Counting points on hyperelliptic curves in large characteristic: algorithms and complexity

Simon Abelard

Université de Lorraine

September 7, 2018

```
                      /* CARAMBA */          E,C,
                      /*         */         c,r,
                      /*         */        u,l,
                                          e,s,
                                         i=5,
   d[5],Q[999         ]={0};main(n      ){for
  (;i--;e=scanf("%"  "d",d+i));for(C    =*d;
 ++i<C       ;++Q[  i*i%        C],c=  i[Q]?
c:i);          for(;i --;)        for(u =C;u
--;n           +=!u*Q [l%C        ],e+= Q[(C
+l*l-         c*s*   s%C)         %C])  for(
  l=i,s=u,r=4;r;E=     i*l+c*u*s,s=(u*l  +i*s)
   %C,l=E%C+r           --[d]);printf     ("%d"
                                           "\n",
                                            (e+n*
                                              n)/2
     /* cc caramba.c; echo f3 f2 f1 f0 p | ./a.out */  -C);}
```

Point-counting 101

An example

How many solutions of $Y^2 = X^7 - 7X^5 + 14X^3 - 7X + 1$?

Point-counting 101

An example

How many solutions of $Y^2 = X^7 - 7X^5 + 14X^3 - 7X + 1$?

But what is a solution ? Where does it live?
Solutions in $\mathbb{Z}$: diophantine equations, undecidable.

Point-counting 101

An example

How many solutions of $Y^2 = X^7 - 7X^5 + 14X^3 - 7X + 1$?

But what is a solution ? Where does it live?
Solutions in $\mathbb{Z}$: diophantine equations, undecidable.

Our problem

Count solutions of $f(X, Y) = 0$ in a finite field $\mathbb{F}_{p^n}$.

Point-counting 101

An example

How many solutions of $Y^2 = X^7 - 7X^5 + 14X^3 - 7X + 1$?

But what is a solution ? Where does it live?
Solutions in $\mathbb{Z}$: diophantine equations, undecidable.

Our problem

Count solutions of $f(X, Y) = 0$ in a finite field $\mathbb{F}_{p^n}$.

Naive approach: try all possibilities for $(x, y) \in \mathbb{F}_{p^n}^2$.
When p large (hundreds of bits), not the best idea.

Complexity of point-counting

Parameters of the problem

Equation $Y^2 = f(X)$ with f polynomial over $\mathbb{F}_{p^n}$.
Input size: $\deg f \times n \log p$.
Question: dependency on n, p and $\deg f$?
Holy grail: polynomial-time algorithm in input size.

Naive approach exponential in all.

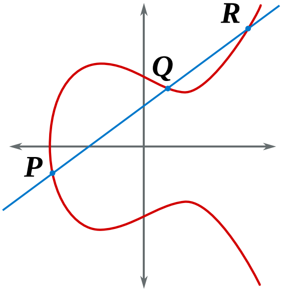
Partly polynomial-time approaches

We will see algorithms polynomial either $n \log p$ or in $\deg f$.
No classical algorithm polynomial (yet) in all (quantum by [Kedlaya'05]).
When fixed f and many p's, polynomial *on average* [Harvey'14].

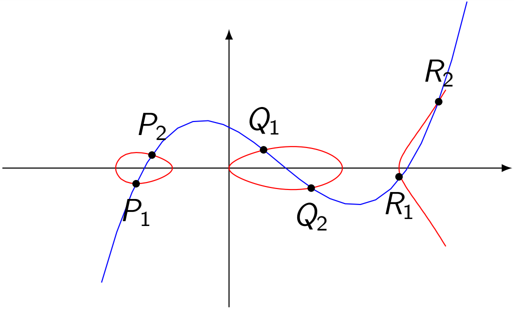
Our favorite geometrical object

The case of hyperelliptic curves

Count solutions of $Y^2 = f(X)$ with $f \in \mathbb{F}_q[X]$ monic squarefree.
Assume $\deg f = 2g + 1$, call g the genus of the curve.
Equation of hyperelliptic curve $\mathcal{C}$, solutions are points on $\mathcal{C}$.



$$\boldsymbol{P + Q + R = 0}$$



$$P_1 + P_2 + Q_1 + Q_2 + R_1 + R_2 = 0$$

Curve of equation $Y^2 = X^5 - 2X^4 - 7X^3 + 8X^2 + 12X$

Point counting II

Let $\mathcal{C}$ be a hyperelliptic curve of genus g.

Weil conjectures to the rescue

Point counting over $\mathbb{F}_q$ is computing the local ζ function of $\mathcal{C}$:

$$\zeta(s) = \exp\left(\sum_k \#\mathcal{C}(\mathbb{F}_{q^k})\frac{s^k}{k}\right) \overset{thm}{=} \frac{\Lambda(s)}{(1-s)(1-qs)}.$$

Where polynomial Λ has degree $2g$ and integer coefficients.

Point counting II

Let $\mathcal{C}$ be a hyperelliptic curve of genus g.

Weil conjectures to the rescue

Point counting over $\mathbb{F}_q$ is computing the local ζ function of $\mathcal{C}$:

$$\zeta(s) = \exp\left(\sum_k \#\mathcal{C}(\mathbb{F}_{q^k})\frac{s^k}{k}\right) \overset{thm}{=} \frac{\Lambda(s)}{(1-s)(1-qs)}.$$

Where polynomial Λ has degree $2g$ and integer coefficients.

Point counting

Input: $f \in \mathbb{F}_q[X]$ defining a hyperelliptic curve $Y^2 = f(X)$.
Output: the polynomial Λ.

Point counting II

Let $\mathcal{C}$ be a hyperelliptic curve of genus g.

Weil conjectures to the rescue

Point counting over $\mathbb{F}_q$ is computing the local ζ function of $\mathcal{C}$:

$$\zeta(s) = \exp\left(\sum_k \#\mathcal{C}(\mathbb{F}_{q^k})\frac{s^k}{k}\right) \overset{thm}{=} \frac{\Lambda(s)}{(1-s)(1-qs)}.$$

Where polynomial Λ has degree $2g$ and integer coefficients.

Point counting

Input: $f \in \mathbb{F}_q[X]$ defining a hyperelliptic curve $Y^2 = f(X)$.
Output: the polynomial Λ.

