

Divisor Class Halving on Hyperelliptic Curves

Peter Birkner

Department of Mathematics, Technical University of Denmark
(currently visiting Fields Institute, Toronto)

Cryptography Seminar, University of Waterloo

(Supported by the Danish Research Council for Technology and Production Sciences, grant no. 274-05-0151)

Motivation

- In cryptography the Discrete Logarithm Problem (DLP) is important: **Given:** A group $(G, +)$, $G = \langle g \rangle$ and $h \in G$.
Find: $k \in \mathbb{Z}$ with $k \cdot g = h$.
- We need groups where the DLP is hard to solve.
- Scalar multiplication $k \cdot g$ is the most important operation in DLP based cryptosystems, which can be performed by double-and-add or windowing algorithms.
- We need very fast group operations, doublings are most critical in windowing methods.
- **Aim of the talk:** Efficient arithmetic on hyperelliptic curves.

Part I: Basic Notions and Introduction

Definition

Let K be a field and \bar{K} an algebraic closure of K . A *hyperelliptic curve C of genus g over K* with at least one K -rational Weierstraß point can be given by an equation of the form

$$C : y^2 + h(x)y = f(x),$$

where

- $h \in K[x]$ is a polynomial of degree at most g ,
- $f \in K[x]$ is a monic polynomial of degree $2g + 1$,
- no point on C over \bar{K} satisfies both partial derivatives $2y + h = 0$ and $h'y - f' = 0$.

Definition

Let C be a hyperelliptic curve of genus g over a field K .

- A **divisor** D on C is a formal sum of points on C :

$$D = \sum_{P \in C(\overline{K})} n_P P \quad (n_P \in \mathbb{Z}, n_P = 0 \text{ for almost all } P \in C(\overline{K}))$$

- The **degree** of a divisor D on C is

$$\deg(D) = \sum_{P \in C(\overline{K})} n_P.$$

The **group of degree zero divisors on C** is denoted $\text{Div}_C^0(\overline{K})$.

Definition

Let C be a hyperelliptic curve of genus g over a field K .

- The **coordinate ring of C over K** is the quotient ring

$$K[C] := K[x,y]/(y^2+h(x)y-f(x)).$$

An element of $K[C]$ is called a **polynomial function on C** .

- The **function field $K(C)$ of C over K** is the field of fractions of $K[C]$. The elements of $K(C)$ are called **rational functions on C** .

Definition

Let C be a hyperelliptic curve of genus g over a field K and let $P \in C(\overline{K})$. A function $u \in \overline{K}(C)$ with $u(P) = 0$ is called a **uniformising parameter for P** , if the following property holds:

For each $0 \neq a \in \overline{K}[C]$ there exists an integer d and a function $s \in \overline{K}(C)$ such that $a = u^d s$ and $s(P) \notin \{\infty, 0\}$.

Theorem

For each $P \in C(\overline{K})$ there exists an uniformising parameter for P .

Definition

- 1 Let C be a hyperelliptic curve of genus g over a field K . Let $P \in C(\bar{K})$ with uniformising parameter $u \in \bar{K}(C)$ and let $0 \neq a \in \bar{K}[C]$ be a **polynomial function on C** . Now, a can be written as

$$a = u^d s.$$

The (unique) integer d is called the **order of a at P** .

- 2 Let $r = a/b \in \bar{K}(C)^*$ be a **rational function on C** and $P \in C(\bar{K})$. The **order of r at P** is defined as

$$\text{ord}_P(r) := \text{ord}_P(a) - \text{ord}_P(b).$$

Definition

Let $r \in \overline{K}(C)^*$ be a rational function on C . The **divisor of r** is

$$\operatorname{div}(r) := \sum_{P \in C(\overline{K})} (\operatorname{ord}_P(r)) P.$$

Definition

A divisor D is called **principal** if $D = \operatorname{div}(r)$ for some rational function $r \in \overline{K}(C)^*$. The set of principal divisors on C is denoted **$\operatorname{Princ}(C)$** .

A principal divisor has degree zero. So, $\operatorname{Princ}(C)$ is a subgroup of $\operatorname{Div}_C^0(\overline{K})$.

