Curve Cryptography
–
suitable primitives for embedded devices

Tanja Lange

Department of Mathematics

Technical University of Denmark

t.lange@mat.dtu-dk

30.05.2005

Overview

- Arithmetic on elliptic curves.
- Arithmetic on hyperelliptic curves.
- Special case of characteristic two.
- Brief overview of side channel attacks.
- Countermeasures for curve based cryptography.

Arithmetic on elliptic curves

Elliptic curve

$$E : y^2 + \underbrace{(a_1 x + a_3)}_{h(x)} y = \underbrace{x^3 + a_2 x^2 + a_4 x + a_6}_{f(x)}, \quad h, f \in \mathbb{F}_q[x].$$

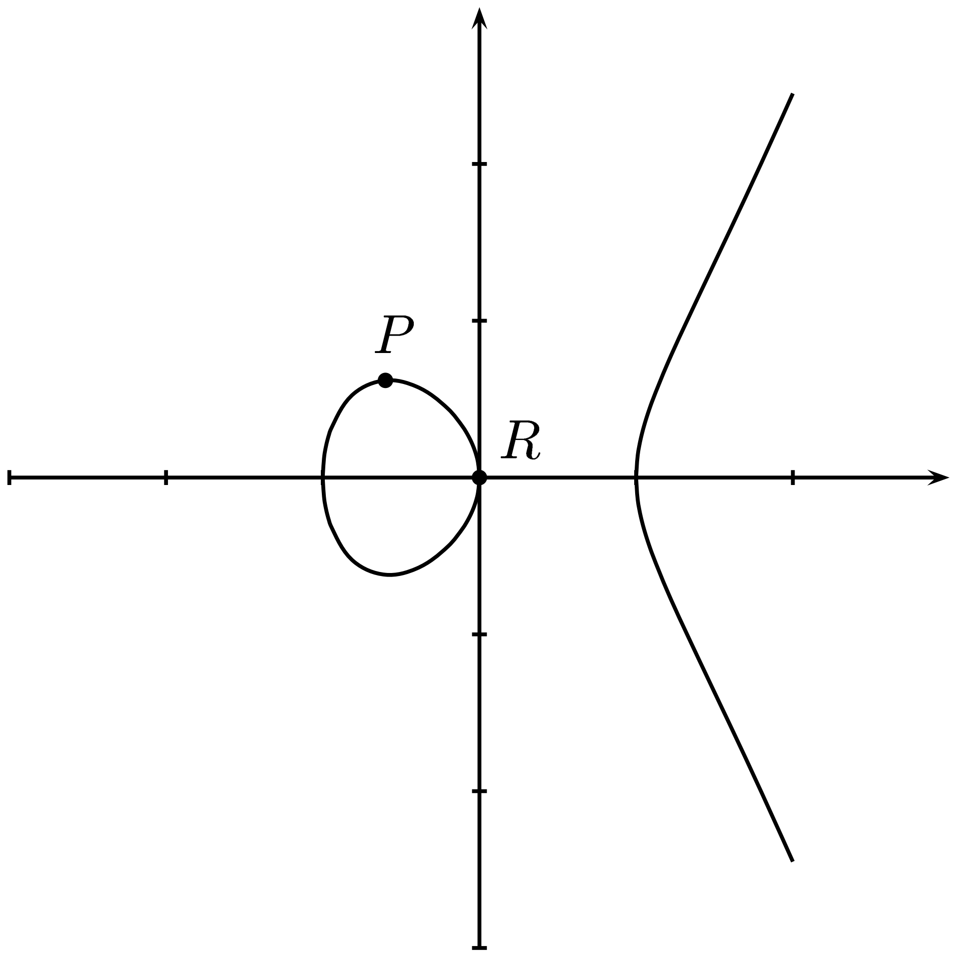
Group: $E(\mathbb{F}_q) = \{ (x, y) \in \mathbb{F}_q^2 : y^2 + h(x)y = f(x) \} \cup \{ P_\infty \}$

Often $q = 2^r$ or $q = p$, prime. Isomorphic transformations lead to

$$\begin{aligned} y^2 &= f(x), & q \text{ odd}, \\ y^2 + xy &= x^3 + ax^2 + b, & q = 2^r, \text{ curve non-supersingular} \end{aligned}$$

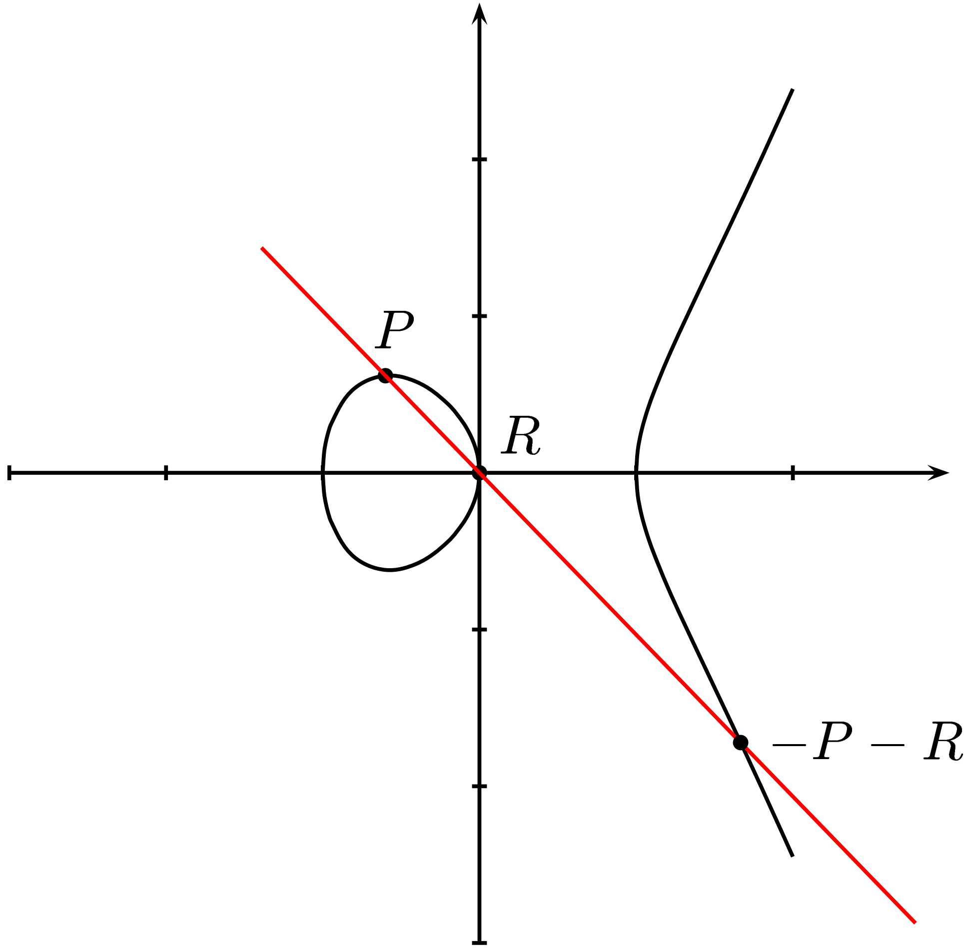
Group Law in $E(\mathbb{R}), h = 0$

$y^2 = x^3 - x$



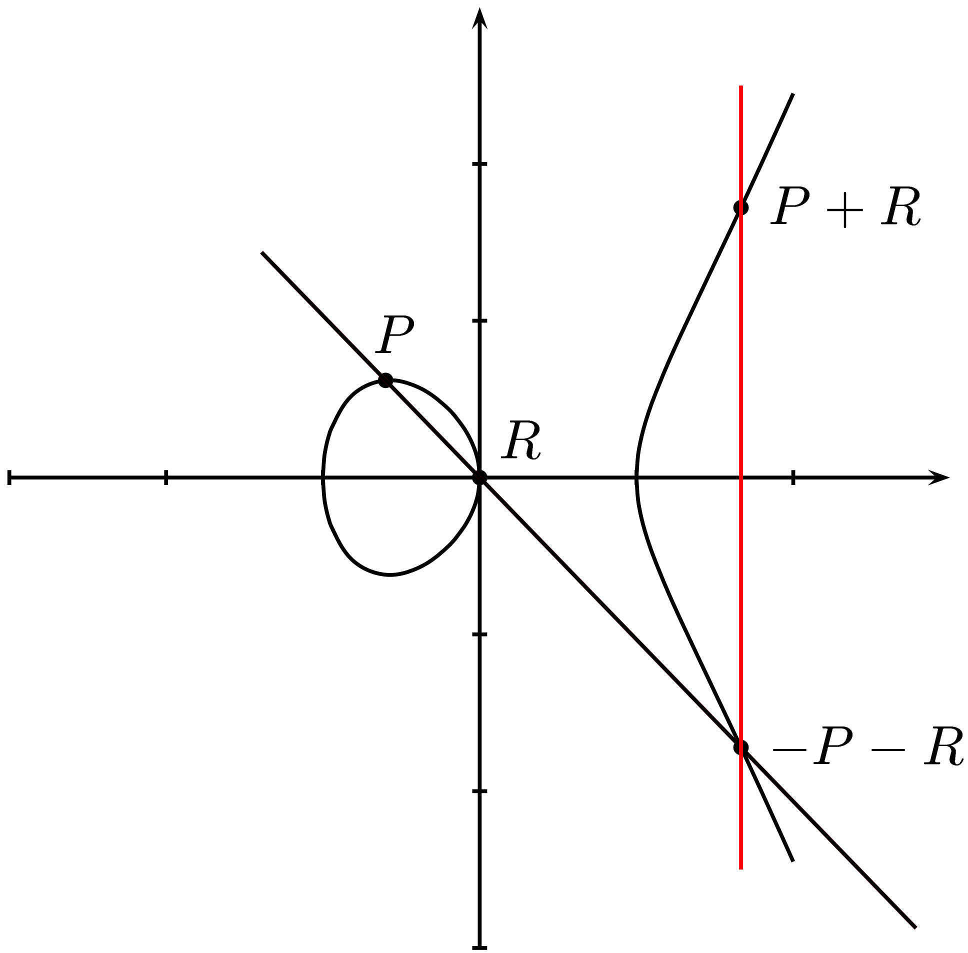
Group Law in $\boldsymbol{E(\mathbb{R}), h = 0}$

