

Chaînes d'addition Euclidiennes Appliquées à la Multiplication de Points sur les Courbes Elliptiques

Nicolas Méloni

ARITH-LIRMM, Université Montpellier2, France
I3M, Université Montpellier2, France

23 janvier 2007

- 1 Rappels: cryptographie et courbes elliptiques
- 2 Chaînes d'addition euclidiennes
- 3 Multiplication de points sur les courbes elliptiques
- 4 Conclusion et perspectives

Principes

- une clé publique pour le chiffrement,
- une clé privée pour le déchiffrement,
- retrouver la clé publique à partir de la clé privée est "dur".

Quelques exemples

- factorisation de grands entiers → RSA,
- plus court vecteur dans un réseau → NTRU,
- logarithme discret dans un groupe → ECC ,
- et bien d'autres...

Données

- G un groupe d'ordre m ,
- $P \in G$,
- k un entier inférieur à m ,
- $Q = k \times P$.

Problème

Retrouver k à partir de P et Q .

Complexité

- $O(m^{\frac{1}{2}})$ sur un groupe générique,
- sous exponentielle sur \mathbb{F}_q^* ,
- $O(m^{\frac{1}{2}})$ sur une courbe elliptique (bien choisie).

Définition

- K corps de caractéristique $p > 3$
- $E(K) : y^2 = x^3 + ax + b, a, b \in K$

Formules d'addition

$$P_1 = (x_1, y_1), P_2 = (x_2, y_2), P_3 = P_1 + P_2$$

- $x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2$
- $y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) - y_1$

Multiplication de point

- $k \in \mathbb{N}, P \in E(K)$
- $[k]P = \underbrace{P + \dots + P}_{k \text{ fois}}$

Exemple

- $k = 21 = 10101_2 = b_4b_3b_2b_1b_0$
- Calcul de $21P$

Double-and-add

- Initialisation: $Q \leftarrow P$
- $b_3 = 0$: $Q \leftarrow 2Q$ ($Q = 2P$)
- $b_2 = 1$: $Q \leftarrow 2Q + P$ ($Q = 5P$)
- $b_1 = 0$: $Q \leftarrow 2Q$ ($Q = 10P$)
- $b_0 = 1$: $Q \leftarrow 2Q + P$ ($Q = 21P$)

Complexité

- $k = (b_{t-1} \dots b_0)_2$,
- En moyenne $t - 1$ doublements + $\frac{t}{2}$ additions.

Représentation signée (NAF)

- $k_i \in \{-1, 1\}$
- $k = 31 = 11111_2 = 10000\bar{1}_{NAF}$,
- $t - 1$ doublements + $\frac{t}{3}$ additions.

w-NAF

- $k = 267 = 100001011_2$,
- 2-NAF: $100010\bar{1}0\bar{1}$,
- 3-NAF: 100001003 ,
- 4-NAF: $10001000\bar{5}$,
- 5-NAF: 1000000011 ,

Complexité

- Toujours $t - 1$ doublements + $\frac{t}{w+1}$ additions.

Equation

- $E(K) : Y^2 = X^3 + aXZ^4 + bZ^6$
- $(X, Y, Z) \sim (X t^2, Y t^3, Z t)$ pour tout $t \neq 0$
- $(X, Y, Z) \sim (\frac{X}{Z^2}, \frac{Y}{Z^3})$ en affine

Formules d'addition

$$P_1 = (X_1, Y_1, Z_1), P_2 = (X_2, Y_2, Z_2), P_3 = P_1 + P_2$$

- $X_3 = (Y_2 Z_1^3 - Y_1 Z_2^3)^2 - (X_1 Z_2^2 + X_2 Z_1^2)(X_2 Z_1^2 - X_1 Z_2^2)^2$
- $Y_3 = (Y_2 Z_1^3 - Y_1 Z_2^3)(X_1 Z_2^2 (X_2 Z_1^2 - X_1 Z_2^2)^2 - X_3) - Y_1 Z_2^3 (X_2 Z_1^2 - X_1 Z_2^2)^3$
- $Z_3 = Z_1 Z_2 (X_2 Z_1^2 - X_1 Z_2^2)$

