^2 \\ a_1^2 \end{pmatrix} \overset{\text{Hadamard}}{\Longrightarrow} \begin{pmatrix} x_0 = a_0^2 + a_1^2 \\ x_1 = a_0^2 - a_1^2 \end{pmatrix}$$

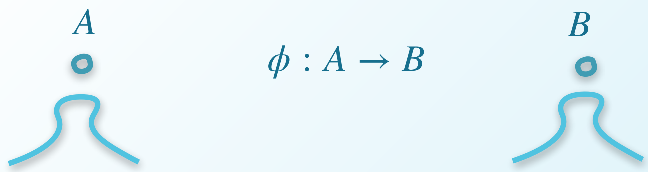
$$\overset{\text{Sqrt}}{\Longrightarrow} \begin{pmatrix} \tilde{b}_0 = \pm\sqrt{x_0} \\ \tilde{b}_1 = \pm\sqrt{x_1} \end{pmatrix} \overset{\text{Hadamard}}{\Longrightarrow} \begin{pmatrix} b_0 = \tilde{b}_0 + \tilde{b}_1 \\ b_1 = \tilde{b}_0 - \tilde{b}_1 \end{pmatrix}$$

$\begin{pmatrix} \tilde{b}_0 \\ \tilde{b}_1 \end{pmatrix}$ dual theta point of $\theta_B(0) : H(\theta_B(0))$

Theta-CGL

- Public: $a = \theta_A(0)$.
- Input: $m = (m_1, \ldots, m_n)$, $m_i \in \{-1,1\}^{g(g+1)/2}$.
- We have 2^g sign choices for square roots (note $k = 2^g - 1$), but it turns out fixing $g(g+1)/2$ square roots determines the others.
- Output: $b = \theta_B(0)$.





$\theta_A(0)$ identifies A $\qquad$ $\theta_B(0)$ identifies B

$$a = \begin{pmatrix} a_0 \\ \cdots \\ a_k \end{pmatrix} \Longrightarrow \begin{pmatrix} a_0^2 \\ \cdots \\ a_k^2 \end{pmatrix} \Longrightarrow \begin{pmatrix} x_0 \\ \cdots \\ x_k \end{pmatrix} = H(a^2)$$

$$\overset{\text{Sqrt}}{\Longrightarrow} \begin{pmatrix} \tilde{b}_0 = \pm\sqrt{x_0} \\ \cdots \\ \tilde{b}_k = \pm\sqrt{x_k} \end{pmatrix} \Longrightarrow H(\begin{pmatrix} \tilde{b}_0 \\ \cdots \\ \tilde{b}_k \end{pmatrix}) = b$$

choosing the signs means choosing the isogeny

Theta-CGL: why go to dimension g > 1?

	g = 1	g = 2	g = 3
superspecial a.v.s.	$\approx p$	$\approx p^3$	$\approx p^6$
solving the isogeny problem	$O(\sqrt{p})$	$O(p)$	$O(p^2)$

Going to higher dimensions allows us to significantly reduce the size of the prime, essentially at no additional cost!

Theta-CGL: Rust implementation

Dimension	2-radical (μs)	4-radical (μs)	8-radical (μs)
g = 1	3153	2037	1737
g = 2	989	742	-
g = 3	432	-	-

Thank you!

PSE Discord