Example $\mathcal{C} : Y^2 = X^7 - 7X^5 + 14X^3 - 7X + 1$ defined over $\mathbb{F}_{23}$.
The associated Λ is $12167X^6 - 198X^3 + 1$.

A first application

Why counting points?

Cryptographic purposes (genus ≤ 2)

Curves provide groups with no known subexponential algorithm for DLP. Size of group determines security level [*Pohlig-Hellman'78*].

In other algorithms

Primality proving with proven complexity [*Adleman-Huang'01*].
Deterministic factorization in $\mathbb{F}_q[X]$? (ongoing [*Kayal'06, Poonen'17*])

Arithmetic geometry

Conjectures in number theory e.g. Sato-Tate in genus ≥ 2.
L-functions associated: $L(s, \mathcal{C}) = \sum_p A_p/p^s$ with $A_p = \#\mathcal{C}(\mathbb{F}_p)/\sqrt{p}$.
Computing them relies on point-counting primitives.

Algorithms for point counting

Let $\mathcal{C}$ be a curve over $\mathbb{F}_q$ with $q = p^n$.

p-adic methods

- elliptic curves: *Satoh'99, Mestre'00*
- hyp. curves: *Kedlaya'01, Denef-Vercauteren'06, Lauder-Wan'06*
- more general curves: *Castryck-Denef-Vercauteren'06, Tuitman'17*

Asymptotic complexity: polynomial in g and n, exponential in $\log p$.

ℓ-adic methods

Elliptic curves (*Schoof'85*) extended to Abelian varieties (*Pila'90*).
Asymptotic complexity: polynomial in $\log p$ and n, exponential in g.

Schoof's algorithm in genus ≤ 2

[Pila'90] is polynomial but with 23-bit exponent for $\log q$ when $g = 2$.

Asymptotic complexities

Genus	Complexity	Authors
$g = 1$	$\widetilde{O}(\log^4 q)$	Schoof-Elkies-Atkin ($\sim$ 1990)
$g = 2$	$\widetilde{O}(\log^8 q)$	Gaudry-Harley-Schost (2000)
$g = 2$ with RM	$\widetilde{O}(\log^5 q)$	Gaudry-Kohel-Smith (2011)

RM: real multiplication

Schoof's algorithm in genus ≤ 2

[Pila'90] is polynomial but with 23-bit exponent for $\log q$ when $g = 2$.

Asymptotic complexities

Genus	Complexity	Authors
$g = 1$	$\widetilde{O}(\log^4 q)$	Schoof-Elkies-Atkin ($\sim$ 1990)
$g = 2$	$\widetilde{O}(\log^8 q)$	Gaudry-Harley-Schost (2000)
$g = 2$ with RM	$\widetilde{O}(\log^5 q)$	Gaudry-Kohel-Smith (2011)

RM: real multiplication

Practical results

In genus 1, SEA record with p a 16645-bit prime (*Sutherland'10*).
In genus 2, heavy computations yield 256-bit cryptographic Jacobian.
In genus 2 with RM, can go up to 1024-bit Jacobians.

Schoof's algorithm in genus ≤ 2

[Pila'90] is polynomial but with 23-bit exponent for $\log q$ when $g = 2$.

Asymptotic complexities

Genus	Complexity	Authors
$g = 1$	$\widetilde{O}(\log^4 q)$	Schoof-Elkies-Atkin ($\sim$ 1990)
$g = 2$	$\widetilde{O}(\log^8 q)$	Gaudry-Harley-Schost (2000)
$g = 2$ with RM	$\widetilde{O}(\log^5 q)$	Gaudry-Kohel-Smith (2011)

RM: real multiplication

Practical results

In genus 1, SEA record with p a 16645-bit prime (*Sutherland'10*).
In genus 2, heavy computations yield 256-bit cryptographic Jacobian.
In genus 2 with RM, can go up to 1024-bit Jacobians.

What about genus 3?

Contributions: Schoof's algorithm in genus 3

Main results

For $\mathcal{C}$ a genus-3 hyperelliptic curve with explicit RM, we give a Las Vegas algorithm to compute Λ in $\widetilde{O}(\log^6 q)$ bit ops.
Without RM, the algorithm runs in $\widetilde{O}(\log^{14} q)$ bit ops.
Experiments: $g = 3$ and $p = 2^{64} - 59$, 192-bit RM-Jacobian.

Complexities

Genus	Complexity	Authors
$g = 1$	$\widetilde{O}(\log^4 q)$	Schoof-Elkies-Atkin
$g = 2$	$\widetilde{O}(\log^8 q)$	Gaudry-Schost
$g = 2$ with RM	$\widetilde{O}(\log^5 q)$	Gaudry-Kohel-Smith
$g = 3$	$\widetilde{O}(\log^{14} q)$	this thesis
$g = 3$ with RM	$\widetilde{O}(\log^6 q)$	this thesis

Contributions: asymptotic complexity in any genus

Asymptotic complexities

Authors (year)	Complexity	Context
Pila (1990)	$O\left((\log q)^{g^{O(g)}}\right)$	Abelian varieties
Huang-Ierardi (1998)	$O\left((\log q)^{g^{O(1)}}\right)$	Plane curves
Adleman-Huang (2001)	$O\left((\log q)^{g^{O(1)}}\right)$	Abelian varieties
Adleman-Huang (2001)	$O\left((\log q)^{O(g^2 \log g)}\right)$	Hyperelliptic curves
this thesis	$O_g\left((\log q)^{O(g)}\right)$	Hyperelliptic curves
this thesis	$\widetilde{O}_g\left((\log q)^8\right)$	with explicit RM

A prototype of Schoof's algorithm

Let $\mathcal{C} : y^2 = f(x)$ be a hyperelliptic curve over $\mathbb{F}_q$.
Let J be its Jacobian and g its genus.

1. (Hasse-Weil) bounds on coeffs of $\Lambda \Rightarrow$ compute $\Lambda \bmod \ell$
2. ℓ-torsion $J[\ell] = \{D \in J | \ell D = 0\} \simeq (\mathbb{Z}/\ell\mathbb{Z})^{2g}$
3. action on Frobenius $\pi : (x, y) \mapsto (x^q, y^q)$ on $J[\ell]$ yields $\Lambda \bmod \ell$

Algorithm *a la* Schoof

For sufficiently many primes ℓ

Describe I_ℓ the ideal of ℓ-torsion

Compute action of π on I_ℓ

Deduce $\Lambda \bmod \ell$

Recover Λ by CRT

A prototype of Schoof's algorithm

Let $\mathcal{C} : y^2 = f(x)$ be a hyperelliptic curve over $\mathbb{F}_q$.
Let J be its Jacobian and g its genus.

