Computing Rational Isogenies with Irrational Kernel Points

Gustavo Banegas[1], **Valerie Gilchrist**[2], Anaëlle Le Devehat[3], Benjamin Smith [3]

Qualcomm France SARL, Valbonne, France

Université Libre de Bruxelles and FNRS, Brussels, Belgium

Inria and Laboratoire d'Informatique de l'École polytechnique, Institut Polytechnique de Paris, Palaiseau, France

April 12, 2023

Separable isogenies—why do we care?

A *separable* isogeny can be defined uniquely (up to isomorphism) by its kernel.

With a cyclic kernel, this means we require only one point (the generator of the kernel) to define an isogeny.

Isogeny-based cryptosystems such as CSIDH and CRS evaluate points at separable isogenies.

Irrational kernels

We want to compute an isogeny of degree ℓ, whose domain is $E(\mathbb{F}_q)$.
Usually, we will look for a point of order ℓ to generate its kernel.

Suppose no point P of order ℓ is defined over $E(\mathbb{F}_q)$, so we go to $E(\mathbb{F}_{q^k})$ to find it.
We say $\langle P \rangle$ is *irrational*.

That is, the kernel of our isogeny will be defined over $\mathbb{F}_q$, but its elements will not be.

Vélu requires the kernel points, so this will be costly to run over an extension field.

Kernel polynomials

The *kernel polynomial* of an isogeny is used in Vélu's formulæ, given by

$$D(X) := \prod_{P \in S} (X - x(P))$$

where $S \subset \langle P \rangle$ is any subset such that

$$S \cap -S = \emptyset \qquad \text{and} \qquad S \cup -S = \langle P \rangle \setminus \{0\}\,.$$

Note, D will not always split over the base field, but its coefficients will always be defined over it.

Evaluating the kernel polynomial

Take $\ell = 13$ and $k = 3$.
We want to choose a set $S \subset \langle P \rangle$, that contains all multiples of P up to negation (excluding the identity) for use in our kernel polynomial.

Some classic examples are taking the first half or odd multiples.

Evaluating the kernel polynomial

Take $\ell = 13$ and $k = 3$.
We want to choose a set $S \subset \langle P \rangle$, that contains all multiples of P up to negation (excluding the identity) for use in our kernel polynomial.

Some classic examples are taking the first half or odd multiples.

$$P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P \quad 7P \quad 8P \quad 9P \quad 10P \quad 11P \quad 12P$$

Evaluating the kernel polynomial

Take $\ell = 13$ and $k = 3$.
We want to choose a set $S \subset \langle P \rangle$, that contains all multiples of P up to negation (excluding the identity) for use in our kernel polynomial.

Some classic examples are taking the first half or odd multiples.

$$\overbrace{P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P}^{P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P} \quad 7P \quad 8P \quad 9P \quad 10P \quad 11P \quad 12P$$

Evaluating the kernel polynomial

Take $\ell = 13$ and $k = 3$.
We want to choose a set $S \subset \langle P \rangle$, that contains all multiples of P up to negation (excluding the identity) for use in our kernel polynomial.

Some classic examples are taking the first half or odd multiples.

$$\overbrace{P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P}^{P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P}$$

$$\begin{array}{cccccccccccc}
P & 2P & 3P & 4P & 5P & 6P & 7P & 8P & 9P & 10P & 11P & 12P \\
\downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & \\
P & & 3P & & 5P & & 7P & & 9P & & 11P &
\end{array}$$

Evaluating the kernel polynomial

For example, choose

$$S = \{P, [2]P, [3]P, \ldots, [(\ell - 1)/2]P\}$$

This method would use one `xDBL` to get $[2]P$, and $(\ell - 1)/2 - 2 = (\ell - 5)/2$ `xADD`'s.

Evaluating the kernel polynomial

Another approach...

$$P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P \quad 7P \quad 8P \quad 9P \quad 10P \quad 11P \quad 12P$$

Evaluating the kernel polynomial

Another approach...

$$
\begin{tikzcd}
P \arrow[d] & 2P \arrow[d] & 3P \arrow[drr] & 4P \arrow[dl] & 5P \arrow[drrrrr] & 6P \arrow[d] & 7P \arrow[drrrrr] & 8P \arrow[dllll] & 9P \arrow[d] & 10P \arrow[dr] & 11P \arrow[dlll] & 12P \arrow[dlllll] \\
P & 2P & 4P & 8P & 3P & 6P & 12P & 11P & 9P & 5P & 10P & 7P
\end{tikzcd}
$$

Evaluating the kernel polynomial

Another approach...

