Hyperelliptic Curve Arithmetic

(A Shifty Affair)

Renate Scheidler

UNIVERSITY OF CALGARY

Joint work with **Michael John Jacobson, Jr.** and **Monireh Rezai Rad**, U Calgary

West Coast Number Theory
18 December 2016

Why Hyperelliptic Curves?

Jacobians of low genus hyperelliptic curves over finite fields represent a very good setting for cryptography and a strong alternative to elliptic curves.

Why Hyperelliptic Curves?

Jacobians of low genus hyperelliptic curves over finite fields represent a very good setting for cryptography and a strong alternative to elliptic curves.

They are also of interest in number theory (arithmetic geometry).

Why Hyperelliptic Curves?

Jacobians of low genus hyperelliptic curves over finite fields represent a very good setting for cryptography and a strong alternative to elliptic curves.

They are also of interest in number theory (arithmetic geometry).

Cryptography deals almost exclusively with odd degree models since their arithmetic is simpler and faster than that of even degree models. They are also more restrictive.

Why Hyperelliptic Curves?

Jacobians of low genus hyperelliptic curves over finite fields represent a very good setting for cryptography and a strong alternative to elliptic curves.

They are also of interest in number theory (arithmetic geometry).

Cryptography deals almost exclusively with odd degree models since their arithmetic is simpler and faster than that of even degree models. They are also more restrictive.

We like even degree models!

Hyperelliptic Curves

Hyperelliptic curve of genus g over $\mathbb{F}_q$ (q odd):

$$y^2 + h(x)y = f(x)$$

with

- $h(x), f(x) \in \mathbb{F}_q[x]$
- Conditions on the degrees and leading coefficients of $h(x)$ and $f(x)$
- Non-singular and absolutely irreducible

Hyperelliptic Curves

Hyperelliptic curve of genus g over $\mathbb{F}_q$ (q odd):

$$y^2 + h(x)y = f(x)$$

with

- $h(x), f(x) \in \mathbb{F}_q[x]$
- Conditions on the degrees and leading coefficients of $h(x)$ and $f(x)$
- Non-singular and absolutely irreducible

Simplest case: q odd:

$$C : y^2 = f(x)$$

with $f(x) \in \mathbb{F}_q[x]$ monic and squarefree of degree

- $2g + 1$ (odd degree model) or
- $2g + 2$ (even degree model).

Hyperelliptic Curves

Hyperelliptic curve of genus g over $\mathbb{F}_q$ (q odd):

$$y^2 + h(x)y = f(x)$$

with

- $h(x), f(x) \in \mathbb{F}_q[x]$
- Conditions on the degrees and leading coefficients of $h(x)$ and $f(x)$
- Non-singular and absolutely irreducible

Simplest case: q odd:

$$C : y^2 = f(x)$$

with $f(x) \in \mathbb{F}_q[x]$ monic and squarefree of degree

- $2g + 1$ (odd degree model) or
- $2g + 2$ (even degree model).

For $g = 1$, the odd degree model yields the Weierstraß equation of an elliptic curve.

Why Bother With Even Degree Models?

Transformation between models via

$$x \to \frac{1}{x - a}$$

for suitable $a \in \overline{\mathbb{F}}_q$.

Why Bother With Even Degree Models?

Transformation between models via

$$x \to \frac{1}{x - a}$$

for suitable $a \in \overline{\mathbb{F}}_q$.

- Odd degree $\to$ even degree: $f(a) \neq 0$ (so $a \in \mathbb{F}_q$)

Why Bother With Even Degree Models?

Transformation between models via

$$x \to \frac{1}{x - a}$$

for suitable $a \in \overline{\mathbb{F}}_q$.

- Odd degree $\to$ even degree: $f(a) \neq 0$ (so $a \in \mathbb{F}_q$)
- Even degree $\to$ odd degree: $f(a) = 0$ (so $a \in$ Splitting field of f).

Why Bother With Even Degree Models?

Transformation between models via

$$x \to \frac{1}{x - a}$$

for suitable $a \in \overline{\mathbb{F}}_q$.

- Odd degree $\to$ even degree: $f(a) \neq 0$ (so $a \in \mathbb{F}_q$)
- Even degree $\to$ odd degree: $f(a) = 0$ (so $a \in$ Splitting field of f).