1. (Hasse-Weil) bounds on coeffs of $\Lambda \Rightarrow$ compute Λ mod ℓ
2. ℓ-torsion $J[\ell] = \{D \in J | \ell D = 0\} \simeq (\mathbb{Z}/\ell\mathbb{Z})^{2g}$
3. action on Frobenius $\pi : (x, y) \mapsto (x^q, y^q)$ on $J[\ell]$ yields Λ mod ℓ

Algorithm *a la* Schoof

For sufficiently many primes ℓ

  Describe I_ℓ the ideal of ℓ-torsion

  Compute action of π on I_ℓ

  Deduce Λ mod ℓ

Recover Λ by CRT

A prototype of Schoof's algorithm

Let $\mathcal{C} : y^2 = f(x)$ be a hyperelliptic curve over $\mathbb{F}_q$.
Let J be its Jacobian and g its genus.

1. (Hasse-Weil) bounds on coeffs of $\Lambda \Rightarrow$ compute $\Lambda \bmod \ell$
2. ℓ-torsion $J[\ell] = \{D \in J | \ell D = 0\} \simeq (\mathbb{Z}/\ell\mathbb{Z})^{2g}$
3. action on Frobenius $\pi : (x, y) \mapsto (x^q, y^q)$ on $J[\ell]$ yields $\Lambda \bmod \ell$

Algorithm *a la* Schoof

For sufficiently many primes ℓ

Describe I_ℓ the ideal of ℓ-torsion

Compute action of π on I_ℓ

Deduce $\Lambda \bmod \ell$

Recover Λ by CRT

Real multiplication

Explicit real multiplication

Famous endomorphisms: scalar multiplications and Frobenius π.
Ask for additional endomorphism η with explicit expression.
Then $\mathbb{Z}[\eta] \hookrightarrow \mathrm{End}(J)$ and we say $\mathcal{C}$ has RM by $\mathbb{Z}[\eta]$.
Real multiplication: $\mathbb{Z}[\eta]$ is in a totally real number field.

Real multiplication

Explicit real multiplication

Famous endomorphisms: scalar multiplications and Frobenius π.
Ask for additional endomorphism η with explicit expression.
Then $\mathbb{Z}[\eta] \hookrightarrow \mathrm{End}(J)$ and we say $\mathcal{C}$ has RM by $\mathbb{Z}[\eta]$.
Real multiplication: $\mathbb{Z}[\eta]$ is in a totally real number field.

An RM family (Mestre'91,Tautz-Top-Verberkmoes'91,Kohel-Smith'06)

Family $\mathcal{C}_t : y^2 = x^7 - 7x^5 + 14x^3 - 7x + t$ with $t \in \mathbb{F}_q$.
$\longrightarrow$ hyperelliptic curves of genus 3.

Real multiplication

Explicit real multiplication

Famous endomorphisms: scalar multiplications and Frobenius π.
Ask for additional endomorphism η with explicit expression.
Then $\mathbb{Z}[\eta] \hookrightarrow \mathrm{End}(J)$ and we say $\mathcal{C}$ has RM by $\mathbb{Z}[\eta]$.
Real multiplication: $\mathbb{Z}[\eta]$ is in a totally real number field.

An RM family (Mestre'91,Tautz-Top-Verberkmoes'91,Kohel-Smith'06)

Family $\mathcal{C}_t : y^2 = x^7 - 7x^5 + 14x^3 - 7x + t$ with $t \in \mathbb{F}_q$.
$\longrightarrow$ hyperelliptic curves of genus 3.

For $P = (x, y)$ generic point on $\mathcal{C}$, $\eta(P - \infty) = P_+ + P_- - 2\infty$ with

$$P_\pm = \left(-\tfrac{11}{4}x \pm \sqrt{\tfrac{105}{16}x^2 + \tfrac{16}{9}}, y\right).$$

Real multiplication

Explicit real multiplication

Famous endomorphisms: scalar multiplications and Frobenius π.
Ask for additional endomorphism η with explicit expression.
Then $\mathbb{Z}[\eta] \hookrightarrow \mathrm{End}(J)$ and we say $\mathcal{C}$ has RM by $\mathbb{Z}[\eta]$.
Real multiplication: $\mathbb{Z}[\eta]$ is in a totally real number field.

An RM family (Mestre'91,Tautz-Top-Verberkmoes'91,Kohel-Smith'06)

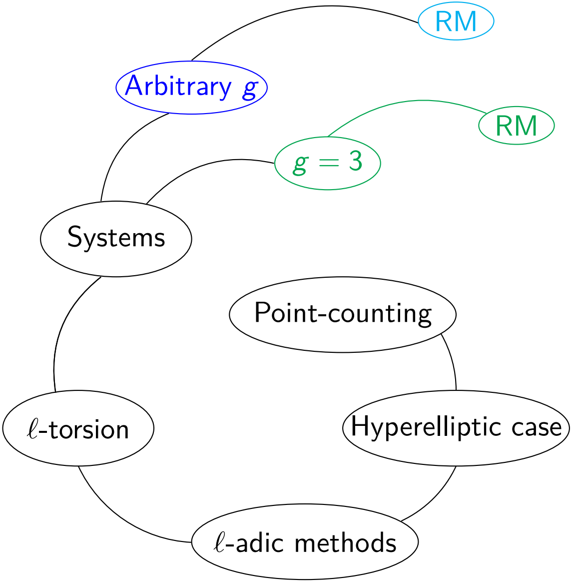
Family $\mathcal{C}_t : y^2 = x^7 - 7x^5 + 14x^3 - 7x + t$ with $t \in \mathbb{F}_q$.
$\longrightarrow$ hyperelliptic curves of genus 3.

For $P = (x, y)$ generic point on $\mathcal{C}$, $\eta(P - \infty) = P_+ + P_- - 2\infty$ with

$$P_\pm = \left(-\tfrac{11}{4}x \pm \sqrt{\tfrac{105}{16}x^2 + \tfrac{16}{9}}, y\right).$$

Element η has minimal polynomial $X^3 + X^2 - 2X - 1$.

Directions



With P. Gaudry and P.-J. Spaenlehauer, presented at ANTS 2018.