Coût

- 12 multiplications + 4 carrés

Cas où $Z_1 = Z_2 = Z$

$$P_1 = (X_1, Y_1, Z), P_2 = (X_2, Y_2, Z), P_3 = P_1 + P_2$$

- $X_3 = (Y_2 - Y_1)^2 - X_1(X_2 - X_1)^2 - X_2(X_2 - X_1)^2$
- $Y_3 = (Y_2 - Y_1)(X_1(X_2 - X_1)^2 - X_3) - Y_1(X_2 - X_1)^3$
- $Z_3 = Z(X_2 - X_1)$

Cas où $Z_1 = Z_2 = Z$

$$P_1 = (X_1, Y_1, Z), P_2 = (X_2, Y_2, Z), P_3 = P_1 + P_2$$

- $X_3 = (Y_2 - Y_1)^2 - X_1(X_2 - X_1)^2 - X_2(X_2 - X_1)^2$
- $Y_3 = (Y_2 - Y_1)(X_1(X_2 - X_1)^2 - X_3) - Y_1(X_2 - X_1)^3$
- $Z_3 = Z(X_2 - X_1)$

Cas où $Z_1 = Z_2 = Z$

$$P_1 = (X_1, Y_1, Z), P_2 = (X_2, Y_2, Z), P_3 = P_1 + P_2$$

- $X_3 = (Y_2 - Y_1)^2 - X_1(X_2 - X_1)^2 - X_2(X_2 - X_1)^2$
- $Y_3 = (Y_2 - Y_1)(X_1(X_2 - X_1)^2 - X_3) - Y_1(X_2 - X_1)^3$
- $Z_3 = Z(X_2 - X_1)$

Résultat

- $P_3 = (X_3, Y_3, Z_3)$ et $P'_1 = (X_1(X_2 - X_1)^2, Y_1(X_2 - X_1)^3, Z(X_2 - X_1))$,
- $P'_1 \sim P_1$,
- P_3 et P_1 ont la même coordonnée z .

Coût

- 5 multiplications + 2 carrés.

Opérateur NewADD

- Entrée: P_1 et P_2 avec la même coordonnée z
- Sortie: $\text{NewADD}(P_1, P_2) = (P_1 + P_2, P_1)$
où $P_1 + P_2$ et P_1 ont la même coordonnée z
- Coût : $5M + 2C$.

Opérateur NewADD

- Entrée: P_1 et P_2 avec la même coordonnée z
- Sortie: $\text{NewADD}(P_1, P_2) = (P_1 + P_2, P_1)$
où $P_1 + P_2$ et P_1 ont la même coordonnée z
- Coût : $5M + 2C$.

Exemple

- $\text{NewADD}(2P, P) = (3P, 2P)$
- $\text{NewADD}(3P, 2P) = (5P, 3P)$
- $\text{NewADD}(3P, 5P) = (8P, 3P)$
- $\text{NewADD}(8P, 3P) = (11P, 8P)$

Opérateur NewADD

- Entrée: P_1 et P_2 avec la même coordonnée z
- Sortie: $\text{NewADD}(P_1, P_2) = (P_1 + P_2, P_1)$
où $P_1 + P_2$ et P_1 ont la même coordonnée z
- Coût : $5M + 2C$.

Exemple

- $\text{NewADD}(2P, P) = (3P, 2P)$
- $\text{NewADD}(3P, 2P) = (5P, 3P)$
- $\text{NewADD}(3P, 5P) = (8P, 3P)$
- $\text{NewADD}(8P, 3P) = (11P, 8P)$

Comparaisons

- Coord Jacobiennes \rightarrow dbl: $4M+6C$, add: $12M+4C$,
- Coord Jacob modifiées \rightarrow , dbl: $4M+4C$, add: $9M+5C$,
- Courbes Montgomery \rightarrow dbl: $3M+2C$, add: $4M+2C$,
- Montgomery généralisées \rightarrow dbl: $6M + 3C$, add: $9M+2C$.