$y^2 = x^3 - x$



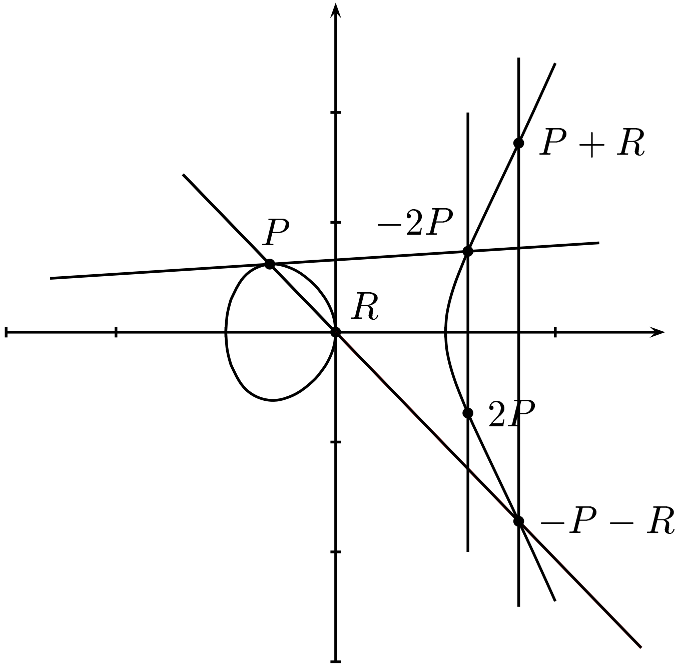
Group Law in $\boldsymbol{E(\mathbb{R}), h = 0}$

$y^2 = x^3 - x$



Group Law (q odd)

$$E : y^2 = x^3 + a_4 x + a_6, \ a_i \in \mathbb{F}_q$$



for $(x_1, y_1) \neq (x_2, -y_2)$:

$$\begin{aligned}(x_1, y_1) + (x_2, y_2) &= \\ &= (x_3, y_3) = \\ &= (\lambda^2 - x_1 - x_2, \lambda(x_1 - x_3) - y_1),\end{aligned}$$

where

$$\lambda = \begin{cases} (y_2 - y_1)/(x_2 - x_1) & \text{if } x_1 \neq x_2, \\ (3x_1^2 + a_4)/(2y_1) & \text{else.} \end{cases}$$

$\Rightarrow$ Addition and Doubling differ considerably.

1 I, 2M, 1S vs. 1 I, 2M, 2S

Projective Coordinates

$P = (X_1 : Y_1 : Z_1)$, $Q = (X_2 : Y_2 : Z_2)$, $P + Q = (X_3 : Y_3 : Z_3)$ on

$$E : Y^2 Z = X^3 + a_4 X Z^2 + a_6 Z^3$$

Addition: $P \neq \pm Q$

$A = Y_2 Z_1 - Y_1 Z_2, B = X_2 Z_1 - X_1 Z_2,$

$C = A^2 Z_1 Z_2 - B^3 - 2B^2 X_1 Z_2$

$X_3 = BC, Z_3 = B^3 Z_1 Z_2$

$Y_3 = A(B^2 X_1 Z_2 - C) - B^3 Y_1 Z_2,$

Doubling $P = Q \neq -P$

$A = a_4 Z_1^2 + 3X_1^2, B = Y_1 Z_1,$

$C = X_1 Y_1 B, D = A^2 - 8C$

$X_3 = 2BD, Z_3 = 8B^3.$

$Y_3 = A(4C - D) - 8Y_1^2 B^2$

No inversion is needed and the computation times are $12M + 2S$ for a general addition and $7M + 5S$ for a doubling.

Jacobian Coordinates

$P = (X_1 : Y_1 : Z_1)$, $Q = (X_2 : Y_2 : Z_2)$, $P + Q = (X_3 : Y_3 : Z_3)$
on

$$Y^2 = X^3 + a_4 X Z^4 + a_6 Z^6$$

Addition: $P \neq \pm Q$

$A = X_1 Z_2^2, B = X_2 Z_1^2, C = Y_1 Z_2^3,$

$D = Y_2 Z_1^3, E = B - A, F = D - C$

$X_3 = -E^3 - 2AE^2 + F^2, Z_3 = Z_1 Z_2 E,$

$Y_3 = -CE^3 + F(AE^2 - X_3),$

Doubling $P = Q \neq -P$

$A = 4X_1 Y_1^2, B = 3X_1^2 + a_4 Z_1^4$

$X_3 = -2A + B^2, Z_3 = 2Y_1 Z_1$

$Y_3 = -8Y_1^4 + B(A - X_3).$

No inversion is needed and the computation times are $12M + 4S$ for a general addition and $4M + 6S$ for a doubling.

Different coordinate systems $y^2 = x^3 + ax + b$

system	points	correspondence
affine ($\mathcal{A}$)	(x, y)	
projective ($\mathcal{P}$)	(X, Y, Z)	$(X/Z, Y/Z)$
jacobian ($\mathcal{J}$)	(X, Y, Z)	$(X/Z^2, Y/Z^3)$
Chudnovsky jacobian ($\mathcal{J}^C$)	(X, Y, Z, Z^2, Z^3)	$(X/Z^2, Y/Z^3)$
modified jacobian ($\mathcal{J}^m$)	(X, Y, Z, aZ^4)	$(X/Z^2, Y/Z^3)$

system	addition			doubling		
affine ($\mathcal{A}$)	2M	1S	1I	2M	2S	1I
projective ($\mathcal{P}$)	12M	2S	–	7M	5S	–
jacobian ($\mathcal{J}$)	12M	4S	–	4M	6S	–
Chudnovsky jacobian ($\mathcal{J}^C$)	11M	3S	–	5M	6S	–
modified jacobian ($\mathcal{J}^m$)	13M	6S	–	4M	4S	–

Mixed coordinates

(Cohen, Miyaji, Ono, Asiacrypt '98)

Idea is to combine different systems to achieve highest speed for scalar multiplication incl. windowing methods.

affordable inversions:

precomputations in $\mathcal{A}$ (with Montgomery's trick),
main doublings in $\mathcal{J}^m$,
final doublings $2\mathcal{J}^m = \mathcal{J}$,
additions $\mathcal{A} + \mathcal{J} = \mathcal{J}^m$

expensive inversions:

precomputations in $\mathcal{J}^C$,
main doublings in $\mathcal{J}^m$,
final doublings $2\mathcal{J}^m = \mathcal{J}$,
additions $\mathcal{J} + \mathcal{J}^C = \mathcal{J}^m$

Arithmetic on hyperelliptic curves

Hyperelliptic Curves

Affine equation of hyperelliptic curve of genus g (with $\mathbb{F}_q$-rational Weierstraß-point at infinity)

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

$h(x), f(x) \in \mathbb{F}_q[x]$, f monic, $\deg f = 2g + 1, \deg h \leq g$
non singular, i. e. not both partial derivatives

$$(2y + h(x) \text{ and } h'(x)y - f'(x))$$

vanish in any in $(a, b) \in C/\overline{\mathbb{F}_q}$

Examples

Concerning the arithmetic properties one can consider elliptic curves as hyperelliptic curves, i. e.

