Side-channel attacks and countermeasures for curve based cryptography

Tanja Lange

Technische Universiteit Eindhoven

tanja@hyperelliptic.org

28.05.2007

Overview

- Elliptic curves
  - Definition and group law
  - Efficient implementations
- Simple side-channel attacks
  - Montgomery ladder
  - Side-channel atomicity
  - Unified group laws
  - Edwards coordinates
  - Comparison
- Countermeasures against DPA
- SCA on pairings

Overview

- Elliptic curves
  - Definition and group law
  - Efficient implementations
- Simple side-channel attacks
  - Montgomery ladder
  - Side-channel atomicity
  - Unified group laws
  - Edwards coordinates
  - Comparison
- Countermeasures against DPA
- SCA on pairings
- Edwards coordinates for speed

Elliptic curves

Elliptic curve

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

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

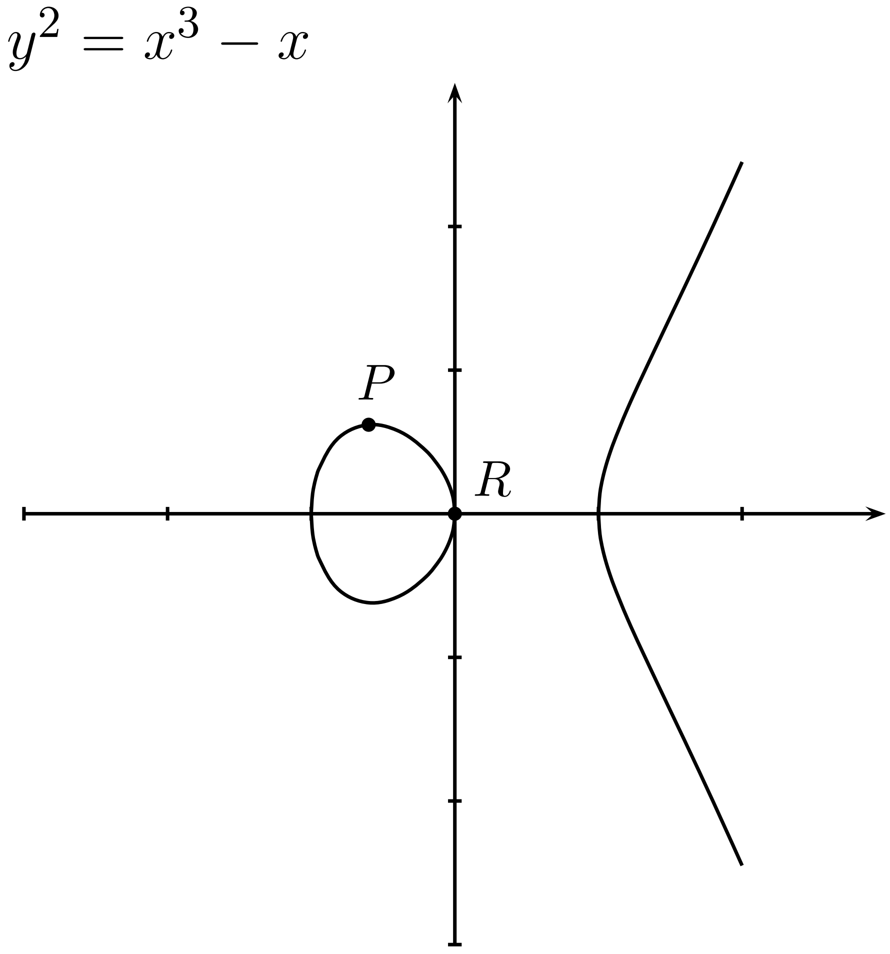
$$y^2 = f(x) \qquad\qquad q \text{ odd},$$

for

$$\begin{aligned} y^2 + xy &= x^3 + a_2 x^2 + a_6 \\ y^2 + y &= x^3 + a_4 x + a_6 \end{aligned} \qquad q = 2^r, \quad \begin{array}{l} \text{curve non-supersingular} \\ \text{curve supersingular} \end{array}$$

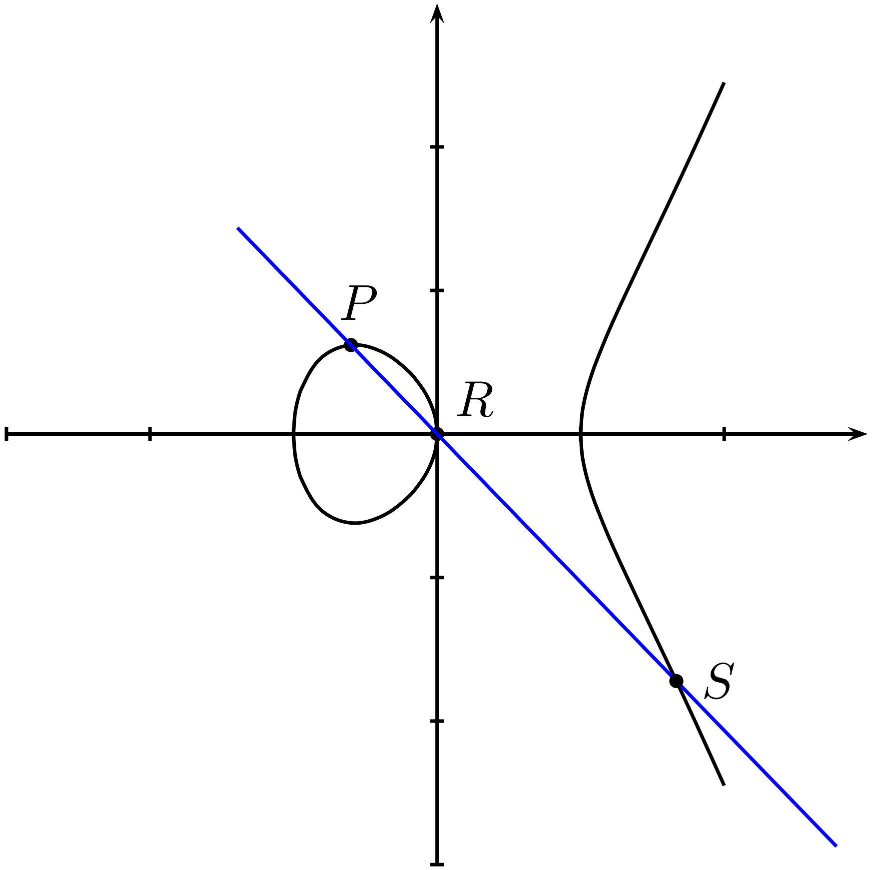
In this talk we consider only fields of odd characteristic and $\mathbb{R}$.

