

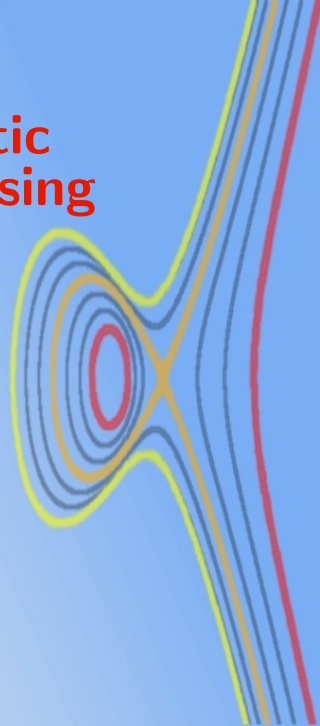
Quasi-quadratic elliptic curve point counting using rigid cohomology

Hendrik Hubrechts

Katholieke Universiteit Leuven

MEGA 2007 - Strobl (Austria)

June 28, 2007



Outline



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

Motivation: cryptography

Elliptic curves and point counting

Rigid cohomology and Kedlaya's algorithm

A new quasi-quadratic algorithm

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

Outline



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

Motivation: cryptography

Motivation:
cryptography

Elliptic curves and point counting

Elliptic curves and
point counting

Rigid cohomology and Kedlaya's algorithm

Rigid cohomology
and Kedlaya's
algorithm

A new quasi-quadratic algorithm

A new
quasi-quadratic
algorithm

Outline



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

Motivation: cryptography

Motivation:
cryptography

Elliptic curves and point counting

Elliptic curves and
point counting

Rigid cohomology and Kedlaya's algorithm

Rigid cohomology
and Kedlaya's
algorithm

A new quasi-quadratic algorithm

A new
quasi-quadratic
algorithm

Outline



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

Motivation: cryptography

Motivation:
cryptography

Elliptic curves and point counting

Elliptic curves and
point counting

Rigid cohomology and Kedlaya's algorithm

Rigid cohomology
and Kedlaya's
algorithm

A new quasi-quadratic algorithm

A new
quasi-quadratic
algorithm

Public key cryptography – one way functions



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ **Cryptography**: studies methods for secure communication
- ▶ **Public Key Cryptography**: secure communication over an unsecure channel

Idea: use **public** information to **encode**,
private information to **decode**

- ▶ Basic component: “**(trapdoor) one way function**”

$$f : D \rightarrow E \quad (\text{suppose bijective})$$

where f is easy to compute, but hard to invert (without secret knowledge)

Examples of one way functions



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

1. **Product/factorization**: p, q large primes, compute

$$n := p \cdot q$$

Protocol: RSA, well-known and widely used
cryptosystem

2. **Shortest vector in a lattice**: NTRU cryptosystem

3. **Discrete exponentiation** (inverse: discrete logarithm)
(Cyclic) group $G = \langle g \rangle, \cdot$

$$\varphi : \frac{\mathbb{Z}}{(\#G)\mathbb{Z}} \rightarrow G : x \mapsto g^x$$

Proposed groups G : \mathbb{F}_q^\times , elliptic curves (EC's) over \mathbb{F}_q ,
Jacobians of hyperelliptic curves over \mathbb{F}_q

Security of these functions



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

1. **Factorization** of n : **subexponential** time in $\log n$
(number field sieve)
2. **Shortest vector in a lattice**: in general **exponential** time
in the dimension
3. **Discrete logarithm problem**:
 - ▶ \mathbb{F}_q^\times : **subexponential** time in $\log q$ (index calculus)
 - ▶ Elliptic curves (EC's), Jacobians of low genus
hyperelliptic curves over \mathbb{F}_q : in general **exponential** time
in $\log(\#G)$.
($q = p^n$ with p and n moderate: subexponential time
algorithm (C. Diem))
 - ▶ If $\#G$ is smooth: **subexponential** time, viz. polynomial
time in $\log(\#G) \times$ the largest prime factor of $\#G$

For this reason we need point counting algorithms!

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

Elliptic curves and zeta functions



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ An EC \bar{E}/\mathbb{F}_{p^n} is a smooth genus 1 curve over \mathbb{F}_{p^n} . It has an affine Weierstrass equation

$$Y^2 + \bar{a}_1 XY + \bar{a}_3 Y = X^3 + \bar{a}_2 X^2 + \bar{a}_4 X + \bar{a}_6,$$

and one point P_∞ at infinity.