With P. Gaudry and P.-J. Spaenlehauer, to appear in FOCM journal.

Chapter VII of the manuscript, to be submitted.

A one-slide summary

Contributions

	hyperelliptic	with RM
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}(\log^{6} q)$
arbitrary g	$O_g((\log q)^{O(g)})$	$\widetilde{O}_g(\log^{8} q)$

A one-slide summary

Contributions

	hyperelliptic	with RM
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}(\log^{6} q)$
arbitrary g	$O_g((\log q)^{O(g)})$	$\widetilde{O}_g(\log^{8} q)$

All our results are based on 3 steps:

- modelling (subgroups of) the ℓ-torsion by polynomial systems
- bounding their sizes (number of variables, degrees)
- solving them (and bounding complexity)

A one-slide summary

Contributions

	hyperelliptic	with RM
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}(\log^{6} q)$
arbitrary g	$O_g((\log q)^{O(g)})$	$\widetilde{O}_g(\log^{8} q)$

All our results are based on 3 steps:

- modelling (subgroups of) the ℓ-torsion by polynomial systems
- bounding their sizes (number of variables, degrees)
- solving them (and bounding complexity)

Keys to each result

Genus 3: use RM to split the torsion $\Rightarrow$ decrease the degrees.
Genus g: different modelling, exploit multihomogeneity.
Genus g with RM: combine both approaches.

A one-slide summary

Contributions

	hyperelliptic	with RM
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}(\log^{6} q)$
arbitrary g	$O_g((\log q)^{O(g)})$	$\widetilde{O}_g(\log^{8} q)$

All our results are based on 3 steps:

- modelling (subgroups of) the ℓ-torsion by polynomial systems
- bounding their sizes (number of variables, degrees)
- solving them (and bounding complexity)

Keys to each result

Genus 3: use RM to split the torsion $\Rightarrow$ decrease the degrees.
Genus g: different modelling, exploit multihomogeneity.
Genus g with RM: combine both approaches.

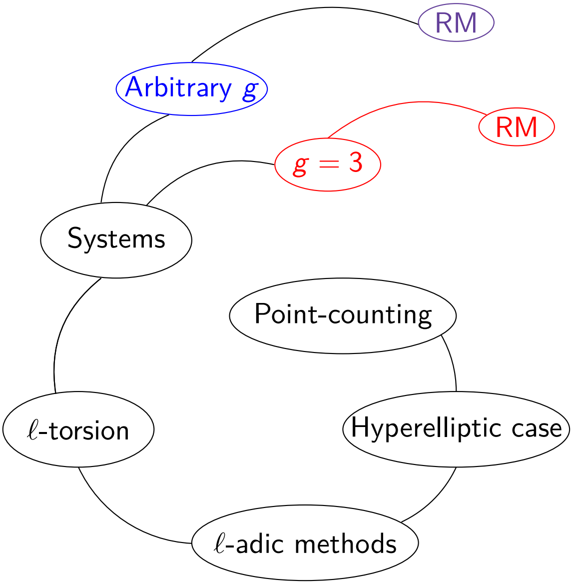
Plan

1 Introduction

2 Genus 3

3 Arbitrary g

4 RM in any genus



Counting points on genus-3 hyperelliptic curves

Contents

- Model the ℓ-torsion
- Use RM to split $J[\ell]$
- Model the 'parts' of $J[\ell]$
- Bound size of input systems
- Solve them with resultants
- Practical results

Modelling the ℓ-torsion

To model the ℓ-torsion, consider a divisor $D = \sum_{i=1}^{g}(P_i - \infty)$.
Compute $\ell D = \sum_{i=1}^{g} \ell(P_i - \infty)$ formally.
Then write a system equivalent to $\ell D = 0$ in J, and 'solve' it.

Modelling the ℓ-torsion

To model the ℓ-torsion, consider a divisor $D = \sum_{i=1}^{g}(P_i - \infty)$.
Compute $\ell D = \sum_{i=1}^{g} \ell(P_i - \infty)$ formally.
Then write a system equivalent to $\ell D = 0$ in J, and 'solve' it.

Bad news

In genus 3, the ideal $J[\ell]$ has degree ℓ^6.
Complexity bound: square of the degree, i.e. ℓ^{12} field ops.
$\Rightarrow$ Even $\ell = 5$ already seems out of reach…

Modelling the ℓ-torsion

To model the ℓ-torsion, consider a divisor $D = \sum_{i=1}^{g}(P_i - \infty)$.
Compute $\ell D = \sum_{i=1}^{g} \ell(P_i - \infty)$ formally.
Then write a system equivalent to $\ell D = 0$ in J, and 'solve' it.

Bad news

In genus 3, the ideal $J[\ell]$ has degree ℓ^6.
Complexity bound: square of the degree, i.e. ℓ^{12} field ops.
$\Rightarrow$ Even $\ell = 5$ already seems out of reach...

Wishful thinking

Can we split $J[\ell]$ into small (π-stable) subspaces?
For curves with explicit RM, it is possible.

Tuning Schoof's algorithm using RM

Let $\mathcal{C}$ be a genus-3 hyperelliptic curve with explicit RM by $\mathbb{Z}[\eta]$.

Splitting $J[\ell]$

For totally split ℓ, decompose $\ell = \mathfrak{p}_1\mathfrak{p}_2\mathfrak{p}_3$ in $\mathbb{Z}[\eta]$.

Find well-chosen ϵ_i in $\mathfrak{p}_i$ (i.e. of 'size' $\ell^{1/3}$).

The action of π on all the $\operatorname{Ker} \epsilon_i$ uniquely determines $\Lambda \bmod \ell$.

Advantage: model $\operatorname{Ker} \epsilon_i$ instead of $J[\ell]$, degree $O(\ell^2)$ vs ℓ^6.

Cantor's division polynomials (*Cantor'94*)

Problem

We have to compute ℓD or $\epsilon_i(D)$ to write our systems.
The ϵ_i are 'close to' multiplication by $\ell^{1/3} \Rightarrow$ scalar multiplication ?

Cantor's division polynomials (*Cantor'94*)

Problem

We have to compute ℓD or $\epsilon_i(D)$ to write our systems.
The ϵ_i are 'close to' multiplication by $\ell^{1/3}$ $\Rightarrow$ scalar multiplication ?