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



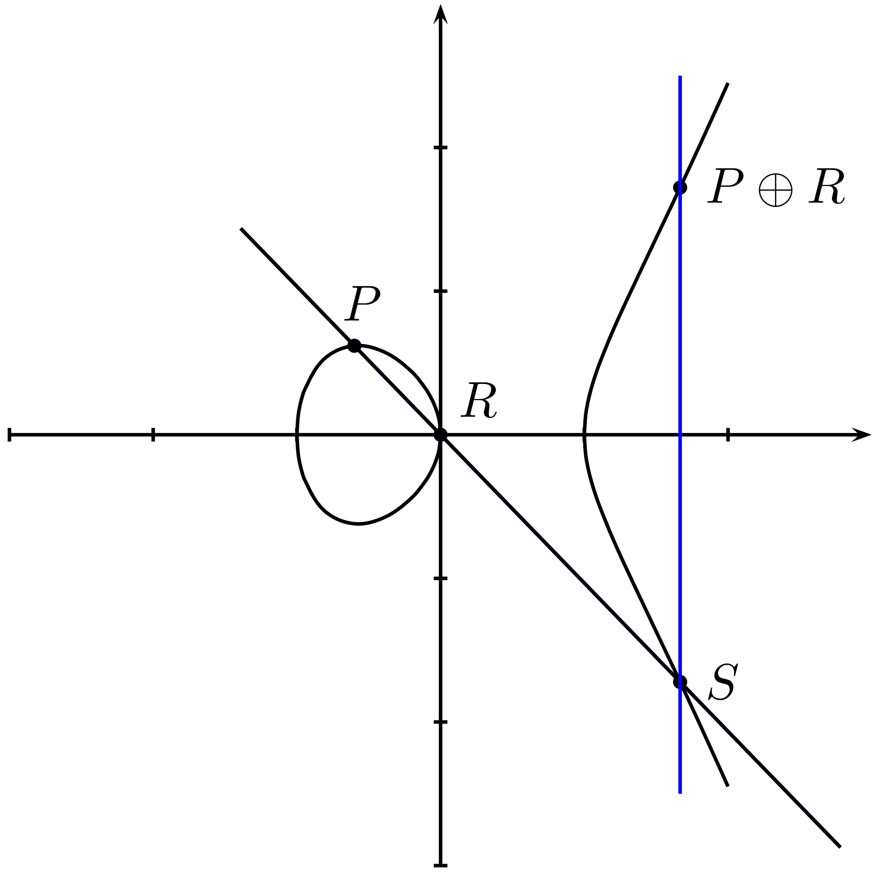
Group Law in $E(\mathbb{R}), h = 0$

$y^2 = x^3 - x$



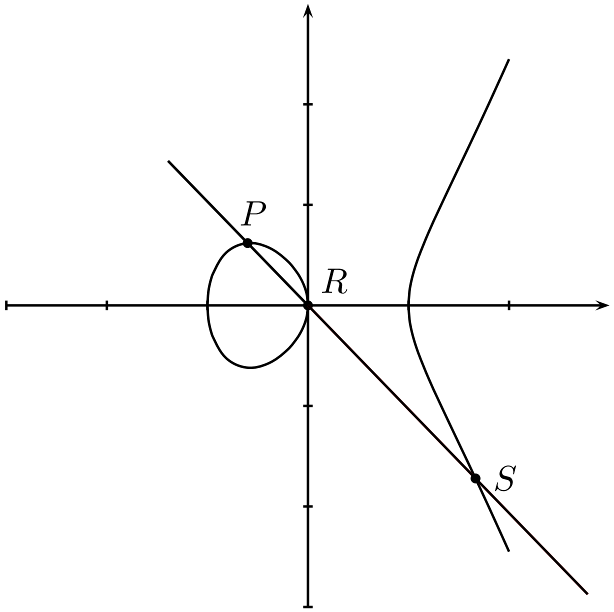
Group Law in $E(\mathbb{R}), h = 0$

$y^2 = x^3 - x$



Group Law (q odd)

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



Line $y = \lambda x + \mu$ has slope

$$\lambda = \frac{y_R - y_P}{x_R - x_P}.$$

Equating gives

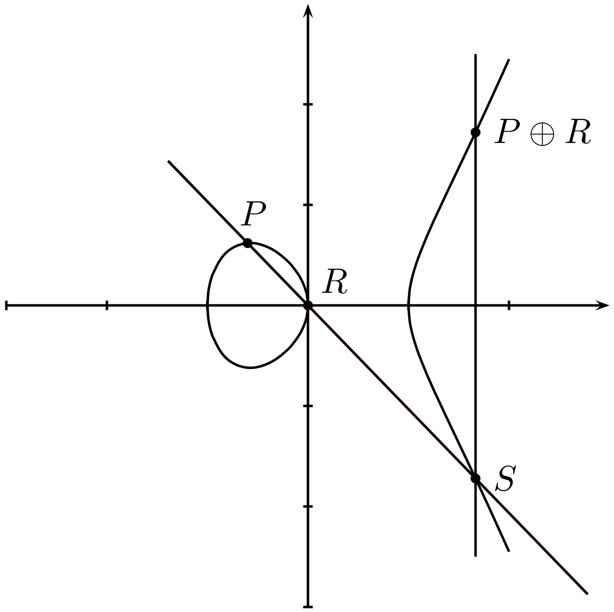
$$(\lambda x + \mu)^2 = x^3 + a_4 x + a_6.$$

This equation has 3 solutions, the x-coordinates of P, R and S, thus

$$\begin{aligned} (x - x_P)(x - x_R)(x - x_S) &= x^3 - \lambda^2 x^2 + (a_4 - 2\lambda\mu)x + a_6 - \mu^2 \\ x_S &= \lambda^2 - x_P - x_R \end{aligned}$$

Group Law (q odd)

