Les corps finis dans PARI/GP

B. Allombert

IMB
Université de Bordeaux/CNRS

30/11/2021

Corps premiers

Pour obtenir un nombre premier aléatoire :

```
? p=randomprime(2^100)
%1 = 792438309994299602682608069491
```

Pour créer un élement de $\mathbb{F}_p$:

```
? a=Mod(17,p);
? a^(p-1) \\ exponentiation
%3 = Mod(1,792438309994299602682608069491)
```

Pour accéder aux composantes de *a* :

```
? a.mod
%4 = 792438309994299602682608069491
? lift(a) \\lift to Z
%5 = 17
```

Corps finis généraux

La fonction `ffinit(p,n)` permet de construire un polynôme irréductible de degré n sur $\mathbb{F}_p$,

```
? P=ffinit(13,2)
%6 = Mod(1,13)*x^2+Mod(1,13)*x+Mod(12,13)
? polisirreducible(P)
%7 = 1
```

Pour construire un élément de $\mathbb{F}_{p^n}$ à partir de son polynôme minimal :

```
? a=ffgen(P,'a)
%8 = a
```

Cela peut être fait directement avec

```
? a=ffgen(13^2,'a);
```

Le symbole `'a` indique que l'élément doit être affiché `a`.

Corps finis généraux

```
? b = a^2+3*a+2
%10 = 2*a+3
```

Pour accéder aux composantes de *b* :

```
? b.pol
%11 = 2*a+3
? b.mod
%12 = a^2+a+12
? [b.p, b.f]
%13 = [13, 2]
```

Pour retrouver *a* à partir de *b*

```
? ffgen(b)
%14 = a
```

Opérations sur les éléments

De nombreuse fonctions génériques sont valides sur les corps finis.

```
? c = ffgen(3^8,'c);
? d = random(c) \\random element in the field
%16 = 2*c^6 + 2*c^5 + 2*c^3 + c^2 + 2*c + 1
? issquare(d)
%17 = 1
? trace(d) \\over F_3
%18 = Mod(2, 3)
? norm(d)
%19 = Mod(1, 3)
? minpoly(d^82)
%20 = Mod(1,3)*x^4+Mod(2,3)*x^3+Mod(1,3)*x^2+Mod(2,
```

Opérations sur les éléments

```
? factor(x^5+x^3+c)
%21 = [x + (2*c^5 + c^4 + 2*c) 1]
%      [x^2 + (c^7 + 2*c^6 + ... + c^2 + 2) 1]
%      [x^2 + (2*c^7 + c^6 + ... + 2*c^2 + 1) 1]
? R=polrootsmod(x^7+x+c)
%22 = [c^7 + 2*c^6 + c^5 + c^3 + 2*c + 2,
%       2*c^7 + c^6 + c^2 + 1]~
? subst(x^7+x+c,x,R)
%23 = [0,0]~
```

Opérations concernant la structure multiplicative

Attention : Le générateur du corps n'est pas forcément un générateur du groupe multiplicatif !

```
? fforder(c)
%24 = 1640
? z = ffprimroot(c)
%25 = 2*c^7+2*c^6+2*c^5+2*c^4+c^3+c^2+c+2
? fforder(z)
%26 = 6560
? n = fflog(c,z)
%27 = 2612
? z^n
%28 = c
```

Rappel : il y a des fonctions identiques pour $\mathbb{Z}/N\mathbb{Z}$: `znorder`, `znprimroot`, `znlog`.

Morphismes entre corps finis

Il est possible de définir des morphismes entre corps finis.

```
? d = ffgen([3,24],'d)
%29 = d
? Mcd = ffembed(c,d); \\calcul un plongement de c d
? ffembed(d,c)
  ***   at top-level: ffembed(d,c)
  ***                 ^------------
  *** ffembed: domain error in ffembed: d is not a
? c2 = ffmap(Mcd,c^5+c+1) \\applique le plongement
%31 = d^21+2*d^19+d^18+d^15+2*d^13+2*d^12+2*d^10+d^
? F = fffrobenius(d,8); \\8-éme puissance du Froben
? ffmap(F, d) == d^(3^8)
%33 = 1
? ffmap(F, c2) == c2
%34 = 1
```

Extension des corps finis

`ffextend` permet de construire une extension de corps finis définie par un polynôme irréductible.

```
? T = x^3+d*x+1; polisirreducible(T)
%35 = 1
? [e,Mde] = ffextend(d, T, 'e);
? e.f
%37 = 72
? fforder(e)
%38 = 1595328861543098787996 86
? ffmap(Mde, d)
%39 = e^66+2*e^64+2*e^60+e^58+2*e^57+e^55+e^54+e^51
%     +e^35+2*e^34+2*e^31+e^27+2*e^25+2*e^23+e^22+2*
%     +e^10+2*e^8+e^7+e^6+2*e^4+2*e^3+e^2+2*e
```

Composition des morphismes

`ffcompomap` permet de composer des morphismes :
`ffcompomap`$(f, g) = f \circ g$.

```
? Mce = ffcompomap(Mde,Mcd);
? ffmap(Mce, c) == ffmap(Mde, ffmap(Mcd, c))
%41 = 1
? ffcompomap(F,Mcd) == Mcd
%42 = 1
? ffcompomap(F,F) == fffrobenius(d,16)
%43 = 1
```

Préimages

`ffinvmap` permet de déterminer l'image inverse d'un morphisme.

```
? Mdc = ffinvmap(Mcd);
? ffmap(Mdc, ffmap(Mcd, c^3+c+1))
%45 = c^3 + c + 1
? Mec = ffcompomap(Mdc, ffinvmap(Mde));
? ffmap(Mec, ffmap(Mce, c))
%47 = c
? ffinvmap(fffrobenius(c,3)) == fffrobenius(c,5)
%48 = 1
```

