楕円曲線暗号の基礎 ♯2

松尾 和人 (IISEC)

2007年10月6日

♣ 楕円曲線 ♣

$Y^2 = F(X)$

$$\begin{aligned} E : Y^2 &= F(X) \\ &= X^3 + AX + B,\ \ A, B \in \mathbb{F}_p \end{aligned}$$

$E(\mathbb{F}_p) = \{(x, y) \in \mathbb{F}_p^2 \mid y^2 = F(x)\} \bigcup \{P_\infty\}$

P_∞: **無限円点**

$E(\mathbb{F}_p)$ **の要素を** E **の** $\mathbb{F}_p$-**有理点という**

位数 $n := \# E(\mathbb{F}_p)$ **：集合** $E(\mathbb{F}_p)$ **の要素数**

n **の範囲：**

$p + 1 - 2\sqrt{p} \leq n \leq p + 1 + 2\sqrt{p}$

- A, B **によって** n **は変わる**
- $n \approx p$

点 $P \in E(\mathbb{F}_p)$ **の位数** $N := \# \langle P \rangle$

$N \mid n$

♣ E 上の離散対数問題 ♣

- $E(\mathbb{F}_p)$ **は有限アーベル群**

- **離散対数問題**
  - Given: $E/\mathbb{F}_p$: EC,
    $P \in E(\mathbb{F}_p)$, $Q \in \langle P \rangle$
  - Find: $x \in \mathbb{Z}/N\mathbb{Z}$ s.t. $Q = [x]P$

- **容易：** $(x, P) \mapsto Q$
  - $x = (x_{k-1} x_{k-2} \ldots x_1 x_0)_2$,
    $Q = \sum_{0 \leq i < k} [2^{x_i}] P$,
    $k = O(\log p)$

- **困難：** $(P, Q) \mapsto x$

♣ 楕円曲線上の離散対数問題の難しさ ♣

- Generic Algo.
  一般に利用可能
  - **全数探索**
    - $N = O(p)$
  - Square-root **法**
    - **計算量は** N **に依存**
    - **多くの曲線上の離散対数問題は これにより現実的に解読可能**
- Non-Generic Algo.
  特殊な構造を利用
  効果が大きいことが多い
  適用範囲は小さい
  - Menezes-Okamoto-Vanstone
  - SSSA
  - Index calculus

♣ Square-root 法 ♣

- **任意の群上の離散対数問題に適用可能**
- Pohlig-Hellman
  $+$
  {baby-step giant-step,
  Pollard's rho, lambda}

♣ Pohlig-Hellman 法 ♣

- Silver **が発見、**
  1978 **年に** Pohlig **と** Hellman **が再発見**
  1. **問題を小さく分解し**
  2. **それぞれを** rho **法などで解く**
  3. **中国の剰余定理・分割統治法により**
  4. **元の問題の解を得る**

♣ 中国の剰余定理 ♣

$m_1, m_2 \in \mathbb{Z}$, $\mathrm{gcd}(m_1, m_2) = 1$,

$x_1, x_2 \in \mathbb{Z}$, $0 \le x_i < m_i$ **としたとき**

$$x \equiv x_i \bmod m_i$$

を満足する $x \in \mathbb{Z}$ s.t.

$0 \le x < m_1 m_2$ **が一意に定まる。**

$$x = ((m_1^{-1} \bmod m_2)(x_2 - x_1) \bmod m_2)m_1 + x_1$$

計算量

$O((\log(m_1 m_2))^2)$ bit-operations

(雑)

♣ 問題の分解 1 ♣

まず、 N **を素因数分解する：**

$$N = \prod_{1 \le i \le r} l_i^{e_i}$$

すると、

求めたい x **は**

$$[N/l_i^{e_i}]Q = [N/l_i^{e_i} x]P, i = 1, \ldots, r$$

を満足

ここで

$$\#\langle [N/l_i^{e_i}]P \rangle = l_i^{e_i}$$

そこで

$Q_i := [N/l_i^{e_i}]Q,$

$P_i := [N/l_i^{e_i}]P$

として

$x_i \in [0, l_i^{e_i} - 1]$ s.t. $Q_i = P_i^{x_i}$

for $i \in [1, r]$

が求まったとすると、

x_i, $l_i^{e_i}$ は

$$x \equiv x_i \bmod l_i^{e_i} \quad \text{for} \quad i \in [1, r]$$
$$\gcd(l_i^{e_i}, l_j^{e_j}) = 1 \quad \text{for} \quad i \neq j$$