- ▶ \bar{E} has a natural and computable group structure
- ▶ Define $N_k := \#\bar{E}/\mathbb{F}_{(p^n)^k}$ and the Weil **zeta function** of \bar{E}

$$Z(\bar{E}/\mathbb{F}_{p^n}; T) := \exp\left(\sum_{k=1}^{\infty} \frac{N_k}{k} T^k\right) = \frac{p^n T^2 - tT + 1}{(1-T)(1-p^n T)},$$

where $t \in \mathbb{Z}$, $|t| \leq 2\sqrt{p^n}$ (Hasse-Weil bound)

- ▶ $N_1 = p^n + 1 - t$ is what we want to compute

A bound on t , the trace of Frobenius



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ We say that \bar{E} is supersingular if $t \equiv 0 \pmod{p}$, equivalently $N_1 \equiv 1 \pmod{p}$

- ▶ Such curves are rare, special and to avoid in cryptography

We always assume that \bar{E} is **not supersingular** (which is easy to verify)

- ▶ Goal: compute t , as $|t| < 2\sqrt{p^n}$ it suffices to compute

$$t \pmod{p^N} \quad \text{with} \quad p^N \geq 4\sqrt{p^n} \quad \Rightarrow \quad N := \lceil n/2 + \log_p 4 \rceil$$

Note that we can assume $N \leq n$, so that $t \pmod{p^N}$ certainly suffices

A few known point counting methods



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

We work with a curve \bar{E} over \mathbb{F}_{p^n}

1. Schoof '85 (SEA): time $\tilde{O}((\log p^n)^4)$
(Still the best for large p)

Idea: compute t modulo some small primes $\ell \neq p$, by considering the ℓ -torsion of the curve

2. Satoh '99, ..., Harley '02: $\tilde{O}(n^2)$
(p -adic \Rightarrow fixed small p)

Idea: compute canonical lift of \bar{E} and determine action of Frobenius on it

3. Rigid cohomology (started with Kedlaya '01): more details immediately
Originally: $\tilde{O}(n^3)$, now $\tilde{O}(n^2)$

Weil cohomology



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

Suggested by Weil in '49 (for proving the Weil conjectures):
Try to find a good cohomology for your variety (defined over \mathbb{F}_{p^n}) with a Frobenius morphism (" $x \mapsto x^{p^n}$ ") on it, then a Lefschetz fixed point formula for this operator gives you the zeta function

Given a curve $\bar{f}(X, Y) = 0$, the de Rham cohomology of $\mathbb{F}_{p^n}[X, Y]/\bar{f}(X, Y)$ does not work, e.g. all $X^{p^n-1}dX$ are non-exact.

Motivation:
cryptography

Elliptic curves and
point counting

**Rigid cohomology
and Kedlaya's
algorithm**

A new
quasi-quadratic
algorithm

p -Adic numbers and rigid lifts



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

$$\mathbb{Q}_p = \left\{ \sum_{i \geq J} a_i p^i \mid J \in \mathbb{Z}, a_i \in \{0, 1, \dots, p-1\} \right\}, \quad \mathbb{Z}_p : \sum_{i \geq 0}$$

\mathbb{Q}_{p^n} = unique unramified degree n extension of \mathbb{Q}_p ,

$$\text{then we have } \mathbb{Z}_{p^n}/p\mathbb{Z}_{p^n} \cong \mathbb{F}_{p^n} \quad (\star)$$

Frobenius automorphism $\sigma : \mathbb{Q}_{p^n} \rightarrow \mathbb{Q}_{p^n}$, lift of $x \mapsto x^p$

Given EC $\bar{E} : Y^2 + \bar{a}_1 XY + \bar{a}_3 Y = X^3 + \bar{a}_2 X^2 + \bar{a}_4 X + \bar{a}_6$
over \mathbb{F}_{p^n} , then we can take a **rigid lift** by

$$E : Y^2 + a_1 XY + a_3 Y = X^3 + a_2 X^2 + a_4 X + a_6,$$

where $a_i \in \mathbb{Z}_{p^n}$ and $(a_i \bmod p) \equiv \bar{a}_i$ using (\star)