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



Point P is on line, thus

$y_P = \lambda x_P + \mu$, i.e.
$\mu = y_P - \lambda x_P$,

and

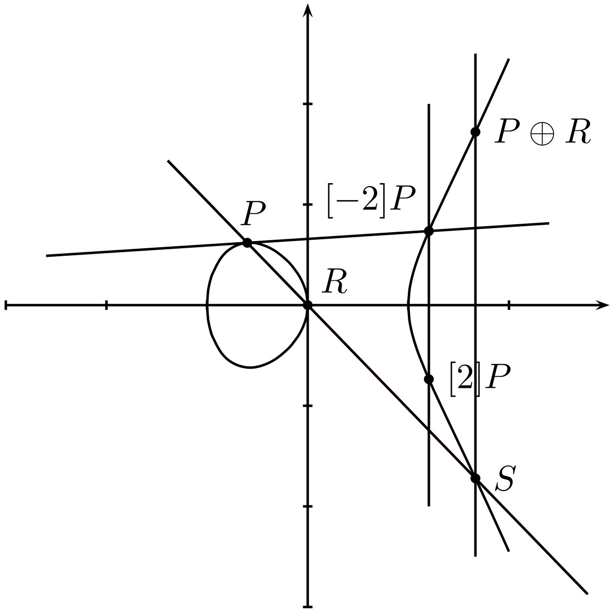
$$\begin{aligned} y_S &= \lambda x_S + \mu \\ &= \lambda x_S + y_P - \lambda x_P \\ &= \lambda(x_S - x_P) + y_P \end{aligned}$$

Point $P \oplus R$ has the same x-coordinate as S but negative y-coordinate:

$$x_{P \oplus R} = \lambda^2 - x_P - x_R, \quad y_{P \oplus R} = \lambda(x_P - x_{P \oplus R}) - y_P$$

Group Law (q odd)

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



In general, for $(x_P, y_P) \neq (x_R, -y_R)$:

$$\begin{aligned}(x_P, y_P) \oplus (x_R, y_R) &= \\ &= (x_{P\oplus R}, y_{P\oplus R}) = \\ &= (\lambda^2 - x_P - x_R, \lambda(x_P - x_{P\oplus R}) - y_P),\end{aligned}$$

where

$$\lambda = \begin{cases} (y_R - y_P)/(x_R - x_P) & \text{if } x_P \neq x_R, \\ (3x_P^2 + a_4)/(2y_P) & \text{else.} \end{cases}$$

$\Rightarrow$ Addition and Doubling need
1 I, 2M, 1S and 1 I, 2M, 2S, respectively
ADD and DBL differ by 1S!

Weierstraß equation

$$E : y^2 + \underbrace{(a_1x + a_3)}_{h(x)} y = \underbrace{x^3 + a_2x^2 + a_4x + a_6}_{f(x)},\ \ h, f \in \mathbb{F}_q[x].$$

- Negative of $P = (x_P, y_P)$ is given by $-P = (x_P, -y_P - h(x_P))$.
- $(x_P, y_P) \oplus (x_R, y_R) = (x_3, y_3) =$ $= (\lambda^2 + a_1\lambda - a_2 - x_P - x_R, \lambda(x_P - x_3) - y_P - a_1x_3 - a_3)$, where

$$\lambda = \begin{cases} (y_R - y_P)/(x_R - x_P) & \text{if } x_P \neq x_R, \\ \frac{3x_P^2 + 2a_2x_P + a_4 - a_1y_P}{2y_P + a_Px_P + a_3} & \text{else.} \end{cases}$$

Projective Coordinates

$P = (X_1 : Y_1 : Z_1)$, $Q = (X_2 : Y_2 : Z_2)$, $P \oplus Q = (X_3 : Y_3 : Z_3)$
on $E : Y^2Z = X^3 + a_4XZ^2 + a_6Z^3$; $(x, y) \sim (X/Z, Y/Z)$

Addition: $P \neq \pm Q$

$A = Y_2Z_1 - Y_1Z_2, B = X_2Z_1 - X_1Z_2,$
$C = A^2Z_1Z_2 - B^3 - 2B^2X_1Z_2$
$X_3 = BC, Z_3 = B^3Z_1Z_2$
$Y_3 = A(B^2X_1Z_2 - C) - B^3Y_1Z_2,$

Doubling $P = Q \neq -P$

$A = a_4Z_1^2 + 3X_1^2, B = Y_1Z_1,$
$C = X_1Y_1B, D = A^2 - 8C$
$X_3 = 2BD, Z_3 = 8B^3$.
$Y_3 = A(4C - D) - 8Y_1^2B^2$

- No inversion is needed – good for most implementations
- General ADD: 12M+2S
- DBL: 7M+5S
- Fast . . . but very different performance of ADD and DBL

Jacobian Coordinates

$P = (X_1 : Y_1 : Z_1)$, $Q = (X_2 : Y_2 : Z_2)$, $P \oplus Q = (X_3 : Y_3 : Z_3)$
on $Y^2 = X^3 + a_4 X Z^4 + a_6 Z^6$; $(x, y) \sim (X/Z^2, Y/Z^3)$

Addition: $P \neq \pm Q$

$A = X_1 Z_2^2, B = X_2 Z_1^2, C = Y_1 Z_2^3,$
$D = Y_2 Z_1^3, E = B - A, F = D - C$
$X_3 = 2(-E^3 - 2AE^2 + F^2)$
$Z_3 = E(Z_1 + Z_2)^2 - Z_1^2 - Z_2^2$
$Y_3 = 2(-CE^3 + F(AE^2 - X_3))$,

Doubling $P = Q \neq -P$

$A = Y_1^2, B = Z_1^2$
$C = 4X_1 A, D = 3X_1^2 + a_4 B^2$
$X_3 = -2C + D^2$
$Z_3 = (Y_1 + Z_1)^2 - A - B$
$Y_3 = -8A^2 + D(C - X_3)$.

- General ADD: 11M+5S
- mixed ADD ($\mathcal{J} + \mathcal{A} = \mathcal{J}$): 8M+3S
- DBL: 3M+7S (one M by a_4); for $a_4 = -3$: 3M+5S
- Even faster . . . even more different performance

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 ($\mathcal{J}$)	11M	5S	–	3M	7S	–
Chudnovsky Jacobian ($\mathcal{J}^C$)	10M	4S	–	4M	7S	–
modified Jacobian ($\mathcal{J}^m$)	12M	7S	–	4M	4S	–

