

Projet de Cryptographie : l'algorithme de Schoof

Comptage de points sur une courbe elliptique

P-Aug. BERTHET, Marc RENARD

Université de Paris

8 juin 2021



Université de Paris

Sommaire

- 1 Introduction
 - Notations
- 2 Fondement théorique
 - Lois sur \mathbb{R}
 - Morphisme de Frobenius
 - Borne de Hasse
 - Polynômes de division
- 3 Algorithme et Implémentation
- 4 Résultats
 - Résultats dans \mathbb{F}_{37}
 - Résultats sur d'autres corps finis de forme \mathbb{F}_p
 - Résultats sur d'autres corps finis de forme \mathbb{F}_q
- 5 Améliorations et applications à la Cryptologie
- 6 Références



Université de Paris

Sommaire

- 1 Introduction
 - Notations
- 2 Fondement théorique
- 3 Algorithme et Implémentation
- 4 Résultats
- 5 Améliorations et applications à la Cryptologie
- 6 Références



Université de Paris

Notations

On utilisera les notations suivantes :

- $E(C, \mathbb{K})$: la courbe elliptique E d'équation C sur le corps \mathbb{K} de cardinal q ,
- $P(x, y)$, un point de la courbe E ,
- (x_k, y_k) , les coordonnées de kP
- $\pi(P)$, le morphisme de Fröbenius, de trace t , appliqué à P ,
- $\tilde{E}[l]$, le groupe des points de l -torsion de la courbe E ,
- ψ_l , le l -ième polynôme de division de la courbe E ,
- O , le point à l'infini de la courbe E ,
- \bar{t}_l , l'estimation de t modulo l ,
- \bar{q} , la valeur de q modulo l .



Sommaire

1 Introduction

2 Fondement théorique

- Lois sur \mathbb{R}
- Morphisme de Frobenius
- Borne de Hasse
- Polynômes de division

3 Algorithme et Implémentation

4 Résultats



Université de Paris

Lois sur R

Definition

Soient $P_1 = a_1(x) + y * a_2(x)$ et $P_2 = b_1(x) + y * b_2(x)$ deux éléments de R . On a :

$$P_1 + P_2 = (a_1(x) + b_1(x)) + y * (a_2(x) + b_2(x))$$

$$P_1 * P_2 = (a_1 * b_1 + a_2 * b_2 * (x^3 + A * x + B)) + y * (a_1 * b_2 + a_2 * b_1)$$



Morphisme de Fröbenius

Definition

On appelle morphisme de Fröbenius sur E , noté π , l'endomorphisme de courbe elliptique (ou encore isogénie) suivant :

$$\pi : E \rightarrow E$$

$$\pi(x, y) \mapsto Q = (x^q, y^q)$$

$$\pi(O) \mapsto O$$

Ce morphisme a pour équation caractéristique le polynôme suivant :

$$X^2 - tX + q = 0,$$

avec t la trace de l'endomorphisme et q son déterminant.



Borne de Hasse

Theorem

Borne de Hasse. Soit E une courbe elliptique sur \mathbb{F}_q . Alors on a l'égalité suivante :

$$|E(\mathbb{F}_q)| = q + 1 - t, \text{ avec } |t| < 2\sqrt{q},$$

avec q le cardinal de \mathbb{F}_q , et t la trace du morphisme de Frobenius sur E .



Polynômes de division

Definition

On appelle l -ième polynôme de division d'une courbe E , noté ψ_l , le polynôme dont les racines sont les points du groupe $\bar{E}[l]$.



Université de Paris

Propriété

Lemma

*Soit $P(x_p, y_p)$ un point de la courbe E et k un scalaire positif.
Alors le point $R(x_r, y_r) = k * P$ est donné par :*

$$x_r = x_p - (\psi_{k-1} * \psi_{k+1}) / (\psi_k^2),$$

$$y_r = (\psi_{2k}) / (2 * \psi_k^4).$$



Sommaire

- 1 Introduction
- 2 Fondement théorique
- 3 Algorithme et Implémentation**
- 4 Résultats
- 5 Améliorations et applications à la Cryptologie
- 6 Références



Université de Paris

Initialisation

```
a = ffggen(q,v);  
e = [0,0,0,E[1],E[2]];  
P = [[a^0*'x',0],[0,a^0]];  
[S,DIV_POL] = gen_s_and_div_pols( q , e  
    );  
L = List(); \\ liste des Mod(t,l)
```



Université de Paris

Cas $l = 2$

```
pgcd2 = gcd('x^q-x', 'x^3+E[1]*'x+E[2]);  
if (poldegree(pgcd2)>0,  
    \\then  
    listput(L,Mod(0,2)),  
    \\else  
    listput(L,Mod(1,2));  
);
```



Université de Paris

Pour $l \in S \setminus \{2\}$ | étape 1 : init tour l

```
l = S[i];  
q_l = q % l;  
if (q_l >= (l/2), q_l = q_l - l);
```

```
phi_l = mapget(DIV_POL, l);
```

```
P_mod_phi_l = Mod(P, phi_l);
```

```
frob2p = frob_ell(frob_ell(P_mod_phi_l, q, E,  
    phi_l), q, E, phi_l);
```

```
qlp = div_pol_mul(P_mod_phi_l, q_l, E, DIV_POL,  
    phi_l);
```



Université de Paris

Pour $l \in S \setminus \{2\}$ | étape 2 : si $\pi^2(P) \neq q_l P$

```
for(t = 1, (l-1)/2,
  Pqt = div_pol_mul(P_mod_phi_l, t, E, DIV_POL,
    phi_l);
  Pqt = frob_ell(Pqt, q, E, phi_l);
  if(mul_law(add_law(frob2p[2], (-1)*Pqt[2]),
    add_law((-1)*qlp[1], frob2p[1]), E, phi_l)
    == mul_law(add_law((-1)*qlp[2], frob2p
      [2]), add_law(frob2p[1], (-1)*Pqt[1]), E,
      phi_l),
    listput(L, 1 + q + Mod(t, l));
    trouve = 1;
    break();
);
```