Hyperelliptic curve construction methods may generate even degree models (which are traditionally just discarded).

Why Bother With Even Degree Models?

Transformation between models via

$$x \to \frac{1}{x - a}$$

for suitable $a \in \overline{\mathbb{F}}_q$.

- Odd degree $\to$ even degree: $f(a) \neq 0$ (so $a \in \mathbb{F}_q$)
- Even degree $\to$ odd degree: $f(a) = 0$ (so $a \in$ Splitting field of f).

Hyperelliptic curve construction methods may generate even degree models (which are traditionally just discarded).

Research into efficient arithmetic on even degree models is far less advanced.

Points, Divisors, Jacobian

Points on C consists of the usual *finite* points plus

- one infinite point, denoted ∞, for odd degree models;
- two infinite points, denoted ∞^+ and ∞^-, for even degree models.

Points, Divisors, Jacobian

Points on C consists of the usual *finite* points plus

- one infinite point, denoted ∞, for odd degree models;
- two infinite points, denoted ∞^+ and ∞^-, for even degree models.

Divisors on C are finite formal sums of points on C: $D = \sum_{i=1}^{r} n_i P_i$.

Points, Divisors, Jacobian

Points on C consists of the usual *finite* points plus

- one infinite point, denoted ∞, for odd degree models;
- two infinite points, denoted ∞^+ and ∞^-, for even degree models.

Divisors on C are finite formal sums of points on C: $D = \sum_{i=1}^{r} n_i P_i$.

Degree of D: $\deg(D) = \sum_{i=1}^{r} n_i$.

Points, Divisors, Jacobian

Points on C consists of the usual *finite* points plus

- one infinite point, denoted ∞, for odd degree models;
- two infinite points, denoted ∞^+ and ∞^-, for even degree models.

Divisors on C are finite formal sums of points on C: $D = \sum_{i=1}^{r} n_i P_i$.

Degree of D: $\deg(D) = \sum_{i=1}^{r} n_i$.

The Jacobian of C, denoted $\mathrm{Jac}_{\mathbb{F}_q}(C)$, is the quotient group of degree zero divisors on C defined over $\mathbb{F}_q$ modulo principal equivalence.

Points, Divisors, Jacobian

Points on C consists of the usual *finite* points plus

- one infinite point, denoted ∞, for odd degree models;
- two infinite points, denoted ∞^+ and ∞^-, for even degree models.

Divisors on C are finite formal sums of points on C: $D = \sum_{i=1}^{r} n_i P_i$.

Degree of D: $\deg(D) = \sum_{i=1}^{r} n_i$.

The Jacobian of C, denoted $\mathrm{Jac}_{\mathbb{F}_q}(C)$, is the quotient group of degree zero divisors on C defined over $\mathbb{F}_q$ modulo principal equivalence.

- AKA *degree zero divisor class group*;
- Similar to the (Arakelov) class group of a quadratic number field;
- For elliptic curves, this is just the group of points over $\mathbb{F}_q$.

Jacobian Arithmetic, Odd Degree Models

For odd degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty$$

where D only contains finite points.

Jacobian Arithmetic, Odd Degree Models

UNIVERSITY OF CALGARY

For odd degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty$$

where D only contains finite points.

Every class $\mathcal{C} \in \mathrm{Jac}_{\mathbb{F}_q}(C)$ is represented uniquely by a reduced divisor R as

$$\mathcal{C} = [R - \deg(R)\infty]$$

- $\deg(R) \leq g$, with equality almost always (generic case)
- R can be represented by two polynomials of degree $\leq g$ with coefficients in $\mathbb{F}_q$ (Mumford representation of R).

Jacobian Arithmetic, Odd Degree Models

UNIVERSITY OF CALGARY

For odd degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty$$

where D only contains finite points.

Every class $\mathcal{C} \in \mathrm{Jac}_{\mathbb{F}_q}(C)$ is represented uniquely by a reduced divisor R as

$$\mathcal{C} = [R - \deg(R)\infty]$$

- $\deg(R) \leq g$, with equality almost always (generic case)
- R can be represented by two polynomials of degree $\leq g$ with coefficients in $\mathbb{F}_q$ (Mumford representation of R).