$$y^2 + (a_1x + a_3)y = x^3 + a_2x^2 + a_4x + a_6$$

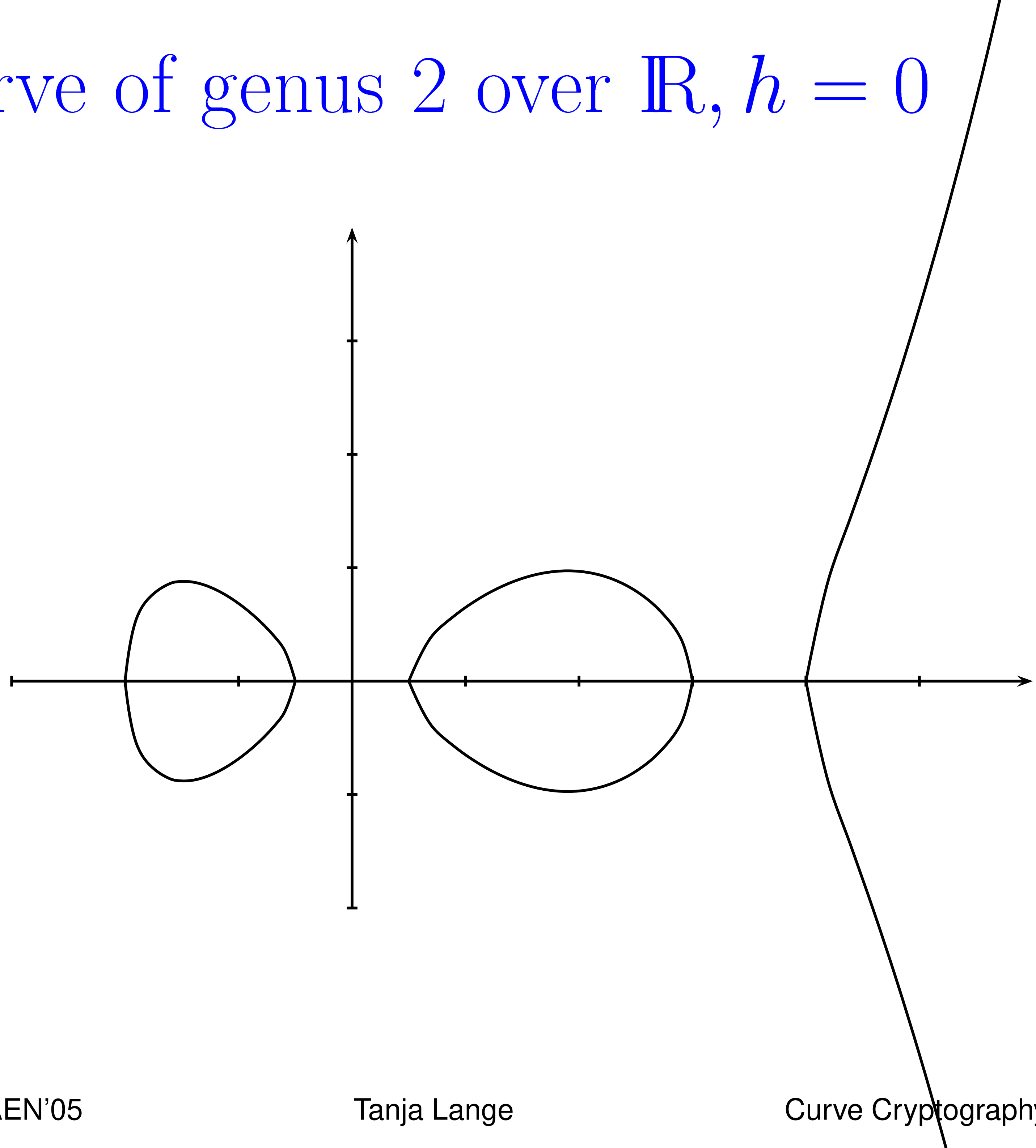
is considered as curve of genus one.

Curve of genus 2 over field of odd characteristic

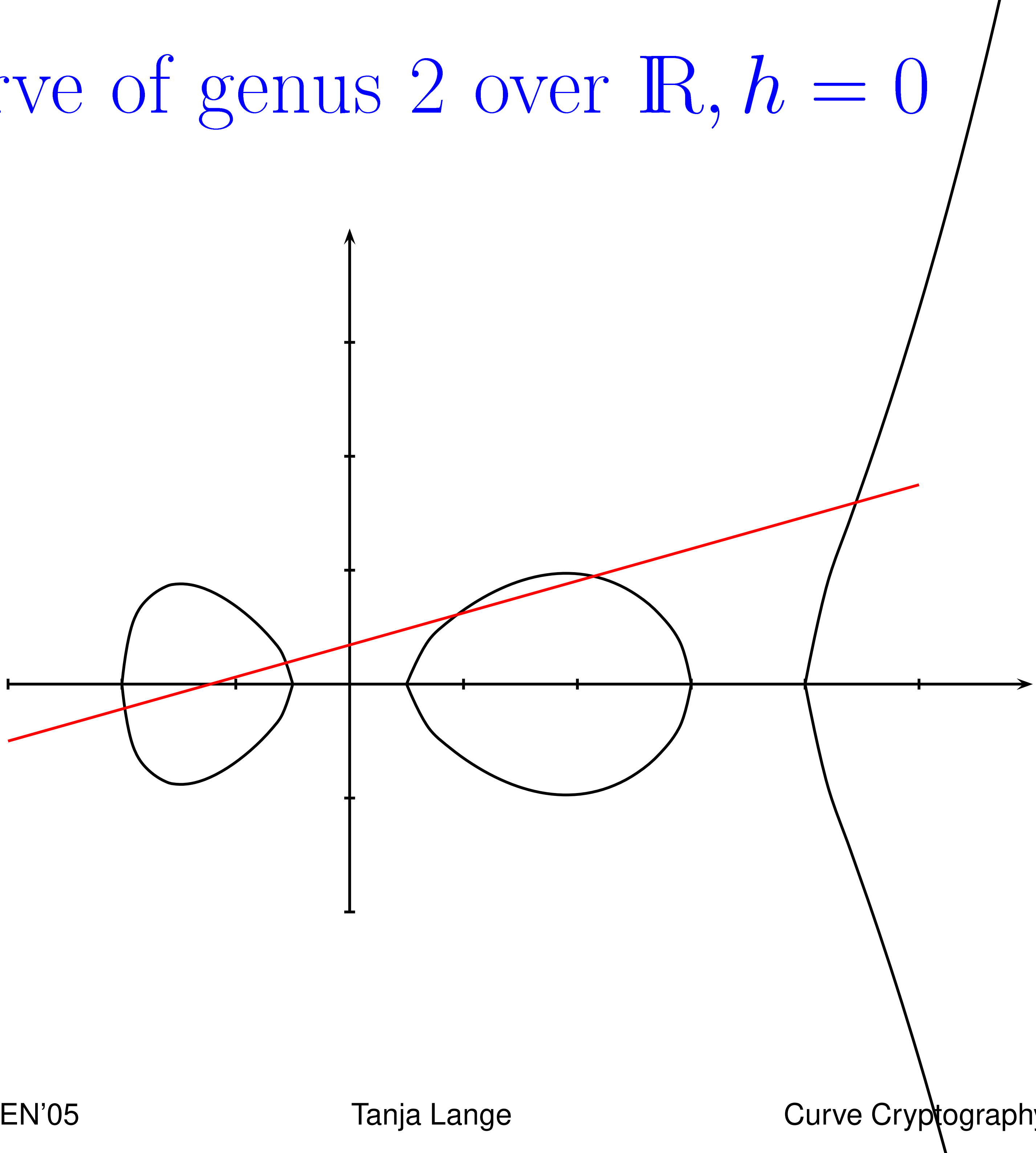
$$y^2 = x^5 + f_3x^3 + f_2x^2 + f_1x + f_0,$$

provided $f(x)$ has no multiple roots.

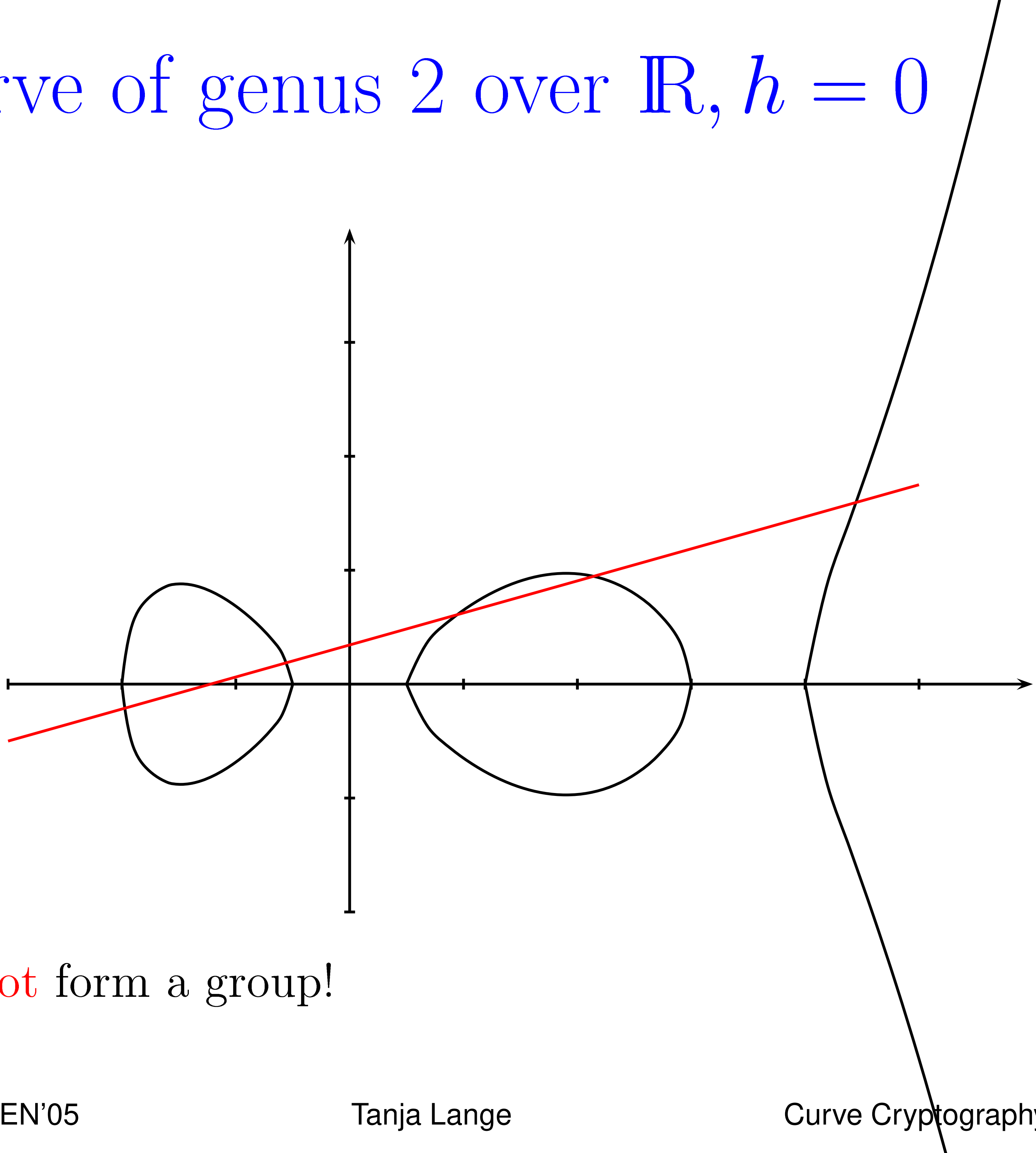
Curve of genus 2 over $\mathbb{R}$, $h = 0$



