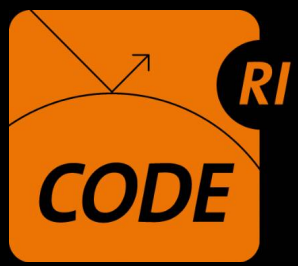
More Efficient Isogeny Proofs of Knowledge via Canonical Modular Polynomials

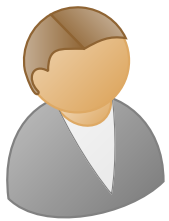
Joint work with Thomas den Hollander,
Sören Kleine, Marzio Mula & Daniel Slamanig

Sebastian A. Spindler • August 18th, 2025



Research Institute
Cyber Defence
Universität der Bundeswehr München

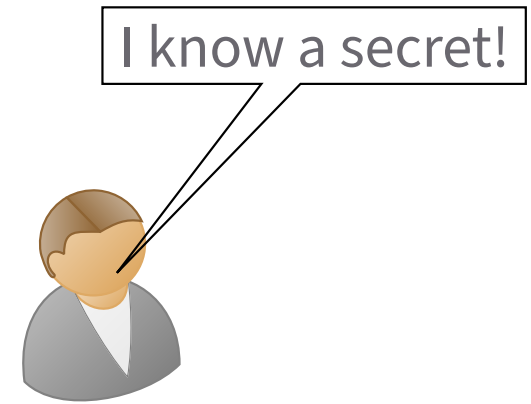
The Scenario



Preston

Veronica

The Scenario

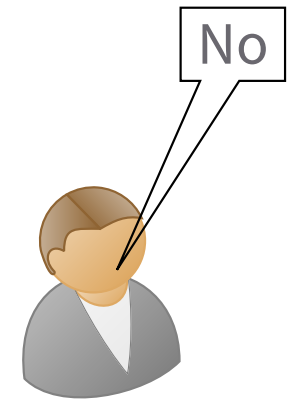


Preston

Veronica

The Scenario

The Scenario

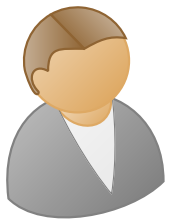


Preston

Veronica

The Scenario

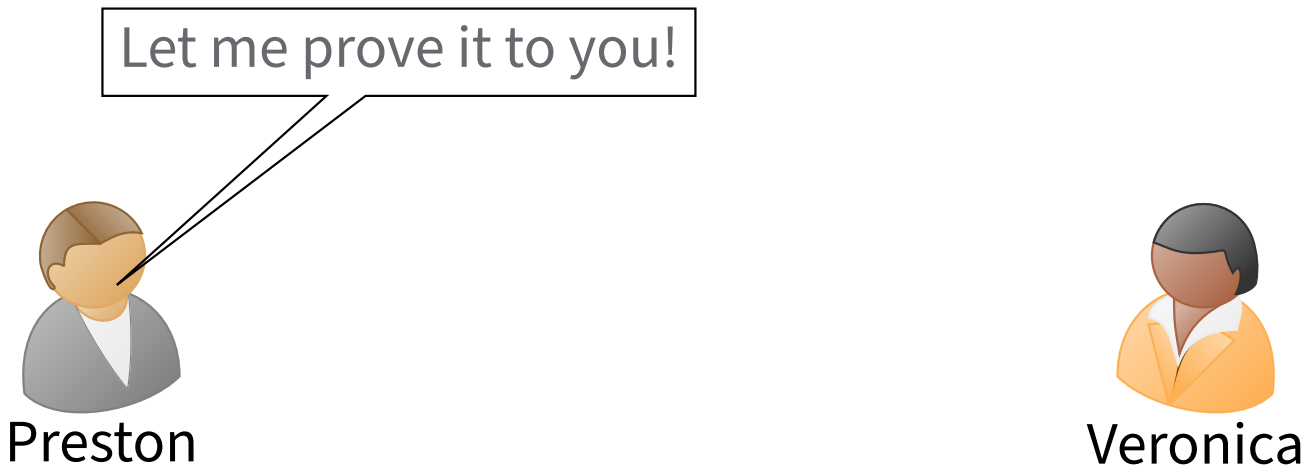
The Scenario



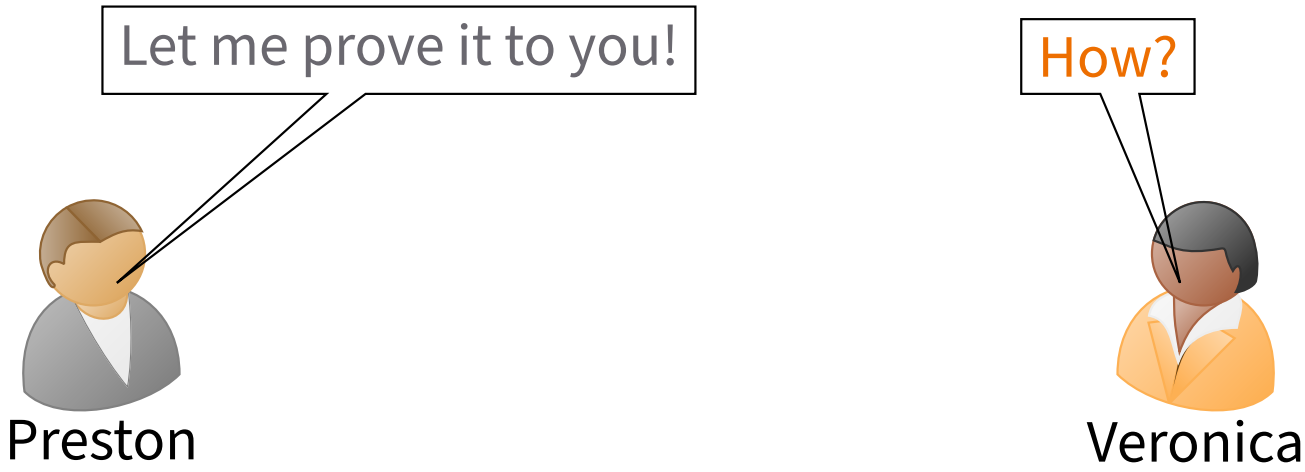
Preston

Veronica

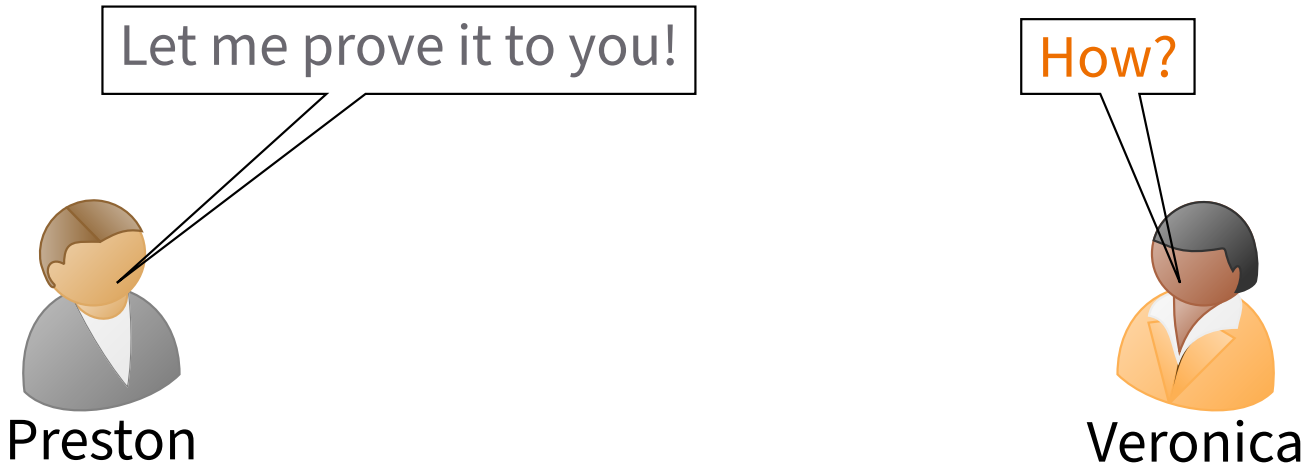
The Scenario



The Scenario

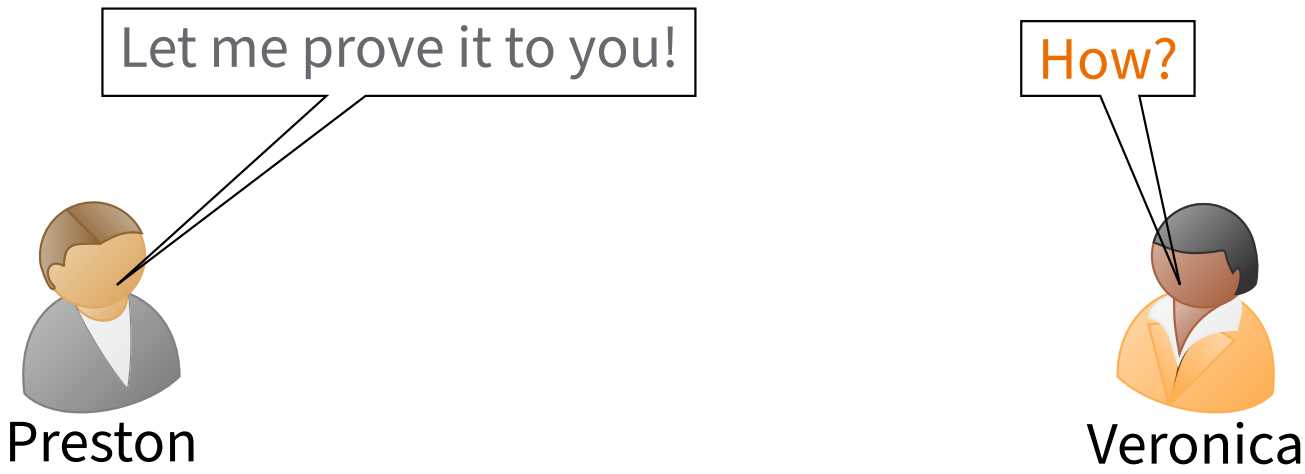


The Scenario