を満足するので、

中国の剰余定理により、x が

$$O\left((\log p)^2\right) \text{ bit-operations}$$

で求まる

♣ 問題の分解 2 ♣

問題は既に書き換えられている：

Given: $E/\mathbb{F}_p$: EC,
l: prime, $e \in \mathbb{N}$ s.t. $l^e \mid N$,
$P \in E(\mathbb{F}_p)$ s.t. $\#\langle P \rangle = l^e$,
$Q \in \langle P \rangle$

Find: $x \in [0, l^e - 1]$ s.t. $Q = [x]P$

P_i, Q_i, x_i, l_i, e_i を

P, Q, x, l, e と置き直した

♣ $e > 1$ のとき ♣

$f \in \mathbb{Z}$ を $0 < f < e$ とする

$$Q = [x]P,\ \ 0 \le x < l^e$$

$\Rightarrow$

$$\exists u, v \in \mathbb{Z} \text{ s.t. } x = l^f v + u,$$
$$0 \le u < l^f, 0 \le v < l^{e-f}$$

$\Rightarrow$

$$\begin{aligned}[l^{e-f}]Q &= [(l^{e-f})x]P \\ &= [(l^{e-f})(l^f v + u)]P \\ &= [(l^{e-f} l^f v + l^{e-f} u)]P \\ &= [(l^e v + l^{e-f} u)]P \\ &= [l^{e-f} u]P \\ &= [u][l^{e-f}]P\end{aligned}$$

ここで

$$\#\langle [l^{e-f}]P \rangle = l^f < l^e$$

$$Q = [l^f v + u]P$$

$\Rightarrow$

$$\begin{aligned}Q - [u]P &= [l^f v]P \\ &= [v][l^f]P\end{aligned}$$

ここで

$$\#\langle b^{l^f} \rangle = l^{e-f} < l^e$$

そこで

1: $[l^{e-f}]Q = [u][l^{e-f}]P$ を解いて、u を得る

2: $Q - [u]P = [v][l^f]P$ を解いて、v を得る

3: $x = l^f v + u$

これを再帰的に用いれば、

$e = 1$ のときに帰着される（分割統治法）

f の選択：今の場合 $f \approx e/2$ が最良

♣ 計算量評価 ♣

1 step での計算：

$[l^{e-f}]Q$, $[l^{e-f}]P$, $[u]P$, $[l^f]P$

$4 \times [l^e]P$:

$$O(\log l^e) = O(e \log l) E(\mathbb{F}_p)\text{-ops.}$$

$+$

より小さな離散対数問題を解くための時間 $\times 2$

$$\begin{aligned} T(l,e) &= O(e \log l) + 2T(l, e/2) \\ &= O(e \log e \log l + eT(l,1)) \end{aligned}$$

$e = 1$ の場合の解法は？

♣ $Q = [x]P$, $\#\langle P \rangle = l$ の計算 ♣

Given: $E/\mathbb{F}_p$: EC,
l: prime,
$P \in E(\mathbb{F}_p)$ s.t. $\#\langle P \rangle = l$,
$Q \in \langle P \rangle$

Find: $x \in [0, l^e - 1]$ s.t. $Q = [x]P$

- 全数探索
  - $O(l)$
- Square-root 法
  - Baby-step giant-step 法
    - Deterministic algo.
    - メモリー必要
  - Pollard の rho 法 / lambda 法
    - Monte Carlo algo.
    - 空間計算量: $O(1)$
    - パラレル計算可能