Extensions relatives

```
? ffmap(Mdc, d)
%49 = []
```

Cela indique qu'il n'y a pas de solutions car *d* n'est pas dans le corps de définition de *c*.
`ffmaprel` permet d'exprimer *d* comme élément algébrique sur le corps de définition de *c* :

```
? rd = ffmaprel(Mdc, d)
%50 = Mod(d,d^3+2*d+(c^7+2*c^5+2*c^4+2*c^3+c^2+2*c)
? sd = ffmaprel(Mdc, d^4+1)
%51 = Mod(d^2+(2*c^7+c^5+c^4+c^3+2*c^2+c)*d+1,d^3+2
```

Extensions relatives

Cela permet de calculer la trace et la norme relative et le polynôme minimal.

```
? trace(rd)
%52 = 0
? norm(rd)
%53 = 2*c^7+c^5+c^4+c^3+2*c^2+c
? [norm(norm(rd)), norm(d)]
%54 = [Mod(1,3), Mod(1,3)]
? minpoly(rd)
%55 = x^3+2*x+(c^7+2*c^5+2*c^4+2*c^3+c^2+2*c)
? minpoly(sd)
%56 = x^3+x^2+(c^5+2*c^4+2*c^2)*x+(c^7+2*c^6+c^5+c^
```

Création à partir des corps de nombres

`nfmodprinit` /`nfmodpr` permet d'identifier le corps résiduel d'un idéal premier comme corps fini.

```
? nf = nfinit(y^8-2*y^7+9*y^6-2*y^5+38*y^4-34*y^3\
              +31*y^2-6*y+1);
? pr = idealprimedec(nf,2)[2]; [pr.e,pr.f]
%58 = [2, 2]
? modpr = nfmodprinit(nf,pr,'z); \\ calcule le morp
? g = nfmodpr(nf,y,modpr)
%60 = z
? nfmodpr(nf,y^2+1,modpr)
%61 = z
? nfmodprlift(nf,g+1,modpr) \\trouve une préimage
%62 = [1, 1, 0, 0, 0, 0, 0, 0]~
```

Courbe elliptique sur les corps finis

Une courbe elliptique donnée par une forme de Weierstrass courte

$$y^2 = x^3 + a_4 x + a_6$$

ou longue

$$y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6$$

est défini par

```
? E=ellinit([a4,a6]);
? E=ellinit([a1,a2,a3,a4,a6]);
```

Courbe elliptique sur les corps finis

Soit *u* un élément de corps finis.

```
? u = ffgen([101,2],'u);
? E = ellinit([10,81*u+94],u);
```

(Le paramètre `u` garantie que la courbe est bien définie sur $\mathbb{F}_{101^2}$ et pas $\mathbb{F}_{101}$).

```
? ellcard(E) \\ cardinal de E(F_q)
%67 = 10116
? P = random(E) \\ point aléatoire sur E(F_q)
%68 = [10*u+16,92*u+5]
? Q = random(E) \\ Un autr epoint aléatoire sur E(F
%69 = [39*u+22,38*u+86]
? ellisoncurve(E, P) \\ vérifie que le point est su
%70 = 1
```

Opérations sur les points

```
? elladd(E, P, Q)    \\ P+Q dans E
%71 = [55*u+78,17*u+71]
? ellmul(E, P, 100) \\ 100.P dans E
%72 = [42*u+39,18]
? ellorder(E,P) \\ordre de P
%73 = 1686
```

Structure du groupe $E(\mathbb{F}_q)$

```
? [d1,d2]=ellgroup(E) \\ structure de E(F_q)
%74 = [1686, 6]
```

Le couple $[d1, d2]$ signifie $\mathbb{Z}/d_1\mathbb{Z} \times \mathbb{Z}/d_2\mathbb{Z}$, avec $d_2 \mid d_1$.

Couplages

```
? [G1,G2] = ellgenerators(E)
%75 = [[31*u+7,81*u+35],[35*u+79,31*u+61]]
? ellorder(E,G1)
%76 = 1686
? w = ellweilpairing(E,G1,G2,d1)
%77 = 100*u
? fforder(w)
%78 = 6
? t = elltatepairing(E,G2,G1,d2)^((101^2-1)/d2)
%79 = u+1
? fforder(t)
%80 = 6
```

Logarithmes discrets

```
? e = random(d1);
? S = ellmul(E,P,e)
%82 = [37*u+45,56*u+29]
? elllog(E,S,P)
%81= 1405
? e
%84 = 1405
```

Tordues

```
? Et = elltwist(E); Et[1..5] \\ retourne le twist n
%85 = [0, 0, 0, 96*u + 25, 62*u + 74]
? ellap(E)
%86 = 86
? ellap(Et)
%87 = -86
```

Isogénies

```
? P3 = ellmul(E,G1,d1/3);
? ellorder(E,P3)
%89 = 3
? [eq,iso] = ellisogeny(E,P3);
? eq
%91 = [0,0,0,55*u+49,88*u+26]
? iso
%92 = [x^3+(64*u+23)*x^2+(4*u+65)*x+(72*u+12)
%       y*x^3+(96*u+85)*y*x^2+(56*u+4)*y*x+(36*u+40)
%       x+(32*u+62)]
? G1q = ellisogenyapply(iso, G1)
%93 = [28*u+12,84*u+24]
? Eq = ellinit(eq); ellorder(Eq, G1q)
%94 = 562
```