With a **zero-knowledge proof of knowledge**!

The Scenario



With a **zero-knowledge proof of knowledge**!

Multiple applications: Multi-party protocols (honest behavior control, trustless setup), signature schemes, …

General zk-Proofs of Knowledge

We want four properties from a zero-knowledge proof of knowledge:

General zk-Proofs of Knowledge

We want four properties from a zero-knowledge proof of knowledge:

- **Completeness**: If Preston knows a secret, he should be able to convince Veronica

General zk-Proofs of Knowledge

We want four properties from a zero-knowledge proof of knowledge:

- **Completeness**: If Preston knows a secret, he should be able to convince Veronica
- **Knowledge Soundness**: Preston cannot deceive Veronica if he doesn't know a secret

General zk-Proofs of Knowledge

We want four properties from a zero-knowledge proof of knowledge:

- **Completeness**: If Preston knows a secret, he should be able to convince Veronica
- **Knowledge Soundness**: Preston cannot deceive Veronica if he doesn't know a secret
- **Zero-Knowledge**: Veronica should not learn Preston's secret

General zk-Proofs of Knowledge

We want four properties from a zero-knowledge proof of knowledge:

- **Completeness**: If Preston knows a secret, he should be able to convince Veronica
- **Knowledge Soundness**: Preston cannot deceive Veronica if he doesn't know a secret
- **Zero-Knowledge**: Veronica should not learn Preston's secret
- **Non-Interactivity**: Preston can produce a proof without Veronica, and Veronica can verify it without Preston

What to prove? Isogenies!

Knowledge of secret should be special

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

Definition

Let $\ell \neq p$ be primes.