Monksy-Washnitzer cohomology



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ p -Adic ring k : the ring of **overconvergent power series** is

$$k\langle X, Y \rangle^\dagger := \left\{ \sum_{i,j \geq 0} b_{ij} X^i Y^j \mid \liminf_{i+j \rightarrow \infty} \frac{\text{ord}_p(b_{ij})}{i+j} > 0 \right\},$$

i.e. all power series converging on a disk strictly bigger than the unit disk

- ▶ For a curve $\bar{E} : \bar{f}(X, Y) = 0$ with rigid lift E given by $f(X, Y) = 0$, we define the **dagger ring**

$$A^\dagger := \frac{\mathbb{Z}_{p^n}\langle X, Y \rangle^\dagger}{f(X, Y)}$$

- ▶ Monksy-Washnitzer cohomology: de Rham coh. of A^\dagger :

$$H_{MW}^1(\bar{E}/\mathbb{F}_{p^n}) := \frac{\Omega^1(A^\dagger)}{dA^\dagger} \otimes \mathbb{Q}_{p^n}$$

Point counting using the Frobenius operator



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hübrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

For simplicity we take EC $\bar{E} : Y^2 = \bar{Q}(X)$ with rigid lift $E : Y^2 = Q(X)$

- ▶ For algorithmic purposes Kedlaya removed the Weierstrass points:

$$E' := E \text{ minus } \{(x, y) \mid Q(x) = 0\}$$

- ▶ Resulting dagger ring:

$$A^\dagger = \mathbb{Z}_{p^n} \langle X, \sqrt{Q}, 1/\sqrt{Q} \rangle^\dagger$$

- ▶ On A^\dagger a p^n th power Frobenius morphism F_{p^n} exists, $H_{MW}^1(E')$ is a \mathbb{Q}_{p^n} -vector space of dimension 5

Let \mathcal{F}_{p^n} be a matrix of F_{p^n} , then

$$\text{Tr}(\mathcal{F}_{p^n}) = \#(\bar{E}/\mathbb{F}_{p^n}) - p^n - \#\{x \in \mathbb{F}_{p^n} \mid \bar{Q}(x) = 0\}$$

In practice: two adaptations



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ For convergence reasons: it is much better to work with F_p , the p th power Frobenius, here given by

$$X \mapsto X^p, \quad dX \mapsto d(X^p) = pX^{p-1}dX,$$

$$\sqrt{Q(X)} \mapsto Q(X)^{p/2} \cdot \left(1 - \frac{Q(X)^p - Q^\sigma(X^p)}{Q(X)^p}\right)^{1/2} \in A^\dagger$$

- ▶ With the hyperelliptic involution $\iota : X \mapsto X, \sqrt{Q} \mapsto -\sqrt{Q}$ we have

$$H_{MW}^1(E') \cong H_{MW}^+(E') \oplus H_{MW}^-(E'),$$

eigenspaces corresponding to the eigenvalues ± 1 of ι

- ▶ The \mathbb{Q}_{p^n} -vector space $H_{MW}^-(E')$ has dimension 2

Conclusion



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ We end up with $H_{MW}^-(E')$, a 2-dimensional \mathbb{Q}_{p^n} -vector space, with the operators

F_p as p th power Frobenius (matrix \mathcal{F}_p),

F_{p^n} as p^n th power Frobenius (matrix \mathcal{F}_{p^n}), and

$$\mathrm{Tr}(\mathcal{F}_{p^n}) = t \quad (\text{recall: } N_1 = p^n + 1 - t)$$

- ▶ As F_p is σ -linear we have (recall $\sigma : \mathbb{Q}_{p^n} \rightarrow \mathbb{Q}_{p^n}$, the lift of $x \mapsto x^p$)

$$\mathcal{F}_{p^n} = \sigma^{n-1}(\mathcal{F}_p) \cdot \sigma^{n-2}(\mathcal{F}_p) \cdots \sigma(\mathcal{F}_p) \cdot \mathcal{F}_p$$

Kedlaya's algorithm



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

(in fact for hyperelliptic curves in odd characteristic)

- ▶ Take a basis $(\{dX/\sqrt{Q^3}, XdX/\sqrt{Q^3}\})$ of $H_{MW}^-(E')$, compute $F_p(b) \in A^\dagger$ for b in the basis, reduce the result back to the basis and find \mathcal{F}_p
- ▶ Compute \mathcal{F}_{p^n} from \mathcal{F}_p via (roughly)

$$M_0 := \mathcal{F}_p, \quad M_{i+1} := \sigma^{2^i}(M_i) \cdot M_i$$

- ▶ Determine the zeta function from \mathcal{F}_{p^n}
- ▶ All computations have to be done modulo p^M for appropriate M
- Result: time $\tilde{O}(n^3)$, space $\mathcal{O}(n^3)$ (for fixed small p and fixed genus)