Explicit addition in $\mathrm{Jac}_{\mathbb{F}_q}(C)$: if R_1 represents $\mathcal{C}_1$ and R_2 represents $\mathcal{C}_2$, then the reduced divisor representing $\mathcal{C}_1 + \mathcal{C}_2$ is efficiently computable via Cantor's algorithm ("add & reduce").

Jacobian Arithmetic, Even Degree Models

For even degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty^+ + n(\infty^- - \infty^+)$$

where D only contains finite points and $n \in \mathbb{Z}$.

Jacobian Arithmetic, Even Degree Models

For even degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty^+ + n(\infty^- - \infty^+)$$

where D only contains finite points and $n \in \mathbb{Z}$.

Every class $\mathcal{C} \in \mathrm{Jac}_{\mathbb{F}_q}(C)$ is represented uniquely by a reduced divisor R as

$$\mathcal{C} = [R - \deg(R)\infty^+ + n(\infty^- - \infty^+)]$$

with R reduced and $n \in \mathbb{Z}$ "small".

Jacobian Arithmetic, Even Degree Models

For even degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty^+ + n(\infty^- - \infty^+)$$

where D only contains finite points and $n \in \mathbb{Z}$.

Every class $\mathcal{C} \in \mathrm{Jac}_{\mathbb{F}_q}(C)$ is represented uniquely by a reduced divisor R as

$$\mathcal{C} = [R - \deg(R)\infty^+ + n(\infty^- - \infty^+)]$$

with R reduced and $n \in \mathbb{Z}$ "small".

- $n = 0$ too restrictive (misses non-generic divisor classes)

Jacobian Arithmetic, Even Degree Models

For even degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty^+ + n(\infty^- - \infty^+)$$

where D only contains finite points and $n \in \mathbb{Z}$.

Every class $\mathcal{C} \in \mathrm{Jac}_{\mathbb{F}_q}(C)$ is represented uniquely by a reduced divisor R as

$$\mathcal{C} = [R - \deg(R)\infty^+ + n(\infty^- - \infty^+)]$$

with R reduced and $n \in \mathbb{Z}$ "small".

- $n = 0$ too restrictive (misses non-generic divisor classes)
- $0 \leq n \leq g - \deg(R)$ works, but needs Cantor *plus* up to $\lfloor g/2 \rfloor$ adjustment steps (Paulus-Rück 1999)

Jacobian Arithmetic, Even Degree Models

For even degree models, every degree zero divisor is of the form

$$D - \deg(D)\infty^+ + n(\infty^- - \infty^+)$$

where D only contains finite points and $n \in \mathbb{Z}$.

Every class $\mathcal{C} \in \mathrm{Jac}_{\mathbb{F}_q}(C)$ is represented uniquely by a reduced divisor R as

$$\mathcal{C} = [R - \deg(R)\infty^+ + n(\infty^- - \infty^+)]$$

with R reduced and $n \in \mathbb{Z}$ "small".

- $n = 0$ too restrictive (misses non-generic divisor classes)
- $0 \leq n \leq g - \deg(R)$ works, but needs Cantor *plus* up to $\lfloor g/2 \rfloor$ adjustment steps (Paulus-Rück 1999)
- $-\lceil g/2 \rceil \leq n \leq \lfloor g/2 \rfloor - \deg(R)$ gets rid of the extra adjustment steps — balanced representation (Galbraith-Harrison-Mireles Morales 2008)

Generic situation

For generic divisors ($\deg(R) = g$):

Paulus-Rück: $D = R - g\infty^+$

Balanced representation: $D = R - \left\lfloor \frac{g}{2} \right\rfloor \infty^- - \left\lceil \frac{g}{2} \right\rceil \infty^+$

Generic situation

For generic divisors ($\deg(R) = g$):

Paulus-Rück: $D = R - g\infty^+$

Balanced representation: $D = R - \left\lfloor \frac{g}{2} \right\rfloor \infty^- - \left\lceil \frac{g}{2} \right\rceil \infty^+$

Cantor's Algorithm:

- *Divisor addition* — yields a finite divisor of degree $2g$

Generic situation

For generic divisors ($\deg(R) = g$):

Paulus-Rück: $D = R - g\infty^+$

Balanced representation: $D = R - \left\lfloor \frac{g}{2} \right\rfloor \infty^- - \left\lceil \frac{g}{2} \right\rceil \infty^+$

Cantor's Algorithm:

- *Divisor addition* — yields a finite divisor of degree $2g$
- *Divisor reduction* — $\lceil g/2 \rceil$ steps:
  - $\lfloor g/2 \rfloor$ subtractions by $\infty^+ - \infty_-$
  - For g odd, one subtraction by ∞^-

Paulus-Rück Arithmetic

For simplicity, assume that g is even. Consider two generic divisors

$$D_1 = R_1 - g\infty^- \qquad \deg(R_1) = g$$
$$D_2 = R_2 - g\infty^- \qquad \deg(R_2) = g$$

Paulus-Rück Arithmetic

For simplicity, assume that g is even. Consider two generic divisors

$$D_1 = R_1 - g\infty^- \qquad \deg(R_1) = g$$
$$D_2 = R_2 - g\infty^- \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

Paulus-Rück Arithmetic

For simplicity, assume that g is even. Consider two generic divisors

$$D_1 = R_1 - g\infty^- \qquad \deg(R_1) = g$$
$$D_2 = R_2 - g\infty^- \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - g\infty^-] = [R_1 - g\infty^-] + [R_2 - g\infty^-] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

Paulus-Rück Arithmetic

For simplicity, assume that g is even. Consider two generic divisors

$$D_1 = R_1 - g\infty^- \qquad \deg(R_1) = g$$
$$D_2 = R_2 - g\infty^- \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - g\infty^-] = [R_1 - g\infty^-] + [R_2 - g\infty^-] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

$$[D_3] = [D_1] + [D_2] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

Paulus-Rück Arithmetic

For simplicity, assume that g is even. Consider two generic divisors

$$D_1 = R_1 - g\infty^- \qquad \deg(R_1) = g$$
$$D_2 = R_2 - g\infty^- \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - g\infty^-] = [R_1 - g\infty^-] + [R_2 - g\infty^-] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

$$[D_3] = [D_1] + [D_2] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

Need $g/2$ additions by $\infty^+ - \infty^-$ to obtain the unique Paulus-Rück representative in the class of $D_1 + D_2$.

Paulus-Rück Arithmetic

For simplicity, assume that g is even. Consider two generic divisors

$$D_1 = R_1 - g\infty^- \qquad \deg(R_1) = g$$
$$D_2 = R_2 - g\infty^- \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - g\infty^-] = [R_1 - g\infty^-] + [R_2 - g\infty^-] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

$$[D_3] = [D_1] + [D_2] - \left[\frac{g}{2}(\infty^+ - \infty^-)\right]$$

Need $g/2$ additions by $\infty^+ - \infty^-$ to obtain the unique Paulus-Rück representative in the class of $D_1 + D_2$.

In general, $\lfloor g/2 \rfloor$ extra steps are required.

Balanced Divisor Arithmetic

Assume again that g is even. Consider two generic balanced divisors

$$D_1 = R_1 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_1) = g$$

$$D_2 = R_2 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_2) = g$$

Balanced Divisor Arithmetic

Assume again that g is even. Consider two generic balanced divisors

$$D_1 = R_1 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_1) = g$$
$$D_2 = R_2 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

Balanced Divisor Arithmetic

Assume again that g is even. Consider two generic balanced divisors

$$D_1 = R_1 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_1) = g$$
$$D_2 = R_2 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - \frac{g}{2}(\infty^- - \infty^+)] = [R_1 - \frac{g}{2}(\infty^- - \infty^+)] + [R_2 - \frac{g}{2}(\infty^- - \infty^+)]$$

Balanced Divisor Arithmetic

Assume again that g is even. Consider two generic balanced divisors

$$D_1 = R_1 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_1) = g$$
$$D_2 = R_2 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - \frac{g}{2}(\infty^- - \infty^+)] = [R_1 - \frac{g}{2}(\infty^- - \infty^+)] + [R_2 - \frac{g}{2}(\infty^- - \infty^+)]$$

$$[D_3] = [D_1] + [D_2]$$

Balanced Divisor Arithmetic

Assume again that g is even. Consider two generic balanced divisors

$$D_1 = R_1 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_1) = g$$
$$D_2 = R_2 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - \frac{g}{2}(\infty^- - \infty^+)] = [R_1 - \frac{g}{2}(\infty^- - \infty^+)] + [R_2 - \frac{g}{2}(\infty^- - \infty^+)]$$

$$[D_3] = [D_1] + [D_2]$$

No additional steps needed.