Definition

The **divisor class group of C** is the quotient group

$$\mathrm{Pic}_C^0(\bar{K}) := \mathrm{Div}_C^0(\bar{K}) / \mathrm{Princ}(C).$$

It is also called the **Picard group of C** .

A divisor class \bar{D} is called **K -rational**, if \bar{D} is invariant under the action of the Galois group $\mathrm{Gal}(\bar{K}/K)$. The set of **K -rational divisor classes** is denoted $\mathrm{Pic}_C^0(K)$.

Since our curves have only one point at infinity, the divisor class group is isomorphic to the **ideal class group of C** .

Theorem (Mumford)

Let C be a hyperelliptic curve of genus g over a field K . Each nontrivial divisor class of C over K can be represented by a unique pair of polynomials $u, v \in K[x]$, where

- 1 u is monic,
- 2 $\deg v < \deg u \leq g$,
- 3 $u \mid v^2 + vh - f$.

In the genus 2 case each divisor class can be represented by the 4 coefficients u_1, u_0, v_1, v_0 of the polynomials u and v .

Composition & Reduction (Cantor/Koblitz)

IN: $\overline{D}_1 = [u_1, v_1], \overline{D}_2 = [u_2, v_2], C : y^2 + h(x)y = f(x)$

OUT: $\overline{D} = [u, v]$ reduced with $\overline{D} = \overline{D}_1 + \overline{D}_2$

- 1 compute $d_1 = \gcd(u_1, u_2) = e_1 u_1 + e_2 u_2$
- 2 compute $d = \gcd(d_1, v_1 + v_2 + h) = c_1 d_1 + c_2 (v_1 + v_2 + h)$
- 3 let $s_1 = c_1 e_1, s_2 = c_1 e_2, s_3 = c_2$
- 4 $u = \frac{u_1 u_2}{d^2} \quad v = \frac{s_1 u_1 v_2 + s_2 u_2 v_1 + s_3 (v_1 v_2 + f)}{d} \pmod{u}$
- 5 let $u' = \frac{f - vh - v^2}{u} \quad b' = (-h - v) \pmod{u'}$
- 6 if $\deg u' > g$ put $u := u', v := v'$ goto step 5
- 7 make u monic

Part II: Efficient Doubling of Divisor Classes in Characteristic 2 and Genus 2

Classification of Binary Genus 2 Curves

Let $C : y^2 + h(x)y = f(x)$. We need to have $h \neq 0$ to avoid singular curves.

The 2-rank of those curves is determined by the **degree of h** :

- **Type I** if $\deg h = 2$. This case splits into **Type Ia** where h has no root in \mathbb{F}_{2^d} and **Type Ib** where such a root does exist. These curves have 2-rank equal to 2.
- **Type II** if $\deg h = 1$. These curves have 2-rank equal to 1.
- **Type III** if $\deg h = 0$. These curves have 2-rank equal to 0, they are supersingular.

In the remainder of this talk we focus on Type II curves.