The slow steps in Kedlaya's algorithm



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

Two main steps in Kedlaya's algorithm:

1. **Computing \mathcal{F}_p** : computing $F_p(b)$ for b in the basis is very expensive $\Rightarrow \tilde{O}(n^3)$ time and $\mathcal{O}(n^3)$ space
2. **Computing \mathcal{F}_{p^n} from \mathcal{F}_p** : computing σ^k on \mathbb{Q}_{p^n} is expensive $\Rightarrow \tilde{O}(n^3)$ time

We will do both steps in time $\tilde{O}(n^2)$:

1. Using **deformation** (EC's: all curves in good families)
2. By **semi-diagonalising \mathcal{F}_p** (EC's: \mathcal{F}_p is 2-dimensional, but has in fact only one invariant)

Deformation: families of curves



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

Assume p is odd ($p = 2$: more complicated, similar results)

Choose family $\bar{E}_\Gamma : Y^2 = \bar{Q}_\Gamma(X)$ of (hyper)elliptic curves,

with $\bar{Q}_\Gamma(X) \in \mathbb{F}_p[X, \Gamma]$, and $\bar{\gamma} \in \mathbb{F}_{p^n}$ s.t. $\mathbb{F}_{p^n} = \mathbb{F}_p[\bar{\gamma}]$

In earlier work we have proven for the curve $\bar{E}_{\bar{\gamma}}$:

Theorem

We can compute $\mathcal{F}_p(\gamma)$ in time $\tilde{O}(n^2)$ and space $\mathcal{O}(n^2)$.

We will immediately explain how this works

Lemma

Given any EC \bar{E} over \mathbb{F}_{p^n} , we can find (efficiently) an equation of the form

$$Y^2 = X^3 + (\bar{a}_2 + \bar{b}_2\bar{\gamma})X^2 + \cdots + (\bar{a}_6 + \bar{b}_6\bar{\gamma}),$$

where $\bar{a}_i, \bar{b}_i \in \mathbb{F}_p$, $\bar{\gamma} \in \mathbb{F}_{p^n}$, for either \bar{E} or its quadratic twist.

Main point: we can always work with a family \bar{E}_Γ as above

Relative rigid cohomology



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

We construct $H_{MW}^-(E')$ as before, but now for the whole family \bar{E}_Γ at once

Concretely: $\bar{E}_\Gamma : Y^2 = \bar{Q}_\Gamma(X)$, rigid lift $E_\Gamma : Y^2 = Q_\Gamma(X)$, then

$$S^\dagger := \mathbb{Q}_p \left\langle \Gamma, \left(\text{Res}_X \left(Q_\Gamma, \frac{\partial}{\partial X} Q_\Gamma \right) \right)^{-1} \right\rangle^\dagger, \text{ the base ring,}$$

with the singular fibres removed from the family,

$$T^\dagger := \mathbb{Q}_p \left\langle X, \sqrt{Q_\Gamma}, 1/\sqrt{Q_\Gamma}, \Gamma, \left(\text{Res}_X \left(Q_\Gamma, \frac{\partial}{\partial X} Q_\Gamma \right) \right)^{-1} \right\rangle^\dagger,$$

$$H_{MW}^-(E'_\Gamma) \subset \frac{\Omega^1(T^\dagger)}{dT^\dagger},$$

which is an S^\dagger -module of rank 2

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

The connection and differential equation



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ Let $\mathcal{F}_p(\Gamma)$ be the matrix of the p th power Frobenius F_p on this module $H_{MW}^-(E'_\Gamma)$

- ▶ We define the **connection**

$$\nabla : T^\dagger \rightarrow T^\dagger d\Gamma : f \mapsto \frac{\partial f}{\partial \Gamma} d\Gamma,$$

with matrix $G(\Gamma)$ on $H_{MW}^-(E'_\Gamma)$

- ▶ From $\nabla \circ F_p = F_p \circ \nabla$ we can deduce

$$\frac{\partial}{\partial \Gamma} \mathcal{F}_p(\Gamma) + \mathcal{F}_p(\Gamma) \cdot G(\Gamma) = G(\Gamma^p) \cdot \mathcal{F}_p(\Gamma) d(\Gamma^p)$$

- ▶ With a few more ideas we can compute $\mathcal{F}_p(\Gamma)$ as power series up to sufficient precision p^N, Γ^M in time $\tilde{O}(n^2)$