Arithmetic

in the time of

Side-channel attacks

Side Channels

Attacker can measure

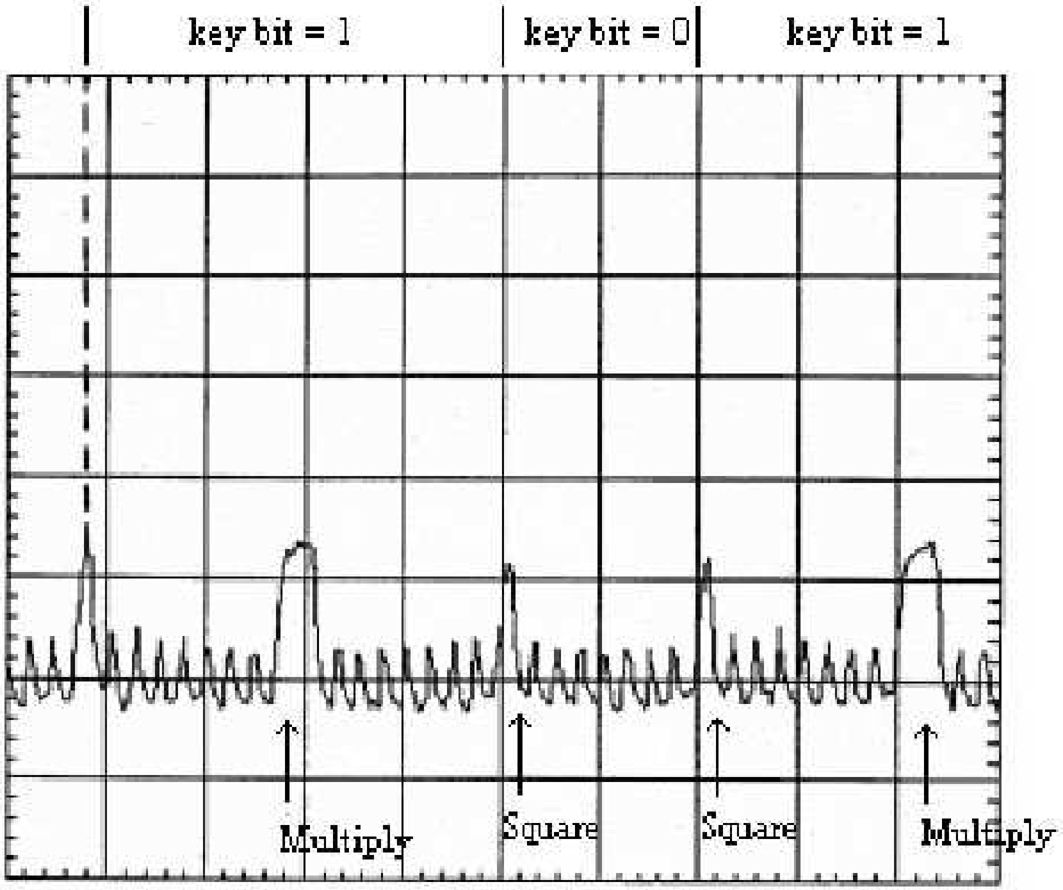
- Time to perform operations,
- Power consumption during operations,
- Electro-magnetic radiation during computation,
- Noise produced during computation.
- ...

Obviously, integer 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

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



$\Rightarrow$ Simple Side-Channel Analysis (SSCA) (often SPA= Simple Power Analysis).

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

S M S S M $\sim$ $(1101)_2 = 13$).

Double-and-always-Add

This is the obvious countermeasure . . .

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

OUT: $Q_0 = nP$

1. $Q_0 = P, Q_1 = [2]P$
2. for $i = l - 1$ down to 0 do
3. $\quad Q_0 = [2]Q_0$
4. $\quad Q_{1-n_i} = Q_{1-n_i} \oplus P$ dummy operation if $n_i = 1$
5. output Q_0

. . . but it is very inefficient.

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

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 $R_i = Q_i \oplus P$, then

$$Q_i = [2]Q_{i+1} \oplus [n_i]P = Q_{i+1} \oplus R_{i+1} \oplus n_i P \ominus P = [2]R_{i+1} \oplus n_i P \ominus [2]P.$$

This implies

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

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

Montgomery Form

Generalized to arbitrary multiples $[n]P = (X_n : Y_n : Z_n), [m]P = (X_m : Y_m : Z_m)$ with known difference $[m-n]P$ on

$$E_M : By^2 = x^3 + Ax^2 + x$$

Addition: $n \neq m$

$$\begin{aligned} X_{m+n} &= Z_{m-n}\big((X_m - Z_m)(X_n + Z_n) + (X_m + Z_m)(X_n - Z_n)\big)^2, \\ Z_{m+n} &= X_{m-n}\big((X_m - Z_m)(X_n + Z_n) - (X_m + Z_m)(X_n - Z_n)\big)^2 \end{aligned}$$

Doubling: $n = m$

$$\begin{aligned} 4X_nZ_n &= (X_n + Z_n)^2 - (X_n - Z_n)^2, \\ X_{2n} &= (X_n + Z_n)^2(X_n - Z_n)^2, \\ Z_{2n} &= 4X_nZ_n\big((X_n - Z_n)^2 + \big((A+2)/4\big)(4X_nZ_n)\big). \end{aligned}$$

An addition takes 4M and 2S whereas a doubling needs only 3M and 2S. Order is divisible by 4.

Montgomery Arithmetic for EC

- Needs only x coordinate (not y)
  $\Rightarrow$ lower storage requirement compared to full Montgomery ladder.
- Projective version (no inversion) extremely efficient for curves in Montgomery form $\Rightarrow$ Montgomery curves (curves were proposed for ECM factoring due to fast group operation).
- Starting with affine base point and using Montgomery ladder, so that $Z_{m-n} = Z_1 = 1$, leads to ADD for 3M+2S.
- Choose A so that $(A+2)/4$ is small, then DBL for only 2M+2S.
- Shielded against SPA (Brier/Joye for arbitrary curves), slower than for general curves but faster than full doubling and addition and less storage needed.

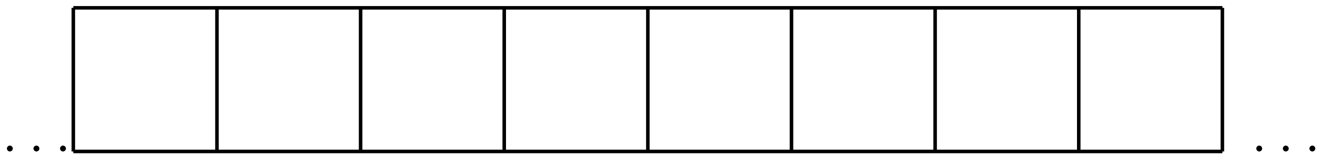
Side-channel atomicity

- Chevallier-Mames, Ciet, Joye 2004
  Idea: build group operation from identical blocks.
- Each block consists of:

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

  fill with cheap dummy additions and negations

  ADD $(\mathcal{A} + \mathcal{J})$ needs 11 blocks
  DBL $(2\mathcal{J})$ needs 10 blocks



- Requires that M and S are indistinguishable from their traces.
- No protection against fault attacks.