Doubling in the General Case

Doubling, deg $u = 2$			
Input	$[u, v], u = x^2 + u_1 x + u_0, v = v_1 x + v_0$		
Output	$[u', v'] = 2[u, v]$		
Step	Expression	odd	even
1	compute $\tilde{v} \equiv (h + 2v) \bmod u = \tilde{v}_1 x + \tilde{v}_0$: $\tilde{v}_1 = h_1 + 2v_1 - h_2 u_1, \tilde{v}_0 = h_0 + 2v_0 - h_2 u_0$;		
2	compute resultant $r = \text{res}(\tilde{v}, u)$: $w_0 = v_1^2, w_1 = u_1^2, w_2 = \tilde{v}_1^2, w_3 = u_1 \tilde{v}_1, r = u_0 w_2 + \tilde{v}_0 (\tilde{v}_0 - w_3)$;	2S, 3M ($w_2 = 4w_0$)	2S, 3M (see below)
3	compute almost inverse $\text{inv}' = \text{inv}r$: $\text{inv}'_1 = -\tilde{v}_1, \text{inv}'_0 = \tilde{v}_0 - w_3$;		
4	compute $k' = (f - hv - v^2)/u \bmod u = k'_1 x + k'_0$: $w_3 = f_3 + w_1, w_4 = 2u_0, k'_1 = 2(w_1 - f_4 u_1) + w_3 - w_4 - h_2 v_1$; $k'_0 = u_1(2w_4 - w_3 + f_4 u_1 + h_2 v_1) + f_2 - w_0 - 2f_4 u_0 - h_1 v_1 - h_2 v_0$;	1M	2M (see below)
5	compute $s' = k' \text{inv}' \bmod u$: $w_0 = k'_0 \text{inv}'_0, w_1 = k'_1 \text{inv}'_1, s'_1 = (\text{inv}'_0 + \text{inv}'_1)(k'_0 + k'_1) - w_0 - w_1(1 + u_1), s'_0 = w_0 - u_0 w_1$;	5M	5M
6	compute $s'' = x + s_0/s_1$ and s_1 : $w_1 = 1/(rs'_1) (= 1/r^2 s_1), w_2 = rw_1 (= 1/s'_1), w_3 = s_1^2 w_1 (= s_1)$; $w_4 = rw_2 (= 1/s_1), w_5 = w_4^2, s_0'' = s_0' w_2$;	I, 2S, 5M	I, 2S, 5M
7	compute $l' = s'' u = x^3 + l'_2 x^2 + l'_1 x + l'_0$: $l'_2 = u_1 + s_0'', l'_1 = u_1 s_0'' + u_0, l'_0 = u_0 s_0''$;	2M	2M
8	compute $u' = s^2 + (h + 2v)s/u + (v^2 + hv - f)/u^2$: $u'_0 = s_0''^2 + w_4(h_2(s_0'' - u_1) + 2v_1 + h_1) + w_5(2u_1 - f_4), u'_1 = 2s_0'' + h_2 w_4 - w_5$;	S, 2M	S, M
9	compute $v' \equiv -h - (l + v) \bmod u' = v'_1 x + v'_0$: $w_1 = l'_2 - u'_1, w_2 = u'_1 w_1 + u'_0 - l'_1, v'_1 = w_2 w_3 - v_1 - h_1 + h_2 u'_1$; $w_2 = u'_0 w_1 - l'_0, v'_0 = w_2 w_3 - v_0 - h_0 + h_2 u'_0$;	4M	4M
total		each I, 5S, 22M	

Doubling $\deg h = 1, \deg u = 2$				
Input	$[u, v], u = x^2 + u_1x + u_0, v = v_1x + v_0; h_1^2, h_1^{-1}$			
Output	$[u', v'] = 2[u, v]$			
Step	Expression	$h_1 = 1$	h_1^{-1} small	h_1 arbitrary
1	<u>compute rs_1:</u> $z_0 = u_0^2, k'_1 = u_1^2 + f_3;$ $w_0 = f_0 + v_0^2 (= rs'_1/h_1^3);$	3S	3S	3S
2	<u>compute $1/s_1$ and s''_0:</u> $w_1 = (1/w_0)z_0 (= h_1/s_1);$ $z_1 = k'_1 w_1, s''_0 = z_1 + u_1;$	I, 2M	I, 2M	I, 2M
3	<u>compute u':</u> $w_2 = h_1^2 w_1, u'_1 = w_2 w_1;$ $u'_0 = s''_0{}^2 + w_2;$	2S	S, 2M	S, 2M
4	<u>compute v':</u> $w_3 = w_2 + k'_1;$ $v'_1 = h_1^{-1}(w_3 z_1 + w_2 u'_1 + f_2 + v_1^2);$ $v'_0 = h_1^{-1}(w_3 u'_0 + f_1 + z_0);$	S, 3M	S, 3M	S, 5M
total		I, 6S, 5M	I, 5S, 7M	I, 5S, 9M

Inversion-free Doubling (1)

Inversion is expensive compared to multiplication (e. g. in hardware applications). Can we avoid inversions?