$$P \quad 2P \quad 4P \quad 8P \quad 3P \quad 6P \quad 12P \quad 11P \quad 9P \quad 5P \quad 10P \quad 7P$$

Evaluating the kernel polynomial

Another approach...

$$P \xrightarrow{\times 2} 2P \xrightarrow{\times 2} 4P \xrightarrow{\times 2} 8P \xrightarrow{\times 2} 3P \xrightarrow{\times 2} 6P \xrightarrow{\times 2} 12P \xrightarrow{\times 2} 11P \xrightarrow{\times 2} 9P \xrightarrow{\times 2} 5P \xrightarrow{\times 2} 10P \xrightarrow{\times 2} 7P$$

Evaluating the kernel polynomial

Another approach...

$$\underbrace{P \xrightarrow{\times 2} 2P \xrightarrow{\times 2} 4P \xrightarrow{\times 2} 8P \xrightarrow{\times 2} 3P \xrightarrow{\times 2} 6P}_{} \xrightarrow{\times 2} 12P \xrightarrow{\times 2} 11P \xrightarrow{\times 2} 9P \xrightarrow{\times 2} 5P \xrightarrow{\times 2} 10P \xrightarrow{\times 2} 7P$$

This uses $(\ell - 3)/2$ `xDBL`'s.

Evaluating the kernel polynomial

We say n is a *primitive root* modulo ℓ if every a coprime to ℓ can be written as $a = n^i$, for some i.

We get the following lemma:

Lemma

If 2 is a primitive root modulo ℓ then

$$S = \{[2^i]P : 0 \leq i < (\ell - 1)/2\}.$$

This Lemma can be extended to the case where 2 is semi-primitive (has order $\frac{\ell-1}{2}$) with added constraints on ℓ.

How (semi)primitive is 2?

How often is 2 (semi)primitive modulo ℓ? For prime $3 \leq \ell < 10000$, roughly 57%.

	Primes $2 < \ell < 600$										
Yes	**3**	**5**	**7**	**11**	**13**	**19**	**23**	**29**	**37**	**47**	**53**
	59	**61**	**67**	**71**	**79**	**83**	**101**	**103**	**107**	**131**	**139**
	149	**163**	**167**	**173**	**179**	**181**	**191**	**197**	**199**	**211**	**227**
	239	**263**	**269**	**271**	**293**	**311**	**317**	**347**	**349**	**359**	**367**
	373	379	383	389	419	421	443	461	463	467	479
	487	491	503	509	523	541	547	557	563	**587**	599
No	**17**	**31**	**41**	**43**	**73**	**89**	**97**	**109**	**113**	**127**	**137**
	151	**157**	**193**	**223**	**229**	**233**	**241**	**251**	**257**	**277**	**281**
	283	**307**	**313**	**331**	**337**	**353**	397	401	409	431	433
	439	449	457	499	521	569	571	577	593		

Table: Primes in bold appear in the CSIDH-512 parameter set.

Evaluating the kernel polynomial

$$\text{Scalar multiplication :} \quad 1\texttt{xDBL} \quad + \tfrac{\ell-5}{2}\texttt{xADD}$$

$$\text{Doubling :} \quad \tfrac{\ell-3}{2}\texttt{xDBL}$$

Model	`xADD`	`xDBL`
Montgomery	$4M + 2S$	$2M + 2S + 1c$
Short Weierstrass (projective)	$11M + 5S$	$1M + 8S + 1c$

Table: Costs of `xADD` and `xDBL`. Here M and S represent multiplication and squaring, respectively, in $\mathbb{F}_{q^k}$, while c represents multiplication by a curve constant in $\mathbb{F}_q$

When $k = 1$, we get $M \approx c$, so we get a saving of around 16%.

Exploiting the action of Frobenius

P is a point of order ℓ defined over $E(\mathbb{F}_{q^k})$ (and not over any proper subfield containing $\mathbb{F}_q$).

$\langle P \rangle$ is Galois stable, so Frobenius acts as an eigenvalue, λ, on $\langle P \rangle$.

$$\begin{array}{ccccccccc} P & \mapsto & \pi(P) & \mapsto & \pi^2(P) & \mapsto & \cdots & \pi^{k-1}(P) \\ P & \mapsto & \lambda P & \mapsto & \lambda^2 P & \mapsto & \cdots & \lambda^{k-1} P \end{array}$$

This λ must be nonzero, and in fact it must be an element of order k in $(\mathbb{Z}/\ell\mathbb{Z})^\times$.

Exploiting the action of Frobenius

Take $\ell = 13$ and $k = 3$.
π must act as either [3] or [9] on $\langle P \rangle$: let's suppose it is [3]...