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

Definition

*Let $\ell \neq p$ be primes. The **supersingular isogeny graph** $G_\ell(p)$ is the connected directed multigraph with:*

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

Definition

*Let $\ell \neq p$ be primes. The **supersingular isogeny graph** $G_\ell(p)$ is the connected directed multigraph with:*

- *Vertices: j-invariants $j(E) \in \mathbb{F}_{p^2}$ of supersingular curves*

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

Definition

*Let $\ell \neq p$ be primes. The **supersingular isogeny graph** $G_\ell(p)$ is the connected directed multigraph with:*

- *Vertices: j-invariants $j(E) \in \mathbb{F}_{p^2}$ of supersingular curves*
- *Edges $j(E) \to j(E')$ correspond* to ℓ-isogenies $E \to E'$*

*Up to some notion of equivalence

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

Definition

*Let $\ell \neq p$ be primes. The **supersingular isogeny graph** $G_\ell(p)$ is the connected directed multigraph with:*

- *Vertices: j-invariants $j(E) \in \mathbb{F}_{p^2}$ of supersingular curves*
- *Edges $j(E) \to j(E')$ correspond* to ℓ-isogenies $E \to E'$*

Our underlying hard problem:

Finding a path between $j(E_0)$ and $j(E_1)$ in $G_\ell(p)$ is hard!

*Up to some notion of equivalence

What to prove? Isogenies!

Knowledge of secret should be special $\rightsquigarrow$ Need hard problem!

Definition

*Let $\ell \neq p$ be primes. The **supersingular isogeny graph** $G_\ell(p)$ is the connected directed multigraph with:*

- *Vertices: j-invariants $j(E) \in \mathbb{F}_{p^2}$ of supersingular curves*
- *Edges $j(E) \to j(E')$ correspond* to ℓ-isogenies $E \to E'$*

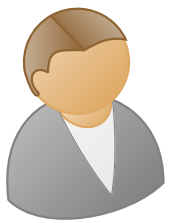
Our underlying hard problem:

Finding a path between $j(E_0)$ and $j(E_1)$ in $G_\ell(p)$ is hard!

(Parameter sizes: ℓ small, $p \approx 2^{2\lambda}$ for λ bits security)