Opérateur NewADD

- Entrée: P_1 et P_2 avec la même coordonnée z
- Sortie: $\text{NewADD}(P_1, P_2) = (P_1 + P_2, P_1)$
où $P_1 + P_2$ et P_1 ont la même coordonnée z
- Coût : $5M + 2C$.

Exemple

- $\text{NewADD}(2P, P) = (3P, 2P)$
- $\text{NewADD}(3P, 2P) = (5P, 3P)$
- $\text{NewADD}(3P, 5P) = (8P, 3P)$
- $\text{NewADD}(8P, 3P) = (11P, 8P)$

Comparaisons

- Coord Jacobiennes \rightarrow dbl: $4M+6C$, add: $12M+4C$,
- Coord Jacob modifiées \rightarrow , dbl: $4M+4C$, add: $9M+5C$,
- Courbes Montgomery \rightarrow dbl: $3M+2C$, add: $4M+2C$,
- Montgomery généralisées \rightarrow dbl: $6M + 3C$, add: $9M+2C$.

Opérateur NewADD

- Entrée: P_1 et P_2 avec la même coordonnée z
- Sortie: $\text{NewADD}(P_1, P_2) = (P_1 + P_2, P_1)$
où $P_1 + P_2$ et P_1 ont la même coordonnée z
- Coût : $5M + 2C$.

Exemple

- $\text{NewADD}(2P, P) = (3P, 2P)$
- $\text{NewADD}(3P, 2P) = (5P, 3P)$
- $\text{NewADD}(3P, 5P) = (8P, 3P)$
- $\text{NewADD}(8P, 3P) = (11P, 8P)$

Comparaisons

- Coord Jacobiennes \rightarrow dbl: $4M+6C$, add: $12M+4C$,
- Coord Jacob modifiées \rightarrow , dbl: $4M+4C$, add: $9M+5C$,
- Courbes Montgomery \rightarrow dbl: $3M+2C$, add: $4M+2C$,
- Montgomery généralisées \rightarrow dbl: $6M + 3C$, add: $9M+2C$.

Questions fondamentales

- Peut-on calculer n'importe quel point grâce aux formules précédentes ?
- Peut trouver facilement une chaîne d'addition calculant un point donné ?
- Est-ce efficace ?

Réponses

Questions fondamentales

- Peut-on calculer n'importe quel point grâce aux formules précédentes ?
- Peut trouver facilement une chaîne d'addition calculant un point donné ?
- Est-ce efficace ?

Réponses

- Oui

Questions fondamentales

- Peut-on calculer n'importe quel point grâce aux formules précédentes ?
- Peut trouver facilement une chaîne d'addition calculant un point donné ?
- Est-ce efficace ?

Réponses

- Oui
- Oui

Questions fondamentales

- Peut-on calculer n'importe quel point grâce aux formules précédentes ?
- Peut trouver facilement une chaîne d'addition calculant un point donné ?
- Est-ce efficace ?

Réponses

- Oui
- Oui
- Pas clair

La suite de Fibonacci

- $F_0 = 0, F_1 = 1, \forall n \geq 0, F_{n+2} = F_{n+1} + F_n$
- $0, 1, 1, 2, 3, 5, 8, 13, 21, 34, \dots$

Algorithme

Algorithm 1: Fibonacci(F_n, P)

Data: $P \in E(K)$ et F_n le nième Fibonacci

Result: $[F_n]P \in E$;

begin

$(U, V) \leftarrow (2P, P)$

for $i = 4 \dots n$ **do**

$(U, V) \leftarrow \text{NewADD}(U, V)$;

end

return U

end

Calculer 31

- $1+2 = 3$
- $2+3 = 5$
- $3+5 = 8$
- $5+8 = 13$
- $5+13 = 18$
- $13+18 = 31$

Calculer 31

- $1+2 = 3$
- $2+3 = 5$
- $3+5 = 8$
- $5+8 = 13$
- $5+13 = 18$
- $13+18 = 31$

Euclide(31,18)