♣ Rho/lambdaの基本アイディア ♣

バースデイパラドックスの利用：

クラスメイトが23**人いれば、**

クラスに同じ誕生日のペアが居る確率は

1/2**以上**

$$1 - 1 \times \frac{364}{365} \times \frac{363}{365} \times \cdots \times \frac{343}{365} = 0.507\ldots$$

$$\sqrt{365} = 19.104\ldots$$

♣ Birthday Paradox ♣

S : set, $n_0 = \#S$

r **個の中に**1**組も同じ値のペアがない確率**:

$$\begin{aligned}
\prod_{i=1}^{r} \frac{n_0 - i + 1}{n_0} &= \prod_{i=1}^{r} \left(1 - \frac{i-1}{n_0}\right) \\
&< \prod_{i=1}^{r} \exp\left(-\frac{i-1}{n_0}\right) \\
&\quad \because 1 + x \le e^x \\
&= \exp\left(\sum_{i=1}^{r} -\frac{i-1}{n_0}\right) \\
&= \exp\left(-\frac{r(r-1)}{2n_0}\right) \\
&\approx \exp\left(-\frac{r^2}{2n_0}\right)
\end{aligned}$$

$$r = \sqrt{2(\log 2)n_0} \Rightarrow \exp\left(-\frac{r^2}{2n_0}\right) = 0.5$$

$\Rightarrow O(\sqrt{n_0})$ **個の中には**
一致するペアがある確率が高い

♣ Rho法の原型 ♣

Algorithm 1 Pollard's rho.alpha

Input: $E/\mathbb{F}_p$: EC, l: prime,
$P \in E(\mathbb{F}_p)$ s.t. $\#\langle P\rangle = l$, $Q \in \langle P\rangle$

Output: $x \in [0, l-1]$ s.t. $Q = [x]P$

1: $i := 0$
2: **repeat**
3: $\quad i := i + 1$
4: $\quad$ Choose $\alpha_i, \beta_i \in [0, l-1]$ randomly
5: $\quad R_i = [\alpha_i]P + [\beta_i]Q$
6: **until** ${}^\exists j$ s.t. $1 \le j < i, R_j = R_i$
7: $x = (\alpha_i - \alpha_j)(\beta_j - \beta_i)^{-1} \bmod l$
/*$\alpha_i + \beta_i x \equiv \alpha_j + \beta_j x \bmod l$*/
8: Output x and terminate

(平均）時間計算量: $O(\sqrt{l})$

(平均）空間計算量: $O(\sqrt{l})$

♣ Beta版の作成 ♣

方針：ランダムをやめる

4: Choose $\alpha_i, \beta_i \in [0, l-1]$ randomly
5: $R_i = [\alpha_i]Q + [\beta_i]P$

$\rightarrow$ ランダムウォーク関数 W を使う

集合 S_1, S_2, S_3 を適当に決める

$\#S_1 \approx \#S_2 \approx \#S_3$,

$\langle P\rangle = S_1 \cup S_2 \cup S_3$,

$S_1 \cap S_2 = S_2 \cap S_3 = S_3 \cap S_1 = \emptyset$

$$
\begin{aligned}
R_i &= W(R_{i-1}) \\
&= \begin{cases} R_{i-1} + P, & \text{if } R_{i-1} \in S_1 \\ [2]R_{i-1}, & \text{if } R_{i-1} \in S_2 \\ R_{i-1} + Q, & \text{if } R_{i-1} \in S_3 \end{cases}
\end{aligned}
$$

♣ ランダムウォーク関数を利用した計算 ♣

$$
\begin{aligned}
R_i &= W(R_{i-1}) \\
&= \begin{cases} R_{i-1} + P, & \text{if } R_{i-1} \in S_1 \\ [2]R_{i-1}, & \text{if } R_{i-1} \in S_2 \\ R_{i-1} + Q, & \text{if } R_{i-1} \in S_3 \end{cases}
\end{aligned}
$$