Université de Paris

Pour $l \in S \setminus \{2\}$ | étape 2 : si $\pi^2(P) \neq q_l P$

```

if (mul_law(add_law(frob2p[2], Pqt[2]),
    add_law(qlp[1], (-1)*frob2p[1]), E, phi_l)
    == mul_law(add_law(qlp[2], (-1)*frob2p
        [2]), add_law(frob2p[1], (-1)*Pqt[1]), E,
        phi_l),
    listput(L, 1 + q - Mod(t, l));
    trouve = 1;
    break();
);
);

```



Université de Paris

Pour $l \in S \setminus \{2\}$ | étape 3 : si $\pi^2(P) = \pm q_l P$
et q n'est pas un carré modulo l

```
w;  
if (issquare(Mod(q_l, l), &w) == 0,  
    listput(L, 1 + q - Mod(0, l));  
    trouve = 1;  
);
```



Université de Paris

Pour $l \in S \setminus \{2\}$ | étape 3 : si $\pi^2(P) = \pm q_l P$
et q est un carré modulo l

```
wP = div_pol_mul(P_mod_phi_l, lift(w), E, DIV_POL,
    phi_l);
wP = frob_ell(wP, q, E, phi_l);
if(wP == frob2p,
    listput(L, 1 + q - 2*w);
    trouve = 1,
    if(wP == [frob2p[1], (-1)*frob2p[2]],
        listput(L, 1 + q + 2*w),
        listput(L, 1 + q - Mod(0, l)));
    );
```



Université de Paris

Pour $l \in S \setminus \{2\}$ | étape 4 : restes chinois sur L

```
t = chinese(L);  
N = t.mod;  
t = lift(t);  
while( t < (q+1-2*sqrt(q)) , t = t + N );  
return(t);
```



Université de Paris

Sommaire

1 Introduction

2 Fondement théorique

3 Algorithme et Implémentation

4 Résultats

- Résultats dans \mathbb{F}_{37}
- Résultats sur d'autres corps finis de forme \mathbb{F}_p
- Résultats sur d'autres corps finis de forme \mathbb{F}_q

5 Améliorations et applications à la Cryptologie



Université de Paris

Résultats dans \mathbb{F}_{37}

$(A \bmod 37, B \bmod 37)$	(5,0)	(0,9)	(0,6)	(1,12)	(2,2)
$\#(E(\mathbb{F}_{37}))$	26	27	28	29	30
$(A \bmod 37, B \bmod 37)$	(2,8)	(3,6)	(1,13)	(1,18)	(1,8)
$\#(E(\mathbb{F}_{37}))$	31	32	33	34	35
$(A \bmod 37, B \bmod 37)$	(1,0)	(0,5)	(1,5)	(0,3)	(1,2)
$\#(E(\mathbb{F}_{37}))$	36	37	38	39	40
$(A \bmod 37, B \bmod 37)$	(1,16)	(1,9)	(2,9)	(1,7)	(2,14)
$\#(E(\mathbb{F}_{37}))$	41	42	43	44	45
$(A \bmod 37, B \bmod 37)$	(1,11)	(3,15)	(0,1)	(0,2)	(2,0)
$\#(E(\mathbb{F}_{37}))$	46	47	48	49	50

Résultats sur d'autres corps finis de forme \mathbb{F}_p

(A,B)	$\#(E(\mathbb{F}_p))$
(Mod(3,1031),Mod(2,1031))	1018
(Mod(3,32771),Mod(2,32771))	33117
(Mod(1,65537),Mod(5,65537))	65460
(Mod(1,100003),Mod(5,100003))	99707
(Mod(1,131101),Mod(5,131101))	131132
(Mod(2,190027),Mod(6,190027))	190183
(Mod(1,333517),Mod(2,333517))	332640



Résultats sur d'autres corps finis de forme \mathbb{F}_q

q	(A,B)	$\#(E(\mathbb{F}_q))$
25	(g^2, g^6)	27
49	(g^2, g^6)	55
125	(g^{10}, g^{15})	108
625	(g^{10}, g^{15})	577
$1331 = 11^3$	(g^{10}, g^{15})	1274
$28561 = 13^4$	(g^{10}, g^{15})	28800
$130321 = 19^4$	(g^{10}, g^{15})	130969



Sommaire

- 1 Introduction
- 2 Fondement théorique
- 3 Algorithme et Implémentation
- 4 Résultats
- 5 Améliorations et applications à la Cryptologie
- 6 Références



Université de Paris






Sommaire

- 1 Introduction
- 2 Fondement théorique
- 3 Algorithme et Implémentation
- 4 Résultats
- 5 Améliorations et applications à la Cryptologie
- 6 **Références**



Université de Paris

Références

-  Gregg Musiker. *Schoof's Algorithm for Counting Points on $E(\mathbb{F}_q)$* . 7 Décembre 2005.
-  René Schoof. *Counting points on elliptic curves over finite fields*. Journal de Théorie des Nombres de Bordeaux 7 , 219-254 , 1995.
-  Luca Defeo. *Algorithmique et Programmation en C, 2014-2019*. Master Algèbre Appliquée à la Cryptologie et au Calcul Formel, UVSQ.
-  Hankerson, Menezes, Vanstone. *Guide to Elliptic Curves Cryptography*. Springer.
-  Mohammed Krir. *Introduction aux Courbes Elliptiques*. Master Algèbre Appliquée à la Cryptologie et au Calcul Formel, UVSQ.