- **Affine** coordinates: A divisor class is given by the coefficients of u and v : $[u_1, u_0, v_1, v_0]$
- **Projective** coordinates: Use $[U_1, U_0, V_1, V_0, Z]$ with $u_i = U_i/Z$ and $v_i = V_i/Z$
- **New** coordinates: $[U_1, U_0, V_1, V_0, Z_1, Z_2, z_1, z_2, z_3, z_4]$ with $u_i = U_i/Z_1^2$, $v_i = V_i/Z_1^3 Z_2$ and precomputations z_1, z_2, z_3, z_4
- **Recent** coordinates: $[U_1, U_0, V_1, V_0, Z, z]$ with $u_i = U_i/Z$ and $v_i = V_i/Z^2$ and the precomputation $z = Z^2$

Inversion-free doubling can be achieved in Recent, New and Projective coordinates!

Inversion-free Doubling (2)

Divisor class doubling in Recent coordinates **without inversions!**

Input:	$\bar{D} = [U_1, U_0, V_1, V_0, Z, z]$, precomputed values h_1^2 and h_1^{-1}	
Output	$[U'_1, U'_0, V'_1, V'_0, Z', z'] = 2[U_1, U_0, V_1, V_0, Z, z]$	
Step	Expression	Complexity
1	Precomputations $Z_4 \leftarrow z^2, y_0 \leftarrow U_0^2, t_1 \leftarrow U_1^2 + f_3 z, w_0 \leftarrow Z_4 f_0 + V_0^2,$ $\bar{Z} \leftarrow z w_0, w_1 \leftarrow y_0 Z_4, y_1 \leftarrow t_1 y_0 z, s_0 \leftarrow y_1 + U_1 w_0 z$ $w_2 \leftarrow h_1^2 w_1, w_3 \leftarrow w_2 + t_1 w_0$	10M + 4S
2	Compute U' $U'_1 \leftarrow w_2 w_1, w_2 \leftarrow w_2 \bar{Z}, U'_0 \leftarrow s_0^2 + w_2$	2M + S
3	Compute V' $Z' \leftarrow \bar{Z}^2, V'_1 \leftarrow h_1^{-1} (w_2 U'_1 + (w_3 y_1 + f_2 Z' + (V_1 w_0)^2) Z')$ $V'_0 \leftarrow h_1^{-1} (\bar{Z} (w_3 U'_0 + y_0 w_0 Z')), z' \leftarrow Z'^2$	11M + 3S
Total		23M + 8S

If $h_1 = 1$, one can even achieve $20M + 8S$.

Inversion Free Coordinates — Overview

In the case $h(x) = x$, i. e. the curve is of Type II, we have the following complexities for adding and doubling:

Doubling		Addition	
Operation	Costs	Operation	Costs
$2\mathcal{N} = \mathcal{N}$	$28M + 5S$	$\mathcal{R} + \mathcal{R} = \mathcal{R}$	$49M + 8S$
$2\mathcal{P} = \mathcal{P}$	$22M + 6S$	$\mathcal{P} + \mathcal{P} = \mathcal{P}$	$49M + 4S$
$2\mathcal{R} = \mathcal{R}$	$20M + 8S$	$\mathcal{N} + \mathcal{N} = \mathcal{N}$	$42M + 6S$
—	—	$\mathcal{A} + \mathcal{R} = \mathcal{R}$	$42M + 7S$
—	—	$\mathcal{A} + \mathcal{P} = \mathcal{P}$	$39M + 4S$
—	—	$\mathcal{A} + \mathcal{N} = \mathcal{N}$	$36M + 6S$
$2\mathcal{A} = \mathcal{A}$	$1 + 5M + 6S$	$\mathcal{A} + \mathcal{A} = \mathcal{A}$	$1 + 2M + 3S$

Part III: Halving of Divisor Classes

What is halving of a divisor class?

Given:

- A genus 2 hyperelliptic curve (HEC) of form
$$C : y^2 + h(x)y = f(x)$$
over an finite field \mathbb{F}_{2^d} .
- Let the divisor class group $\text{Pic}_C^0(K)$ of C have order $2^k r$ where r **is odd**.
- A divisor class $\bar{D} = [u_1, u_0, v_1, v_0]$ in the order- r subgroup S of $\text{Pic}_C^0(K)$ to be halved.