Side-channel atomicity

- Chevallier-Mames, Ciet, Joye 2004
  Idea: build group operation from identical blocks.
- Each block consists of:

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

  fill with cheap dummy additions and negations
  ADD $(\mathcal{A}+\mathcal{J})$ needs 11 blocks
  DBL $(2\mathcal{J})$ needs 10 blocks

...	ADD_9	ADD_{10}	ADD_{11}	DBL_1	DBL_2	DBL_3	DBL_4	DBL_5	...

- Requires that M and S are indistinguishable from their traces.
- No protection against fault attacks.

Uniform Projective coordinates

- Brier, Joye 2002
  Idea: unify how the slope is computed.
- improved in Brier, Déchène, and Joye 2004
- $$\begin{aligned}\lambda &= \frac{(x_1+x_2)^2 - x_1x_2 + a_4 + y_1 - y_2}{y_1 + y_2 + x_1 - x_2}\\ &= \begin{cases} \frac{y_1-y_2}{x_1-x_2} & (x_1,y_1) \neq \pm(x_2,y_2) \\ \frac{3x_1^2+a_4}{2y_1} & (x_1,y_1) = (x_2,y_2)\end{cases}\end{aligned}$$

  Multiply numerator & denominator by $x_1 - x_2$ to see this.
- Proposed formulae can be generalized to projective coordinates.
- Some special cases may occur, but with very low probability, e. g. $x_2 = y_1 + y_2 + x_1$. Alternative equation for this case.

Jacobi intersection and quartic

- Liardet and Smart CHES 2001: Jacobi intersection
- Billet and Joye AAECC 2003: Jacobi-Model

$$E_J : Y^2 = \epsilon X^4 - 2\delta X^2 Z^2 + Z^4.$$

$$\begin{aligned} X_3 &= X_1 Z_1 Y_2 + Y_1 X_2 Z_2 \\ Z_3 &= (Z_1 Z_2)^2 - \epsilon (X_1 X_2)^2 \\ Y_3 &= (Z_3 + 2\epsilon (X_1 X_2)^2)(Y_1 Y_2 - 2\delta X_1 X_2 Z_1 Z_2) + \\ & \quad 2\epsilon X_1 X_2 Z_1 Z_2 (X_1^2 Z_2^2 + Z_1^2 X_2^2). \end{aligned}$$

- Unified formulas need 10M+3S+D+2E
- Can have ϵ or δ small
- Needs point of order 2; for $\epsilon = 1$ the group orsder is divisible by 4.

Hessian curves

$$E_H : X^3 + Y^3 + Z^3 = cXYZ.$$

Addition: $P \neq \pm Q$	Doubling $P = Q \neq -P$
$X_3 = X_2Y_1^2Z_2 - X_1Y_2^2Z_1$	$X_3 = Y_1(X_1^3 - Z_1^3)$
$Y_3 = X_1^2Y_2Z_2 - X_2^2Y_1Z_1$	$Y_3 = X_1(Z_1^3 - Y_1^3)$
$Z_3 = X_2Y_2Z_1^2 - X_1Y_1Z_2^2$	$Z_3 = Z_1(Y_1^3 - X_1^3)$

- Curves were first suggested for speed
- Joye and Quisquater suggested Hessian Curves for unified group operations using

$$[2](X_1 : Y_1 : Z_1) = (Z_1 : X_1 : Y_1) \oplus (Y_1 : Z_1 : X_1)$$

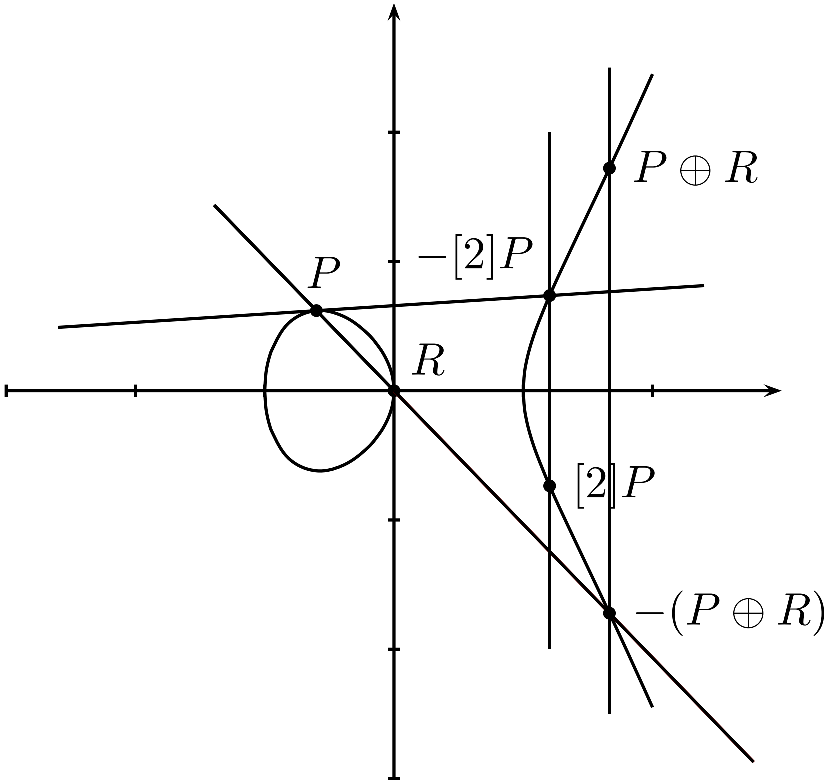
- Unified formulas need 12M.
- Needs point of order 3.

Edwards coordinates

Addition on Elliptic Curves

At Mathematics: Algorithms and Proofs in Leiden, January 2007, Harold M. Edwards gave a talk on Addition on Elliptic Curves

So Dan and I expected . . .



Addition on Elliptic Curves

At Mathematics: Algorithms and Proofs in Leiden, January 2007, Harold M. Edwards gave a talk on Addition on Elliptic Curves

But there it was – the elliptic curve:

$$x^2 + y^2 = a^2(1 + x^2y^2).$$

Addition on Elliptic Curves

At Mathematics: Algorithms and Proofs in Leiden, January 2007, Harold M. Edwards gave a talk on Addition on Elliptic Curves

But there it was – the elliptic curve:

$$x^2 + y^2 = a^2(1 + x^2y^2).$$

Nonsingular if and only if $a^5 \neq a$.