- $31 = 18 + 13$
- $18 = 13 + 5$
- $13 = 8 + 5$
- $8 = 5 + 3$
- $5 = 3 + 2$
- $3 = 2 + 1$

Calculer 31

- $1+2 = 3$
- $2+3 = 5$
- $3+5 = 8$
- $5+8 = 13$
- $5+13 = 18$
- $13+18 = 31$

Euclide(31,18)

- $31 = 18 + 13$
- $18 = 13 + 5$
- $13 = 8 + 5$
- $8 = 5 + 3$
- $5 = 3 + 2$
- $3 = 2 + 1$

Euclide(31,21)

- $31 = 21 + 10$ (petit pas)
- $21 = 11 + 10$ (grand pas)
- $11 = 10 + 1$ (petit pas)
- $10 = 9 + 1$ (petit pas)
- $9 = 8 + 1$ (petit pas)
- $8 = 7 + 1$ (petit pas)
- $7 = 6 + 1$ (petit pas)
- $6 = 5 + 1$ (petit pas)
- $5 = 4 + 1$ (petit pas)
- $4 = 3 + 1$ (petit pas)
- $3 = 2 + 1$

Calculer 31

- $1+2 = 3$
- $2+3 = 5$
- $3+5 = 8$
- $5+8 = 13$
- $5+13 = 18$
- $13+18 = 31$

Euclide(31,18)

- $31 = 18 + 13$
- $18 = 13 + 5$
- $13 = 8 + 5$
- $8 = 5 + 3$
- $5 = 3 + 2$
- $3 = 2 + 1$

Euclide(31,21)

- $31 = 21 + 10$ (petit pas)
- $21 = 11 + 10$ (grand pas)
- $11 = 10 + 1$ (petit pas)
- $10 = 9 + 1$ (petit pas)
- $9 = 8 + 1$ (petit pas)
- $8 = 7 + 1$ (petit pas)
- $7 = 6 + 1$ (petit pas)
- $6 = 5 + 1$ (petit pas)
- $5 = 4 + 1$ (petit pas)
- $4 = 3 + 1$ (petit pas)
- $3 = 2 + 1$

Conclusion

Il suffit de choisir un entier k' premier avec k pour obtenir une chaîne calculant k .
On identifiera une chaîne d'addition calculant un entier k au vecteur
 $c = (c_1, \dots, c_n) \in \{\text{'grand pas'}, \text{'petit pas'}\}^n$ correspondant à la chaîne.

Algorithme

Algorithm 2: Euclid-Exp(c, P)

Data: $P \in E(K)$ une chaîne $c = (c_1, \dots, c_l)$ calculant k ;

Result: $[k]P \in E$;

begin

$(U, U) \leftarrow (2P, P)$

for $i = 1 \dots l$ **do**

if $c_i = \text{'grand pas'}$ **then**

$(U, V) \leftarrow \text{NewADD}(U, V)$;

else

$(U, V) \leftarrow \text{NewADD}(V, U)$;

end

end

$(U, V) \leftarrow \text{NewADD}(U, V)$;

return U

end

- Algorithme résistant aux attaques par canaux cachés,
- Aucun précalcul,
- coût: 1 doublement + l additions.

Calculer $31P$ avec $c = (\textit{grand}, \textit{grand}, \textit{grand}, \textit{petit}, \textit{grand})$

Initialisation. $(U, V) \leftarrow (2P, P)$

- $c_1 = \textit{grand pas}$: $(U, V) \leftarrow \text{NewADD}(2P, P) = (3P, 2P)$
- $c_2 = \textit{grand pas}$: $(U, V) \leftarrow \text{NewADD}(3P, 2P) = (5P, 3P)$
- $c_3 = \textit{grand pas}$: $(U, V) \leftarrow \text{NewADD}(5P, 3P) = (8P, 5P)$
- $c_4 = \textit{petit pas}$: $(U, V) \leftarrow \text{NewADD}(5P, 8P) = (13P, 5P)$
- $c_5 = \textit{grand pas}$: $(U, V) \leftarrow \text{NewADD}(13P, 5P) = (18P, 13P)$
- $(U, V) \leftarrow \text{NewADD}(18P, 13P)$
- $U = 31P$