Computing the small Frobenius



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

- ▶ As $\sigma(\Gamma) = \Gamma^p$, hence $\sigma(\gamma) = \gamma^p$, we need for $\gamma \in \mathbb{Q}_{p^n}$ the **Teichmüller lift** of $\bar{\gamma} \in \mathbb{F}_{p^n}$, i.e. the unique root of unity congruent to $\bar{\gamma} \pmod{p}$
- ▶ Let $\bar{\varphi}(x)$ be the minimal polynomial of $\bar{\gamma}$, then we take $\varphi(x)$ as the **Teichmüller modulus lift** of $\bar{\varphi}(x)$, i.e. the **minimal polynomial of the Teichmüller lift γ** , or equivalently $\varphi(x) \mid x^{p^n} - x$
- ▶ Conclusion (with some abuse of notation):

$$\mathbb{Q}_{p^n} = \frac{\mathbb{Q}_p[\gamma]}{\varphi(\gamma)}, \quad \text{and} \quad \mathcal{F}_p(\gamma) = (\mathcal{F}_p(\Gamma) \bmod \varphi(\Gamma)),$$

which can be computed in time $\tilde{O}(n^2)$

- ▶ An algorithm of Harley allows us to find $\varphi(x) \bmod p^N$ in time $\tilde{O}(n^2)$

Integral matrices



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

A slight complication: in the next step we need that $\mathcal{F}_p(\gamma)$ is integral, i.e. is defined over \mathbb{Z}_p^n

▶ p odd: basis $\left\{ \frac{dX}{\sqrt{Q_\Gamma}^3}, \frac{XdX}{\sqrt{Q_\Gamma}^3} \right\}$ suffices

▶ $p = 2$: we can compute (efficiently) a matrix of basis transformation such that the transformed $\mathcal{F}_p(\gamma)$ is integral

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

**A new
quasi-quadratic
algorithm**

The slow steps in Kedlaya's algorithm (again)

Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Two main steps in Kedlaya's algorithm:

1. Computing \mathcal{F}_p : computing $F_p(b)$ for b in the basis is very expensive $\Rightarrow \tilde{O}(n^3)$ time and $\mathcal{O}(n^3)$ space
2. **Computing \mathcal{F}_{p^n}** from \mathcal{F}_p : computing σ^k on \mathbb{Q}_{p^n} is expensive $\Rightarrow \tilde{O}(n^3)$ time

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

We will do both steps in time $\tilde{O}(n^2)$:

1. Using deformation (EC's: all curves in good families)
2. By **semi-diagonalising \mathcal{F}_p** (EC's: \mathcal{F}_p is 2-dimensional, but has in fact only one invariant)

An eigenvalue of the big Frobenius

The Weil conjectures imply for a nonsupersingular EC that

$$\det(\mathcal{F}_{p^n}(\gamma)) = p^n \quad \text{and} \quad \text{Tr}(\mathcal{F}_{p^n}(\gamma)) = t \not\equiv 0 \pmod{p}$$

and hence that $\mathcal{F}_{p^n}(\gamma)$ has p -adic eigenvalues

$$\lambda \text{ and } \frac{p^n}{\lambda}, \quad \text{with } \lambda \in \mathbb{Z}_p^\times$$

As we work mod p^N with $N \leq n$, we have

$$t = \text{Tr}(\mathcal{F}_{p^n}(\gamma)) = \lambda + \frac{p^n}{\lambda} \equiv \lambda \pmod{p^N},$$

so we have to compute

$$t \equiv \lambda \pmod{p^N}$$



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptology

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

From the p th to the p^n th power Frobenius



Recall: we need λ , the p -adic unit eigenvalue of $\mathcal{F}_{p^n}(\gamma)$

- ▶ Suppose we can solve the equation

$$\mathcal{F}_p(\gamma) \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \mu \cdot \begin{pmatrix} \sigma(x) \\ \sigma(y) \end{pmatrix}, \quad (*)$$

$x, y, \mu \in \mathbb{Z}_{p^n}$, μ a unit, then we find the factorization

$$\mathcal{F}_p(\gamma) = C^\sigma \cdot \begin{pmatrix} \mu & * \\ 0 & * \end{pmatrix} \cdot C^{-1}, \quad \text{with } C = \begin{pmatrix} x & * \\ y & * \end{pmatrix}$$

