# Courbes elliptiques

Tiffanie Jolivet - Arthur Froger

21 mai 2020

Dans ce projet, nous allons aborder le sujet des courbes elliptiques.
Ces courbes sont intéressantes car, munies d'une loi que l'on précisera, elles peuvent former un groupe. Pour ce faire, on devra montrer l'existence d'un élément neutre, d'un inverse pour chaque point et pour finir, l'associativité qui constitue une grosse partie de notre projet. Mais avant de montrer tout cela, afin de faciliter nos calculs, nous allons d'abord simplifier l'équation d'une courbe elliptique en faisant apparaître la forme réduite, dite de Weierstrass. Elles sont utilisées dans le domaine de la cryptographie et peuvent notamment permettre de faire des échanges de clés. Grâce au problème elliptique du logarithme discret que nous allons découvrir, le secret sera bien gardé !

## Table des matières

# 1 Introduction aux courbes elliptiques

**Définition 1.** *Une courbe algébrique réelle plane est l'ensemble des zéros d'un polynôme en deux variables réelles.*

**Définition 2.** *Soit $E$ une courbe algébrique réelle plane définie par $P \in \mathbb{R}[x, y]$. On dit qu'un point $M$ sur $E$ est régulier si les dérivées partielles de $P$ par rapport à $x$ et à $y$ ne sont pas simultanément nulles lorsqu'on les évalue en $M$. Dans le cas contraire, le point $M$ est dit singulier.*

**Définition 3.** *Une courbe est dite non singulière si tous ses points sont réguliers.*

**Définition 4.** *Une courbe elliptique réelle est une courbe algébrique non singulière qui a pour équation :*

$$y^2 + a_2xy + a_4y = x^3 + a_1x^2 + a_3x + a_5$$

*où $a_i \in \mathbb{R}$ pour tout $i \in [\![1, 5]\!]$*

Afin de faciliter nos calculs, nous allons d'abord réduire notre équation et ensuite voir les conditions de non singularité.

## 1.1 Réduction de Weierstrass

**Proposition 1.** *A un isomorphisme affine près, l'équation d'une courbe elliptique s'écrit de manière unique sous la forme réduite (dite de Weierstrass) suivante :*

$$\boxed{y^2 = x^3 + ax + b}$$

*Démonstration.* On va faire une première simplification en remarquant que :

$$\left(y + \frac{a_2}{2}x + \frac{a_4}{2}\right)^2 = y^2 + a_2xy + a_4y + \frac{a_2^2x^2}{4} + \frac{a_2a_4x}{2} + \frac{a_4^2}{4}$$

On pose alors le changement de variable linéaire en $y$ suivant :

$$Y = y + \frac{a_2}{2}x + \frac{a_4}{2} \tag{1}$$

Notre équation devient :

$$Y^2 = x^3 + \frac{4a_1 + a_2^2}{4}x^2 + \frac{2a_3 + a_2a_4}{2}x + \frac{4a_5 + a_4^2}{4}$$

Autrement écrit :

$$Y^2 = x^3 + b_1x^2 + b_2x + b_3$$

avec

$$b_1 = \frac{4a_1 + a_2^2}{4}, \quad b_2 = \frac{2a_3 + a_4a_2}{2}, \quad b_3 = \frac{4a_5 + a_4^2}{4}$$

Nous allons maintenant faire disparaître le terme en $x^2$ grâce à un autre changement linéaire sur $x$ cette fois-ci.
Comme :

$$\left(x + \frac{b_1}{3}\right)^3 = x^3 + b_1x^2 + \frac{b_1^2x}{3} + \frac{b_1^3}{27}$$

On pose :

$$X = x + \frac{b_1}{3} \tag{2}$$

Notre équation devient :

$$Y^2 = X^3 + X\left(b_2 - \frac{{b_1}^3}{3}\right) + b_3 - \frac{b_1b_2}{3} - \frac{2{b_1}^3}{27}$$

On a bien une équation de la forme :

$$Y^2 = X^3 + aX + b$$

Ainsi, on vient de montrer que toute courbe elliptique, dans un repère adapté, peut s'écrire sous la forme réduite de Weierstrass : $y^2 = x^3 + ax + b$

□

**Exemple 1.** *Voici une application du changement de variable sur la courbe* $y^2 + 2xy + 4y = x^3 - 2x^2 + 5x + 1$ *pour arriver à la forme réduite de Weierstrass.*
*D'abord, on pose* $Y = y + x + 2$*, ce qui nous donne* $Y^2 = x^3 - x^2 + 9x + 5$.
*Ensuite, on pose* $X = x - \frac{1}{3}$*, ce qui nous donne* $Y^2 = X^3 + \frac{28}{3}X + \frac{218}{27}$.
*Les images suivantes nous illustrent le changement de repère lors du passage de l'équation de base à l'équation réduite de Weierstrass.*

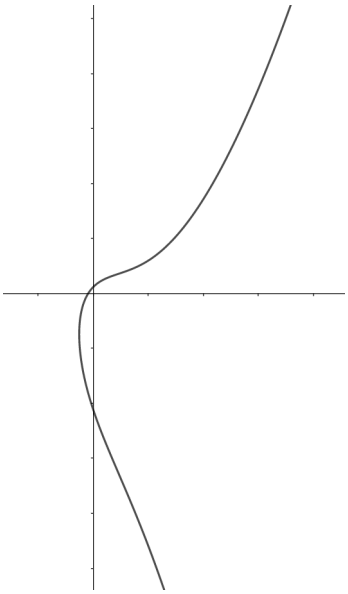

FIGURE 1. $y^2 + 2xy + 4y = x^3 - 2x^2 + 5x + 1$

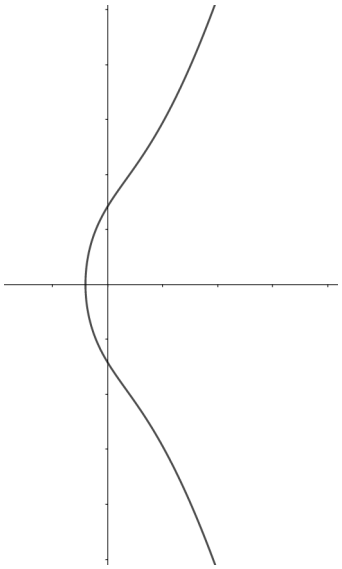

FIGURE 2. $y^2 = x^3 + \frac{28}{3}x + \frac{218}{27}$

## 1.2 Conditions de non singularité

**Proposition 2.** *Une courbe réelle algébrique de la forme* $y^2 = x^3 + ax + b$ *où* $a, b \in \mathbb{R}$ *est une courbe elliptique si, et seulement si,*

$$\boxed{27b^2 + 4a^3 \neq 0}$$

*Démonstration.* Soit

$$\begin{array}{rccc} f & : & \mathbb{R}^2 & \to & \mathbb{R} \\ & & (x,y) & \mapsto & y^2 - x^3 - ax - b \end{array}$$

On a $\dfrac{\partial f}{\partial y} = 2y$ et $\dfrac{\partial f}{\partial x} = -3x^2 - a$

Regardons les conditions lorsque les deux dérivées partielles s'annulent. Soit $P = (x, y)$, le point où les deux dérivées partielles s'annulent. On a :

$$\begin{cases} 2y = 0 \\ -a - 3x^2 = 0 \end{cases} \Leftrightarrow \begin{cases} y = 0 \\ x^2 = \frac{-a}{3} \end{cases}$$

— Si $a = 0$, on a :

$$\begin{cases} 2y = 0 \\ 3x^2 = 0 \end{cases} \Leftrightarrow \begin{cases} y = 0 \\ x = 0 \end{cases}$$

et donc la courbe $y^2 = x^3 + b$ est singulière en $(0, 0)$. Comme pour $b \neq 0$ le point $(0, 0)$ ne vérifie pas $y^2 = x^3 + b$, on a $a, b = 0$. Pour $a = 0$, la courbe est singulière si, et seulement si, $b = 0$.