Balanced Divisor Arithmetic

Assume again that g is even. Consider two generic balanced divisors

$$D_1 = R_1 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_1) = g$$
$$D_2 = R_2 - \frac{g}{2}(\infty^- + \infty^+) \qquad \deg(R_2) = g$$

Cantor's algorithm:

$$[R_3] = [(R_1 + R_2) - \frac{g}{2}(\infty^+ + \infty^-)] \qquad \deg(R_3) = g$$

$$[R_3 - \frac{g}{2}(\infty^- - \infty^+)] = [R_1 - \frac{g}{2}(\infty^- - \infty^+)] + [R_2 - \frac{g}{2}(\infty^- - \infty^+)]$$

$$[D_3] = [D_1] + [D_2]$$

No additional steps needed.

For g odd, one additional *balancing* step is needed.

Infrastructure

Assume for simplicity that $\mathrm{Jac}_{\mathbb{F}_q}(C) = \langle [\infty^- - \infty^+] \rangle$.

Infrastructure

Assume for simplicity that $\mathrm{Jac}_{\mathbb{F}_q}(C) = \langle [\infty^- - \infty^+] \rangle$.

Then the infrastructure is the set of divisors

$$R - \deg(R)\infty^- \text{ with } R \text{ reduced}$$

Infrastructure

Assume for simplicity that $\mathrm{Jac}_{\mathbb{F}_q}(C) = \langle [\infty^- - \infty^+] \rangle$.

Then the infrastructure is the set of divisors

$$R - \deg(R)\infty^- \text{ with } R \text{ reduced}$$

- Arithmetic is done like Paulus-Rück

Infrastructure

Assume for simplicity that $\mathrm{Jac}_{\mathbb{F}_q}(C) = \langle [\infty^- - \infty^+] \rangle$.

Then the infrastructure is the set of divisors

$$R - \deg(R)\infty^- \text{ with } R \text{ reduced}$$

- Arithmetic is done like Paulus-Rück
- Same speed disadvantage, although fixed in the context of scalar multiplication using the same "shift by $\lceil g/2 \rceil$" trick (Jacobson-S.-Stein 2007)
- Misses an expected proportion of $1/q$ divisor classes (Fontein 2009)

Infrastructure

Assume for simplicity that $\mathrm{Jac}_{\mathbb{F}_q}(C) = \langle [\infty^- - \infty^+] \rangle$.

Then the infrastructure is the set of divisors

$$R - \deg(R)\infty^- \text{ with } R \text{ reduced}$$

- Arithmetic is done like Paulus-Rück
- Same speed disadvantage, although fixed in the context of scalar multiplication using the same "shift by $\lceil g/2 \rceil$" trick (Jacobson-S.-Stein 2007)
- Misses an expected proportion of $1/q$ divisor classes (Fontein 2009)
- No cryptographic security advantage over the Jacobian (Mireles Morales 2008)

Infrastructure

Assume for simplicity that $\mathrm{Jac}_{\mathbb{F}_q}(C) = \langle [\infty^- - \infty^+] \rangle$.

Then the infrastructure is the set of divisors

$$R - \deg(R)\infty^- \text{ with } R \text{ reduced}$$

- Arithmetic is done like Paulus-Rück
- Same speed disadvantage, although fixed in the context of scalar multiplication using the same "shift by $\lceil g/2 \rceil$" trick (Jacobson-S.-Stein 2007)
- Misses an expected proportion of $1/q$ divisor classes (Fontein 2009)
- No cryptographic security advantage over the Jacobian (Mireles Morales 2008)

Mireles Morales declared this speed disadvantage in essence the final nail in the infrastructure arithmetic coffin (he was likely unaware of JSS 2007).