*Up to some notion of equivalence

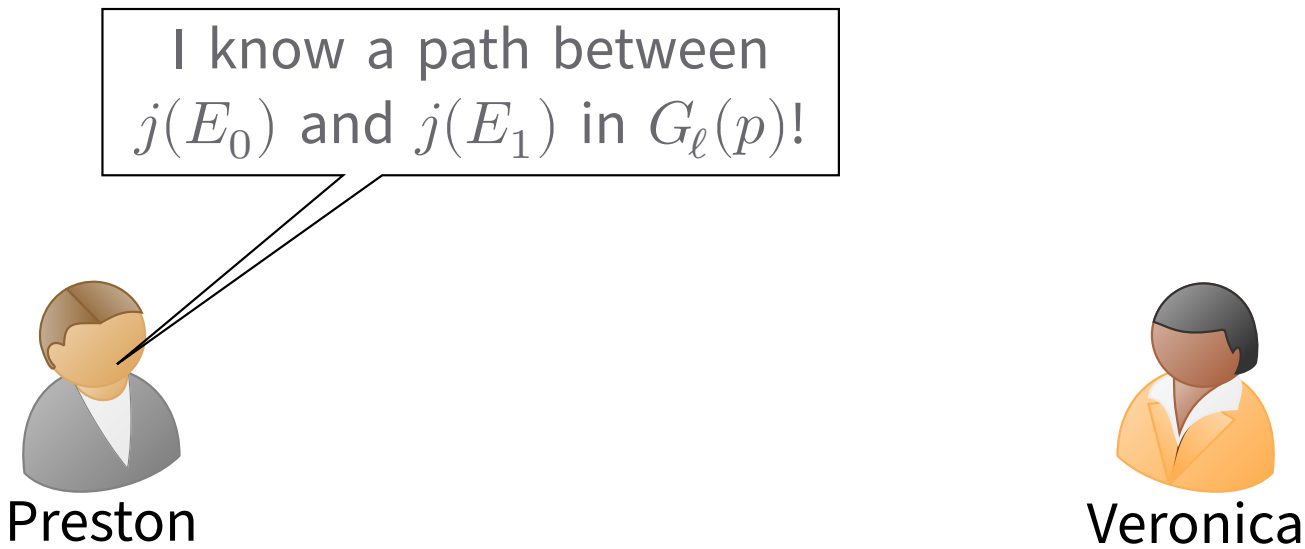
The Isogeny Scenario



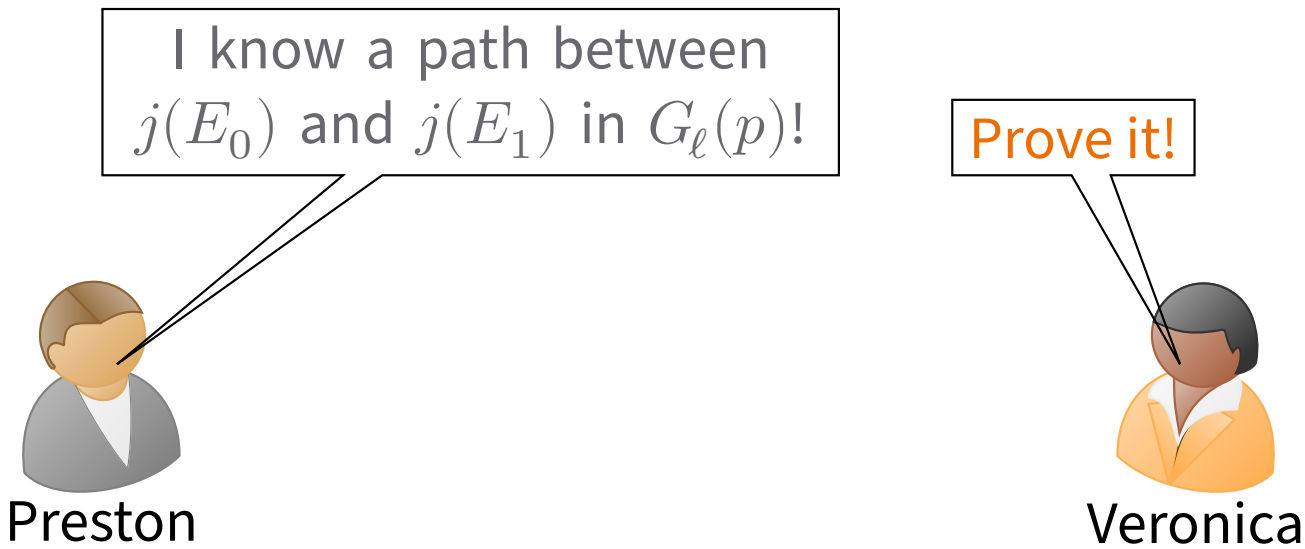
Preston

Veronica

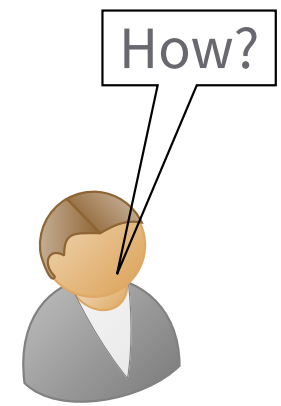
The Isogeny Scenario



The Isogeny Scenario



The Isogeny Scenario



Preston

Veronica

Generic Machinery: R1CS-based zk-SNARKs

- A rank-1 constraint system is of the form

$$(\mathsf{A}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} \bullet (\mathsf{B}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} = (\mathsf{C}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix}$$

  with statement matrices $\mathrm{A}, \mathrm{B}, \mathrm{C} \in \mathbb{F}^{m \times (1+n)}$ and witness vector $w = (1, w_1, \ldots, w_n)$

Generic Machinery: R1CS-based zk-SNARKs

- A rank-1 constraint system is of the form

$$(\mathsf{A}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} \bullet (\mathsf{B}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} = (\mathsf{C}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix}$$

  with statement matrices $\mathrm{A}, \mathrm{B}, \mathrm{C} \in \mathbb{F}^{m \times (1+n)}$ and witness vector $w = (1, w_1, \ldots, w_n)$

Generic Machinery: R1CS-based zk-SNARKs

- A rank-1 constraint system is of the form

$$(\mathsf{A}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} \bullet (\mathsf{B}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} = (\mathsf{C}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix}$$

  with statement matrices $\mathrm{A}, \mathrm{B}, \mathrm{C} \in \mathbb{F}^{m \times (1+n)}$ and witness vector $w = (1, w_1, \ldots, w_n)$