$$
\begin{aligned}
\alpha_i &= W_\alpha(\alpha_{i-1}) \\
&= \begin{cases} \alpha_{i-1} + 1, & \text{if } R_{i-1} \in S_1 \\ 2\alpha_{i-1}, & \text{if } R_{i-1} \in S_2 \\ \alpha_{i-1}, & \text{if } R_{i-1} \in S_3 \end{cases}
\end{aligned}
$$

$$
\begin{aligned}
\beta_i &= W_\beta(\beta_{i-1}) \\
&= \begin{cases} \beta_{i-1}, & \text{if } R_{i-1} \in S_1 \\ 2\beta_{i-1}, & \text{if } R_{i-1} \in S_2 \\ \beta_{i-1} + 1, & \text{if } R_{i-1} \in S_3 \end{cases}
\end{aligned}
$$

Algorithm 2 Pollard's rho.beta

Input: $E/\mathbb{F}_p$: EC, l: prime,
$P \in E(\mathbb{F}_p)$ s.t. $\#\langle P \rangle = l$, $Q \in \langle P \rangle$

Output: $x \in [0, l-1]$ s.t. $Q = [x]P$

```
i := 1
Choose α_1, β_1 ∈ [0, l − 1] randomly
R_1 = [α_1]P + [β_1]Q
repeat
    i := i + 1
    R_i = W(R_{i−1})
    α_i = W_α(α_{i−1}), β_i = W_β(β_{i−1})
until ∃j s.t. 1 ≤ j < i, c_j = c_i
x = (α_i − α_j)(β_j − β_i)^{−1} mod l
Output x and terminate
```

（平均）時間計算量: $O(\sqrt{l})$

（平均）空間計算量: $O(\sqrt{l})$

Beta版の空間計算量はalpha版と同じ

ところが、
ランダムウォークさせた場合、一度衝突するとずっと衝突したままになる

しかも、
いつか必ず$j=2i$となる

(計算量は変わらない)

i	3	4	5	6	7	8	$\boxed{9}$
j	12	13	14	15	16	17	$\boxed{18}$

i	10	11	12	13	14	15	16	17
j	18	19	20	21	22	23	24	25

10	11	12	13	14	15	$\boxed{16}$
26	27	28	29	30	31	$\boxed{32}$

Algorithm 3 Pollard's rho method

Input: $E/\mathbb{F}_p$: EC, l: prime,
$P \in E(\mathbb{F}_p)$ s.t. $\#\langle P\rangle = l$, $Q \in \langle P\rangle$

Output: $x \in [0, l-1]$ s.t. $Q = [x]P$

Choose $\alpha_1, \beta_1 \in [0, l-1]$ randomly
$R_1 = [\alpha_1]P + [\beta_1]Q$
$R_2 = W(R_1)$
$\alpha_2 = W_\alpha(\alpha_1)$, $\beta_2 = W_\beta(\beta_1)$
while $R_1 \neq R_2$ **do**
  $R_1 = W(R_1)$
  $\alpha_1 = W_\alpha(\alpha_1)$, $\beta_1 = W_\beta(\beta_1)$
  $R_2 = W(W(c_2))$
  $\alpha_2 = W_\alpha(W_\alpha(\alpha_2))$,
  $\beta_2 = W_\beta(W_\beta(\beta_2))$
$x = (\alpha_2 - \alpha_1)(\beta_1 - \beta_2)^{-1} \bmod l$
Output x and terminate

時間計算量: $O(\sqrt{l})$

空間計算量: $O(1)$

♣ 例題 ♣

$P \in E(\mathbb{F}_p)$

$$P \in \begin{cases} S_1 & ; 0 \le X(P) < p/3 \\ S_2 & ; p/3 \le X(P) < 2p/3 \\ S_3 & ; 2p/3 \le X(P) < p \end{cases}$$

