

Class Number Computation in Cubic Function Fields

Eric Landquist (UIUC)

Joint work with:

Renate Scheidler (U. of Calgary)

Andreas Stein (U. of Wyoming)

November 3, 2007

And what does this have to do with:

My wife's first car?

Australian marsupials?

The Nobel Peace Prize?

A talk show/podcast based in Columbia, MO?

Overview

- * Motivation
- * Background
- * Idea of the Algorithm
- * The Set-up: Zeta and L -functions
- * Estimating the Class Number
- * Optimization: Running Time
- * Optimization: Computations
- * Examples
- * Future Work and Open Problems
- * Questions?

Motivation

* “The determination of the structure of $Cl(K)$ and in particular of the class number $h(K)$ is one of the main problems in algorithmic algebraic number theory.” [pg. 208, Cohen]

* Compute the order of the Jacobian** of a cubic function field, K , over \mathbb{F}_q .

* An interesting and difficult computational problem.

* Cryptography:

- Group, efficient representation and arithmetic
- Group order with large prime divisor
- Efficient computation of the group order
- Security based on a “hard problem”

* Distribution of zeroes of the zeta function of K : Katz-Sarnak Heuristics.

** Not named after Jacoby Ellsbury, the Red Sox' rookie outfielder.

Background - Definitions and Notation

* $\mathbb{F}_q(x)$ is the field of rational functions over \mathbb{F}_q .

* Let $K = \mathbb{F}_q(C) = \mathbb{F}_q(x, y)$ be a separable cubic extension of genus g , where

$$C : y^3 - A(x)y + B(x) = 0 .$$

* $h = |\text{Pic}(K)|$ is the divisor class number.

$$(\sqrt{q} - 1)^{2g} < h < (\sqrt{q} + 1)^{2g}$$

* $h_K = |\text{Cl}(K)|$ is the (ideal) class number.

If $h_K = 1$, then \mathcal{O}_K is a PID** and hence a UFD.

* R is the regulator, a quantity based on the fundamental units of K .

* $fh = Rh_K$, where f is usually 1, sometimes 3.

* Three cases: unit rank 0, 1, and 2.

* In unit rank 0, $R = 1$; otherwise h_K is small.

** Not to be confused with PID Radio, which occasionally discusses UFOs.

Idea of the Algorithm*

1. Find: A good estimate E of h and a sharp upper bound U on the error.

$$E - U \leq h \leq E + U$$

The Hasse-Weil bounds are the default, but if $g \geq 3$, we can do better.

2. Find information about h , e.g. $h \equiv 1 \pmod{3}$.

3. Search the interval in time $\mathcal{O}(\sqrt{U})$ using:

– Shank's baby step-giant step method:
Deterministic, faster, storage: $\mathcal{O}(\sqrt{U})$

– Pollard's kangaroo** method:
Monte Carlo***, parallelizable, little storage

* Not what happened after the Nobel Peace Prize was announced.

** No kangaroos were harmed in the course of this research.

*** Not to be confused with Fritz, my wife's old '83 Chevy Monte Carlo.

The Set-up: Zeta and L -Functions

Let $u = q^{-s}$, $P \in \mathbb{F}_q[x]$ be a prime polynomial, and $\mathfrak{p} | (P)$. Then

$$\begin{aligned}
 \zeta_K(s) &= \prod_{\mathfrak{p}} \left(1 - u^{\deg(\mathfrak{p})}\right)^{-1} \\
 &= \frac{L_K(u)}{(1-u)(1-qu)} = \frac{\prod_{i=1}^{2g} (1 - \pi_i u)}{(1-u)(1-qu)} \\
 &= \zeta_{K,\infty}(s) \zeta_{K,X}(s) = Z_{K,\infty}(u) Z_{K,X}(u) \\
 &= \frac{1}{(1-u)(1-x_1 u)(1-x_2 u)} \\
 &\quad \cdot \frac{1}{1-qu} \prod_{\nu=1}^{\infty} \prod_{\deg(P)=\nu} \prod_{i=1}^2 \frac{1}{1 - z_i(P) u^\nu} ,
 \end{aligned}$$

where the x_i are determined by the splitting of ∞

and the $z_i(P)$ are determined by the splitting of (P) .

Key point: $h = L_K(1)$.