— Si $a \neq 0$ :
On injecte dans notre équation et on a :

$$
\begin{aligned}
0 &= x^3 + ax + b \\
\Leftrightarrow 0 &= \frac{-a}{3}x + ax + b \\
\Leftrightarrow 0 &= \frac{2a}{3}x + b \\
\Leftrightarrow -b &= \frac{2a}{3}x \\
\Leftrightarrow \frac{-3b}{2a} &= x \\
\Leftrightarrow \frac{9b^2}{4a^2} &= x^2
\end{aligned}
$$

On sait que $x^2 = -\frac{a}{3}$
On a donc $\frac{9b^2}{4a^2} = \frac{-a}{3}$, c'est à dire $27b^2 + 4a^3 = 0$.
Donc la courbe est singulière si, et seulement si, $27b^2 + 4a^3 = 0$

□

**Exemple 2.** *Voici deux représentations de courbes singulières.*

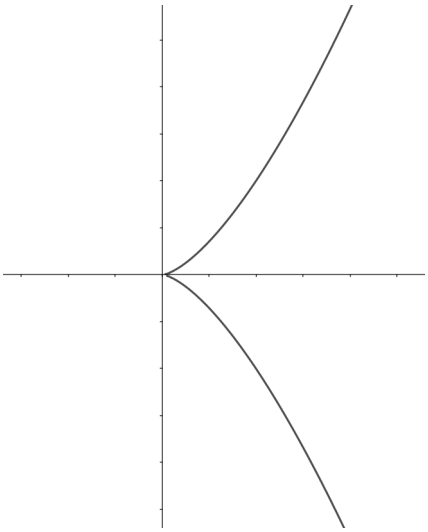

FIGURE 3. courbe singulière d'équation $y^2 = x^3$

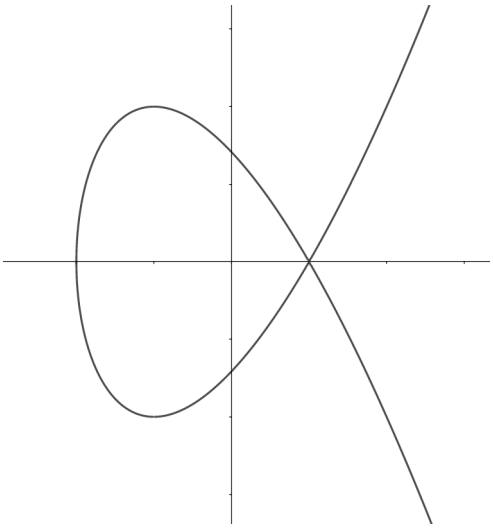

FIGURE 4. courbe singulière $y^2 = x^3 - 3x + 2$

## 1.3 Loi de groupe

Soit $(E)$ une courbe elliptique d'équation $y^2 = x^3 + ax + b$ avec $a$ et $b$ vérifiant $27b^2 + 4a^3 \neq 0$.
On aimerait construire une loi de groupe. Pour cela, on veut munir l'ensemble des points $\mathcal{A} = (E) \cup O$ d'une addition, ici $O$ est un point qui vit à l'infini [1]. Soit $* : \mathcal{A} \times \mathcal{A} \to \mathcal{A}$ qui à deux points de $\mathcal{A}$ fait correspondre un troisième point de $\mathcal{A}$ en prenant l'intersection entre la droite formée par les deux premiers points et $\mathcal{A}$. La loi qui pourrait faire de $(\mathcal{A}, *)$ un groupe ne possède pas toutes les propriétés requises (pas d'élément neutre). En effet, on suppose qu'il existe $O$ un élément de $(\mathcal{A}, *)$ qui vérifie $P * O = P$ pour tout point $P \in \mathcal{A}$. Autrement dit, $(OP)$ et $\mathcal{A}$ s'intersectent deux fois en $P$ et une fois en $O$. Comme $P$ est un point double, alors $(OP)$ est la droite tangente en $P$ de la courbe définie par $\mathcal{A}$. Donc, si cet élément existe, il n'est pas unique. L'exemple qui suit nous le montre bien :

1. $\mathbb{R}^2$ est la carte affine de $\mathbb{P}^2$ donnée par $z = 1$, le point à l'infini correspond au point $z = 0$. Par manque de temps, nous ne l'avons pas traité.

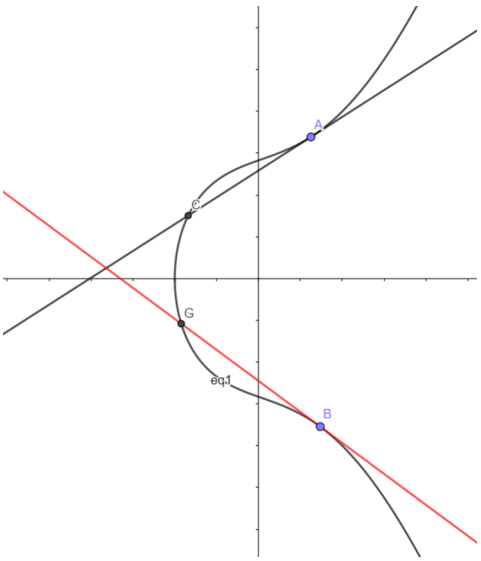


FIGURE 5. $y^2 = x^3 + x + 2$

Sur cette image, les points $A$ et $B$ représentent deux points quelconques de la courbe, et $C$ et $G$ les points d'intersection entre la courbe et les tangentes en $A$ et en $B$ respectivement. On a bien deux points distincts, ce qui montre que O n'est pas unique. Ce n'est donc pas un élément neutre. On a montré qu'il n'existe aucun élément $O$ unique qui vérifie $P * O = P$ pour tout $P \in \mathcal{A}$, donc $(\mathcal{A}, *)$ n'admet pas d'élément neutre, on doit alors définir une nouvelle loi.

On va définir la loi $+$ telle que $P + Q$ est la réflexion de $P * Q$ par rapport à l'axe des abscisses.

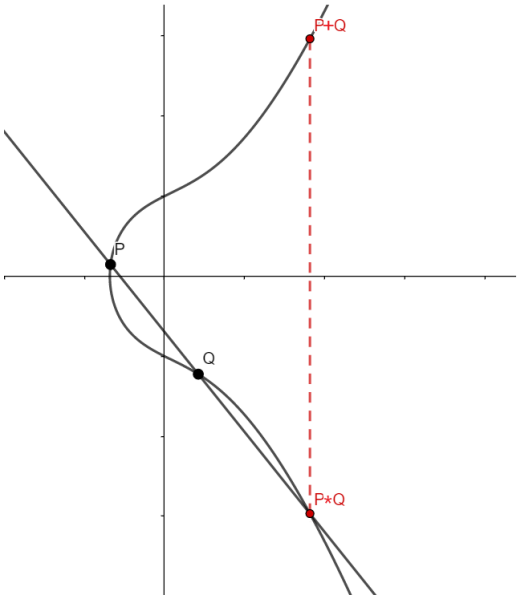


FIGURE 6. construction de la loi $+$ sur $y^2 = x^3 + x + 1$

Pour couvrir tous les cas :

— Si $y_p \neq 0$, $P + P$ est défini comme le symétrique du point d'intersection de la tangente à $E$ en $P$ qui n'est pas $P$.

— Si $y_p = 0$, la tangente en $P$ est parallèle à l'axe des $y$ et on a $P + P = O$.

— Si $P = O$, on a $P + Q = Q$. Si $Q = O$ alors $P + Q = P$.

— Si $P$ et $Q$ ont la même abscisse et $P \neq Q$, alors on a $P = (x_p, y_p)$ et $Q = (x_p, -y_p)$. On trace la droite parallèle à l'axe des $y$ passant par $P$ et $Q$. On obtient $P + Q = O$.

**Proposition 3.** *L'opération + est une loi de composition interne*

*Démonstration.* On veut montrer que la loi + est une loi de composition interne sur l'ensemble $\mathcal{A} = (E) \cup O$, c'est à dire que + est une application de $\mathcal{A} \times \mathcal{A}$ dans $\mathcal{A}$.