Answer: Cantor's *n*-division polynomials

For $n > g$ and $P = (x, y)$ a generic point on $\mathcal{C}$, $n(P - \infty)$ is described by $2g + 2$ univariate polynomials in x.

In genus 1 and 2, it is known that their degrees are in $O(n^2)$.

Cantor's division polynomials (*Cantor'94*)

Problem

We have to compute ℓD or $\epsilon_i(D)$ to write our systems.
The ϵ_i are 'close to' multiplication by $\ell^{1/3} \Rightarrow$ scalar multiplication ?

Answer: Cantor's n-division polynomials

For $n > g$ and $P = (x, y)$ a generic point on $\mathcal{C}$, $n(P - \infty)$ is described by $2g + 2$ univariate polynomials in x.

In genus 1 and 2, it is known that their degrees are in $O(n^2)$.

Quadratic bound (this thesis)

In genus 3, Cantor's n-division polynomials have degrees in $O(n^2)$.

Counting points on genus-3 hyperelliptic curves

Contents

- Model the ℓ-torsion
- Use RM to split $J[\ell]$
- Model the 'parts' of $J[\ell]$
- Bound size of input systems
- Solve them with resultants
- Practical results

Counting points on genus-3 hyperelliptic curves

Contents

- Model the ℓ-torsion
- Use RM to split $J[\ell]$
- Model the 'parts' of $J[\ell]$
- Bound size of input systems
- Solve them with resultants
- Practical results

Solving the systems, in theory

Successive elimination by resultants

System modelling kernel: trivariate with degrees bounded by some d.
Compute tri- then bi-variate resultants to put in triangular form.
Final complexity in $\widetilde{O}(d^6)$ field operations.

Solving the systems, in theory

Successive elimination by resultants

System modelling kernel: trivariate with degrees bounded by some d.
Compute tri- then bi-variate resultants to put in triangular form.
Final complexity in $\widetilde{O}(d^6)$ field operations.

Complexities

For ℓ inert, $d = O(\ell^2)$ and $J[\ell]$ is computed in $\widetilde{O}(\ell^{12})$ field ops.
For ℓ totally split, $d = O(\ell^{2/3})$ and cost decreased to $\widetilde{O}(\ell^4)$ field ops.
(The ϵ_i amount to multiplication by $\ell^{1/3}$)

Solving the systems, in theory

Successive elimination by resultants

System modelling kernel: trivariate with degrees bounded by some d.
Compute tri- then bi-variate resultants to put in triangular form.
Final complexity in $\widetilde{O}(d^6)$ field operations.

Complexities

For ℓ inert, $d = O(\ell^2)$ and $J[\ell]$ is computed in $\widetilde{O}(\ell^{12})$ field ops.
For ℓ totally split, $d = O(\ell^{2/3})$ and cost decreased to $\widetilde{O}(\ell^4)$ field ops.
(The ϵ_i amount to multiplication by $\ell^{1/3}$)

Overall complexities of $\widetilde{O}(\log^{14} q)$ in general and $\widetilde{O}(\log^6 q)$ with RM.

A practical example

$\mathcal{C} : y^2 = x^7 - 7x^5 + 14x^3 - 7x + 42$ over $\mathbb{F}_p$ with $p = 2^{64} - 59$.

Retrieving modular information

With general (non-RM related) techniques: Λ modulo $12 = 3 \times 4$.
Smallest totally-split prime: Λ modulo $\ell = 13$.

From theory to practice

Timing estimates for resultants

Evaluation/Interpolation: many not-so-small univariate resultants.

ℓ	Cost (NTL)	Cost (FLINT)
13	1,850 days	735 days
29	310,000 days	190,000 days

From theory to practice

Timing estimates for resultants

Evaluation/Interpolation: many not-so-small univariate resultants.

ℓ	Cost (NTL)	Cost (FLINT)
13	1,850 days	735 days
29	310,000 days	190,000 days

Successful attempt (F4, FGLM in Magma)

mod ℓ^k	#var	degree bounds	time	memory
2	—	—	—	—
4 (inert^2)	6	15	1 min	negl.
3 (inert)	5	55	14 days	140 GB
$13 = \mathfrak{p}_1\mathfrak{p}_2\mathfrak{p}_3$	5	52	3×3 days	41 GB

A practical example

$\mathcal{C} : y^2 = x^7 - 7x^5 + 14x^3 - 7x + 42$ over $\mathbb{F}_p$ with $p = 2^{64} - 59$.

Retrieving modular information

With general (non-RM related) techniques: Λ modulo $12 = 3 \times 4$.
Smallest totally-split prime: $\ell = 13$

We deduce Λ modulo $m = 156$, still far from sufficient...

A practical example

$\mathcal{C} : y^2 = x^7 - 7x^5 + 14x^3 - 7x + 42$ over $\mathbb{F}_p$ with $p = 2^{64} - 59$.

Retrieving modular information

With general (non-RM related) techniques: Λ modulo $12 = 3 \times 4$.
Smallest totally-split prime: $\ell = 13$

We deduce Λ modulo $m = 156$, still far from sufficient...

Finishing the computation

Action of π on J (not on $J[\ell]$), by collision search.
[Matsuo-Chao-Tsujii'02,Gaudry-Schost'04,Galbraith-Ruprai'09].
Main drawback: exponential complexity.
Advantages: memory efficient, massively run in parallel.
And a factor $156^{3/2} \simeq 1950$ speed-up via modular info.
In our experiments, it represents 105 CPU-days done in a few hours.

Summary of hyperelliptic genus-3 case

Complexities

	Genus 3 hyperelliptic	with RM
Object to model	ℓ-torsion $J[\ell]$	Ker ϵ_i where $\ell = \prod \epsilon_i$
Equation	$\ell D = 0$	$\epsilon_i(D) = 0$
Degrees	$O(\ell^2)$	$O(\ell^{2/3})$
Complexity	$\widetilde{O}\left((\log q)^{14}\right)$	$\widetilde{O}((\log q)^6)$

Summary of hyperelliptic genus-3 case

Complexities

	Genus 3 hyperelliptic	with RM
Object to model	ℓ-torsion $J[\ell]$	Ker ϵ_i where $\ell = \prod \epsilon_i$
Equation	$\ell D = 0$	$\epsilon_i(D) = 0$
Degrees	$O(\ell^2)$	$O(\ell^{2/3})$
Complexity	$\widetilde{O}\left((\log q)^{14}\right)$	$\widetilde{O}((\log q)^6)$

Experiments

We count points in a 192-bit hyperelliptic Jacobian with RM.
Previously: 183-bit by Sutherland (generic group methods).
Both are for particular cases, although RM is less likely.