Addition on Elliptic Curves

At Mathematics: Algorithms and Proofs in Leiden, January 2007, Harold M. Edwards gave a talk on Addition on Elliptic Curves

But there it was – the elliptic curve:

$$x^2 + y^2 = a^2(1 + x^2y^2).$$

Nonsingular if and only if $a^5 \neq a$.
To see that this is indeed an elliptic curve, use $z = y(1 - a^2x^2)/a$ to obtain

$$z^2 = x^4 - (a^2 + 1/a^2)x^2 + 1.$$

Edwards' Addition Formulae

- $P = (x_P, y_P), Q = (x_Q, y_Q)$ on $x^2 + y^2 = a^2(1 + x^2y^2)$.
- $$P \oplus Q = \left(\frac{x_P y_Q + y_P x_Q}{a(1 + x_P x_Q y_P y_Q)}, \frac{y_P y_Q - x_P x_Q}{a(1 - x_P x_Q y_P y_Q)} \right).$$
- $$\begin{aligned} [2]P &= \left(\frac{x_P y_P + y_P x_P}{a(1 + x_P x_P y_P y_P)}, \frac{y_P y_P - x_P x_P}{a(1 - x_P x_P y_P y_P)} \right) \\ &= \left(\frac{2x_P y_P}{a(1 + (x_P y_P)^2)}, \frac{y_P^2 - x_P^2}{a(1 - (x_P y_P)^2)} \right). \end{aligned}$$
- For much more information on elliptic curves in this shape see Edwards 2007 paper in Bull. AMS., electronic April 9.

Results on Edwards coordinates are ongoing joint work with

Daniel J. Bernstein

Edwards coordinates

Introduce further parameter and relabel

$$x^2 + y^2 = c^2(1 + dx^2y^2), \ c, d \neq 0, dc^4 \neq 1.$$

- Neutral element is $(0, c)$, this is an affine point!
- $-(x_P, y_P) = (-x_P, y_P)$.
- $P \oplus Q = \left(\frac{x_P y_Q + y_P x_Q}{c(1 + dx_P x_Q y_P y_Q)}, \frac{y_P y_Q - x_P x_Q}{c(1 - dx_P x_Q y_P y_Q)} \right).$

Edwards coordinates

Introduce further parameter and relabel

$$x^2 + y^2 = c^2(1 + dx^2y^2),\ c, d \neq 0, dc^4 \neq 1.$$

- Neutral element is $(0, c)$, this is an affine point!
- $-(x_P, y_P) = (-x_P, y_P)$.
- $P \oplus Q = \left(\frac{x_P y_Q + y_P x_Q}{c(1 + d x_P x_Q y_P y_Q)}, \frac{y_P y_Q - x_P x_Q}{c(1 - d x_P x_Q y_P y_Q)}\right)$.
- $[2]P = \left(\frac{x_P y_P + y_P x_P}{c(1 + d x_P x_P y_P y_P)}, \frac{y_P y_P - x_P x_P}{c(1 - d x_P x_P y_P y_P)}\right)$.

Edwards coordinates

Introduce further parameter and relabel

$$x^2 + y^2 = c^2(1 + dx^2y^2), \; c, d \neq 0, dc^4 \neq 1.$$

- Neutral element is $(0, c)$, this is an affine point!
- $-(x_P, y_P) = (-x_P, y_P)$.
- $P \oplus Q = \left(\frac{x_P y_Q + y_P x_Q}{c(1 + d x_P x_Q y_P y_Q)}, \frac{y_P y_Q - x_P x_Q}{c(1 - d x_P x_Q y_P y_Q)} \right).$
- $[2]P = \left(\frac{x_P y_P + y_P x_P}{c(1 + d x_P x_P y_P y_P)}, \frac{y_P y_P - x_P x_P}{c(1 - d x_P x_P y_P y_P)} \right).$
- Unified group operations!

Edwards coordinates

Introduce further parameter and relabel

$$x^2 + y^2 = c^2(1 + dx^2y^2),\ c, d \neq 0, dc^4 \neq 1.$$

- Neutral element is $(0, c)$, this is an affine point!
- $-(x_P, y_P) = (-x_P, y_P)$.
- $P \oplus Q = \left(\frac{x_P y_Q + y_P x_Q}{c(1 + d x_P x_Q y_P y_Q)}, \frac{y_P y_Q - x_P x_Q}{c(1 - d x_P x_Q y_P y_Q)} \right).$

$$\begin{aligned}
A &= Z_P \cdot Z_Q;\ B = A^2;\ C = X_P \cdot X_Q;\ D = Y_P \cdot Y_Q;\\
E &= (X_P + Y_P) \cdot (X_Q + Y_Q) - C - D;\ F = d \cdot C \cdot D;\\
X_{P\oplus Q} &= A \cdot E \cdot (B - F);\ Y_{P\oplus Q} = A \cdot (D - C) \cdot (B + F);\\
Z_{P\oplus Q} &= c \cdot (B - F) \cdot (B + F).
\end{aligned}$$

Edwards coordinates

Introduce further parameter and relabel

$$x^2 + y^2 = c^2(1 + dx^2y^2),\ c, d \neq 0, dc^4 \neq 1.$$

- Neutral element is $(0, c)$, this is an affine point!
- $-(x_P, y_P) = (-x_P, y_P)$.
- $P \oplus Q = \left(\dfrac{x_P y_Q + y_P x_Q}{c(1 + dx_P x_Q y_P y_Q)}, \dfrac{y_P y_Q - x_P x_Q}{c(1 - dx_P x_Q y_P y_Q)} \right).$

$$\begin{aligned}
A &= Z_P \cdot Z_Q;\ B = A^2;\ C = X_P \cdot X_Q;\ D = Y_P \cdot Y_Q; \\
E &= (X_P + Y_P) \cdot (X_Q + Y_Q) - C - D;\ F = d \cdot C \cdot D; \\
X_{P\oplus Q} &= A \cdot E \cdot (B - F);\ Y_{P\oplus Q} = A \cdot (D - C) \cdot (B + F); \\
Z_{P\oplus Q} &= c \cdot (B - F) \cdot (B + F).
\end{aligned}$$

Needs 10M + 1S + 1C + 1D + 7A. At least one of c, d small.