$p = 31$

$\Rightarrow S_1$: $[0, 10]$, S_2: $[11, 20]$, S_3: $[21, 30]$

$E/\mathbb{F}_p : Y^2 = X^3 + 10X + 22$

$\#E(\mathbb{F}_p) = 37$: **素数**

$P = (10, 3)$, $Q = (9, 10)$

$\alpha_1 = 35$, $\beta_1 = 36$

$R_1 = [\alpha]P + [\beta]Q$

i	R_i	α_i	β_i
1	$(30, 13) \in S_3$	35	36
2	$(11, 3) \in S_2$	35	0
3	$(3, 19) \in S_1$	33	0
4	$(15, 4) \in S_2$	34	0
5	$(21, 17) \in S_3$	31	0
6	$(5, 13) \in S_1$	31	1
7	$(20, 17) \in S_2$	32	1
8	$(29, 11) \in S_3$	27	2
9	$(3, 12) \in S_1$	27	3
10	$(7, 2) \in S_1$	28	3
11	$(21, 14) \in S_3$	29	3
12	$(8, 11) \in S_1$	29	4
13	$(29, 11) \in S_3$	30	4
14	$(3, 12) \in S_1$	30	5
15	$(7, 2) \in S_1$	31	5
16	$(21, 14) \in S_3$	32	5
17	$(8, 11) \in S_1$	32	6
18	$(29, 11) \in S_3$	33	6
19	$(3, 12) \in S_1$	33	7
20	$(7, 2) \in S_1$	34	7
21	$(21, 14) \in S_3$	35	7

$$
\begin{aligned}
\frac{\alpha_{14}-\alpha_9}{\beta_9-\alpha_{14}} &\equiv \frac{\alpha_{15}-\alpha_{10}}{\beta_{10}-\alpha_{15}} \\
&\equiv \frac{\alpha_{15}-\alpha_{11}}{\beta_{11}-\alpha_{15}} \\
&\equiv \frac{\alpha_{20}-\alpha_{10}}{\beta_{10}-\alpha_{20}} \mod 37
\end{aligned}
$$

$$
\begin{aligned}
\frac{30-27}{3-5} &\equiv \frac{31-28}{3-5} \\
&\equiv \frac{32-29}{3-5} \\
&\equiv \frac{34-28}{3-7} \\
&\equiv 17 \mod 37
\end{aligned}
$$

$[17]P = Q$

♣ l^e に対する計算量 ♣

$$
\begin{aligned}
T(l,e) &= O(e \log l) + T(l, e/2) \\
&= O(e \log e \log l + eT(l,1)) \\
&= O\left(e\left(\log e \log l + \sqrt{l}\right)\right) \\
&= \begin{cases} e\sqrt{l} \\ \text{for} e = O(2^{\sqrt{l}/\log l}) \\ e \log e \log l \\ \text{for} e = \Omega(2^{\sqrt{l}/\log l}) \end{cases}
\end{aligned}
$$

ここで、$\#\langle P\rangle$ に対して T を評価すると、$\#\langle P\rangle = l^e$ より

$$
\begin{cases} O(e\sqrt{l}) & ;\#\langle b\rangle \text{ に対し指数時間} \\ O(e \log e \log l) & ;\#\langle b\rangle \text{ に対し多項式時間} \end{cases}
$$

♣ Square-root 法の計算量 ♣

$$N = \prod_{1\le i\le r} l_i^{e_i}$$

$$T(N) = \sum_{1\le i\le r} O\left(e_i\sqrt{l_i}\right) E(\mathbb{F}_p)\text{-ops.}$$

ワーストケースは各 i に対し $e_i = 1$ のとき：

$$\begin{aligned} T(N) &= \sum_{1\le i\le r} O\left(\sqrt{l_i}\right) \\ &= O(\sqrt{l})E(\mathbb{F}_p)\text{-ops.}, \\ & \quad l := \max(l_i) \end{aligned}$$

$\Rightarrow$
$l \approx \#E(\mathbb{F}_p)$ のとき、
離散対数問題は難しくなる
$\Rightarrow$
暗号に用いるときは、$\#E(\mathbb{F}_p)$ が素数に近いものを選ぶ