Bref rappel

- $q \in [0, 1[$, $q = \frac{k'}{k}$, $q = \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$,

- On note $q = [0, a_1, a_2, a_3, \dots]$,
- si $q \in \mathbb{Q}$ alors sa decomp en FC est finie: $q = [0, a_1, a_2, \dots, a_n]$,
- dans ce cas les a_i sont exactement les quotients partiels de l'algo d'euclide appliqué à k et k' .

Fractions continues/chaînes euclidiennes

- On lit la chaîne de a_n à a_2 ,
- si $i < n$ $a_i \rightarrow 1$ grand pas et $a_i - 1$ petits pas,
- $a_n \rightarrow a_i - 2$ petits pas.

$$\frac{18}{31} = [0, 1, 1, 2, 1, 1, 2]$$

- [0,
- 1,
- 1, → (grand pas,
- 2, → petit pas, grand pas,
- 1, → grand pas,
- 1, → grand pas,
- 2] →)

Conclusion

- 31 = (grand, grand, grand, petit, grand)

Théorème (Knuth, Yao 1975)

Nombre moyen d'étapes de l'algorithme d'Euclide soustractif :

$$I(k) = \mathcal{O}(\ln(k)^2)$$

Conjecture de Zaremba

Pour tout k il existe k' premier avec k tel que:

$$\frac{k'}{k} = [a_0, a_1, \dots, a_n] \text{ avec } \forall i, a_i \leq 3$$

"Sur-conjecture"

En admettant la conjecture précédente alors pour k il existe k' tel que:

$$\sum a_i < 1.77 \log_2(k)$$

Techniques de réduction

- Choisir k' proche de $\frac{k}{\phi}$ où $\phi = \frac{1+\sqrt{5}}{2}$ est le nombre d'or.
(Assure que les premiers quotients partiels sont égaux à 1)
- Tester plusieurs k' !

Nombre d'essais pour trouver une chaîne "raisonnable":

Longueur de la chaîne	320	300	280	260
cas moyent	29	121	2353	7,795,840
cas pire	521	3,454	44,254	79,402,210

D'après la sur-conjecture

On peut espérer des chaînes de tailles ≤ 280 .

Représentation de Zeckendorf

Tout entier k peut s'écrire comme somme de nombres de Fibonacci:

$$k = \sum_{i=2}^n d_i F_i \text{ avec } \forall i \geq 2, d_i d_{i+1} = 0$$

Exemple

- $31 = 21 + 8 + 2$
- $= F_8 + F_6 + F_3$
- $= 1010010_Z$

$$k = 31$$

$$31 = 1010010_{\mathcal{Z}} = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$

$$k = 31$$

$$31 = 1010010_Z = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$
- $d_6 = 1$: $F_3 + 1$ puis $(F_3 + 1) + F_2 = F_4 + 1$

$k = 31$

$$31 = 1010010_Z = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$
- $d_6 = 1$: $F_3 + 1$ puis $(F_3 + 1) + F_2 = F_4 + 1$
- $d_5 = 0$: $(F_4 + 1) + (F_3 + 1) = F_5 + F_3$

$k = 31$

$$31 = 1010010_Z = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$
- $d_6 = 1$: $F_3 + 1$ puis $(F_3 + 1) + F_2 = F_4 + 1$
- $d_5 = 0$: $(F_4 + 1) + (F_3 + 1) = F_5 + F_3$
- $d_4 = 0$: $(F_5 + F_3) + (F_4 + 1) = F_6 + F_4$

$k = 31$

$$31 = 1010010_Z = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$
- $d_6 = 1$: $F_3 + 1$ puis $(F_3 + 1) + F_2 = F_4 + 1$
- $d_5 = 0$: $(F_4 + 1) + (F_3 + 1) = F_5 + F_3$
- $d_4 = 0$: $(F_5 + F_3) + (F_4 + 1) = F_6 + F_4$
- $d_3 = 1$: $(F_6 + F_4) + 1$ puis $(F_6 + F_4 + 1) + (F_5 + F_3) = F_7 + F_5 + 1$