- Can plug this into a zk-SNARK for R1CS (e.g. Aurora, Ligero) to obtain compact & efficient zero-knowledge proof of knowledge

Generic Machinery: R1CS-based zk-SNARKs

- A rank-1 constraint system is of the form

$$(\mathsf{A}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} \bullet (\mathsf{B}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix} = (\mathsf{C}) \begin{pmatrix} 1 \\ w_1 \\ \vdots \\ w_n \end{pmatrix}$$

  with statement matrices $\mathrm{A}, \mathrm{B}, \mathrm{C} \in \mathbb{F}^{m \times (1+n)}$ and witness vector $w = (1, w_1, \dots, w_n)$

- Can plug this into a zk-SNARK for R1CS (e.g. Aurora, Ligero) to obtain compact & efficient zero-knowledge proof of knowledge
- This approach was first pursued in [CLL23]

Canonical Modular Polynomials

Can we find small polynomials to represent ℓ-isogenies?

Canonical Modular Polynomials

Can we find small polynomials to represent ℓ-isogenies?

Yes! But we have to restrict to $\ell \in \{2, 3, 5, 7, 13\}$ for technical reasons.

Canonical Modular Polynomials

Can we find small polynomials to represent ℓ-isogenies?

Yes! But we have to restrict to $\ell \in \{2, 3, 5, 7, 13\}$ for technical reasons.

$$\begin{aligned}
\Phi_2^c(X, j) &= X^3 + 48X^2 + 768X + 4096 - X \cdot j, \\
\Phi_3^c(X, j) &= X^4 + 36X^3 + 270X^2 + 756X + 729 - X \cdot j, \\
\Phi_5^c(X, j) &= X^6 + 30X^5 + 315X^4 + 1300X^3 \\
&\quad + 1575X^2 + 750X + 125 - X \cdot j, \\
\Phi_7^c(X, j) &= X^8 + 28X^7 + 322X^6 + 1904X^5 + 5915X^4 \\
&\quad + 8624X^3 + 4018X^2 + 748X + 49 - X \cdot j, \\
\Phi_{13}^c(X, j) &= X^{14} + 26X^{13} + 325X^{12} + 2548X^{11} + 13832X^{10} \\
&\quad + 54340X^9 + 157118X^8 + 333580X^7 + 509366X^6 \\
&\quad + 534820X^5 + 354536X^4 + 124852X^3 \\
&\quad + 15145X^2 + 746X + 13 - X \cdot j.
\end{aligned}$$

Canonical Modular Polynomials

Can we find small polynomials to represent ℓ-isogenies?

Yes! But we have to restrict to $\ell \in \{2, 3, 5, 7, 13\}$ for technical reasons.

$$\Phi_2^c(X, j) = X^3 + 48X^2 + 768X + 4096 - X \cdot j,$$

$$\Phi_3^c(X, j) = X^4 + 36X^3 + 270X^2 + 756X + 729 - X \cdot j,$$

$$\Phi_5^c(X, j) = X^6 + 30X^5 + 315X^4 + 1300X^3 + 1575X^2 + 750X + 125 - X \cdot j,$$

$$\Phi_7^c(X, j) = X^8 + 28X^7 + 322X^6 + 1904X^5 + 5915X^4 + 8624X^3 + 4018X^2 + 748X + 49 - X \cdot j,$$

$$\Phi_{13}^c(X, j) = X^{14} + 26X^{13} + 325X^{12} + 2548X^{11} + 13832X^{10} + 54340X^9 + 157118X^8 + 333580X^7 + 509366X^6 + 534820X^5 + 354536X^4 + 124852X^3 + 15145X^2 + 746X + 13 - X \cdot j.$$

Canonical Modular Polynomials

Can we find small polynomials to represent ℓ-isogenies?