Pour commencer, on regarde les cas particuliers :

— Le cas d'un double point $P = (x_P, y_P)$ avec $y_P \neq 0$ :

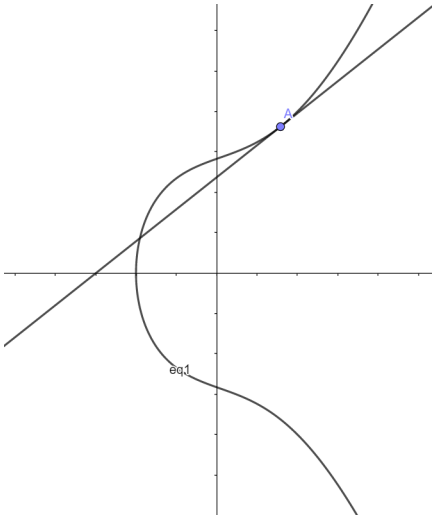


FIGURE 7. Tangente au point $P = (x_P, y_P)$ sur $y^2 = x^3 + x + 2$

On cherche l'équation de la tangente : $y = mx + p$.

-On suppose $y > 0$, on a $y = \sqrt{x^3 + ax + b} = f(x)$.
La formule de la tangente est $T : y = (x - x_P)f'(x_P) + f(x_P)$, c'est à dire :

$$T : y = \frac{(x - x_P)(3{x_P}^2 + a)}{2\sqrt{{x_P}^3 + ax_P + b}} + \sqrt{{x_P}^3 + ax_P + b} = \frac{(x - x_P)(3{x_P}^2 + a)}{2y_P} + y_P$$

D'où $m = \dfrac{3x_P^2 + a}{2y_P}$.

On résout le système suivant :

$$\begin{cases} y = (x - x_P)m + y_P \\ y^2 = x^3 \ + \ ax \ + \ b \end{cases} \Leftrightarrow \begin{cases} y = (x - x_P)m + y_P \\ ((x - x_P)m + y_P)^2 = x^3 \ + \ ax \ + \ b \end{cases}$$

On s'intéresse uniquement au coefficient devant le terme de degré 2. Ici, on a $-m^2$ lorsque l'on passe tous les termes du même coté.
Or la somme des racines d'un polynôme de degré 3 est égale au coefficient devant le $x^2$ divisé par celui devant le $x^3$.
On sait déjà que $x_P$ est solution réelle double de l'équation. Il existe alors $x_1$ réelle telle que :
$m^2 = 2x_P + x_1 \Leftrightarrow x_1 = m^2 - 2x_P$
Ainsi : $y_1 = (x_1 - x_P)m + y_P = (m^2 - 2x_P - x_P)m + y_P$
Si on note $P + P = Q$, alors

$$Q = (m^2 - 2x_P, (3x_P - m^2)m - y_P) \tag{3}$$

-Si on choisit $y < 0$, $y = -\sqrt{x^3 + ax + b} = f(x)$, l'équation de la tangente est

$$T : y = -\frac{(x - x_P)(3{x_P}^2 + a)}{2\sqrt{{x_P}^3 + ax_P + b}} - \sqrt{{x_P}^3 + ax_P + b} = \frac{(x - x_P)(3{x_P}^2 + a)}{2y_P} + y_P$$

On résout le système suivant :

$$\begin{cases} y = (x - x_P)m + y_P \\ y^2 = x^3 \ + \ ax \ + \ b \end{cases} \Leftrightarrow \begin{cases} y = (x - x_P)m + y_P \\ ((x - x_P)m + y_P)^2 = x^3 \ + \ ax \ + \ b \end{cases}$$

Les calculs nous donnent

$$P + P = (m^2 - 2x_P, (3x_P - m^2)m - y_P) \tag{4}$$

— Le cas de la tangente en $P$ où $P = (x_P, 0)$ :

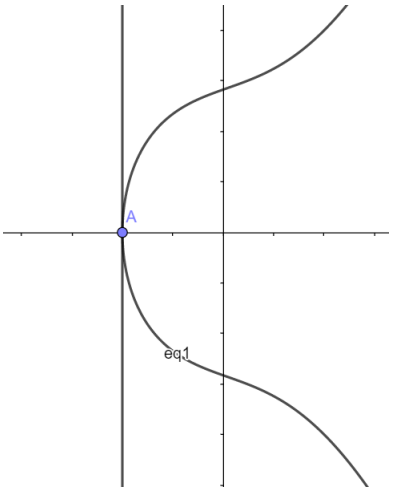


FIGURE 8. Tangente au point $P = (x_P, 0)$ sur $y^2 = x^3 + x + 2$

Comme $y_p = 0$ et que $P$ est un point double, on se trouve dans le même cas que précédemment et $\lim\limits_{y_P \to 0} \frac{3x_P^2 + a}{2y_P} = +\infty$. D'où $P + P = O \in \mathcal{A}$

— Le cas de la droite $(PP')$ où $P = (x_P, y_P)$ avec $y_P \neq 0$ et $P' = (x_P, -y_P)$ :

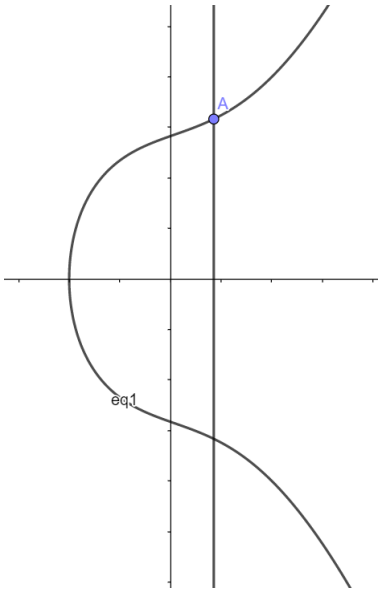


FIGURE 9. $y^2 = x^3 + x + 2$

Il n'y a pas de troisième point d'intersection entre la droite et la courbe, mais comme nous l'avons défini, $O$ vit sur toute droite verticale et donc $P + P' = O \in \mathcal{A}$.

Maintenant que nous avons traité les cas particuliers, nous allons considérer le cas générique, où toutes les opérations sont bien définies.

Soit $P = (x_P, y_P)$ et $Q = (x_Q, y_Q)$ avec $x_P \neq x_Q$ :

L'équation de la droite $(PQ)$ est $y = mx + p$ avec $m = (\frac{y_P - y_Q}{x_P - x_Q})$ et $p = y_P - mx_P$.
On a alors le système :

$$\begin{cases} y = mx + p \\ y^2 = x^3 + ax + b \end{cases} \Leftrightarrow \begin{cases} y = mx + p \\ (mx+p)^2 = x^3 + ax + b \end{cases}$$

On regarde l'équation $x^3 - m^2x^2 + (a - 2mp)x + b - p^2 = 0$.
On sait déjà que $x_P$ et $x_Q$ sont solutions réelles et que l'équation est de degré 3. Il y a alors $x_1$ réelle qui vérifie $m^2 = x_P + x_Q + x_1$, c'est à dire $x_1 = m^2 - x_P - x_Q$.
Si on note $R = (x_R, y_R)$ l'intersection de $(PQ)$ avec $\mathcal{A}$ alors

$$R = (x_1, mx_1 + p) = \big(m^2 - x_P - x_Q, m(m^2 - x_P - x_Q) + p\big)$$

et donc $P + Q = (x_R, -y_R) \in \mathcal{A}$.