Comparison of unified formulae

System	Cost of unified addition-or-doubling
Projective	11M+6S+1D; see Brier/Joye '02
Projective if $a_4 = -1$	13M+3S; see Brier/Joye '02
Jacobi intersection	13M+2S+1D; see Liardet/Smart '01
Jacobi quartic	10M+3S+3D; see Billet/Joye '03
Hessian	12M; see Joye/Quisquater '01
Edwards ($c = 1$)	10M+1S+1D

- Exactly the same formulae for doubling (no re-arrangement like in Hessian; no if-else)
- No exceptional cases if d is not a square. Formulae correct for all affine inputs (incl. $(0, c), -P$).
- Caveat: Edwards curves have a point of order 4, namely $(c, 0)$.

Countermeasures against DPA

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. This changes the scalar & binary representation.
- Split the scalar $n = k_1 + k_2$
- Change the representation of the scalar (Aigner Oswald) using redundancy in signed representation – often too little randomness.
- Randomize group representation, e. g. use isomorphic or isogenous curve; alternative field representation.
- Randomize element representation.

Randomized Points

- Compute $[n]Q$ as $[n](Q \oplus R) \ominus [n]R$.
- For efficiency $[n]R$ should be known, e.g. precompute short list

$$\{(R_1, [n]R_1), (R_2, [n]R_2), \ldots, (R_k, [n]R_k)\}$$

  use

$$[n]Q = [n](Q \oplus R_l) \ominus [n]R_l$$

  for random l.
- Use new results to refill list.

Randomized Coordinates

(Coron's third countermeasure)

- Let the affine base point be $P_j = (x_j, y_j)$. Start computation with

$$P_j = (X_j : Y_j : Z_j) = (rx_j : ry_j : r) \sim (x_j : y_j : 1)$$

  for random r. Randomization can also be used at intermediate steps
- Same idea works for Jacobian coordinates.
- Can be used also for unified addition formulae.
- Make sure to transfer back to affine $(x'_j, y'_j) = [n]P_j$ after computation (Naccache, Smart, Stern Eurocrypt 2004)
- Note that zero values are not changed . Use only in combination with others to avoid Goubin attacks.

Edwards coordinates for speed

Fastest addition formulae

- Unified formulas are valid for addition
- Edwards ADD takes 10M+1S+1D, mixed 9M+1S+1D.

System	Cost of addition
Jacobian	12M+2S; HECC
Jacobi intersection	13M+2S+1D; see Liardet/Smart ’01
Projective	12M+2S; HECC
Jacobi quartic	10M+3S+3D; see Billet/Joye ’03
Hessian	12M; see Joye/Quisquater ’01
Edwards ($c = 1$)	10M+1S+1D

How about non-unified doubling?

$$[2]P = \left(\frac{x_P y_P + y_P x_P}{c(1 + d x_P x_P y_P y_P)}, \frac{y_P y_P - x_P x_P}{c(1 - d x_P x_P y_P y_P)}\right)$$

$$= \left(\frac{2 x_P y_P}{c(1 + d(x_P y_P)^2)}, \frac{y_P^2 - x_P^2}{c(1 - d(x_P y_P)^2)}\right)$$

How about non-unified doubling?

$$[2]P = \left(\frac{x_P y_P + y_P x_P}{c(1 + d x_P x_P y_P y_P)}, \frac{y_P y_P - x_P x_P}{c(1 - d x_P x_P y_P y_P)}\right)$$

$$= \left(\frac{2 x_P y_P}{c(1 + d(x_P y_P)^2)}, \frac{y_P^2 - x_P^2}{c(1 - d(x_P y_P)^2)}\right)$$

$$= \left(\frac{2 c x_P y_P}{c^2(1 + d(x_P y_P)^2)}, \frac{c(y_P^2 - x_P^2)}{c^2(2 - (1 + d(x_P y_P)^2))}\right)$$

Use curve equation $x^2 + y^2 = c^2(1 + d x^2 y^2)$.

How about non-unified doubling?

$$[2]P = \left(\frac{x_P y_P + y_P x_P}{c(1 + d x_P x_P y_P y_P)}, \frac{y_P y_P - x_P x_P}{c(1 - d x_P x_P y_P y_P)}\right)$$

$$= \left(\frac{2 x_P y_P}{c(1 + d(x_P y_P)^2)}, \frac{y_P^2 - x_P^2}{c(1 - d(x_P y_P)^2)}\right)$$

$$= \left(\frac{2c x_P y_P}{c^2(1 + d(x_P y_P)^2)}, \frac{c(y_P^2 - x_P^2)}{c^2(2 - (1 + d(x_P y_P)^2))}\right)$$

$$= \left(\frac{2c x_P y_P}{x_P^2 + y_P^2}, \frac{c(y_P^2 - x_P^2)}{2c^2 - (x_P^2 + y_P^2)}\right)$$

How about non-unified doubling?

$$[2]P = \left(\frac{x_Py_P + y_Px_P}{c(1 + dx_Px_Py_Py_P)}, \frac{y_Py_P - x_Px_P}{c(1 - dx_Px_Py_Py_P)}\right)$$

$$= \left(\frac{2x_Py_P}{c(1 + d(x_Py_P)^2)}, \frac{y_P^2 - x_P^2}{c(1 - d(x_Py_P)^2)}\right)$$

$$= \left(\frac{2cx_Py_P}{c^2(1 + d(x_Py_P)^2)}, \frac{c(y_P^2 - x_P^2)}{c^2(2 - (1 + d(x_Py_P)^2))}\right)$$

$$= \left(\frac{2cx_Py_P}{x_P^2 + y_P^2}, \frac{c(y_P^2 - x_P^2)}{2c^2 - (x_P^2 + y_P^2)}\right)$$

Can always choose $c = 1$!

Doubling in Edwards coordinates

$P = (X_1 : Y_1 : Z_1)$, $Q = (X_2 : Y_2 : Z_2)$, $P \oplus Q = (X_3 : Y_3 : Z_3)$
on $E_E : (X^2 + Y^2)Z^2 = c^2(Z^4 + dX^2Y^2)$; $(x, y) \sim (X/Z, Y/Z)$

$$\begin{aligned}
A &= X_1 + Y_1;\ B = A^2;\ C = X_1^2;\ D = Y_1^2;\ E = C + D;\\
F &= B - E;\ G = c \cdot Z_1;\ H = G^2;\ I = H + H;\ J = E - I;\\
X_3 &= c \cdot F \cdot J;\ Y_3 = c \cdot E \cdot (C - D);\ Z_3 = E \cdot J.
\end{aligned}$$

- 3M + 4S + 3C + 6A.
- For $c = 1$ this gives the fastest known doubling formulas!