Yes! But we have to restrict to $\ell \in \{2, 3, 5, 7, 13\}$ for technical reasons.

$$\Phi_2^c(X, j) = X^3 + 48X^2 + 768X + 4096 - X \cdot j,$$

$$\Phi_3^c(X, j) = X^4 + 36X^3 + 270X^2 + 756X + 729 - X \cdot j,$$

$$\Phi_5^c(X, j) = X^6 + 30X^5 + 315X^4 + 1300X^3 + 1575X^2 + 750X + 125 - X \cdot j,$$

$$\Phi_7^c(X, j) = X^8 + 28X^7 + 322X^6 + 1904X^5 + 5915X^4 + 8624X^3 + 4018X^2 + 748X + 49 - X \cdot j,$$

$$\begin{aligned}\Phi_{13}^c(X, j) = {} & X^{14} + 26X^{13} + 325X^{12} + 2548X^{11} + 13832X^{10} \\ & + 54340X^9 + 157118X^8 + 333580X^7 + 509366X^6 \\ & + 534820X^5 + 354536X^4 + 124852X^3 \\ & + 15145X^2 + 746X + 13 - X \cdot j.\end{aligned}$$

How to use Φ_ℓ^c? Our main Results [dH+24]

Motivated by the modular theory in the background, we proved:

How to use Φ_ℓ^c? Our main Results [dH+24]

Motivated by the modular theory in the background, we proved:

Theorem

For $\ell \in \{2, 3, 5, 7, 13\}$ the edges $j_0 \to j_1$ in $G_\ell(p)$ correspond to the solutions of the system

$$\Phi_\ell^c(X, j_0) = 0 = \Phi_\ell^c(\ell^s/X, j_1)$$

where $s = 12/(\ell - 1)$.

How to use Φ_ℓ^c? Our main Results [dH+24]

Motivated by the modular theory in the background, we proved:

Theorem

For $\ell \in \{2, 3, 5, 7, 13\}$ the edges $j_0 \to j_1$ in $G_\ell(p)$ correspond to the solutions of the system

$$\Phi_\ell^c(X, j_0) = 0 = \Phi_\ell^c(\ell^s / X, j_1)$$

where $s = 12/(\ell - 1)$.

Theorem

If $j_0 \in \mathbb{F}_{p^2}$ is supersingular, then all roots of $\Phi_\ell^c(X, j_0)$ lie in $\mathbb{F}_{p^2}$.

How to use Φ_ℓ^c? Our main Results [dH+24]

Motivated by the modular theory in the background, we proved:

Theorem

For $\ell \in \{2, 3, 5, 7, 13\}$ the edges $j_0 \to j_1$ in $G_\ell(p)$ correspond to the solutions of the system

$$\Phi_\ell^c(X, j_0) = 0 = \Phi_\ell^c(\ell^s / X, j_1)$$

where $s = 12/(\ell - 1)$.

Theorem

If $j_0 \in \mathbb{F}_{p^2}$ is supersingular, then all roots of $\Phi_\ell^c(X, j_0)$ lie in $\mathbb{F}_{p^2}$.

Why? Isogenies between supersingular curves can be defined over $\mathbb{F}_{p^2}$!

Our Approach: Canonical Modular Polynomial [dH+24]

For each step $j_i \xrightarrow{f_i} j_{i+1}$ rephrase system

$$\Phi_\ell^c(f_i, j_i) = 0 = \Phi_\ell^c(\ell^s / f_i, j_i)$$

as an R1CS.

Our Approach: Canonical Modular Polynomial [dH+24]

For each step $j_i \xrightarrow{f_i} j_{i+1}$ rephrase system

$$\Phi_\ell^c(f_i, j_i) = 0 = \Phi_\ell^c(\ell^s / f_i, j_i) \cdot f_i^{\ell+1} / \ell^s$$

as an R1CS.

Our Approach: Canonical Modular Polynomial [dH+24]

For each step $j_i \xrightarrow{f_i} j_{i+1}$ rephrase system

$$\Phi_\ell^c(f_i, j_i) = 0 = \Phi_\ell^c(\ell^s / f_i, j_i) \cdot f_i^{\ell+1} / \ell^s$$

as an R1CS. For $\ell = 2$:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ f_i \\ f_i^2 \\ j_i - c_1' \\ j_{i+1} - c_1' \end{pmatrix} \bullet \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & c_2' & c_3' & -1 & 0 \\ 0 & c_3'' & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ f_i \\ f_i^2 \\ j_i - c_1' \\ j_{i+1} - c_1' \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ -c_0' & 0 & 0 & 0 & 0 \\ -c_0'' & -c_1'' & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ f_i \\ f_i^2 \\ j_i - c_1' \\ j_{i+1} - c_1' \end{pmatrix}$$