Estimating the Class Number

Putting it together:

$$\begin{aligned} h &= L_K(1) = \frac{1}{(1-x_1)(1-x_2)} \\ &\cdot \prod_{\nu=1}^{\infty} \prod_{\deg(P)=\nu} \frac{1}{(1-z_1(P))(1-z_2(P))} \\ &= q^g L_K(1/q) = \frac{q^{g+2}}{(q-x_1)(q-x_2)} \\ &\cdot \prod_{\nu=1}^{\infty} \prod_{\deg(P)=\nu} \frac{q^{2\nu}}{(q^\nu - z_1(P))(q^\nu - z_2(P))} \end{aligned}$$

To find an estimate E of h , then, we determine the splitting behavior of ∞ and every prime polynomial P up to some degree bound λ .

To find U , we bound the tail of the infinite product.

We need to optimize λ so that the time to compute E is balanced with the time to search the interval $[E - U, E + U]$.

Optimizing the Running Time

* Set:

$$\lambda = \begin{cases} \lfloor \frac{2g-1}{5} \rfloor & \text{if } g \equiv 2 \pmod{5} \\ \text{round}\left(\frac{2g-1}{5}\right) & \text{otherwise.} \end{cases}$$

* Running time: $O\left(q^{\text{round}((2g-1)/5)+\eta(g)}\right)$,
where

$$\eta(g) = \begin{cases} 0 & \text{if } g \equiv 0, 3 \pmod{5} \\ 1/4 & \text{if } g \equiv 1 \pmod{5} \\ -1/4 & \text{if } g \equiv 2 \pmod{5} \\ 1/2 & \text{if } g \equiv 4 \pmod{5} \end{cases}$$

* Comparing running times:

Hasse-Weil interval vs. Scheidler-Stein interval

g	λ	H-W	S-S	H-W	S-S
1	0	$O\left(q^{1/4}\right)$	$O\left(q^{1/4}\right)$	$O\left(h^{0.25}\right)$	$O\left(h^{0.25}\right)$
2	0	$O\left(q^{3/4}\right)$	$O\left(q^{3/4}\right)$	$O\left(h^{0.375}\right)$	$O\left(h^{0.375}\right)$
3	1	$O\left(q^{5/4}\right)$	$O\left(q^{4/4}\right)$	$O\left(h^{0.417}\right)$	$O\left(h^{0.333}\right)$
4	1	$O\left(q^{7/4}\right)$	$O\left(q^{6/4}\right)$	$O\left(h^{0.438}\right)$	$O\left(h^{0.375}\right)$
5	2	$O\left(q^{9/4}\right)$	$O\left(q^{8/4}\right)$	$O\left(h^{0.45}\right)$	$O\left(h^{0.4}\right)$
6	2	$O\left(q^{11/4}\right)$	$O\left(q^{9/4}\right)$	$O\left(h^{0.458}\right)$	$O\left(h^{0.375}\right)$
7	2	$O\left(q^{13/4}\right)$	$O\left(q^{11/4}\right)$	$O\left(h^{0.464}\right)$	$O\left(h^{0.393}\right)$

Examples

* **Genus 3:** $q = 100000039 \approx 10^8$

$$y^3 = 72689039 + 40601482x + 80354454x^2 \\ + 39760243x^3 + x^4$$

$$h = 1000018372353203578299247 \\ = 19 \cdot 43 \cdot 1224012695658755909791$$

* Time: 6098 s + 114720 s = 33 hrs, 33 min, 38 sec

$$* \alpha = \frac{|h-E|}{L^2} = 0.2602615784$$

* **Genus 4:** $q = 1000003 \approx 10^6$

$$y^3 = 79247 + 602740x + 387330x^2 \\ + 146921x^3 + 531472x^4 + x^5$$

$$h = 1001264259802134080148796 \\ = 2^2 \cdot 4549 \cdot 55026613530563534851$$

* Time: 3 s + 312772 s = 3 d, 14 hrs, 52 min, 55 s.

$$* \alpha = \frac{|h-E|}{L^2} = 0.3835039858$$

* Used 20 kangaroos: Total Machine Time: 72.4 days

Future Work and Open Problems

Next steps:

- * Extend the arithmetic to work over singular curves.
- * Extend this to compute regulators and class numbers in unit rank 1.

Open Problems:

- * Figure out unit rank 2 arithmetic to compute in that setting. (It looks like a donut.)
- * Extend the arithmetic to compute h for general cubic function fields.
- * Extend these methods to higher degree function fields.

Open Questions:

- * Are there good ways to get a better approximation of h ?
- * Are there faster ways to compute class numbers in function fields?