- ▶ This gives for the big Frobenius $\mathcal{F}_{p^n}(\gamma) =$

$$C^{[\sigma^n]} \cdot \begin{pmatrix} \mu & * \\ 0 & * \end{pmatrix}^{\sigma^{n-1}} \cdots \begin{pmatrix} \mu & * \\ 0 & * \end{pmatrix}^\sigma \cdot \begin{pmatrix} \mu & * \\ 0 & * \end{pmatrix} \cdot C^{-1}$$

hence also

$$\lambda = \mu^{\sigma^{n-1}} \cdots \mu^\sigma \cdot \mu = \mathcal{N}_{\mathbb{Q}_{p^n}/\mathbb{Q}_p}(\mu)$$

- ▶ By known fast generalized Newton lifting methods, (*) can be solved in time $\tilde{O}(n^2)$

Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hübrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

Computing the norm



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

We want to compute $t \equiv \mathcal{N}_{\mathbb{Q}_{p^n}/\mathbb{Q}_p}(\mu) \bmod p^N$

- ▶ (Idea of Harley) recall $\mathbb{Q}_{p^n} \cong \mathbb{Q}_p[x]/\varphi(x)$, then with $\mu(x) \in \mathbb{Q}_{p^n}$

$$\text{Res}_x(\varphi(x), \mu(x)) = \prod_{\alpha \text{ root of } \varphi} \mu(\alpha) =$$

$$\prod_{\tau \in \text{Gal}(\mathbb{Q}_{p^n}/\mathbb{Q}_p)} \mu(\tau(x)) = \mathcal{N}_{\mathbb{Q}_{p^n}/\mathbb{Q}_p}(\mu(x))$$

- ▶ This resultant can be computed in time $\tilde{O}(n^2)$ using an adaptation of Moenck's fast gcd algorithm

Overview of the algorithm



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

INPUT: EC \bar{E} over \mathbb{F}_{p^n}

OUTPUT: Number of points on \bar{E}

Let $N := \lceil \log_p 4 + n/2 \rceil$

1. Place \bar{E} in a good family over \mathbb{F}_p
2. Compute $\mathcal{F}_p(\gamma)$ by solving the differential equation
3. Compute a unit semi-eigenvalue μ of $\mathcal{F}_p(\gamma)$
4. Compute $t \equiv \mathcal{N}_{\mathbb{Q}_{p^N}/\mathbb{Q}_p}(\mu) \pmod{p^N}$ s.t. $|t| < 2\sqrt{p^n}$
5. Output $p^n + 1 - t$

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

A new
quasi-quadratic
algorithm

Theorem

We can compute the number of points on an elliptic curve \bar{E} over \mathbb{F}_{p^n} in time $\tilde{O}(n^2)$ and space $O(n^2)$.

Note: this is only relevant for fixed small p

Implementation results



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

**Hendrik
Hubrechts**

Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

**A new
quasi-quadratic
algorithm**

We do not use Moenck's algorithm for the norm computation, but Satoh, Skjernaa and Taguchi's method (which is far easier to implement)

For a random elliptic curve over \mathbb{F}_{p^n} , time in seconds (AMD Athlon 64 3000+):

$p \backslash n$	50	100	250	500	1000	2000
3	.18	.50	2.55	10.05	46	229
5	.58	1.38	6.48	27.08	117	610
7	2.16	5.51	34.13	156.21	800	4454

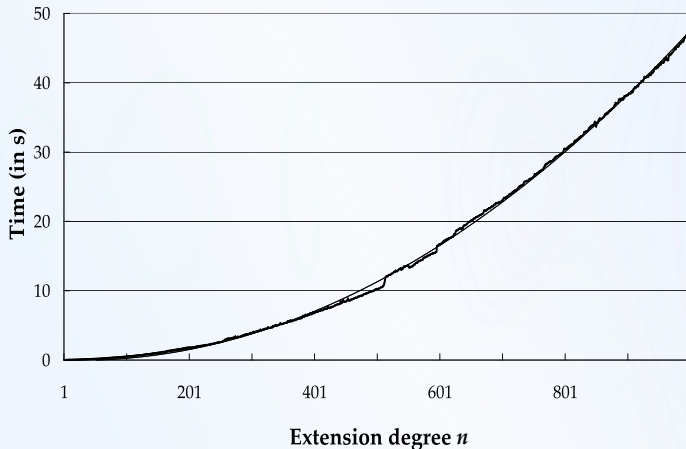
Implementation results



Quasi-quadratic
elliptic curve point
counting using
rigid cohomology

Hendrik
Hubrechts

For field sizes 3^n with $n = 1, \dots, 1000$:



Motivation:
cryptography

Elliptic curves and
point counting

Rigid cohomology
and Kedlaya's
algorithm

**A new
quasi-quadratic
algorithm**