Curve of genus 2 over $\mathbb{R}, h = 0$



Curve of genus 2 over $\mathbb{R}$, $h = 0$



points do not form a group!

Group

Use Divisors, i. e. finite sums of points (elements of the free abelian group), of degree zero

$$\sum_{P \in C(\overline{\mathbb{F}_q})} n_P P, \; n_P \in \mathbb{Z}$$

with $\sum_{P \in C(\overline{\mathbb{F}_q})} n_P = 0$ a $n_P = 0$ for almost all P.
One associates the divisor

$$\operatorname{div}(F) = \sum_{P \in C(\overline{\mathbb{F}_q})} v_P(F) P$$

to the function $F \in \mathbb{F}_q(C)$, where v_P is a valuation in the point P. Such divisors are called principal divisors Princ_C. One can show that they have degree zero.

Divisor class group

Consider the factor group of the group of divisors of degree zero Div_C^0 modulo the principal divisors. This way one constructs the divisor class group of degree zero.

$$\mathrm{Pic}_C^0 = \mathrm{Div}_C^0 / \mathrm{Princ}_C.$$

The $\mathbb{F}_q$-rational group elements $\mathrm{Pic}_C^0(\mathbb{F}_q)$ are those which remain fixed under $\mathrm{Gal}(\overline{\mathbb{F}_q}/\mathbb{F}_q)$.

This construction provides an isomorphism between $\mathrm{Pic}_C^0(\mathbb{F}_q)$ and the $\mathbb{F}_q$-rational points of the Jacobian of C.

Representation of group elements

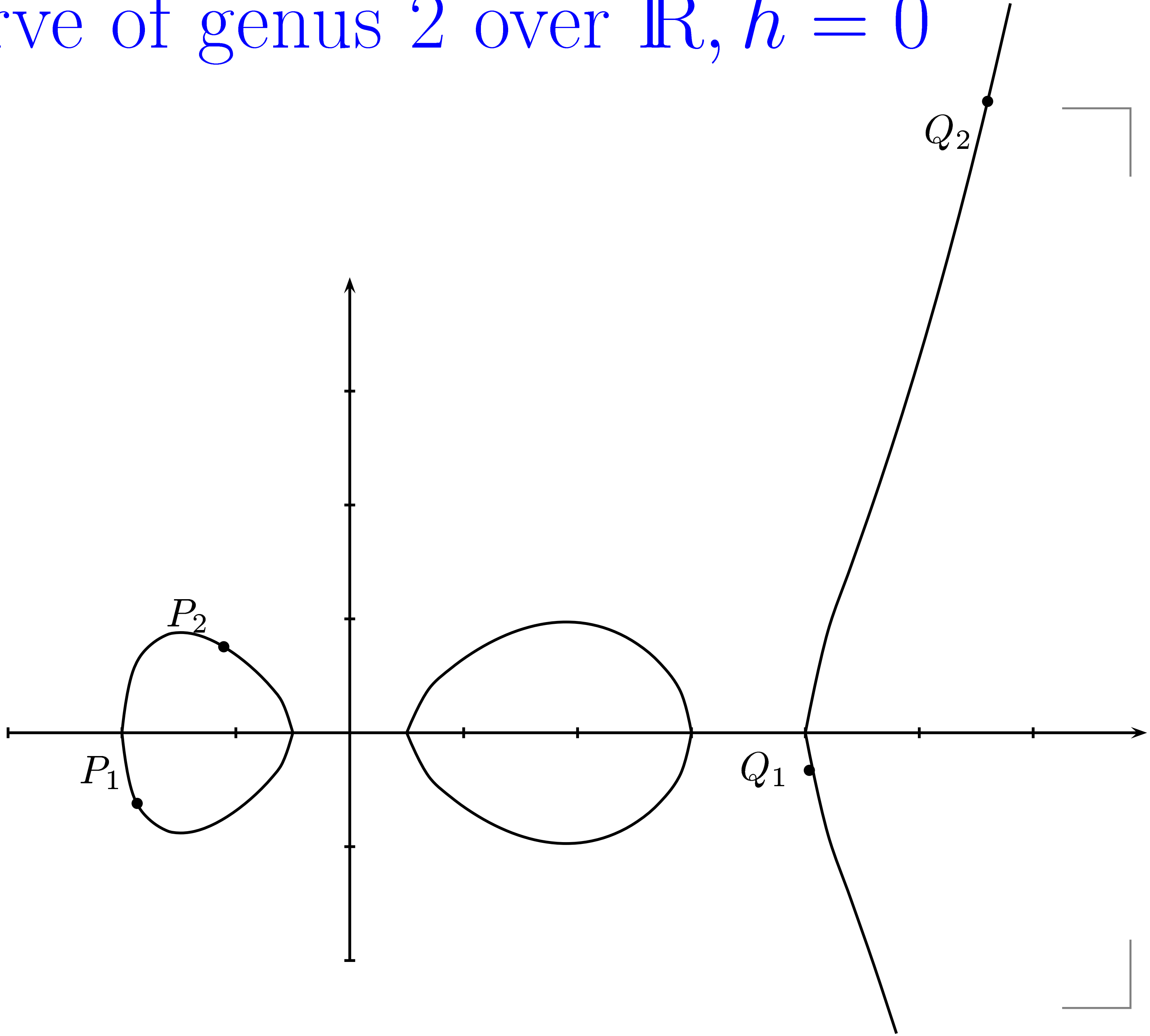
Riemann-Roch allows to find a unique reduced representation by means of a divisor of degree zero with $m \leq g$

$$\bar{D} = \sum_{\substack{i=1 \\ P_i \in C(\overline{\mathbb{F}_q}) \setminus \{P_\infty\}}}^{m} P_i - mP_\infty$$

and $P_i \neq -P_j$ for $i \neq j$.

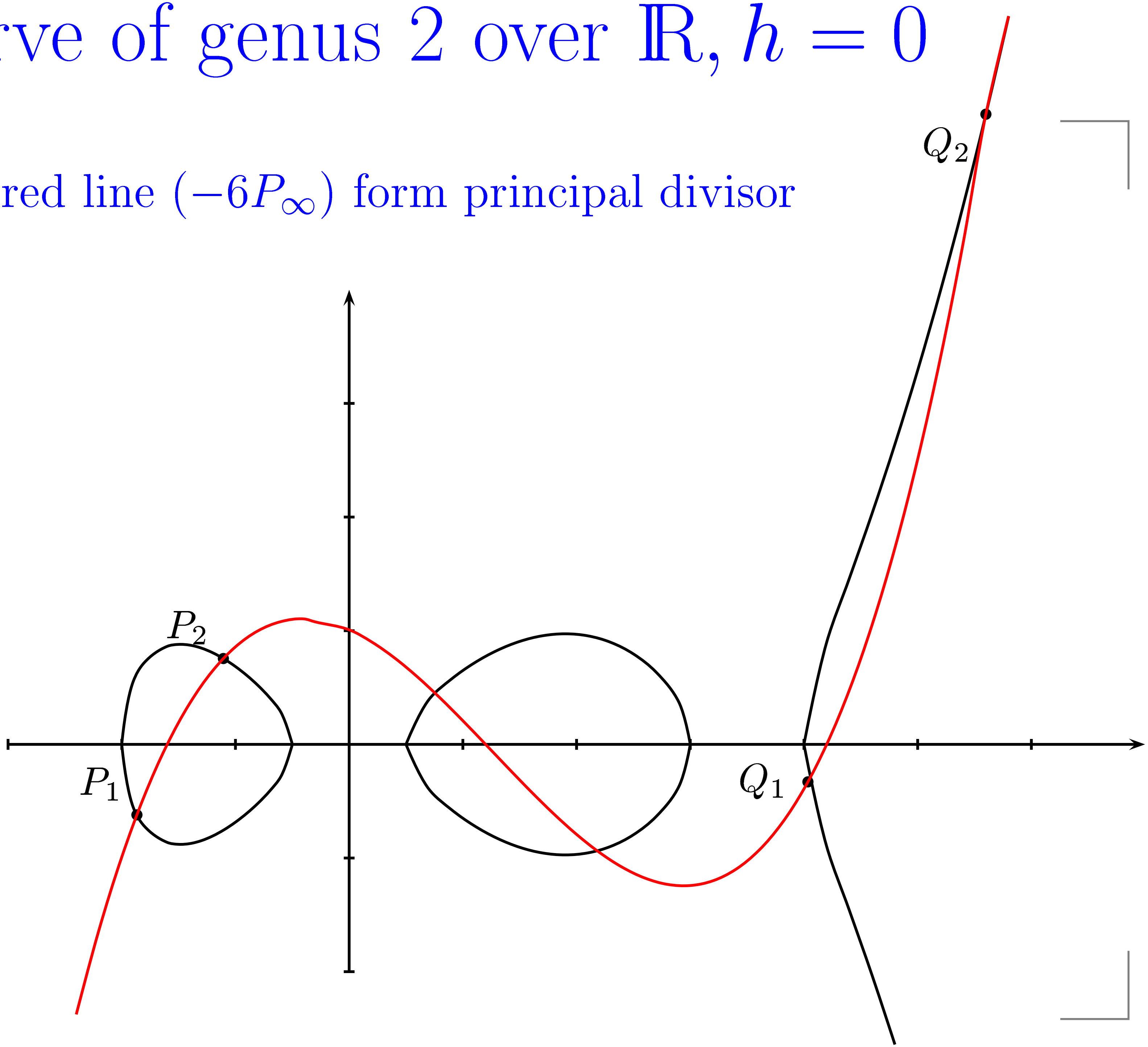
If $\bar{D}$ is defined over $\mathbb{F}_q$, the extension degree of the field of definition of the P_l is bounded.