Summary of hyperelliptic genus-3 case

Complexities

	Genus 3 hyperelliptic	with RM
Object to model	ℓ-torsion $J[\ell]$	$\operatorname{Ker}\epsilon_i$ where $\ell = \prod \epsilon_i$
Equation	$\ell D = 0$	$\epsilon_i(D) = 0$
Degrees	$O(\ell^2)$	$O(\ell^{2/3})$
Complexity	$\widetilde{O}\left((\log q)^{14}\right)$	$\widetilde{O}((\log q)^6)$

Experiments

We count points in a 192-bit hyperelliptic Jacobian with RM.
Previously: 183-bit by Sutherland (generic group methods).
Both are for particular cases, although RM is less likely.
$\longrightarrow$ genus-3 point-counting in large characteristic is challenging.

Perspective on Schoof's algorithm for $g \leq 3$

Villard's algorithm for bivariate resultant (ISSAC 2018)

Genus	Usual resultants	Villard's algorithm	With $\omega = 2.8$
$g = 2$	$\widetilde{O}(\log^8 q)$	$\widetilde{O}((\log q)^{8-2/\omega})$	$\widetilde{O}((\log q)^{7.3})$
$g = 2$ RM	$\widetilde{O}(\log^5 q)$	$\widetilde{O}((\log q)^{5-1/\omega})$	$\widetilde{O}((\log q)^{4.6})$
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}((\log q)^{14-4/\omega})$	$\widetilde{O}((\log q)^{12.6})$
$g = 3$ RM	$\widetilde{O}(\log^6 q)$	$\widetilde{O}((\log q)^{6-4/(3\omega)})$	$\widetilde{O}((\log q)^{5.5})$

Perspective on Schoof's algorithm for $g \leq 3$

Villard's algorithm for bivariate resultant (ISSAC 2018)

Genus	Usual resultants	Villard's algorithm	With $\omega = 2.8$
$g = 2$	$\widetilde{O}(\log^8 q)$	$\widetilde{O}((\log q)^{8-2/\omega})$	$\widetilde{O}((\log q)^{7.3})$
$g = 2$ RM	$\widetilde{O}(\log^5 q)$	$\widetilde{O}((\log q)^{5-1/\omega})$	$\widetilde{O}((\log q)^{4.6})$
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}((\log q)^{14-4/\omega})$	$\widetilde{O}((\log q)^{12.6})$
$g = 3$ RM	$\widetilde{O}(\log^6 q)$	$\widetilde{O}((\log q)^{6-4/(3\omega)})$	$\widetilde{O}((\log q)^{5.5})$

Further improvements

Extension of the SEA algorithm using modular polynomials.
Work of Milio and Martindale, in particular in RM case.
Still large objects (both degrees and coefficients).
Ongoing in genus 2, not tomorrow in genus 3.

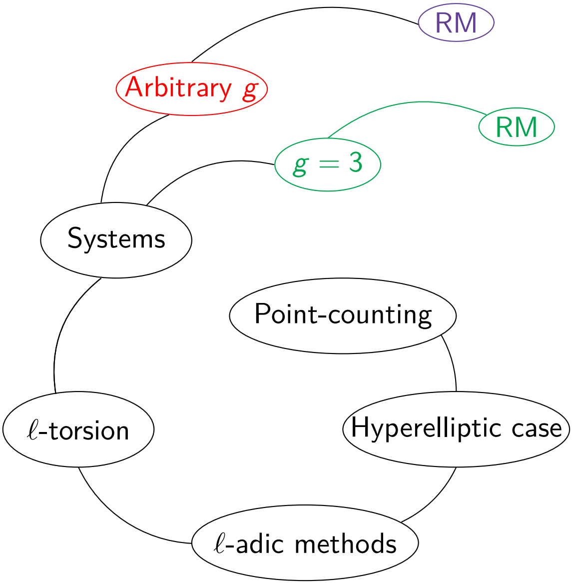
Plan

1 Introduction

2 Genus 3

3 Arbitrary g

4 RM in any genus



Hyperelliptic point-counting in any genus

Strategy

- Extend degree bounds for Cantor's polynomials
- New modelling for $J[\ell]$ with multihomogeneous structure
- Exploit multihomogeneity with geometric resolution

Complexity result

hyperelliptic	with RM
$\widetilde{O}(\log^{14} q)$	$\widetilde{O}(\log^{6} q)$
$O_g((\log q)^{O(g)})$	$\widetilde{O}_\eta(\log^{8} q)$

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- extend degree-bounds on Cantor's polynomials to any g

Cantor's division polynomials II

For $\ell > g$ and $P = (x, y)$ a generic point on $\mathcal{C}$,
Recall that $\ell(P - \infty)$ is given by Cantor's polynomials.

Cubic bound for any g (this thesis)

Cantor's ℓ-division polynomials have degrees in $O_g(\ell^3)$.

Cantor's division polynomials II

For $\ell > g$ and $P = (x, y)$ a generic point on $\mathcal{C}$,
Recall that $\ell(P - \infty)$ is given by Cantor's polynomials.

Cubic bound for any g (this thesis)

Cantor's ℓ-division polynomials have degrees in $O_g(\ell^3)$.

Conjecture: quadratic bound

Cantor proved two of the polynomials had degrees $g\ell^2 + O_g(1)$.
Experiments: the degrees of Cantor's polynomials are consecutive.

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- need to bound the degrees of Cantor's polynomials

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- need to bound the degrees of Cantor's polynomials
- degrees grow at each composition of $\ell(P_i - \infty) + \ell(P_j - \infty)$

Another look at the ℓ-torsion

Writing $\ell D = 0$

Still write $D = P_1 + \cdots + P_g - g\infty$ and compute $\ell(P_i - \infty)$.

Another look at the ℓ-torsion

Writing $\ell D = 0$

Still write $D = P_1 + \cdots + P_g - g\infty$ and compute $\ell(P_i - \infty)$.
Adding the $\ell(P_i - \infty)$ is avoided by different modelling.
But this introduces additional variables.