Fastest doubling formulae

System	Cost of doubling
Projective	6M+5S+1D; HECC
Hessian	6M+6S; see Joye/Quisquater '01
Jacobi quartic	1M+9S+3D; see Billet/Joye '03
Jacobian	2M+7S+1D; HECC
Jacobian if $a_4 = -3$	3M+5S; see DJB '01
Jacobi intersection	4M+3S+1D; see Liardet/Smart '01
Edwards ($c = 1$)	3M+4S

- Edwards faster than Jacobian in DBL & ADD.
- Edwards coordinates allow to use windowing methods
- Montgomery takes 5M+4S+1D per bit.

Multi-scalar multiplication

Idea of joint doublings

- To compute $[n_1]P_1 \oplus [n_2]P_2 \oplus \cdots \oplus [n_m]P_m$ compute the doublings together, i.e. write scalars n_i in binary:

$$
\begin{array}{lclllllll}
n_1 & = & n_{1,l-1}2^{l-1} & +n_{1,l-2}2^{l-2} & +n_{1,l-3}2^{l-3} & \dots & +n_{1,1}2 & +n_{1,0} \\
n_2 & = & n_{2,l-1}2^{l-1} & +n_{2,l-2}2^{l-2} & +n_{2,l-3}2^{l-3} & \dots & +n_{2,1}2 & +n_{2,0} \\
\vdots & = & \vdots & \vdots & \vdots & & \vdots & \vdots \\
n_m & = & n_{m,l-1}2^{l-1} & +n_{m,l-2}2^{l-2} & +n_{m,l-3}2^{l-3} & \dots & +n_{m,1}2 & +n_{m,0}
\end{array}
$$

Idea of joint doublings

- To compute $[n_1]P_1 \oplus [n_2]P_2 \oplus \cdots \oplus [n_m]P_m$ compute the doublings together, i.e. write scalars n_i in binary:

$$\begin{array}{cccccccc}
n_1 & = & n_{1,l-1}2^{l-1} & +n_{1,l-2}2^{l-2} & +n_{1,l-3}2^{l-3} & \ldots & +n_{1,1}2 & +n_{1,0} \\
n_2 & = & n_{2,l-1}2^{l-1} & +n_{2,l-2}2^{l-2} & +n_{2,l-3}2^{l-3} & \ldots & +n_{2,1}2 & +n_{2,0} \\
\vdots & = & \vdots & \vdots & \vdots & & \vdots & \vdots \\
n_m & = & n_{m,l-1}2^{l-1} & +n_{m,l-2}2^{l-2} & +n_{m,l-3}2^{l-3} & \ldots & +n_{m,1}2 & +n_{m,0}
\end{array}$$

- Compute as
$$[2](\underbrace{[n_{1,l-1}]P_1 \oplus [n_{2,l-1}]P_2 \oplus [n_{3,l-1}]P_3 \oplus \cdots \oplus [n_{m,l-1}]P_m}_{\text{first column}})$$

Idea of joint doublings

- To compute $[n_1]P_1 \oplus [n_2]P_2 \oplus \cdots \oplus [n_m]P_m$ compute the doublings together, i.e. write scalars n_i in binary:

$$\begin{array}{lllllll}
n_1 & = & n_{1,l-1}2^{l-1} & +n_{1,l-2}2^{l-2} & +n_{1,l-3}2^{l-3} & \ldots & +n_{1,1}2 & +n_{1,0} \\
n_2 & = & n_{2,l-1}2^{l-1} & +n_{2,l-2}2^{l-2} & +n_{2,l-3}2^{l-3} & \ldots & +n_{2,1}2 & +n_{2,0} \\
\vdots & = & \vdots & \vdots & \vdots & & \vdots & \vdots \\
n_m & = & n_{m,l-1}2^{l-1} & +n_{m,l-2}2^{l-2} & +n_{m,l-3}2^{l-3} & \ldots & +n_{m,1}2 & +n_{m,0}
\end{array}$$

- Compute as

$$[2]\left([2]([n_{1,l-1}]P_1 \oplus [n_{2,l-1}]P_2 \oplus [n_{3,l-1}]P_3 \oplus \cdots \oplus [n_{m,l-1}]P_m)\oplus \right.$$
$$\left.([n_{1,l-2}]P_1 \oplus [n_{2,l-2}]P_2 \oplus [n_{3,l-2}]P_3 \oplus \cdots \oplus [n_{m,l-2}]P_m\right) \oplus$$
$\cdots$ etc.

- Needs many more additions than doublings, even with precomputations.

Applications

- ECDSA verification uses 2 scalar multiplications ... just to add the results.
- If base point P is fixed, precompute $R = [2^{l/2}]P$ and include in the curve parameters. Split scalar $n = n_1 2^{l/2} + n_0$ and compute

$$[n_1]R \oplus [n_0]P.$$

- GLV curves split scalar in two halves to get faster scalar multiplication.
- Verification in accelerated ECDSA can be extended to use 4 or even 6 scalars. Splitting of the scalar is done by LLL techniques.
- Further applications in batch verification of signatures – many scalars – by taking random linear combinations.

Comparison – 1 DBL & 0.5 mixed ADD

System	Cost of 1 DBL & 0.5 mixed ADD
Projective	10.5M+6S+1D
Jacobi quartic	5M+10.5S+4.5D
Hessian	11M+3S
Jacobian	6M+8.5S+1D
Jacobi intersection	9.5M+4S+0.5D
Jacobian if $a_4 = -3$	7M+6.5S
Edwards	7.5M+4.5S+0.5D

1 DBL & 0.75 ADD & 0.75 mixed ADD

System	1DBL & 0.75 ADD & 0.75 mixed ADD
Projective	21.75M+8S+1D
Jacobi intersection	22M+6S+1.5D
Jacobian	16.25M+13S+1D
Jacobian if $a_4 = -3$	17.25M+11S
Jacobi quartic	14.5M+13.5S+7.5D
Hessian	22.5M+3S
Chudnovsky if $a_4 = -3$	16.5M+10.25S
Edwards	17.25M+5.5S+1.5D

Chudnovsky refers to the case that the second input to the addition is of the form $(X_2 : Y_2 : Z_2 : Z_2^2 : Z_2^3)$.
Note that $Z_2 = 1$ is possible here.

The end

`http://www.hyperelliptic.org/tanja/newelliptic/`