Curve of genus 2 over $\mathbb{R}$, $h = 0$



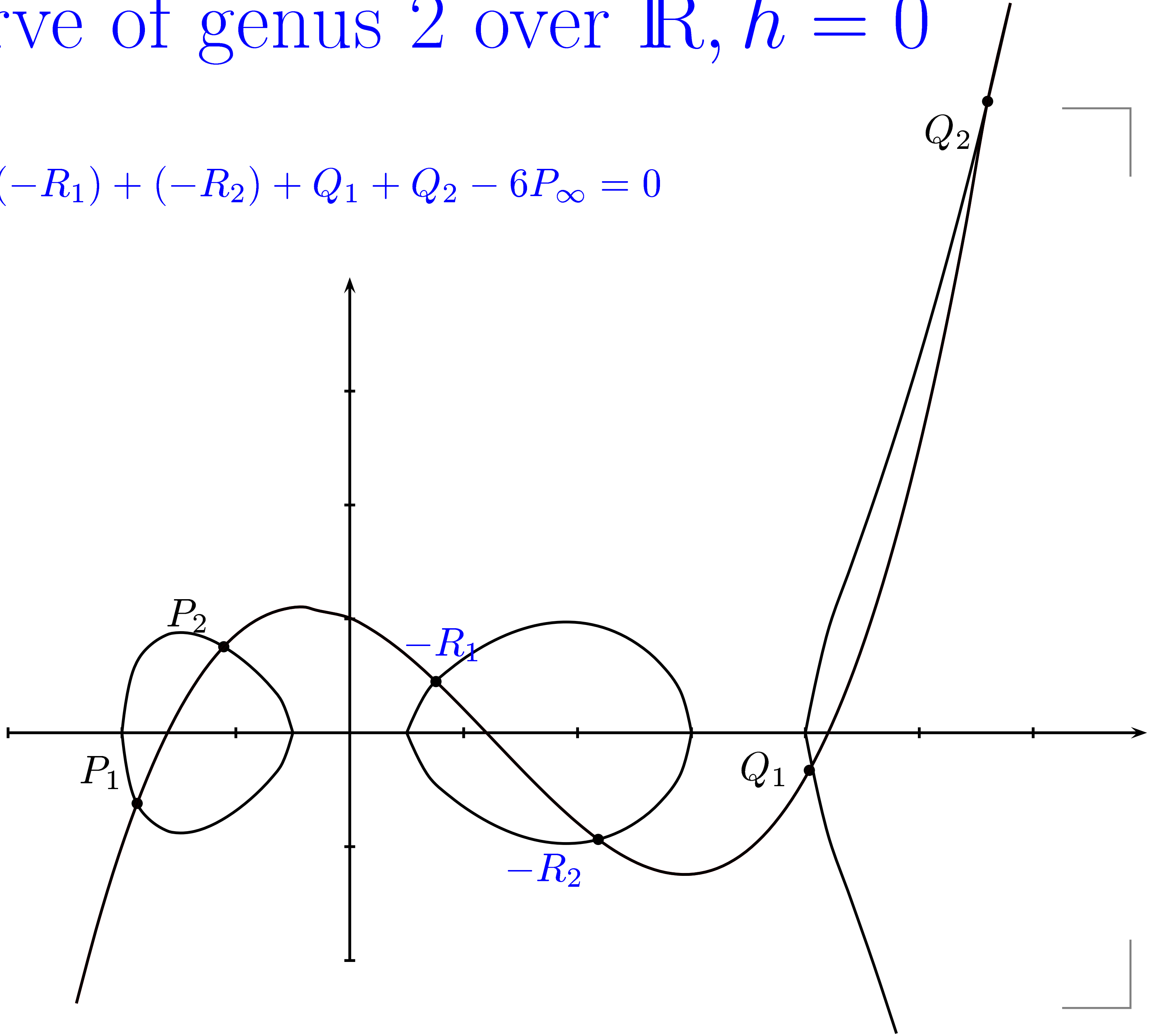
Curve of genus 2 over $\mathbb{R}$, $h = 0$

points on red line $(-6P_\infty)$ form principal divisor



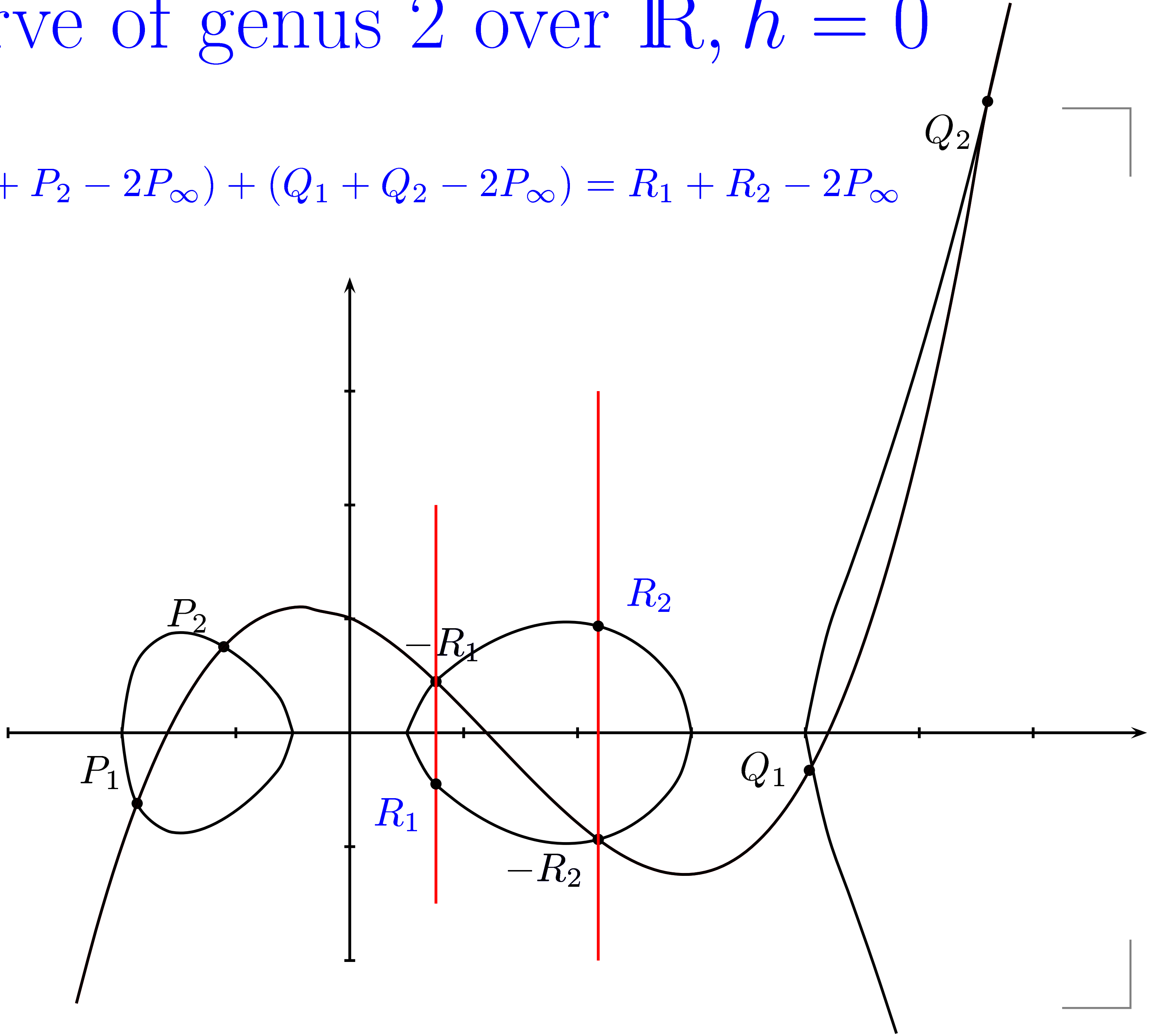
Curve of genus 2 over $\mathbb{R}$, $h = 0$

$$P_1 + P_2 + (-R_1) + (-R_2) + Q_1 + Q_2 - 6P_\infty = 0$$



Curve of genus 2 over $\mathbb{R}$, $h = 0$

$$(P_1 + P_2 - 2P_\infty) + (Q_1 + Q_2 - 2P_\infty) = R_1 + R_2 - 2P_\infty$$



Mumford Representation

Divisor class group is isomorphic to ideal class group $\mathbf{Cl}(C/\mathbb{F}_q)$ of $\mathbb{F}_q(x,y)/(C)$.
Elements D represented by two polynomials

$$D = [u(x), v(x)]; u, v \in \mathbb{F}_q[x],$$

u monic, $\deg v < \deg u \leq g, \quad u | v^2 + vh - f.$

$\Rightarrow$ compact representation.