Some Stats [dH+24]

ℓ	Equations m		Variables n		Non-zero entries	
	[CLL23]	Ours	[CLL23]	Ours	[CLL23]	Ours
2	$4\lambda + 2$	3λ	$4\lambda + 3$	$3\lambda + 1$	$21\lambda + 6$	13λ
3		2.524λ		$2.524\lambda + 1$		11.357λ
5		2.584λ		$2.584\lambda + 1$		12.059λ
7		2.493λ		$2.493\lambda + 1$		12.467λ
13		2.702λ		$2.702\lambda + 1$		15.133λ

Table: Parameters of the R1CS where $\lambda = \log_2(\ell^k)$

Some Stats [dH+24]

ℓ	Equations m		Variables n		Non-zero entries	
	[CLL23]	Ours	[CLL23]	Ours	[CLL23]	Ours
2	$4\lambda + 2$	3λ	$4\lambda + 3$	$3\lambda + 1$	$21\lambda + 6$	13λ
3		2.524λ		$2.524\lambda + 1$		11.357λ
5		2.584λ		$2.584\lambda + 1$		12.059λ
7		2.493λ		$2.493\lambda + 1$		12.467λ
13		2.702λ		$2.702\lambda + 1$		15.133λ

Table: Parameters of the R1CS where $\lambda = \log_2(\ell^k)$

Some Stats [dH+24]

ℓ	Equations m		Variables n		Non-zero entries	
	[CLL23]	Ours	[CLL23]	Ours	[CLL23]	Ours
2	$4\lambda + 2$	3λ	$4\lambda + 3$	$3\lambda + 1$	$21\lambda + 6$	13λ
3		2.524λ		$2.524\lambda + 1$		11.357λ
5		2.584λ		$2.584\lambda + 1$		12.059λ
7		2.493λ		$2.493\lambda + 1$		12.467λ
13		2.702λ		$2.702\lambda + 1$		15.133λ

Table: Parameters of the R1CS where $\lambda = \log_2(\ell^k)$

Some Stats [dH+24]

ℓ	Equations m		Variables n		Non-zero entries	
	[CLL23]	Ours	[CLL23]	Ours	[CLL23]	Ours
2	$4\lambda + 2$	3λ	$4\lambda + 3$	$3\lambda + 1$	$21\lambda + 6$	13λ
3		2.524λ		$2.524\lambda + 1$		11.357λ
5		2.584λ		$2.584\lambda + 1$		12.059λ
7		2.493λ		$2.493\lambda + 1$		12.467λ
13		2.702λ		$2.702\lambda + 1$		15.133λ

Table: Parameters of the R1CS where $\lambda = \log_2(\ell^k)$

Field		Aurora		Ligero	
		[CLL23]	Ours	[CLL23]	Ours
$\mathbb{F}_{p^2}$	Prover time (ms)	934	669	587	420
	Verifier time (ms)	99	74	847	634
	Proof size (kB)	194	178	1849	1599

Table: Benchmarks for $\ell = 2$

Future Work (In Progress)

Can we use other modular polynomials to

- obtain smaller/more efficient proofs?

Future Work (In Progress)

Can we use other modular polynomials to

- obtain smaller/more efficient proofs?
- allow for a bigger range of primes ℓ?

Future Work (In Progress)

Can we use other modular polynomials to

- obtain smaller/more efficient proofs?
- allow for a bigger range of primes ℓ?

Yes! Atkin modular polynomials, Weber modular polynomials, …

Future Work (In Progress)

Can we use other modular polynomials to

- obtain smaller/more efficient proofs?
- allow for a bigger range of primes ℓ?

Yes! Atkin modular polynomials, Weber modular polynomials, …

- Other arithmetizations?

Thank you for your attention!

[dH+24] T. den Hollander, S. Kleine, M. Mula, D. Slamanig, and S. A. Spindler. *More Efficient Isogeny Proofs of Knowledge via Canonical Modular Polynomials*. Cryptology ePrint Archive, Paper 2024/1738. 2024. To appear at CRYPTO 2025.