□

**Lemme 1.** *La loi + est commutative.*

*Démonstration.* La droite passant par deux points quelconques $P$ et $Q$ est la même que celle qui passe par $Q$ et $P$ donc $P * Q = Q * P$. Par conséquent, on a $P + Q = Q + P$. □

**Lemme 2.** *L'élément neutre de la loi + est $O$.*

*Démonstration.* La droite passant par un point $P$ quelconque et par le point à l'infini est la droite verticale passant par P. Elle recoupe la courbe en $Q$, le symétrique de $P$ par rapport à l'axe des abscisses où $Q = (x, -y)$. Ainsi, pour avoir $P + O$, on prend le symétrique de $Q$, c'est-à-dire $P$. On a bien $P + O = P$

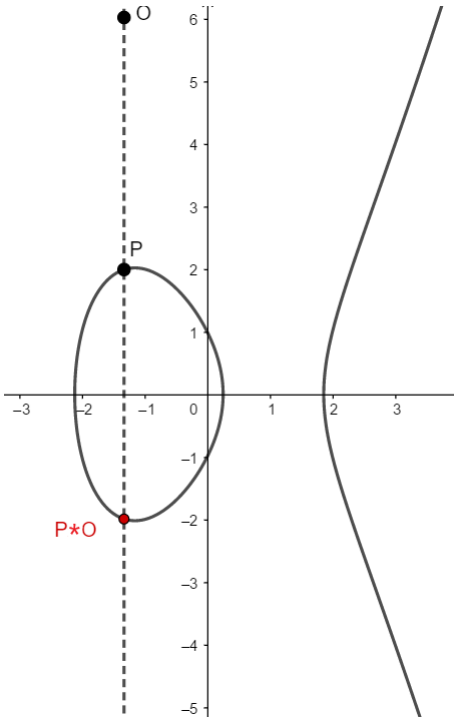


FIGURE 10. Représentation géométrique du neutre sur $y^2 = x^3 - 4x + 1$

□

**Lemme 3.** *Soit $P = (x_P, y_P) \in \mathcal{A}$. Ce point a pour symétrique $-P = (x_P, -y_P)$ qui est l'inverse pour la loi +.*

*Démonstration.* Soit $P = (x, y)$, on note $-P = (x, -y)$ son symétrique. Comme la loi est commutative, on sait déjà que $P + (-P) = (-P) + P$. Il suffit donc de montrer que $P + (-P) = O$ : Comme $P$ et $-P$ ont la même abscisse, alors la droite $(P(-P))$ est verticale et donc $P * (-P) = O$ par définition du point à l'infini. Comme le point à l'infini est l'élément neutre, il est son propre symétrique. C'est pourquoi $P + (-P) = O$.

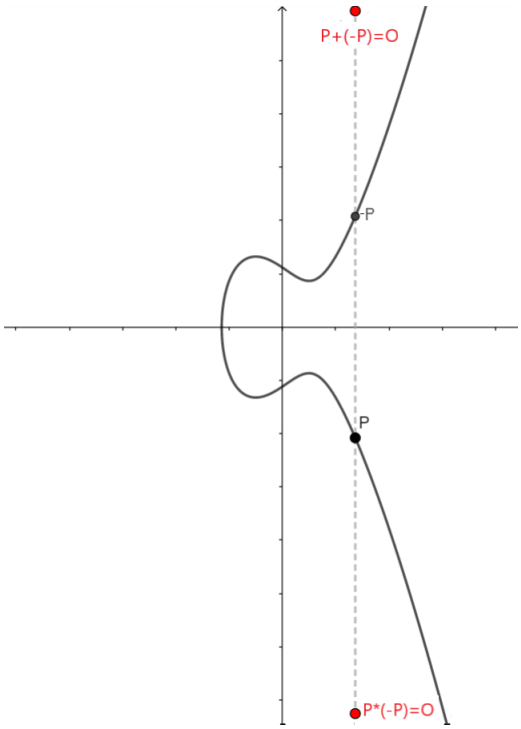


FIGURE 11. Représentation du symétrique d'un point $P$ sur $y^2 = x^3 - 3x + 5$

□

**Lemme 4.** *La loi + est associative.*

*Démonstration.* Soient $P = (x_P, y_P)$, $Q = (x_Q, y_Q)$ et $R = (x_R, y_R)$ des points appartenant à notre courbe elliptique. On veut montrer que $(P + Q) + R = P + (Q + R)$.

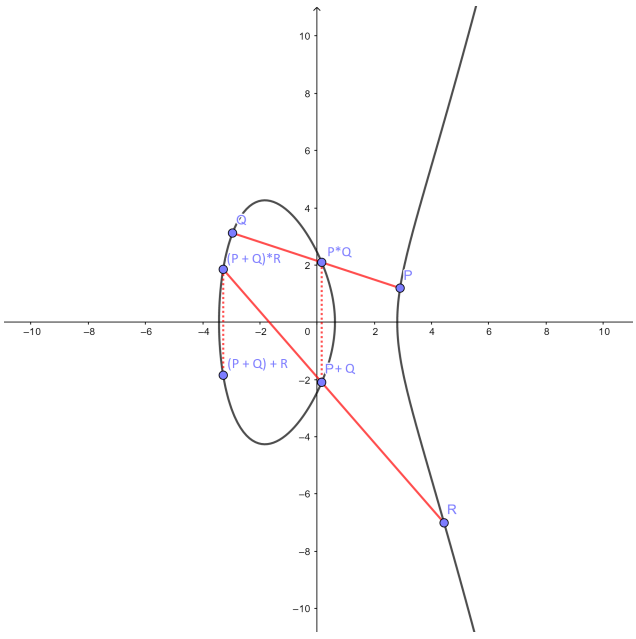


FIGURE 12. $(P + Q) + R$ sur $y^2 = x^3 - 10x + 5$

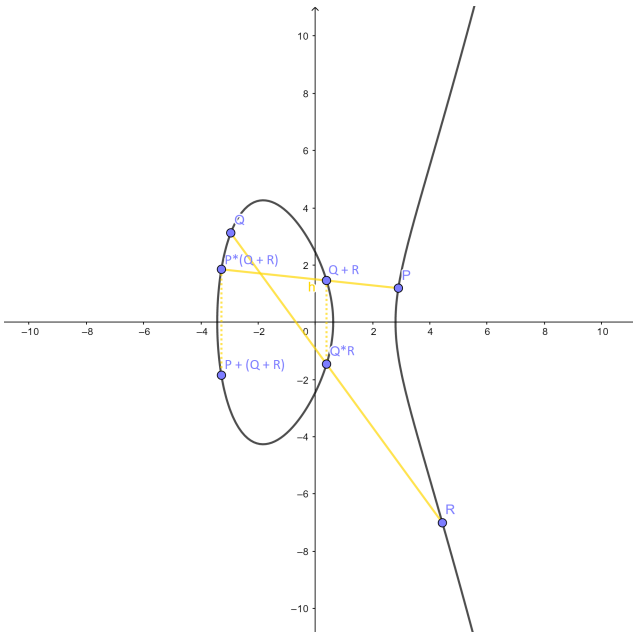


FIGURE 13. $P + (Q + R)$ sur $y^2 = x^3 - 10x + 5$

Démontrons-le par le calcul.

- Supposons dans un premier temps $P$, $Q$, $R$ distincts et n'ayant pas la même abscisse. On suppose également qu'aucun des points n'est égal au point à l'infini. Ces différents cas seront traités après.