Isomorphism is explicit via

$$P_i = (x_i, y_i) \Leftrightarrow u(x_i) = 0, v(x_i) = y_i \text{ with multiplicity}$$

Arithmetic uses "only" arithmetic on polynomials ... but is far less efficient than on elliptic curves (if applied directly).

Arithmetic on hyperelliptic curves

Composition & Reduction (Cantor/Koblitz)

`IN:` $D_1 = [u_1, v_1], D_2 = [u_2, v_2], C : y^2 + h(x)y = f(x)$
`OUT:` $D = [u, v]$ reduced with $D \sim D_1 + D_2$

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. $u = \frac{u_1 u_2}{d^2} \quad v = \frac{s_1 u_1 v_2 + s_2 u_2 v_1 + s_3(v_1 v_2 + f)}{d} \bmod u$
5. let $u' = \frac{f - vh - v^2}{u} \quad b' = (-h - v) \bmod u'$
6. if $\deg u' > g$ put $u := u', v := v'$ goto step 5
7. make u monic

Arithmetic in $\mathbf{Cl}(C/\mathbb{F}_q)$

2000	Harley (odd char.)	$\Rightarrow$ 2 inversions
2001	L. (arbitrary char.)	
2001	Matsuo, Chao, Tsujii (faster)	
2002	Miyamoto, Doi, Matsuo, Chao, Tsujii	$\Rightarrow$ 1 inv.
2002	Takahashi	
2002	L. (arbitrary char.)	
2002	Sugizaki, Matsuo, Chao, Tsujii (even char.)	

genus 3

2002	Kuroki, Gonda, Matsuo, Chao, Tsujii	$\Rightarrow$ 1 inv.
2002	Pelzl	
2003	Guyot & Patankar	

Addition, $g = 2$

Addition, $\deg u_1 = \deg u_2 = 2$	
Input	$[u_1, v_1], [u_2, v_2], u_i = x^2 + u_{i1}x + u_{i0}, v_i = v_{i1}x + v_{i0}$
Output	$[u', v'] = [u_1, v_1] + [u_2, v_2]$

Step	Expression	Operations
1	compute resultant r of u_1, u_2: $z_1 = u_{11} - u_{21},\ z_2 = u_{20} - u_{10},\ z_3 = u_{11}z_1 + z_2;$ $r = z_2 z_3 + z_1^2 u_{10};$	1S, 3M
2	compute almost inverse of u_2 modulo u_1 ($inv = r/u_2 \bmod u_1$): $inv_1 = z_1,\ inv_0 = z_3;$	
3	compute $s' = rs \equiv (v_1 - v_2)inv \bmod u_1$: $w_0 = v_{10} - v_{20},\ w_1 = v_{11} - v_{21},\ w_2 = inv_0 w_0,\ w_3 = inv_1 w_1;$ $s'_1 = (inv_0 + inv_1)(w_0 + w_1) - w_2 - w_3(1 + u_{11}),\ s'_0 = w_2 - u_{10}w_3;$ if $s'_1 = 0$ see below	5M
4	compute $s'' = x + s_0/s_1 = x + s'_0/s'_1$ and s_1: $w_1 = (rs'_1)^{-1}(= 1/r^2 s_1),\ w_2 = rw_1(= 1/s'_1),\ w_3 = s'^2_1 w_1(= s_1);$ $w_4 = rw_2(= 1/s_1),\ w_5 = w_4^2,\ s''_0 = s'_0 w_2;$	I, 2S, 5M
5	compute $l' = s''u_2 = x^3 + l'_2x^2 + l'_1x + l'_0$: $l'_2 = u_{21} + s''_0,\ l'_1 = u_{21}s''_0 + u_{20},\ l'_0 = u_{20}s''_0$	2M
6	compute $u' = (s(l + h + 2v_2) - k)/u_1 = x^2 + u'_1x + u'_0$: $u'_0 = (s''_0 - u_{11})(s''_0 - z_1 + h_2w_4) - u_{10} + l'_1 + (h_1 + 2v_{21})w_4 + (2u_{21} + z_1 - f_4)w_5;$ $u'_1 = 2s''_0 - z_1 + h_2w_4 - w_5;$	3M
7	compute $v' \equiv -h - (l + v_2) \bmod u' = v'_1x + v'_0$: $w_1 = l'_2 - u'_1,\ w_2 = u'_1w_1 + u'_0 - l'_1,\ v'_1 = w_2w_3 - v_{21} - h_1 + h_2u'_1;$ $w_2 = u'_0w_1 - l'_0,\ v'_0 = w_2w_3 - v_{20} - h_0 + h_2u'_0;$	4M
total		I, 3S, 22M

Different Coordinates $g = 2$, odd q

affine $\mathcal{A}$:

$$D \sim [x^2 + u_1 x + u_0, v_1 x + v_0] \sim [u_1, u_0, v_1, v_0]$$

projective $\mathcal{P}$: Miyamoto, Doi, Matsuo, Chao, and Tsujii; L.; all 2002

$$D \sim [U_1, U_0, V_1, V_0, Z] \sim [U_1/Z, U_0/Z, V_1/Z, V_0/Z]$$

new $\mathcal{N}$: L. 2002, contains also mixed coordinates

$$D \sim [U_1, U_1, V_1, V_0, Z_1, Z_2] \sim [U_1/Z_1^2, U_0/Z_1^2, V_1/(Z_1^3 Z_2), V_0/(Z_1^3 Z_2)]$$

$\mathcal{P}$ and $\mathcal{N}$: no inversions

Comparison - odd characteristic

Doubling		Addition	
operation	costs	operation	costs
$2\mathcal{N} = \mathcal{P}$	7S, 38M	$\mathcal{N} + \mathcal{N} = \mathcal{P}$	7S, 51M
$2\mathcal{P} = \mathcal{P}$	6S, 38M	$\mathcal{N} + \mathcal{P} = \mathcal{P}$	4S, 51M
$2\mathcal{N} = \mathcal{N}$	7S, 34M	$\mathcal{N} + \mathcal{N} = \mathcal{N}$	7S, 47M
$2\mathcal{P} = \mathcal{N}$	6S, 34M	$\mathcal{N} + \mathcal{P} = \mathcal{N}$	4S, 48M
		$\mathcal{P} + \mathcal{P} = \mathcal{P}$	4S, 47M
		$\mathcal{P} + \mathcal{P} = \mathcal{N}$	4S, 44M
		$\mathcal{A} + \mathcal{N} = \mathcal{P}$	5S, 40M
		$\mathcal{A} + \mathcal{P} = \mathcal{P}$	3S, 40M
		$\mathcal{A} + \mathcal{N} = \mathcal{N}$	5S, 36M
		$\mathcal{A} + \mathcal{P} = \mathcal{N}$	3S, 37M
$2\mathcal{A} = \mathcal{A}$	1I, 5S, 22M	$\mathcal{A} + \mathcal{A} = \mathcal{A}$	1I, 3S, 22M

Implementations and Comparison

- software: for $\log_2 p \sim 40 - 60$ inversions cheap,
  use $\mathcal{A}$ for $g = 1, 2$.
  Comparison depends on I/M genus
  Paper by Roberto Avanzi (Bochum) at CHES '04,
  (tailored implementation for the small p used here)
  shows:

  small advantage of EC over $g = 2$, genus 3 too slow.

- hardware: usually expensive inversions,
  compute multiple of affine point via $2\mathcal{N} = \mathcal{N}, \mathcal{A} + \mathcal{N} = \mathcal{N}$,
  compute multiple of non-normalized point via
  $2\mathcal{N} = \mathcal{N}, \mathcal{N} + \mathcal{P} = \mathcal{N}$.
  - First implementation by Kim Nguyen at ECC'02
  - Many results at RUB, ... close to break even of ECC and $g = 2$.

Field of even characteristic

Curves over $\mathbb{F}_{2^n}$

$$C : y^2 + (h_2x^2 + h_1x + h_0)y = f(x), \deg h \geq 1.$$

Above formulas hold for general equation. Get higher efficiency by specializing to type of h.

If $\deg h = 1$ the curve has 2-rank 1, which is a special property but does not weaken the system up to our present knowledge. For odd n one can make h monic.