Exploiting the action of Frobenius

Take $\ell = 13$ and $k = 3$.
π must act as either $[3]$ or $[9]$ on $\langle P \rangle$: let's suppose it is $[3]$...

$$P \quad 2P \quad 3P \quad 4P \quad 5P \quad 6P \quad 7P \quad 8P \quad 9P \quad 10P \quad 11P \quad 12P$$

Exploiting the action of Frobenius

Take $\ell = 13$ and $k = 3$.
π must act as either [3] or [9] on $\langle P \rangle$: let's suppose it is [3]...

$$\begin{tikzcd}
P \arrow[d] & 2P \arrow[drr] & 3P \arrow[dl] & 4P \arrow[drrr] & 5P \arrow[dr] & 6P \arrow[dl] & 7P \arrow[drrr] & 8P \arrow[drrr] & 9P \arrow[dllllll] & 10P \arrow[dl] & 11P \arrow[dr] & 12P \arrow[dllll] \\
P & 3P & 9P & 2P & 6P & 5P & 4P & 12P & 10P & 8P & 11P & 7P
\end{tikzcd}$$

Exploiting the action of Frobenius

Take $\ell = 13$ and $k = 3$.
π must act as either [3] or [9] on $\langle P \rangle$: let's suppose it is [3]...

$$P \quad 3P \quad 9P \quad 2P \quad 6P \quad 5P \quad 4P \quad 12P \quad 10P \quad 8P \quad 11P \quad 7P$$

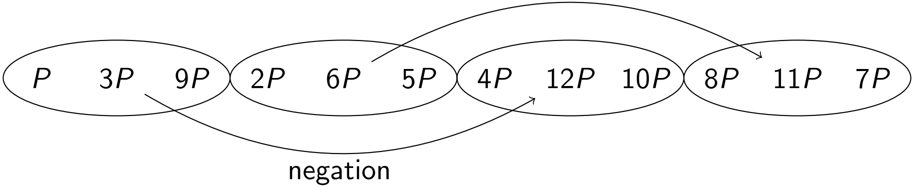
Exploiting the action of Frobenius

Take $\ell = 13$ and $k = 3$.
π must act as either [3] or [9] on $\langle P \rangle$: let's suppose it is [3]...

$$P \quad 3P \quad 9P \qquad 2P \quad 6P \quad 5P \qquad 4P \quad 12P \quad 10P \qquad 8P \quad 11P \quad 7P$$

Exploiting the action of Frobenius

Take $\ell = 13$ and $k = 3$.
π must act as either [3] or [9] on $\langle P \rangle$: let's suppose it is [3]...



Exploiting the action of Frobenius

So we can write

$$S = S_0 \sqcup \pi(S_0) \sqcup \cdots \sqcup \pi^{k'-1}(S_0)$$

where S_0 is the set of points that generate the galois orbits (up to negation) and

$$k' := \begin{cases} k & \text{if } k \text{ is odd}, \\ k/2 & \text{if } k \text{ is even}. \end{cases}$$

Lemma

If 2 is primitive modulo ℓ (or some other conditions) then $S_0 = \{[2^i]P : 0 \leq i < (\ell - 1)/2k'\}$.

From before, let $\ell = 13, k = 3$.
This gives us $S_0 = \{[2^i]P : 0 \leq i < 12/6\} = \{P, 2P\}$.

Evaluating the kernel polynomial with S_0

$$D(X) = \prod_{P\in S}(X - x(P)) = \prod_{P\in S_0}\prod_{i=0}^{k'-1}(X - x(\pi^i(P)))$$

$$= \prod_{P\in S_0}\prod_{i=0}^{k'-1}(X - x(P)^{q^i}),$$

Transposing the order of the products, if we let

$$D_0(X) := \prod_{P\in S_0}(X - x(P))$$

then

$$D(\alpha) = (D_0(\alpha))^{1+2+\cdots(k'-1)} \quad \text{for all } \alpha \in \mathbb{F}_q .$$

Some preliminary results

We use an algorithm due to Costello and Hisil (2017) as a base for comparison.

They derive the following affine formula

$$f(x) = x \cdot \left(\prod_{i=1}^{(\ell-1)/2} \left(\frac{x \cdot x_{[i]P} - 1}{x - x_{[i]P}} \right) \right)^2$$

which results in this projective formula

$$X' = X \cdot \left(\prod_{i=1}^{(\ell-1)/2} (X \cdot X_i - Z_i \cdot Z) \right)^2, Z' = Z \cdot \left(\prod_{i=1}^{(\ell-1)/2} (X \cdot Z_i - X_i \cdot Z) \right)^2$$

This work was later generalized to the even case by Renes (2018).