Problem: Find a divisor class $\bar{E} \in S$ such that $2\bar{E} = \bar{D}$.

What is halving of divisor classes good for?

- Provide an efficient halving algorithm to HEC cryptography
- A step towards equipping HEC with all the features that EC cryptosystems have
- Efficient halving can lead to faster scalar multiplication using **half-and-add** instead of double-and-add
- Halving can be used by all windowing methods

Only **one** result about halving of divisor classes yet!

- Kitamura, Katagi, Takagi: A Complete Divisor Class Halving Algorithm for Hyperelliptic Curve Cryptosystems of Genus Two (ACISP 2005, ePrint 2004/255).
- They cover Type I curves and some special Type II curves.
- Complexity (Type I): 1I, 21M, 2S, 3SR, 2HT, 2TR

- We focus on Type II curves over \mathbb{F}_{2^d} , where d is odd. Isomorphic transformations lead to a unique equation
$$C : y^2 + xy = x^5 + f_3x^3 + f_2x^2 + f_0,$$
with $f_0 \in \mathbb{F}_{2^d}^*$, $f_3 \in \mathbb{F}_{2^d}$ and $f_2 \in \mathbb{F}_2$.
- We have $2 \cdot (2^d - 1) \cdot 2^d \approx 2^{2d+1}$ isomorphism classes of curves.
- Since the 2-rank of these curves is one, we can find curves C with $\#\text{Pic}_C^0(\mathbb{F}_{2^d}) = 2r$, where r is odd.
- Halving algorithm developed by inverting doubling formulae.

From Doubling to Halving (1)

- Explicit formulae (by Lange and Stevens) to double a divisor class
- Given a divisor class $\bar{E} = [u'_1, u'_0, v'_1, v'_0]$

$$u_1 = \left(\frac{u_0'^2}{f_0 + v_0'^2} \right)^2 \quad (1)$$

$$u_0 = \left((u_1'^2 + f_3)\sqrt{u_1} + u_1' \right)^2 + \sqrt{u_1} \quad (2)$$

$$v_0 = \left(\sqrt{u_1} + u_1'^2 + f_3 \right) u_0 + u_0'^2 \quad (3)$$

$$v_1 = \left(\sqrt{u_1} + u_1'^2 + f_3 \right) + \sqrt{u_1} u_1 + f_2 + v_1'^2 \quad (4)$$

- Resulting divisor class $\bar{D} = 2\bar{E} = [u_1, u_0, v_1, v_0]$

From Doubling to Halving (2)

- **Reverse doubling formulae** to perform halving!

- Start with (2) from doubling:

$$u_0 = \left((u_1'^2 + f_3)\sqrt{u_1} + u_1' \right)^2 + \sqrt{u_1}$$

- Expand right-hand side and substitute $U = u_1'^2$:

$$U^2 u_1 + U = u_0 + f_3^2 u_1 + \sqrt{u_1}$$

- Solve quadratic equation in U over \mathbb{F}_{2^d} and get **two solutions** of u_1' by re-substituting $u_1' = \sqrt{U}$

From Doubling to Halving (3)

2 solutions \Rightarrow **2 possible divisor classes**. Which one is the right one?

- 2 divides the group order \Rightarrow 2 preimages of \bar{D} under doubling: \bar{E} and $\bar{E} + \zeta$, where ζ has order 2
- $2\bar{E} = \bar{D}$ and $2(\bar{E} + \zeta) = 2\bar{E} + 0 = \bar{D}$
- Distinguish between \bar{E} and $\bar{E} + \zeta$ by checking, if can be halved again (ζ cannot be halved!)
- **Lemma:** A divisor class $\bar{D} = [u_1, u_0, v_1, v_0]$ can be halved $\iff \text{Tr}(u_1(u_0 + \sqrt{u_1} + f_3^2 u_1)) = 0$.