If $\deg h = 2$ and h has a root then one can achieve $h(x) = x^2 + h_1x$, otherwise $h(x) = x^2 + h_1x + h_0$. In the former case the group order is divisible by 4, in the latter by 2.

Some more isomorphic transformations are possible.

Joint work with Marc Stevens

presented at SAC 2004.

Overview

	$h = X$	$h = h_1 X$, h_1^{-1} small	$h = h_1 X$	$h = X^2 + h_1 X$, h_1^{-1} small
f_4 small	I, 6S, 5M	I, 5S, 7M	I, 5S, 9M	I, 6S, 10M
f_4 not small	n. a.	n. a.	n. a.	I, 6S, 12M

	$h = X^2 + h_1 X$	$h = X^2 + h_1 X + h_0$	all
f_4 small	I,5S, 15M	I, 6S, 21M	I, 5S, 22M
f_4 not small	I, 5S, 17M	n. a.	,n. a.

Inversion Free Coordinates – $h = x$

$\mathcal{P}$: projective $u_i = U_i/Z, v_i = V_i/Z$

$\mathcal{N}$: new $u_i = U_i/Z_1^2, v_i = V_i/Z_1^3 Z_2$

$\mathcal{R}$: recent $u_i = U_i/Z, v_i = V_i/Z^2$

Doubling		Addition	
Operation	Costs	Operation	Costs
$2\mathcal{N} = \mathcal{N}$	$28\text{M} + 5\text{S}$	$\mathcal{R} + \mathcal{R} = \mathcal{R}$	$49\text{M} + 8\text{S}$
$2\mathcal{P} = \mathcal{P}$	$22\text{M} + 6\text{S}$	$\mathcal{P} + \mathcal{P} = \mathcal{P}$	$49\text{M} + 4\text{S}$
$2\mathcal{R} = \mathcal{R}$	$20\text{M} + 8\text{S}$	$\mathcal{N} + \mathcal{N} = \mathcal{N}$	$42\text{M} + 6\text{S}$
—	—	$\mathcal{A} + \mathcal{R} = \mathcal{R}$	$42\text{M} + 7\text{S}$
—	—	$\mathcal{A} + \mathcal{P} = \mathcal{P}$	$39\text{M} + 4\text{S}$
—	—	$\mathcal{A} + \mathcal{N} = \mathcal{N}$	$36\text{M} + 6\text{S}$
$2\mathcal{A} = \mathcal{A}$	$\text{I} + 5\text{M} + 6\text{S}$	$\mathcal{A} + \mathcal{A} = \mathcal{A}$	$\text{I} + 21\text{M} + 3\text{S}$

Side-Channel Attacks

Side-Channels

Attacker can measure

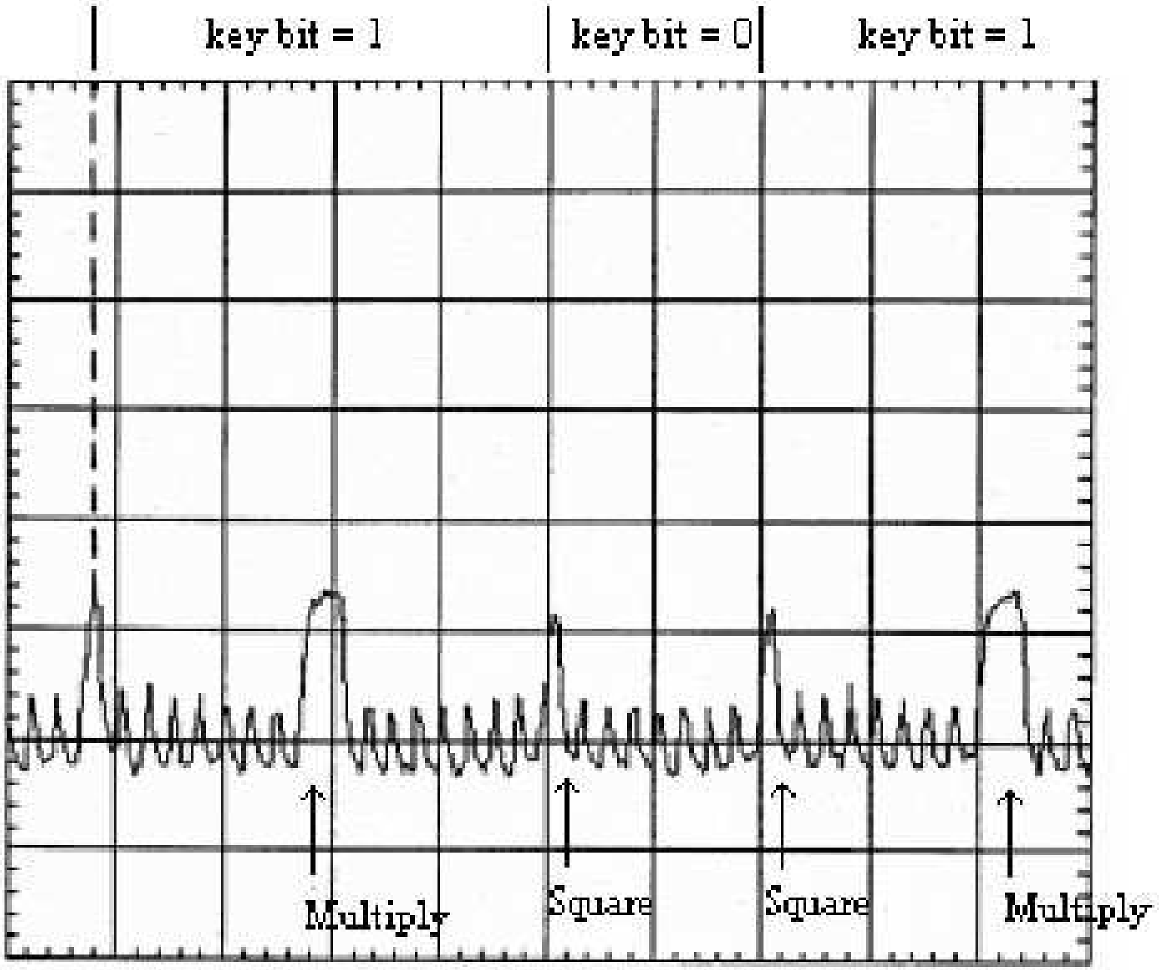
- Time to perform operations,
- Power consumption during operations,
- Electro-magnetic radiation during computation,
- Noise produced during computation.

Obviously, addition is cheaper than multiplication
$\Rightarrow$ needs more clock cycles, different characteristics of power trace.

Attacker might be able to reconstruct sequence of operations (power & EM) or at least learn how many of each kind were performed (timing).

Consequences I

If sequence of operations depends on the secret key and this is directly translated to the observed data, one can reconstruct the key



⇒ Simple Side-Channel Analysis (SSCA) (often SPA= Simple Power Analysis).

(e. g. in binary square-and-multiply one has

S M S S M ∼ $(1101)_2 = 13$).

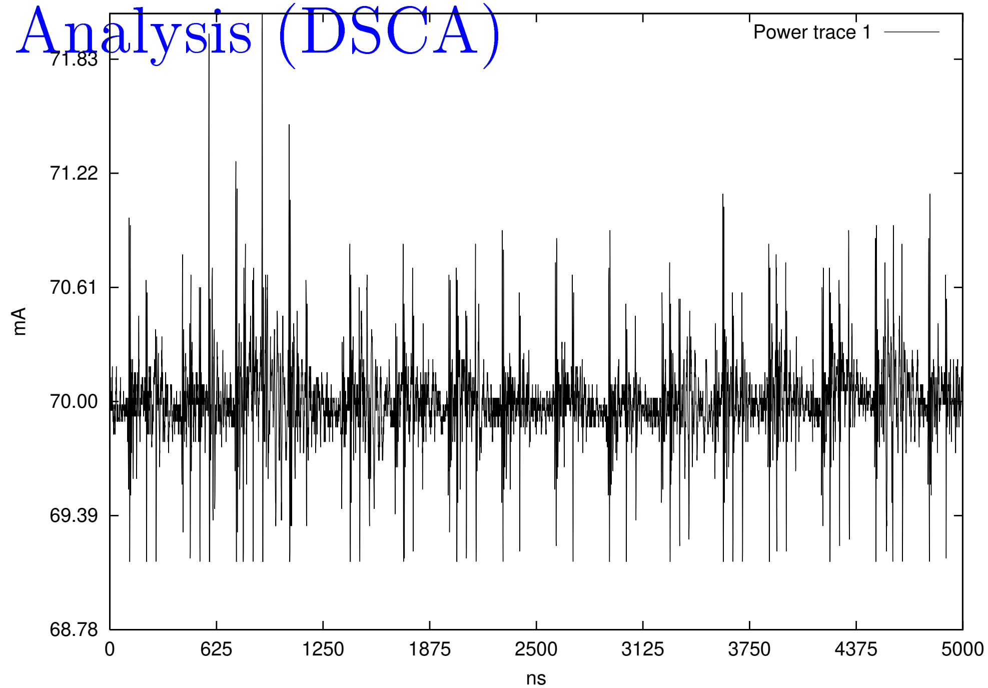
Consequences II

For algorithm shielded against SCA (e. g. by using uniform group operations – topic of these talks) the performance might still depend on the internal representation of the intermediate group elements. Averaging over different inputs reveals intermediate results

$\Rightarrow$ secret key is recovered bit-by-bit from most-significant bit.