更に、
実際には $\#\langle P\rangle = l$ である P を用いる

♣ Square-root 法に対する安全性 ♣

l を $\#E(\mathbb{F}_p)$ の最大素因子とすると、
G 上の離散対数問題を $O(\sqrt{l})$ $E(\mathbb{F}_p)$-ops. で解くことができる
$\Rightarrow$
80 bit の安全性が必要であれば $l \approx 2^{160}$ が、
128 bit の安全性が必要であれば $l \approx 2^{256}$ が必要である

実装効率を考慮すれば、$l \approx \#E(\mathbb{F}_p)$ が望ましい
$\Rightarrow$
80 bit の安全性が必要であれば、
l を 160 bit 素数、
128 bit の安全性が必要であれば、
l を 256 bit 素数とし、

$$\#E(\mathbb{F}_p) = cl,$$

c: small constant.

♣ 実例 ♣

$p = 2^{160} - 47$

$E/\mathbb{F}_p : Y^2 = X^3 + AX + B$

$A = 1419587478389183342895449$
$\quad 5567034801779119991818 32$

$B = 1370276796320878164248991$
$\quad 6604447824844952837371 7$

$E(\mathbb{F}_p) = \{P_\infty,$
$(3, 5590791294558787911099016 6$
$\quad 1839949961707046542132),$
$(3, -5590791294558787911099016 6$
$\quad 1839949961707046542132), \ldots\}$

$$
\begin{aligned}
\#E(\mathbb{F}_p) &= 1461501637330902918 20 \\
&\quad 368702342303847964898 7 \\
&\quad 002960 \text{ (161bit)} \\
&= 2^4 \cdot 5 \cdot 23 \cdot 1277 \cdot 711713 \cdot \\
&\quad 1801867 \cdot 2045011 \cdot \\
&\quad 4738031 \cdot 38408653 \cdot \\
&\quad 1303287809
\end{aligned}
$$

i	l_i	e_i	$\log_2 l_i$
1	2	4	1
2	5	1	3
3	23	1	5
4	1277	1	11
5	711713	1	20
6	1801867	1	21
7	2045011	1	21
8	4738031	1	23
9	38408653	1	26
10	1303287809	1	31

♣ 実例：問題 ♣

$$P = (68877725316734917430550420\\ 3634807500170063433889,\\ 108181954955246312214561 71\\ 5376247940072982167 0608)$$

$N := \#\langle P\rangle = \#E(\mathbb{F}_p)$

$\because [N/l_i]P \neq P_\infty$ for $i = 1,\ldots,10$

$$Q = (3, 55907912945587879110990 1\\ 661839949961707046542132)$$

Find x s.t. $Q = [x]P$

♣ For $l_1 = 2$, $e_1 = 4$ ♣

$$\begin{aligned} P_1 &= [N/l_1^{e_1}]P \\ &= (121808580546805219226 33\\ &\quad 896712682978666377850707 83,\\ &\quad 137024357844751571954304\\ &\quad 663799269715774858384458 2) \end{aligned}$$

$$\begin{aligned} Q_1 &= [N/l_1^{e_1}]Q \\ &= (130535747564222644363 71\\ &\quad 500758743464490665992861 39,\\ &\quad 918498003406678847422986\\ &\quad 552214801965456185529002) \end{aligned}$$

Find $x_1 \in [0, 2^4 - 1]$ s.t. $Q_1 = [x_1]P_1$

♣ 実例：分割統治法 ♣

1: $[l^{e-f}]Q = [u][l^{e-f}]P$ **を解いて、**
u **を得る**

2: $Q - [u]P = [v][l^f]P$ **を解いて、**
v **を得る**

3: $x = l^f v + u$

$f = e_1/2 = 2$

$$\begin{aligned} P_{11} &= [l_1^{e_1-f}]P_1 = [4]P_1 \\ &= (1013707282970583568490612136378940057879000334514, \\ &\quad 10633666673059511999496017743674509237951686312 84) \end{aligned}$$

$$\begin{aligned} Q_{11} &= [l_1^{e_1-f}]Q_1 = [4]Q_1 \\ &= (1013707282970583568490612136378940057879000334514, \\ &\quad 1063366667305951199949601774367450923795168631284) \end{aligned}$$