Some preliminary results

We compare the algorithms using operation counts over the extension field.

Some examples **when 2 is primitive**...

ℓ	k		multiplies	adds	squarings
7	3	Costello-Hisil	20	17	7
		Galois orbits	2	4	8
29	7	Costello-Hisil	108	105	29
		Galois orbits	8	12	48
53	13	Costello-Hisil	204	201	53
		Galois orbits	8	18	107

Computing the kernel generator

We want to find a point $P \in E(\mathbb{F}_{q^k})$ of order ℓ.

One approach would be:

1. sample a random $P \in E(\mathbb{F}_{q^k})$
2. compute $P_\ell = [\#E(\mathbb{F}_{q^k})/\ell]P$
3. compute $\mathrm{Order}(P_\ell)$, which is either 0 or ℓ

Computing the kernel generator

Take $k = 3$.

$$E(\mathbb{F}_{q^3}) \cong E(\mathbb{F}_q) \oplus H_3$$

Note, π^3 fixes all points in $E(\mathbb{F}_{q^3})$, and π fixes all the points in $E(\mathbb{F}_q)$.
That is, $E(\mathbb{F}_{q^i}) = \ker(\pi^i - [1])$.

So we proceed by:

1. sample $P \in E(\mathbb{F}_{q^k})$
2. compute $P_H = (\pi - 1)P$
3. compute $P_\ell = [\#H_3/\ell]P_H$
4. check the order of P_ℓ

This saves us around $1/3$ of the multiplications.

Computing the kernel generator

Take $k = 6$.

$$\begin{tikzcd}[row sep=small] & E(\mathbb{F}_{q^2}) \arrow[dr, phantom, "\subset" sloped] & \\ E(\mathbb{F}_q) \arrow[ur, phantom, "\subset" sloped] \arrow[dr, phantom, "\subset" sloped] & & E(\mathbb{F}_{q^6}) \\ & E(\mathbb{F}_{q^3}) \arrow[ur, phantom, "\subset" sloped] & \end{tikzcd}$$

$$E(\mathbb{F}_{q^2}) = \ker(\pi^2 - [1])$$
$$E(\mathbb{F}_{q^3}) = \ker(\pi^3 - [1])$$

$$\delta_6 := (\pi + [1])(\pi^3 - [1])$$

As before, we do the following:

1. sample $P \in E(\mathbb{F}_{q^k})$
2. compute $P_H = \delta_6 P$
3. compute $P_\ell = [\#H_6/\ell]P_H$
4. check the order of P_ℓ

This saves around $2/3$ of the multiplications.

Computing the kernel generator

Formalizing...

$$\eta_k := \Phi_k(\pi) \in \mathrm{End}(E)$$

where $\Phi_k(X)$ is the k-th cyclotomic polynomial (that is, the minimal polynomial over $\mathbb{Z}$ of the primitive k-th roots of unity in $\bar{\mathbb{Q}}$).

So we define

$$\delta_k := (\pi^k - [1])/\eta_k \in \mathrm{End}(E)$$

and get that $\delta_k(E(\mathbb{F}_{q^k})) \subset H_k$.

Computing the kernel generator

Lemma

If k is even, then every point P in H_k has $x(P)$ in $\mathbb{F}_{q^{k/2}}$.

In the case $k = 2$, this is known as the "twist trick".

Computing the kernel generator

k	$\#H_k$	δ_k
1	1	
2	$q + O(\sqrt{q})$	$\pi - [1]$
3	$q^2 + O(q^{3/2})$	$\pi - [1]$
4	$q^2 + O(q)$	$\pi^2 - [1]$
5	$q^4 + O(q^{7/2})$	$\pi - [1]$
6	$q^2 + O(q^{3/2})$	$(\pi + [1])(\pi^3 - [1])$
7	$q^6 + O(q^{7/2})$	$\pi - [1]$
8	$q^4 + O(q^2)$	$\pi^4 - [1]$
9	$q^6 + O(q^{4/2})$	$\pi^3 - [1]$
10	$q^4 + O(q^{7/2})$	$(\pi + [1])(\pi^5 - [1])$
11	$q^{10} + O(q^{19/2})$	$\pi - [1]$
12	$q^4 + O(q^3)$	$(\pi^2 + [1])(\pi^6 - [1])$

In summary we...

Evaluated kernel polynomials

- doubling trick (and its cost)
- exploiting the action of frobenius

Computed the kernel generator

eprint coming soon...