Calculons d'abord $P+Q=I_1$.
La droite passant par $P$ et $Q$ est définie par :

$$(PQ) : y = \frac{y_Q - y_P}{x_Q - x_P}x + \frac{x_Q y_P - x_P y_Q}{x_Q - x_P}$$

Déterminons l'intersection de $(PQ)$ avec $(E)$. On note $A=(s,T)$ ce point.

$$(PQ) \cap (E) : \begin{cases} y = \frac{y_Q - y_P}{x_Q - x_P}x + \frac{x_Q y_P - x_P y_Q}{x_Q - x_P} \\ y^2 = x^3 + ax + b \end{cases}$$

$x_P$, $x_Q$ et $s$ sont solutions de cette intersection. On a ainsi :

$$(x-x_P)(x-x_Q)(x-s) = x^3 + ax + b - \left(\frac{y_Q - y_P}{x_Q - x_P}x + \frac{x_Q y_P - x_P y_Q}{x_Q - x_P}\right)^2$$

Par identification, on obtient :

$$\left(\frac{y_Q - y_P}{x_Q - x_P}\right)^2 = x_P + x_Q + s \iff s = \left(\frac{y_Q - y_P}{x_Q - x_P}\right)^2 - x_P - x_Q$$

et on a :

$$\begin{aligned} T &= \frac{y_Q - y_P}{x_Q - x_P}s + \frac{x_Q y_P - x_P y_Q}{x_Q - x_P} \\ &= \frac{y_Q - y_P}{x_Q - x_P}\left(\left(\frac{y_Q - y_P}{x_Q - x_P}\right)^2 - x_P - x_Q\right) + \frac{x_Q y_P - x_P y_Q}{x_Q - x_P} \\ &= -\frac{y_Q - y_P}{x_Q - x_P}(x_P + x_Q) + \left(\frac{y_Q - y_P}{x_Q - x_P}\right)^3 + \frac{x_Q y_P - x_P y_Q}{x_Q - x_P} \end{aligned}$$

On a alors $A=(s,T)=P*Q$ et donc si on note $I_1=P+Q$, le symétrique de $A$ par rapport à l'axe des abscisses, on a $I_1=(s,-T)=(s,t)$ où :

$$\begin{aligned} s &= \left(\frac{y_Q - y_P}{x_Q - x_P}\right)^2 - (x_P + x_Q) \\ t &= \frac{y_Q - y_P}{x_Q - x_P}(x_P + x_Q) - \left(\frac{y_Q - y_P}{x_Q - x_P}\right)^3 + \frac{x_P y_Q - x_Q y_P}{x_Q - x_P} \end{aligned}$$

On note $(P+Q)+R=I_2$ avec $I_2=(u,v)$.
Par le même raisonnement que fait précédemment, on obtient :

$$\begin{aligned} u &= \left(\frac{y_R - t}{x_R - s}\right)^2 - (x_R + s) \\ v &= \frac{y_R - t}{x_R - s}(x_R + s) - \left(\frac{y_R - t}{x_R - s}\right)^3 + \frac{s \times y_R - t \times x_R}{x_R - s} \end{aligned}$$

On a donc trouvé, sans développer, $(P+Q)+R$.
On cherche donc à calculer $P+(Q+R)$.
On note $Q+R=L_1$ où $L_1=(d,e)$.
Après calculs, on obtient :

$$\begin{aligned} d &= \left(\frac{y_R - y_Q}{x_R - x_Q}\right)^2 - (x_R + x_Q) \\ e &= \frac{y_R - y_Q}{x_R - x_Q}(x_R + x_Q) - \left(\frac{y_R - y_Q}{x_R - x_Q}\right)^3 + \frac{x_Q y_R - y_Q x_R}{x_R - x_Q} \end{aligned}$$

On note $P+(Q+R)=L_2$ où $L_2=(f,g)$.

$$\begin{aligned} f &= \left(\frac{e - y_P}{d - x_P}\right)^2 - (x_P + d) \\ g &= \frac{e - y_P}{d - x_P}(x_P + d) - \left(\frac{e - y_P}{d - x_P}\right)^3 + \frac{x_P \times e - d \times y_P}{d - x_P} \end{aligned}$$

Pour savoir si on a l'égalité $(P+Q)+R = P+(Q+R)$, nous allons utiliser SageMath, un logiciel de calcul. On écrit le programme ci-dessous.

```
#fonction permettant d'additionner 2 points
def addition(M,N): # M=(x1,y1), N=(x2,y2)
    x=((N[1]-M[1])/(N[0]-M[0]))**2-(M[0]+N[0]) #((y_2-y_1)\(x_2-x_1))^2-(x_1+x_2)
    y=-((N[1]-M[1])/(N[0]-M[0]))**3+((N[1]-M[1])/(N[0]-M[0]))*(2*M[0]+N[0])-M[1]
    return (x,y)

var('xp','yp','xq','yq','xr','yr') #initialise les variables
P=(xp,yp), Q=(xq,yq), R=(xr,yr)  #associe les coordonnees a chaque point

I_1=addition(P,Q) #P+Q = I_1
I_2=addition(I_1,R) #(P+Q)+R = I_2 = (u,v)
L_1=addition(Q,R) #(Q+R) = L_1
L_2=addition(P,L_1) #P+(Q+R) = L_2 = (f,g)

factor(L_2[0]-I_2[0]) # decompose (f-u) en facteurs irreductibles
factor(L_2[1]-I_2[1]) # meme chose pour (g-v)
```

La fonction $factor()$ permet de décomposer nos variables en produits de facteurs irréductibles. On obtient une longue expression mais on constate que : $x_{L_2} - x_{I_2} = \frac{r \times C}{a^2 \times b^2}$ et $y_{L_2} - y_{I_2} = \frac{r \times D}{a^3 \times b^3}$ où $C$ et $D$ des polynômes et
$r = (x_Q - x_R)y_P^2 + (x_R - x_P)y_Q^2 + (x_P - x_Q)y_R^2 + (x_P - x_Q)(x_Q - x_R)(x_R - x_P)(x_P + x_Q + x_R)$
$a = (y_P - y_Q)^2 - (x_P - x_Q)^2(x_P + x_Q + x_R)$
$b = (y_Q - y_R)^2 - (x_Q - x_R)^2(x_P + x_Q + x_R)$.
On observe que nos deux expressions ont $r$ comme facteur commun, cela veut dire que s'il est égal à 0, on aura montré l'associativité.
Nous n'avons pas encore utilisé le fait que $P, Q, R \in (E)$. On a donc le système suivant :

$$\begin{cases} {y_P}^2 = {x_P}^3 + ax_P + b \\ {y_Q}^2 = {x_Q}^3 + ax_Q + b \\ {y_R}^2 = {x_R}^3 + ax_R + b \end{cases}$$

On remplace les équations de notre système dans $r$ :

$$\begin{aligned} r =& (x_Q - x_R)y_P^2 + (x_R - x_P)y_Q^2 + (x_P - x_Q)y_R^2 + (x_P - x_Q)(x_Q - x_R)(x_R - x_P)(x_P + x_Q + x_R) \\ =& (x_Q - x_R)({x_P}^3 + ax_P + b) + (x_R - x_P)(x_Q^3 + ax_Q + b) + (x_P - x_Q)(x_R^3 + ax_R + b) \\ &+ (x_P - x_Q)(x_Q - x_R)(x_R - x_P)(x_P + x_Q + x_R) \\ =& 0 \end{aligned}$$

Comme $r$ est un facteur commun à $f-u$ et $g-v$, ces quantités sont toutes les deux égales à 0. On a donc $P+(Q+R)-((P+Q)+R) = (0,0)$ avec $-$ la soustraction usuelle, d'où l'associativité.

- On considère le cas où $P = Q$.
  - Soit $y_p \neq 0$ :
  On utilise la formule (3) trouvée précédemment .
  On a : $P+P = (m^2 - 2x_P, (3x_P - m^2)m - y_P)$ où $m = \dfrac{3x_P^2 + a}{2y_P}$.
  On crée une nouvelle fonction qui permet de calculer $2P$ sur SageMath.

```
def double(P):
    #permet de calculer P+P
    var('a') # vient de la formule y^2=x^3+ax+b
    m=(3*P[0]**2+a)/(2*P[1])
    x=m**2-2*P[0]
    y=-P[1]+(3*P[0]-m**2)*m
    return (x,y)

var('xp','yp','xr','yr')
P=(xp,yp), R=(xr,yr)
I_1=double(P) #P+P
I_2=addition(I_1,R) #(P+P)+R
L_1=addition(P,R) #P+R
L_2=addition(P,L_1) #P+(P+R)

factor(L_2[0]-I_2[0]) # factor( x_L1 - x_I2)
factor(L_2[1]-I_2[1]) # factor( y_L1 - y_I2)
```

Soit $(P+(P+R))-((P+P)+R)=(x,y)$, avec $-$ la soustraction usuelle. On observe que $x$ et $y$ admettent pour facteur commun la quantité $x_P^3 - x_R^3 + ax_P - ax_R - y_P^2 + y_R^2$

$$\begin{cases} {y_P}^2 = {x_P}^3 + ax_P + b \\ {y_R}^2 = {x_R}^3 + ax_R + b \end{cases}$$

Lorsqu'on utilise les équations ci-dessus, on obtient $x=0$ et $y=0$. D'où l'associativité

-Soit $y_p = 0$, d'une part on a $(P+P)+R = O+R = R$ et d'autre part en utilisant SageMath et en remplaçant $y_p$ par 0 dans le programme précédent, on obtient directement $x_{P+(P+R)} = x_R$ et $y_{P+(P+R)} = y_R$. On a bien $(P+P)+R = P+(P+R)$.