From Doubling to Halving (4)

Check, if first or second solution of QE leads to \bar{E}

- Compute u'_0 and u'_1 using the **first** solution
- Then use lemma to check if u'_1 and u'_0 belong to a divisor class that can be halved (\bar{E}) or not ($\bar{E} + \zeta$)
- If \bar{E} , continue computing v'_0 and v'_1 (**Done!**)
- If $\bar{E} + \zeta$, re-compute u'_1 and u'_0 with the **second** solution and then compute v'_0 and v'_1 (**Done!**)

The Halving Algorithm

Algorithm 1 Divisor Class Halving (HLV)

INPUT: Divisor class $\bar{D} = [u, v]$, where $u = x^2 + u_1x + u_0$, $v = v_1x + v_0$ and the pre-computed values $f_3^2, \sqrt{f_0}$

OUTPUT: Halved divisor class $\bar{E} = [u', v']$ such that $\bar{D} = 2\bar{E}$

- 1: $q_1 \leftarrow \sqrt{u_1}$, $q_2 \leftarrow 1/q_1$, $q_3 \leftarrow q_2^2$, $q_4 \leftarrow u_0q_3$, $q_5 \leftarrow \sqrt{q_2}$ ▷ 1I, 1M, 2SR, 1S
 - 2: $q_6 \leftarrow \sqrt{q_4}$, $c \leftarrow u_1(q_6 + q_5 + f_3)$ ▷ 1SR, 1M
 - 3: $t' \leftarrow \sum_{i=0}^{(d-3)/2} c^{2(2i+1)}$ ▷ 1HT
 - 4: $u'_1 \leftarrow t'q_2$, $t \leftarrow u_1^2$, $s_1 \leftarrow v_0 + (q_1 + t + f_3)u_0$ ▷ 2M, 1S
 - 5: $u'_0 \leftarrow \sqrt{s_1}$, $b \leftarrow \text{Trace}(u'_1(u'_0 + t + f_3))$ ▷ 1M, 1SR, 1TR
 - 6: **if** $b = 0$ **then**
 - 7: $v'_0 \leftarrow q_5u'_0 + \sqrt{f_0}$ ▷ 1M
 - 8: **else**
 - 9: $t \leftarrow t + q_3$, $u'_1 \leftarrow u'_1 + q_2$
 - 10: $u'_0 \leftarrow u'_0 + q_6$, $v'_0 \leftarrow q_5u'_0 + \sqrt{f_0}$ ▷ 1M
 - 11: **end if**
 - 12: $v'_1 \leftarrow \sqrt{v_1 + q_1 \left((q_1 + t + f_3)(t + f_3) + u_1 \right) + f_2}$ ▷ 2M, 1SR
 - 13: **return** $[x^2 + u'_1x + u'_0, v'_1x + v'_0]$ ▷ Total: 1I, 8M, 5SR, 2S, 1HT, 1TR
-

Complexity of Halving — Genus 2 Case

- Efficient halving algorithm for divisor classes
- Type II curves with equation $y^2 + xy = x^5 + f_3x^3 + f_2x^2 + f_0$
- Base field is \mathbb{F}_{2^d} (d odd)

	Type II curve $y^2 + xy = x^5 + f_3x^3 + f_2x^2 + f_0$	Special curve (see KKT, Appendix D) $y^2 + xy = x^5 + f_0$
Kitamura Katagi, Takagi	1I, 15M, 3SR, 3S, 2HT, 2TR	1I, 12M, 5SR, 2S, 1HT, 1TR
Our work	1I, 8M, 5SR, 2S, 1HT, 1TR	

Outlook: The Genus 3 Case

Halving on Genus 3 Curves

- For genus 3, the curve $y^2 + cy = f(x)$ over \mathbb{F}_{2^d} is **not** supersingular.
- The 2-rank of these curves is zero. Hence, doubling and halving can be performed in the whole Picard group and no test is necessary because each divisor class has a unique preimage under doubling.
- The most frequent case of divisor class doubling on genus 3 curves was done by Guyot, Kaveh and Patankar and by Pelzl, Wollinger, Guajardo and Paar.
- Joint work on genus 3 halving is in progress together with N. Thériault.

Thank you for your attention!