Infrastructure Revised

It turns out that redefining the infrastructure as the set of divisors

$$R - \deg(R)\infty^- - \left\lceil \frac{g}{2} \right\rceil (\infty^+ - \infty^-)$$

fixes things.

Infrastructure Revised

It turns out that redefining the infrastructure as the set of divisors

$$R - \deg(R)\infty^- - \left\lceil \frac{g}{2} \right\rceil (\infty^+ - \infty^-)$$

fixes things.

- Same "shift by $\lceil g/2 \rceil$" trick

Infrastructure Revised

It turns out that redefining the infrastructure as the set of divisors

$$R - \deg(R)\infty^- - \left\lceil \frac{g}{2} \right\rceil (\infty^+ - \infty^-)$$

fixes things.

- Same "shift by $\lceil g/2 \rceil$" trick
- Arithmetic is *identical* to balanced divisor arithmetic

Infrastructure Revised

It turns out that redefining the infrastructure as the set of divisors

$$R - \deg(R)\infty^- - \left\lceil \frac{g}{2} \right\rceil (\infty^+ - \infty^-)$$

fixes things.

- Same "shift by $\lceil g/2 \rceil$" trick
- Arithmetic is *identical* to balanced divisor arithmetic
- Obviates need to work in infrastructure (which is a weird "almost-group" structure that on rare occasions fails associativity)

Infrastructure Revised

It turns out that redefining the infrastructure as the set of divisors

$$R - \deg(R)\infty^- - \left\lceil \frac{g}{2} \right\rceil (\infty^+ - \infty^-)$$

fixes things.

- Same "shift by $\lceil g/2 \rceil$" trick
- Arithmetic is *identical* to balanced divisor arithmetic
- Obviates need to work in infrastructure (which is a weird "almost-group" structure that on rare occasions fails associativity)
- Final nail in infrastructure arithmetic coffin after all — but for a somewhat different reason

Conclusion

We implemented arithmetic in three frameworks for hyperelliptic curves:

- Jacobian, odd degree models
- Jacobian via balanced divisors, even degree models
- Infrastructure with JSS 2007 improvements, even degree models

Conclusion

We implemented arithmetic in three frameworks for hyperelliptic curves:

- Jacobian, odd degree models
- Jacobian via balanced divisors, even degree models
- Infrastructure with JSS 2007 improvements, even degree models

The polynomial arithmetic underlying Cantor is somewhat slower for even degree models (since $\deg(f)$ is larger by one).

Conclusion

We implemented arithmetic in three frameworks for hyperelliptic curves:

- Jacobian, odd degree models
- Jacobian via balanced divisors, even degree models
- Infrastructure with JSS 2007 improvements, even degree models

The polynomial arithmetic underlying Cantor is somewhat slower for even degree models (since $\deg(f)$ is larger by one).

The Diffie-Hellman key agreement protocol for example requires the same number of Cantor operations in all three settings.

Conclusion

We implemented arithmetic in three frameworks for hyperelliptic curves:

- Jacobian, odd degree models
- Jacobian via balanced divisors, even degree models
- Infrastructure with JSS 2007 improvements, even degree models

The polynomial arithmetic underlying Cantor is somewhat slower for even degree models (since $\deg(f)$ is larger by one).

The Diffie-Hellman key agreement protocol for example requires the same number of Cantor operations in all three settings.

However, implementation of DH using state-of-the-art explicit formulas shows that

- in genus 2, even degree is about 7-8% slower than odd degree;
- in genus 3, even degree is about 19-20% slower than odd degree.

Conclusion

We implemented arithmetic in three frameworks for hyperelliptic curves:

- Jacobian, odd degree models
- Jacobian via balanced divisors, even degree models
- Infrastructure with JSS 2007 improvements, even degree models

The polynomial arithmetic underlying Cantor is somewhat slower for even degree models (since $\deg(f)$ is larger by one).