$Q_{11} = [u]P_{11}$

$\#\langle P_{11}\rangle = l_1^f = 4$

$\hat{f} = f/2 = 1$

$$\begin{aligned} P_{12} &= [l_1^{f-\hat{f}}]P_{11} = [2]P_{11} \\ &= (4226055817864884638391297862392104045250298532 09, 0) \end{aligned}$$

$$\begin{aligned} Q_{12} &= [l_1^{f-\hat{f}}]Q_{11} = [2]Q_{11} \\ &= (422605581786488463839129786239210404525029853209, 0) \end{aligned}$$

$Q_{12} = [\hat{u}]P_{12} \Rightarrow \hat{u} = 1$

$P_{13} = [l_1^{\hat{f}}]P_{11} = [2]P_{11} = P_{12}$

$Q_{13} = Q_{12} - [\hat{u}]P_{12} = P_\infty$

$Q_{13} = [\hat{v}]P_{13} \Rightarrow \hat{v} = 0$

$\Rightarrow u = l_1^{\hat{f}}\hat{v} + \hat{u} = 1$

1: $[l^{e-f}]Q = [u][l^{e-f}]P$ を解いて、
u を得る

2: $Q - [u]P = [v][l^f]P$ を解いて、
v を得る

3: $x = l^f v + u$

$$
\begin{aligned}
P_{14} &= [l_1^f]P_1 = [4]P_1 \\
&= (1013707282970583568490612136378940057879000334514, \\
&\quad 10633666673059511999496017743674509237951686312 84)
\end{aligned}
$$

$$
\begin{aligned}
Q_{14} &= Q_1 - [u]P_1 = Q_1 - P_1 \\
&= (1013707282970583568490612136378940057879000334514, \\
&\quad 1063366667305951199949601774367450923795168631284)
\end{aligned}
$$

$Q_{14} = [v]P_{14} \Rightarrow v = 1$

$\Rightarrow x_1 = l_1^f v + u = 2^2 + 1 = 5$

♣ 実例：Rho part ♣

i	$l_i^{e_1}$/ $\log_2 l_i$	x_i	sec./stps.
1	2^4 1	5	−−
2	5 3	3	.07 3
3	23 5	20	.10 7
4	1277 11	293	.11 55
5	711713 20	532349	.20 387
6	1801867 21	1686283	.44 1457
7	2045011 21	531538	.34 1105
8	4738031 23	1517838	.62 2350
9	38408653 26	22671833	3.2 14945
10	1303287809 31	1094987215	8.9 42617

$.07 + .10 + .11 + .20 + .44 + .34 + .62 + 3.2 + 8.9 \approx 14$sec.

on Magma/efficēon 1G

♣ 実例：CRT part ♣

$$
\begin{aligned}
x &\equiv 5 \bmod 16 \\
&\equiv 3 \bmod 5 \\
&\equiv 20 \bmod 23 \\
&\equiv 293 \bmod 1277 \\
&\equiv 532349 \bmod 711713 \\
&\equiv 1686283 \bmod 1801867 \\
&\equiv 531538 \bmod 2045011 \\
&\equiv 1517838 \bmod 4738031 \\
&\equiv 22671833 \bmod 38408653 \\
&\equiv 1094987215 \bmod 1303287809
\end{aligned}
$$

$\Rightarrow$

$$
\begin{aligned}
x \equiv\ & 7437997324363661365463626638 \\
& 0124606492914920982\,13 \bmod N
\end{aligned}
$$

♣ 参考文献 ♣

I. Blake, G. Seroussi, and N. Smart. *Elliptic Curves in Cryptography*. Number 265 in London Mathematical Society Lecture Note Series. Cambridge U. P., 1999.

H. Cohen, G. Frey, R. Avanzi, C. Doche, T. Lange, K. Nguyen, and F. Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography*. Chapman & Hall/CRC, 2005.

A. Menezes, P. van Oorschot, and S. Vanstone. *Handbook of applied cryptography*. CRC Press, 1997.

V. Shoup. *A computational introduction to number theory and algebra*. Cambridge University Press, 2005.