- On considère le cas où $Q=-P$.
  On a : $(P+(-P))+R = O+R = R$.
  On utilise Sagemath pour calculer $P+(-P+R) = I_2$ où $I_2 = (x_{I_2}, y_{I_2})$

```
from sympy import together
var('xP','yP','xR','yR')
P=(xP,yP)
Q=(xP,-yP) # Q = -P
R=(xR,yR)

# on reutilise la meme fonction addition que dans les programmes precedents
I_1=addition(Q,R) # Q+R = - Q+R
I_2=addition(P,I_1) # P+(Q+R) = P+(-P+R)

together(I_2[0]) # simplifie x_I2
together(I_2[1]) # simplifie y_I2
```

  Les résultats de SageMath nous donnent directement :
  $x_{I_2} = x_R$
  $y_{I_2} = \dfrac{x_P(y_P+y_R) - x_R(y_P+y_R) - y_P(x_P-x_R)}{x_P - x_R} = y_R$
  Ainsi $(P+(-P))+R = P+(-P+R)$

- On considère le cas où $Q=R$. Il revient au même que celui de $P=Q$. En effet, $(P+Q)+R = R+(P+Q) = R+(Q+P) = R+(R+P)$. A chaque égalité, on a utilisé la commutativité sauf à la dernière égalité où on a juste changé $Q$ par $R$.

- On considère le cas où $Q=-R$. Il revient au même que celui de $P=-Q$, par le même raisonnement que ci-dessus.

- Si un des points est égal au point à l'infini, il est facile de montrer l'associativité.
  Si $P=O$, alors $(O+Q)+R = Q+R$ et $O+(Q+R) = Q+R$, on a bien $(O+Q)+R = O+(Q+R)$.
  Si $Q=O$, alors $(P+O)+R = P+R$ et $P+(O+R) = P+R$, on a bien $(P+O)+R = P+(O+R)$.

□

On rappelle que $\mathcal{A}$ est l'union du point à l'infini $O$ et de l'ensemble des points de la courbe elliptique $(E): y^2 = x^3 + ax + b$.

**Théorème 1.** *L'ensemble $\mathcal{A}$ muni de la loi $+$ est un groupe abélien.*

*Démonstration.* Tous les axiomes ont été démontrés : l'associativité, l'existence d'un élément neutre et chaque élément admet un symétrique. On a également montré la commutativité. □

# 2 Courbes elliptiques sur $\mathbb{F}_p$

## 2.1 Quelques généralités

**Proposition 4.** *L'anneau $\frac{\mathbb{Z}}{p\mathbb{Z}}$, où $p$ est premier, est un corps fini que l'on notera $\mathbb{F}_p$.*

*Démonstration.* $\frac{\mathbb{Z}}{p\mathbb{Z}}$ est un anneau commutatif. Soit $x, y$ des entiers de 0 à $p-1$, représentant des classes $[x]$ et $[y]$ de $\mathbb{F}_p$, $xy \equiv 0[p]$ si et seulement si $xy$ est divisible par $p$, mais $p$ est premier. D'après Gauss, $p$ divise $x$ ou $p$

divise $y$. Comme $x$ et $y$ sont inférieurs stricts à $p$, on a forcément $x \equiv 0[p]$ ou $y \equiv 0[p]$. Donc pour tout $x, y \in \mathbb{F}_p$, $xy \equiv 0[p] \Leftrightarrow x \equiv 0[p]$ ou $y \equiv 0[p]$, c'est à dire $\mathbb{F}_p$ est intègre. Comme $\mathbb{F}_p$ est fini, car il admet un nombre fini d'éléments, on utilise la proposition suivante vu en cours d'anneaux : tout anneau commutatif intègre fini est un corps. Donc $\mathbb{F}_p$ est un corps fini.

□

**Définition 5.** *Une courbe elliptique $E(\mathbb{F}_p)$ est l'ensemble des $\mathbb{F}_p$-points qui vérifient l'équation $y^2 = x^3 + ax + b$ où $a, b \in \mathbb{F}_p$ et tel que $4a^3 + 27b^2 \neq 0$ avec $p \neq 2, 3$.*

**Remarque 1.** *$p \neq 2, 3$ car sinon on aurait 2 ou 3 comme caractéristique de $\mathbb{F}_p$. Les changements de variables effectués ne seraient donc plus les mêmes. En effet, dans le premier changement de variable (1), on divise par 2, puis par 3 dans le second (2). Or si la caractéristique est 2 ou 3, cela revient à diviser par 0.*

Pour la suite, on considère le corps $\mathbb{F}_p$ avec $p > 3$.

On note l'ensemble des points de $E$ à coordonnées dans $\mathbb{F}_p$ :

$$E(\mathbb{F}_p) : \{(x, y) : x, y \in \mathbb{F}_p \,|\, y^2 = x^3 + ax + b\} \cup O$$

On a le même théorème que dans $\mathbb{R}$.

**Théorème 2.** *$(E(\mathbb{F}_p), +)$ est un groupe, où $+$ est l'addition que l'on a définie sur les courbes elliptiques.*

**Remarque 2.** *La démonstration est la même que précédemment, à l'exception qu'ici on travaille sur $\mathbb{F}_p$.*

On peut chercher les points de $E(\mathbb{F}_p)$ manuellement. En voici un exemple.

**Exemple 3.** *On se donne une courbe elliptique $y^2 = x^3 + 4x + 2$ dans $\mathbb{F}_7$, et on va chercher les points $P = (x, y) \in E(\mathbb{F}_7)$ avec $x, y \in \mathbb{F}_7$.*

*Après calculs (cf Annexe A), on trouve 9 points : $A = (0, 3)$, $B = (0, 4)$, $C = (1, 0)$, $D = (2, 2)$, $E = (2, 5)$, $F = (5, 0)$, $G = (6, 2)$, $H = (6, 5)$ et $O$ qui est le point à l'infini. Voici le tableau de la loi $+$ de ces 9 éléments.*

TABLE 1 – représentation de la loi $+$

| *Somme* | $O$ | $A$ | $B$ | $C$ | $D$ | $E$ | $F$ | $G$ | $H$ |
|---|---|---|---|---|---|---|---|---|---|
| $O$ | $O$ | $A$ | $B$ | $C$ | $D$ | $E$ | $F$ | $G$ | $H$ |
| $A$ | $A$ | $E$ | $O$ | $C$ | $B$ | $H$ | $G$ | $D$ | $F$ |
| $B$ | $B$ | $O$ | $D$ | $C$ | $G$ | $A$ | $H$ | $F$ | $E$ |
| $C$ | $C$ | $C$ | $C$ | $O$ | $C$ | $C$ | $C$ | $C$ | $C$ |
| $D$ | $D$ | $B$ | $G$ | $C$ | $F$ | $O$ | $E$ | $E$ | $A$ |
| $E$ | $E$ | $B$ | $A$ | $C$ | $O$ | $F$ | $D$ | $B$ | $G$ |
| $F$ | $F$ | $G$ | $H$ | $C$ | $E$ | $D$ | $O$ | $A$ | $B$ |
| $G$ | $G$ | $D$ | $F$ | $C$ | $E$ | $B$ | $A$ | $E$ | $O$ |
| $H$ | $H$ | $F$ | $E$ | $C$ | $A$ | $G$ | $B$ | $O$ | $D$ |