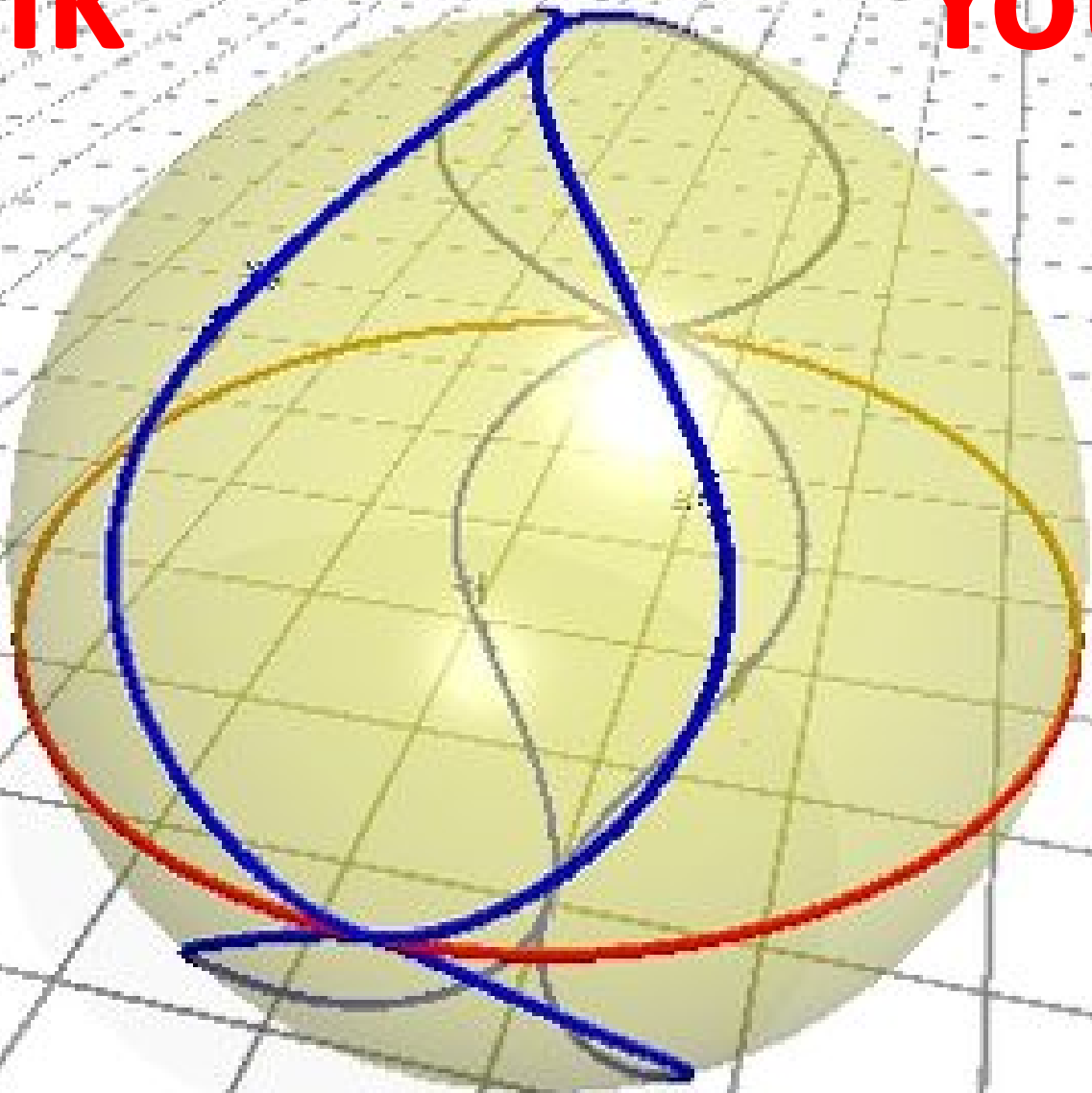
The Diffie-Hellman key agreement protocol for example requires the same number of Cantor operations in all three settings.

However, implementation of DH using state-of-the-art explicit formulas shows that

- in genus 2, even degree is about 7-8% slower than odd degree;
- in genus 3, even degree is about 19-20% slower than odd degree.

It may well be possible to improve the genus 3 formulas for even degree.

Thank You !



$$y^2=x^6+x^2+x$$

http://voltage.typepad.com/superconductor/2011/09/a-projective-imaginary-hyperelliptic-curve.html