Another look at the ℓ-torsion

Writing $\ell D = 0$

Still write $D = P_1 + \cdots + P_g - g\infty$ and compute $\ell(P_i - \infty)$.
Adding the $\ell(P_i - \infty)$ is avoided by different modelling.
But this introduces additional variables.

Our polynomial system

Degrees are bounded by $O_g(\ell^3)$ (Cantor's polynomials).
About g^2 equations in g^2 variables $\Rightarrow$ Bézout bound in ℓ^{g^2}.
$\Rightarrow$ seems hard to improve previous bound in $(\log q)^{O(g^2)}\ldots$
But not all these variables appear with high degrees.

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- need to bound the degrees of Cantor's polynomials
- degrees grow at each composition of $\ell(P_i - \infty) + \ell(P_j - \infty)$

$\Rightarrow$ Different model, more variables but multihomogeneous structure.

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- need to bound the degrees of Cantor's polynomials
- degrees grow at each composition of $\ell(P_i - \infty) + \ell(P_j - \infty)$

$\Rightarrow$ Different model, more variables but multihomogeneous structure.

Solving the system

Resultants: exponential degree growth.

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- need to bound the degrees of Cantor's polynomials
- degrees grow at each composition of $\ell(P_i - \infty) + \ell(P_j - \infty)$

$\Rightarrow$ Different model, more variables but multihomogeneous structure.

Solving the system

Resultants: exponential degree growth.
Gröbner bases: unusable complexity bounds.

Outline

Modelling the ℓ-torsion

Write $\ell D = 0$ with $D = P_1 + \cdots + P_g - g\infty$.
Use Cantor's polynomials for $\ell(P_i - \infty)$ and add them.

- need to bound the degrees of Cantor's polynomials
- degrees grow at each composition of $\ell(P_i - \infty) + \ell(P_j - \infty)$

$\Rightarrow$ Different model, more variables but multihomogeneous structure.

Solving the system

Resultants: exponential degree growth.
Gröbner bases: unusable complexity bounds.
Geometric resolution: takes advantage of structure.

Multihomogeneity and complexity

g variables x_i
$O(g^2)$ equations
degree $O_g(\ell^3)$ in x_i

g variables y_i
$g^2 - g$ variables for φ
$O(g^2)$ equations
degrees in $O_g(1)$

Multihomogeneity and complexity

g variables x_i
$O(g^2)$ equations
degree $O_g(\ell^3)$ in x_i

g variables y_i
$g^2 - g$ variables for φ
$O(g^2)$ equations
degrees in $O_g(1)$

Geometric resolution (*Giusti-Lecerf-Salvy'01, Cafure-Matera'06*)

Assume $f_1, \cdots, f_n$ have degrees $\leq d$ and form a reduced regular sequence, and let $\delta = \max_i \deg\langle f_1, \ldots, f_i\rangle$. There is an algorithm computing a geometric resolution in time polynomial in δ, d, n.

Multihomogeneity and complexity

g variables x_i
$O(g^2)$ equations
degree $O_g(\ell^3)$ in x_i

g variables y_i
$g^2 - g$ variables for φ
$O(g^2)$ equations
degrees in $O_g(1)$

Geometric resolution
(*Giusti-Lecerf-Salvy'01, Cafure-Matera'06*)

Assume $f_1, \cdots, f_n$ have degrees $\leq d$ and form a reduced regular sequence, and let $\delta = \max_i \deg\langle f_1, \ldots, f_i \rangle$. There is an algorithm computing a geometric resolution in time polynomial in δ, d, n.

With $\delta = O_g\left(\ell^{3g}\right)$ bounded by multihomogeneous Bézout bound. Both $d = O_g(\ell^3)$ and $n = O_g(1)$ are harmless for our complexity result.

Overall complexity bound

Overall result

Model the ℓ-torsion with complexity $O_g(\ell^{O(g)})$.
Recall the largest ℓ is in $O_g(\log q)$.

$\Rightarrow$ **we compute the local zeta function in $O_g((\log q)^{O(g)})$.**

Overall complexity bound

Overall result

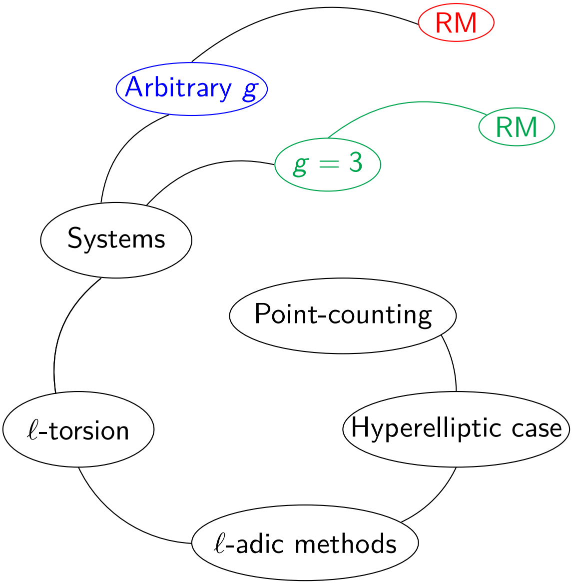
Model the ℓ-torsion with complexity $O_g(\ell^{O(g)})$.
Recall the largest ℓ is in $O_g(\log q)$.

$\Rightarrow$ **we compute the local zeta function in $O_g((\log q)^{O(g)})$.**

State of the art

	hyperelliptic case	plane curves	Abelian var
Adleman-Huang'01	$(\log q)^{O(g^2 \log g)}$	$(\log q)^{g^{O(1)}}$	$(\log q)^{g^{O(1)}}$
This thesis	$O_g\left((\log q)^{O(g)}\right)$	-	-

Plan



RM
Arbitrary g
RM
$g = 3$
Systems
Point-counting
ℓ-torsion
Hyperelliptic case
ℓ-adic methods

Hyperelliptic point-counting with RM in any genus

Contents

- Extend genus-3 case
- Use multihomogeneous modelling for $\operatorname{Ker} \epsilon_i$
- Dependency on g ?

Complexity result

	hyperelliptic	with RM
$g = 3$	$\widetilde{O}(\log^{14} q)$	$\widetilde{O}(\log^{6} q)$
any g	$O_g((\log q)^{cg})$	$\widetilde{O}_\eta(\log^{8} q)$

Explicit RM for arbitrary large g

RM families in any genus (Tautz-Top-Verberkmoes'91)

Consider curves with affine model $\mathcal{C}_{n,t} : Y^2 = D_n(X) + t$.
With t a parameter and D_n the n-th Dickson polynomial.
For $n = 2g + 1$, yields genus-g imaginary hyperelliptic curves.
Explicit expression for η is computable in $\widetilde{O}_\eta(\log q)$ (Kohel-Smith'06).