*Et voici la courbe elliptique :*

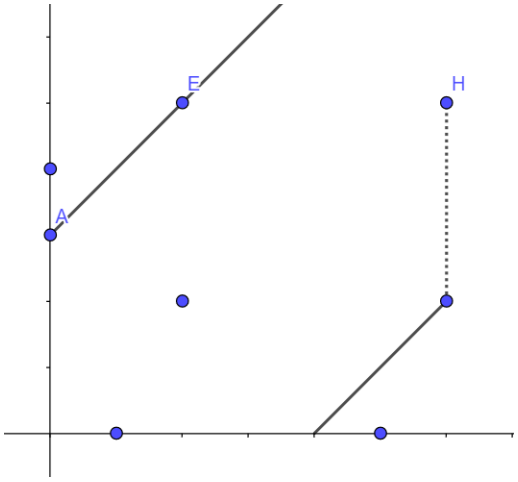


FIGURE 14. On peut trouver l'addition géométriquement, ici $A + E = H$

## 2.2 Problème du logarithme elliptique discret sur $E(\mathbb{F}_p)$

**Définition 6.** *Soient $E$ une courbe elliptique sur $\mathbb{F}_p$, un point $P \in E(\mathbb{F}_p)$ et un point $Q$ tel que $Q = nP$ où $nP = \overbrace{P + P + \ldots + P}^{n fois}$.*
*Le problème du logarithme discret est de trouver le $n$ de l'équation ci-dessus. On note $n =: log_P(Q)$ le logarithme elliptique discret de $Q$ par rapport à $P$*

Ce problème est très utilisé en cryptographie car il est très compliqué de trouver le $n$ et il n'existe pas encore d'algorithme assez efficace pour le résoudre lorsque $n$ est très grand.

Il s'avère rapidement très complexe d'additionner $n$ fois le point $P$ où $n >> 0$. Pour remédier à cette situation, on va créer l'algorithme qui suit.

**Définition 7.** *Le Double-and-add est un algorithme permettant de calculer $Q = nP$ à partir de $n \in \mathbb{N}$ et $P \in E(\mathbb{F}_p)$. Si $n$ a pour décomposition $n = n_0 + n_1 \times 2^1 + n_2 \times 2^2 + .... + n_r \times 2^r$ avec $n_i = \{0, 1\}$.*
*Alors $n$ est égal à $n_0 n_1 n_2 ... n_r$ en base 2. L'algorithme va se servir de cette écriture binaire :*

$$nP = (\sum_{i=0}^{r} 2^i n_i)P = (n_0 + 2 \times (n_1 + ...2 \times (n_{r-2} + 2 \times (n_{r-1} + 2 \times n_r))))P$$
$$= (n_0 P + 2 \times (n_1 P + ...2 \times (n_{r-2} P + 2 \times (n_{r-1} P + 2 \times n_r P))))$$

Voici l'algorithme :

```
def daa (self,n,pt): # Q=nP
    """algorithme double-and-add"""
    assert n>=1
    assert self.app(pt)==True
    Q=pt.copie()
    R=pointInf()
    R._init_()
    while n> 0:
        if n%2==1:
            R=self.addition(Q,R)
        Q=self.addition(Q,Q)
        n=n//2 #division entiere
    return R
```

**Exemple 4.** *Soit $E(\mathbb{F}_{17}) : \{(x, y) : x, y \in \mathbb{F}_p \,|\, y^2 = x^3 + 2x + 4\} \cup O$.*
*On veut calculer $13P$, avec $P = (2, 4) \in E(\mathbb{F}_{17})$. On va donc utiliser l'algorithme Double-And-Add. Pour mieux le comprendre, on note les résultats obtenus à chaque étape :*

TABLE 2 – étapes du double-and-add

| | | |
|---|---|---|
| $n = 13$ | $R = O$ | $Q = P = (2, 4)$ |
| $n = 6$ | $R = P = (2, 4)$ | $Q = 2P = (15, 3)$ |
| $n = 3$ | $R = P = (2, 4)$ | $Q = 4P = (0, 15)$ |
| $n = 1$ | $R = 5P = (7, 15)$ | $Q = 8P = (13, 0)$ |
| $n = 0$ | $R = 13P = (16, 16)$ | $Q = 16P = O$ |

*Ainsi, on obtient $13P = (16, 16)$.*

# 3 Cryptographie sur une courbe elliptique

Comme nous l'avons dit précédemment, les courbes elliptiques sont beaucoup utilisées dans le domaine de la cryptographie. On peut notamment faire des échanges de clés avec le procédé Diffie-Hellman. Supposons qu'Alice et Bob veuillent échanger une clé :

**Méthode 1.**
1. *Alice et Bob se mettent d'accord sur une courbe elliptique, un point $P$ de celle-ci et un corps $\mathbb{F}_p$. Ces informations sont publiques.*
2. *Alice choisit un nombre $n_A$ comme clé privée et calcule $n_A P = Q_A$ à l'aide de l'algorithme "double and Add" par exemple.*
   *Bob fait de même et calcule $n_B P = Q_B$.*

3. *Ils s'échangent publiquement les points $Q_A$ et $Q_B$.*
4. *Pour obtenir leur clé privée commune, Alice va calculer $n_A Q_B = n_A n_B P$ et Bob $n_B Q_A = n_B n_A P$. Ils auront donc tous les deux obtenu la clé privée $n_A n_B P$.*

Pour que quelqu'un puisse trouver leur clé, il faudrait qu'il trouve $n_A$ ou $n_B$, ce qui reviendrait à résoudre le problème elliptique du logarithme discret, qui est très complexe pour des nombres très grands.

**Exemple 5.** *Alice et Bob choisissent une équation au hasard dans $\mathbb{F}_{73}$, par exemple $y^2 = x^3 + 39x + 17$. Ils prennent un point quelconque $P = (12, 13)$ qu'ils vont rendre public. Alors, si Alice choisit un nombre privé $n_A = 120$ avec $n_A P = (64, 18)$ et si Bob choisit un nombre privé $n_B = 317$, avec $n_B P = (53, 56)$, et qu'Alice et Bob s'échangent publiquement leur point, ils obtiendront tout les deux $n_A n_B P = (53, 56)$.*

**Remarque 3.** *On pourrait se demander si, connaissant les clés publiques $Q_A$ et $Q_B$ transmises par Alice et Bob, il serait possible de trouver la clé secrète $n_A n_B P$, sans avoir à calculer $n_A$ et $n_B$. C'est le problème elliptique de Diffie-Hellman. Actuellement, nous n'avons trouvé aucun moyen de trouver $n_A n_B P$ sans au préalable avoir résolu le problème elliptique du logarithme discret.*

Il existe également d'autres méthodes de cryptage sur courbe elliptique comme celle d'ElGamal qui permet d'envoyer un message.

# A Calcul de l'exemple 3

(calcul des points) On veut trouver les points de $\mathbb{F}_7$ qui vérifient $y^2 = x^3 + 4x + 2$, pour cela on procède valeur par valeur :

- Si $x = 0$, alors $y^2 \equiv 2[7]$ or $3^2 \equiv 2[7]$ et $4^2 \equiv 2[7]$, donc $A = (0, 3)$ et $B = (0, 4) \in E(\mathbb{F}_7)$.
- Si $x = 1$, alors $y^2 = 1 + 4 + 2 \equiv 0[7]$, $C = (1, 0) \in E(\mathbb{F}_7)$.
- Si $x = 2$, alors $y^2 = 8 + 8 + 2 \equiv 4[7]$ or $2^2 \equiv 4[7]$ et $5^2 \equiv 4[7]$, donc $D = (2, 2)$ et $E = (2, 5) \in E(\mathbb{F}_7)$ .
- Si $x = 3$, alors $y^2 = 27 + 12 + 2 \equiv 6[7]$ n'admet pas de solution sur $\mathbb{F}_7$.
- Si $x = 4$, alors $y^2 = 64 + 16 + 2 \equiv 5[7]$ n'admet pas de solution sur $\mathbb{F}_7$.
- Si $x = 5$, alors $y^2 = 125 + 20 + 2 \equiv 0[7]$, $F = (5, 0) \in E(\mathbb{F}_7)$.
- Et enfin, si $x = 6$, alors $y^2 = 216 + 24 + 2 \equiv 4[7]$ or $2^2 \equiv 4[7]$ et $5^2 \equiv 4[7]$, donc $G = (6, 2)$, $H = (6, 5)$ $\in E(\mathbb{F}_7)$. On a donc trouvé 8 points dans $E(\mathbb{F}_7)$ mais il ne faut pas oublier le point à l'infini.