$\Rightarrow$ Differential Side-Channel Analysis (DSCA)

(often DPA= Differential Power Analysis)



Scalar Multiplication – Double-and-Add

```
IN: P ∈ E(F_q), n ∈ Z, n = Σ_{i=0}^{l} n_i 2^i
OUT: Q = nP
```

IN: $P \in E(\mathbb{F}_q)$, $n \in \mathbb{Z}$, $n = \sum_{i=0}^{l} n_i 2^i$
OUT: $Q = nP$

1. $Q = P$
2. for $i = l - 1$ down to 0 do
3. $\quad Q = 2Q$
4. $\quad$ if $(n_i = 1)$ then $Q = Q + P$
5. output Q

If ADD $\neq$ DBL one can easily determine n from the sequence of ADD and DBL:

DBL DBL ADD DBL ADD DBL DBL $\Leftrightarrow (101100)_2 = 44$

Countermeasures against SPA

Double-and-always-Add

IN: $P \in E(\mathbb{F}_q)$, $n \in \mathbb{Z}$, $n = \sum_{i=0}^{l} n_i 2^i$
OUT: $Q = nP$

1. $Q = P$
2. for $i = l - 1$ down to 0 do
3. $\quad Q = 2Q$
4. $\quad$ if $n_i == 1$ then $Q = Q + P$
   else $R = Q + P$ //dummy operation
5. output Q

If an active adversary is allowed, the dummy operations might be detected (fault attacks)

Side-Channel Atomicity

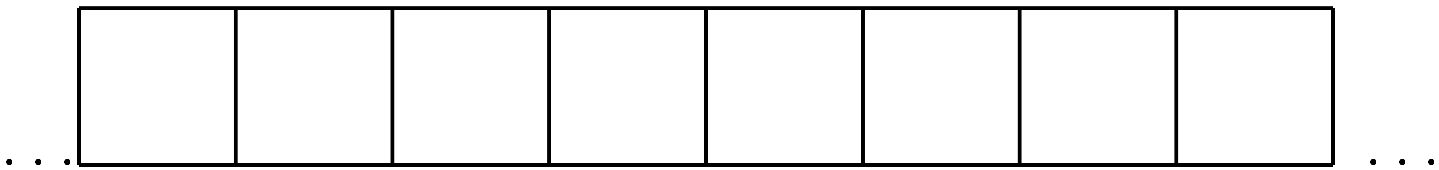
Chevallier-Mames, Ciet, and Joye 2003:
add only cheap dummy operations
$\Rightarrow$ build group operation from identical blocks.

Each block consists of:

1 multiplication, 1 addition, 1 negation, 1 addition;

fill with dummy operations
(only addition and negation $\Rightarrow$ cheap!).

ADD $(\mathcal{A} + \mathcal{P})$ needs 11 blocks
DBL $(2\mathcal{P})$ needs 10 blocks
(same as number of multiplications).



Can be used in parallel/pipelining (Mishra 2004)

Side-Channel Atomicity

Chevallier-Mames, Ciet, and Joye 2003:
add only cheap dummy operations
$\Rightarrow$ build group operation from identical blocks.

Each block consists of:

1 multiplication, 1 addition, 1 negation, 1 addition;

fill with dummy operations
(only addition and negation $\Rightarrow$ cheap!).

ADD $(\mathcal{A} + \mathcal{P})$ needs 11 blocks
DBL $(2\mathcal{P})$ needs 10 blocks
(same as number of multiplications).

...	ADD$_9$	ADD$_{10}$	ADD$_{11}$	DBL$_1$	DBL$_2$	DBL$_3$	DBL$_4$	DBL$_5$	...

Can be used in parallel/pipelining (Mishra 2004)

Uniform Group Operations

$$\left.\begin{array}{l}\text{Liardet and Smart CHES 2001}\\ \text{Billet and Joye AAECC 2003}\end{array}\right\} \Rightarrow \text{Jacobi-Model}$$

Joye and Quisquater $\Rightarrow$ Hessian Curve

Proposed formulae still might involve inversions, can be generalized to projective coordinates.

(Some special cases may occur, but with very low probability.)

Montgomery Ladder (Arbitrary Group)

Idea: Make used addition per round.

Consider the intermediate results (i is decreasing).

$Q_i = \sum_{j=i}^{l} n_j 2^{j-i} P$, put $R_i = Q_i + P$, then

$$Q_i = 2Q_{i+1} + n_i P = Q_{i+1} + R_{i+1} + n_i P - P = 2R_{i+1} + n_i P - 2P.$$

This implies

$$(Q_i, R_i) = \begin{cases} (2Q_{i+1}, Q_{i+1} + R_{i+1}) & \text{if } n_i = 0 \\ (Q_{i+1} + R_{i+1}, 2R_{i+1}) & \text{if } n_i = 1 \end{cases}$$

$$13 = (1101)_2 \sim \begin{array}{rcl} (Q_3, R_3) & = & (P, 2P) \\ (Q_2, R_2) & = & (3P, 4P) \\ (Q_1, R_1) & = & (6P, 7P) \\ (Q_0, R_0) & = & (13P, 14P) \end{array}$$

Montgomery Arithmetic for EC

Advantages:

- Needs only x coordinate (not y)
  $\Rightarrow$ lower storage requirement compared to full Montgomery ladder.
- Projective version (no inversion) extremely efficient for special curves (uses x and z) $\Rightarrow$ Montgomery curves (curves were proposed for ECM factoring due to fast group operation).
- Shielded against SPA (Brier/Joye for arbitrary curves), slower than for general curves but faster than full doubling and addition and less storage needed.

Countermeasures for HECC (SPA)

- Double-and-Always-Add (folklore)
- Atomicity: Lange and Mishra 2004
- Unified group law: still open
- Montgomery arithmetic: Lange, Duquesne (both ANTS 2004),
  Gaudry still unpublished but much quicker

Countermeasures against DPA

DPA I

Observe scalar multiplication $Q_j = kP_j$ for fixed k and different (known) P_j
(e.g. signature generation)

Let $k = \sum_{i=0}^{l} k_i 2^i$.
If $k_{l-1} = 0$ we observe $DBL\ DBL$,
if $k_{l-1} = 1$ we observe $DBL\ ADD\ DBL$,

Take some property, e.g. bits of the internal representation, that should have influence on speed/powerconsumption/EM radiation in computation of DBL (e.g. last bits)

DPA II

$$\mathcal{S}_0 = \{P_j | \text{ low consumption in } 2(2P_j)\}$$

$$\mathcal{S}_1 = \{P_j | \text{ high consumption in } 2(2P_j)\}$$

(compute $2(2P_i)$ on own device to get sets).

Take difference of power traces of $\mathcal{S}_0$ and $\mathcal{S}_1$.
Correlation visible by pike in the difference $\Rightarrow$ guess of $k_{l-1} = 0$ is correct. Otherwise $k_{l-1} = 1$.

$$\mathcal{S}_0 = \{P_j | \text{ low consumption in } 2(2(2P_j + k_{l-1}P_i))\}$$

$$\mathcal{S}_1 = \{P_j | \text{ high consumption in } 2(2(2P_j + k_{l-1}P_i))\}$$

$$\vdots$$

Countermeasure – Main Idea

Differential methods need to simulate group operations on known input. Guess bits one by one and test for correlation between groupings depending on internal representation.
$\Rightarrow$ introduce randomness!

- Choose different n' in equivalence class of n, e. g. use $n' = n + k\ell$, for group order ℓ.
- Modify group representation, e. g. use isomorphic curve.
- Modify element representation.

We present case of elliptic curves, Generalizations to HECC are collected in Avanzi's CHES 2003 paper, considers also Goubin type attacks.

Randomized Coordinate

(Coron's third countermeasure)
Let $P_j = (x_j, y_j)$.
Start computation with

$$P_j = [X_j : Y_j : Z_j] = [rx_j : ry_j : r] \sim [x_j : y_1 : 1]$$

for random r.
Same idea works for Jacobian and Lopez-Dahab coordinates.
Make sure to transfer back to affine $(x'_j, y'_j) = kP_j$ after operation (Naccache, Smart, Stern Eurocrypt 2004)

Add Point

Precompute short list

$$\{(R_1, kR_1), (R_2, kR_2), \ldots, (R_n, kR_n)\}$$

use

$$kP_j = k(P_j + R_l) \ominus kR_l$$

for random l

Use Isomorphism

Assume that there is a family Φ of easy to compute and to invert isomorphisms $\phi \in \Phi$ from C to $\tilde{C}$.
Then the following commutative diagram inserts randomness to the scalar multiplication.

$$\begin{tikzcd}
P \in C(\mathbb{F}_q) \arrow[r, bend left, "\text{multiplication by } n"] \arrow[d, bend right, "\phi"'] & C(\mathbb{F}_q) \ni n \cdot P \\
\phi(P) \in \tilde{C}(\mathbb{F}_q) \arrow[r, bend right, "\text{multiplication by } n"'] & \tilde{C}(\mathbb{F}_q) \ni n \cdot \phi(P) \arrow[u, bend right, "\phi^{-1}"']
\end{tikzcd}$$

The End

The END

Thanks!