Explicit RM for arbitrary large g

RM families in any genus (Tautz-Top-Verberkmoes'91)

Consider curves with affine model $\mathcal{C}_{n,t} : Y^2 = D_n(X) + t$.
With t a parameter and D_n the n-th Dickson polynomial.
For $n = 2g + 1$, yields genus-g imaginary hyperelliptic curves.
Explicit expression for η is computable in $\widetilde{O}_\eta(\log q)$ (Kohel-Smith'06).

	Genus 3	Genus g with RM
Split ℓ	$\mathfrak{p}_1\mathfrak{p}_2\mathfrak{p}_3$	$\prod_{i=1}^{g} \mathfrak{p}_i$
Degree bounds	in $O(\ell^{1/3})$	in $O(\ell^{1/g})$

Explicit RM for arbitrary large g

RM families in any genus (Tautz-Top-Verberkmoes'91)

Consider curves with affine model $\mathcal{C}_{n,t} : Y^2 = D_n(X) + t$.
With t a parameter and D_n the n-th Dickson polynomial.
For $n = 2g + 1$, yields genus-g imaginary hyperelliptic curves.
Explicit expression for η is computable in $\widetilde{O}_\eta(\log q)$ (Kohel-Smith'06).

	Genus 3	Genus g with RM
Split ℓ	$\mathfrak{p}_1\mathfrak{p}_2\mathfrak{p}_3$	$\prod_{i=1}^{g} \mathfrak{p}_i$
Degree bounds	in $O(\ell^{1/3})$	in $O(\ell^{1/g})$

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\operatorname{Ker} \epsilon$ (degree ℓ^2)

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\operatorname{Ker}\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\mathrm{Ker}\,\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\operatorname{Ker}\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$
Variables	g with degree $O_g(\ell^3)$	g with degree $O_g(\ell^{3/g})$
	$O(g^2)$ with degree $O_g(1)$	$O(g^2)$ with degree $O_\eta(1)$

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\operatorname{Ker}\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$
Variables	g with degree $O_g(\ell^3)$	g with degree $O_g(\ell^{3/g})$
	$O(g^2)$ with degree $O_g(1)$	$O(g^2)$ with degree $O_\eta(1)$
Complexity	$O_g((\log q)^{O(g)})$	$\widetilde{O}_\eta(\log^8 q)$

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\operatorname{Ker}\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$
Variables	g with degree $O_g(\ell^3)$	g with degree $O_g(\ell^{3/g})$
	$O(g^2)$ with degree $O_g(1)$	$O(g^2)$ with degree $O_\eta(1)$
Complexity	$O_g((\log q)^{O(g)})$	$\widetilde{O}_\eta(\log^8 q)$

Remark: assuming quadratic degrees for Cantor's polynomials, we get a complexity in $\widetilde{O}_\eta(\log^6 q)$ similar to the case $g = 3$.

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\mathrm{Ker}\,\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$
Variables	g with degree $O_g(\ell^3)$	g with degree $O_g(\ell^{3/g})$
	$O(g^2)$ with degree $O_g(1)$	$O(g^2)$ with degree $O_\eta(1)$
Complexity	$O_g((\log q)^{O(g)})$	$\widetilde{O}_\eta(\log^8 q)$

Remark: assuming quadratic degrees for Cantor's polynomials, we get a complexity in $\widetilde{O}_\eta(\log^6 q)$ similar to the case $g = 3$.

Practical use? Smallest case: $g = 5$ and $\ell = 23$.

Modelling kernels of endomorphisms

	Genus g	with RM
Model	$J[\ell]$ (degree ℓ^{2g})	$\mathrm{Ker}\,\epsilon$ (degree ℓ^2)
Equations	$\ell D = 0$	$\epsilon(D) = 0$
Variables	g with degree $O_g(\ell^3)$	g with degree $O_g(\ell^{3/g})$
	$O(g^2)$ with degree $O_g(1)$	$O(g^2)$ with degree $O_\eta(1)$
Complexity	$O_g((\log q)^{O(g)})$	$\widetilde{O}_\eta(\log^8 q)$

Remark: assuming quadratic degrees for Cantor's polynomials, we get a complexity in $\widetilde{O}_\eta(\log^6 q)$ similar to the case $g = 3$.

Practical use? Smallest case: $g = 5$ and $\ell = 23$.

Warning: even the size of the system is exponential in g.

Summary of results

Three questions to address:

- modelling (subgroups of) the ℓ-torsion by polynomial systems
- bounding their sizes (number of variables, degrees)
- solving them (and bounding complexity)

Answers provided

- quadratic and cubic bounds for Cantor's polynomials
- multihomogeneous modelling for $J[\ell]$ (includes non-genericity)
- exploiting structure via geometric resolution
- when possible (RM) model subgroups of $J[\ell]$

Future work

Beyond the hyperelliptic case

Goal: explicit value for the $g^{O(1)}$, maybe even reach $O_g\left((\log q)^{O(g)}\right)$.
Main obstacle: need analogue of Cantor's polynomials.

Future work

Beyond the hyperelliptic case

Goal: explicit value for the $g^{O(1)}$, maybe even reach $O_g\left((\log q)^{O(g)}\right)$.
Main obstacle: need analogue of Cantor's polynomials.

Splitting the torsion without RM

Model kernels of ℓ-isogenies, as in SEA.
Fast evaluation of modular polynomials? ($g = 1$ in *Sutherland'12*)

Future work

Beyond the hyperelliptic case

Goal: explicit value for the $g^{O(1)}$, maybe even reach $O_g\left((\log q)^{O(g)}\right)$.
Main obstacle: need analogue of Cantor's polynomials.

Splitting the torsion without RM

Model kernels of ℓ-isogenies, as in SEA.
Fast evaluation of modular polynomials? ($g = 1$ in *Sutherland'12*)

Better handling non-genericity?

Elements of $J[\ell]$ of weight $< g$ and other pathological cases?
Problem: when these elements contain a proper subgroup of $J[\ell]$.
Can this happen for any curve or any ℓ? In what proportions?

Thanks for your attention

Credits: @fuzzberta