Maintenant on va calculer la somme de deux points :

- On choisit de calculer $A + C$ : $A = (0, 3)$ et $C = (1, 0)$, alors la droite $(AC)$ est de la forme $y = mx + p$, on a $m = \frac{0-3}{1-0} = -3 \equiv 4[7]$ et donc $x_{A+C} = 4^2 - 1 \equiv 1[7]$ et $y_{A+C} = -(4 + 3 - 4 \times 0) \equiv 0[7]$, d'où $A + C = C$.
- Et on choisit de calculer $A + D$ : $A = (0, 3)$ et $D = (2, 2)$, on calcule $m = \frac{2-3}{2-0}$ or $2 \times 4 \equiv 1[7]$, d'où $m = (-1) \times 4 \equiv 3[7]$ modulo 7. Donc $x_{A+D} = 3^2 - 2 \equiv 0[7]$ et $y_{A+D} = -(3 - 3 \times 0) \equiv 4[7]$, d'où $A + D = B$.

De la même manière, on obtient les autres points, d'où le tableau 1.

# B Programme en python

```
def inv (n,d,p): #n=numerateur, d=denominateur p=modulo
    """permet de calculer l inverse d'une valeur x """

    assert p>0 # condition qui doit etre verifiee
    r=d%p # transforme la valeur sur F(p)
    inv=-1 #inverse du denominateur
    i=0 #condition qui permet de verifier si c'est bien l inverse a*(a)^-1=1
    while i!=1:
        inv=inv+1
        i=(r*inv)%p
    val=n*inv #multiplie le numerateur par l'inverse du den, le dem =1
    return val %p #modulo p pour rester dans f(p)



class point :
    """ initialise et definit les methodes utilisees pour un point"""

    def _init_(self,x,y):
    """initialise un point sur la courbe en lui attribuant ses coordonees"""
        self.x=x
        self.y=y

    def estInf(self):
    """dit que ce n est pas le point a l infini"""
        return False

    def _str_(self):
    """permet d afficher les coordonnees du point"""
        if self.estInf()==True:
            return "O"
        else :
            return "(%d, %d)" % (self.x, self.y)

    def copie(self):
    """copie le point"""
        if self.estInf()==True:
            return pointInf()
        else :
            P=point()
            P._init_(self.x,self.y)
            return P
```

```
""" sous-classe de point"""
class pointInf (point):
"""on definit le point a l infini"""

    def _init_(self):
    """initialise le point a l infini"""
        point._init_(self,None,None) #fait appel a la fonction _init_ de la classe point

    def estInf(self):
    """cette fonction nous permet de verifier que cest le point a l'infini"""
        return True


class courbe:
    """initialise et donne les methodes utilisees
    pour les courbes elliptiques"""

    def _init_ (self,a,b,p):
    """Initialise une courbe elliptique y^2=x^3+ax+b sur F_p"""
        assert p>0 #condition qui doit etre verifiee
        self.a=a
        self.b=b
        self.p=p

    def app (self,p):
        """fonction qui nous dit si un point appartient a E(f_p)"""
        if p.estInf()==True:
            return True
        else :
            y_2=((p.x)**3+self.a*(p.x) + self.b)% self.p  #on fait modulo p car on est sur
                z/pz
            return y_2==(p.y)**2 %self.p #on fait modulo p car on est sur z/pz

        return True

    def addition (self,M,N) :
    """additionne deux points M et N"""
        assert self.app(M)==True # verifie que le point M appartient bien a la courbe
        assert self.app(N)==True

        if M.estInf()==True: #cas ou le premier point est le point a l infini
            return N.copie()

        elif N.estInf()==True: #cas ou le deuxieme point est le point a l infini
            return M.copie()

        elif (M.x==N.x) and (M.y!=N.y) : #cas ou on a une droite verticale, le y_N est l
            inverse de y_M modulo p
            O=pointInf()
            pointInf._init_(O)
            return O

        elif (M.x==N.x) and(M.y==N.y) :#cas ou M=N
            if M.y!= 0:
                Xlamb=3*(M.x)**2+ self.a #numerateur du lambda
                Ylamb=2*(M.y) #denominateur du lambda
                lamb=inv(Xlamb,Ylamb,self.p) #calcule l inverse de Xlamb/Ylamb
                x=(lamb**2-2*M.x) %self.p
                y=(lamb*(M.x-x)-M.y) %self.p
                P=point()
                P._init_(x,y)
                return P
            else: #tangente parallele a l axe des y
                O=pointInf()
                pointInf._init_(O)
                return O

        else:
            Xlamb=(N.y-M.y) #numerateur du lambda
            Ylamb=(N.x-M.x) #denominateur du lambda
            lamb=inv(Xlamb,Ylamb,self.p) #calcule l inverse de Xlamb/Ylamb
            x=((lamb)**2-(M.x+N.x)) % self.p
            y=((-1)*(lamb)**3+lamb*(2*M.x+N.x)-M.y) %self.p
            P=point()
```

```
            P._init_(x,y)
            return P

    def daa (self,n,pt):
        """algorithme double-and-add: permet de calculer Q=nP"""
        assert n>=1
        assert self.app(pt)==True

        Q=pt.copie()
        R=pointInf()
        R._init_()
        while n> 0:
            if n%2==1:
                R=self.addition(Q,R)
            Q=self.addition(Q,Q)
            n=n//2 #division entiere
        return R



def diffie_hellman(pt,E):
"""permet de faire un echange de cles """
    nA=input("Alice choisit un nombre prive: " )
    QA=E.daa(int(nA),pt) #QA=nA*pt
    print("Elle envoie un point QA=nA*P comme cle publique :", (QA)._str_())
    nB=input("Bob choisit un nombre prive: " )
    QB=E.daa(int(nB),pt) #QB=nB*pt
    print("Il envoie un point QB=nB*P comme cle publique :", QB._str_())
    print("Pour avoir leur cle prive commune, Alice va calculer nA*QB et Bob nB*QA. Ici on
        obtient :",(E.daa(int(nA),QB))._str_())
    return E.daa(int(nB),QA) #renvoie la cle prive commune de Bob et Alice.



"""
lignes de code qui nous ont permis de verifier les calculs de l exemple 3 que nous avons fait
    a la main

A=point()
point._init_(A,0,3)
B=point()
point._init_(B,0,4)
C=point()
point._init_(C,1,0)
D=point()
point._init_(D,2,2)
E=point()
point._init_(E,2,5)
F=point()
point._init_(F,5,0)
G=point()
point._init_(G,6,2)
H=point()
point._init_(H,6,5)


e=courbe() #definit que e est de type courbe
courbe._init_(e,4,2,7) #initialise la courbe
print(e.addition(H,A)._str_() )
"""
"""
teste l algorithme double-and-add sur l exemple 4

e=courbe()
courbe._init_(e,2,4,17)
P=point()
P._init_(2,4)
print(e.daa(13,P)._str_())
"""
"""
teste l echange de cles Diffie Hellman sur une courbe e

e=courbe()
courbe._init_(e,4,2,7)
print(diffie_hellmann(P,e)) #affiche la cle secrete commune de Alice et Bob
"""
```