$k = 31$

$$31 = 1010010_Z = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$
- $d_6 = 1$: $F_3 + 1$ puis $(F_3 + 1) + F_2 = F_4 + 1$
- $d_5 = 0$: $(F_4 + 1) + (F_3 + 1) = F_5 + F_3$
- $d_4 = 0$: $(F_5 + F_3) + (F_4 + 1) = F_6 + F_4$
- $d_3 = 1$: $(F_6 + F_4) + 1$ puis $(F_6 + F_4 + 1) + (F_5 + F_3) = F_7 + F_5 + 1$
- $d_2 = 0$: $(F_7 + F_5 + 1) + (F_6 + F_4 + 1) = F_8 + F_6 + F_3$

$k = 31$

$$31 = 1010010_Z = d_8 d_7 d_6 d_5 d_4 d_3 d_2$$

Fibonacci-and-add

- Initialisation: $F_2 = 1, F_1 = 1,$
- $d_7 = 0$: $F_2 + F_1 = F_3$
- $d_6 = 1$: $F_3 + 1$ puis $(F_3 + 1) + F_2 = F_4 + 1$
- $d_5 = 0$: $(F_4 + 1) + (F_3 + 1) = F_5 + F_3$
- $d_4 = 0$: $(F_5 + F_3) + (F_4 + 1) = F_6 + F_4$
- $d_3 = 1$: $(F_6 + F_4) + 1$ puis $(F_6 + F_4 + 1) + (F_5 + F_3) = F_7 + F_5 + 1$
- $d_2 = 0$: $(F_7 + F_5 + 1) + (F_6 + F_4 + 1) = F_8 + F_6 + F_3$

Coût

- k nombre de t bits:
- coût : $1.44 \times t$ Fibonacci et $\frac{1.44 \times t}{3} \simeq \frac{t}{2}$ additions

Représentation signée

- $31 = F_8 + F_6 + F_3 = 21 + 8 + 2 = 34 - 3 = F_9 - F_4$
- $31 = 1000\bar{1}00_Z$

Simplifications

- $F_{n+3} + F_n = 2F_{n+2} \rightarrow 1001_Z = 0200_Z$
- $F_{n+3} - F_n = 2F_{n+1} \rightarrow 100\bar{1}_Z = 0020_Z$
- $F_{n+4} + F_n = 3F_{n+2} \rightarrow 10001_Z = 00300_Z$
- $F_{n+6} - F_n = 4F_{n+3} \rightarrow 100000\bar{1}_Z = 0004000_Z$

Densité

- k nombre de t bits:
nombre de bits non nuls = $\frac{t}{5}$

Nombres d'opérations pour k de t bits

	Binaire	Zeckendorf
Classique	$1.5 \times t$	$1.92 \times t$
4-NAF	$1.2 \times t$	$1.64 \times t$

Mini conclusion

- Entre 28 et 34 % de calculs en plus.

Calcul de kP avec k entier de 160 bits

	Zeckendorf	Coord jacobiennes	Coord mixées	
Double-and-add	2597 M	2470 M	2312 M	
4-NAF	2046 M	1976 M	1727 M	
	CE l=320	CE l=260	Montgomery ladder	4-NAF
Weierstraß	2240 M	1820 M	–	1983
Montgomery	1920 M	1560 M	1749	–

Bilan

- Représentation de Zeckendorf devient presque rentable: entre 5 et 16 % seulement d'opérations en plus.
- La méthode par chaînes euclidiennes est plus rapide à exigence de mémoire équivalente.

Conclusion

- Nouvelles formules d'addition,
- un algo de Fibonacci-and-Add,
- l'exponentiation par chaîne euclidienne offre à la fois vitesse et résistance aux attaques par canaux cachés,
- trouver des chaînes efficaces reste très compliqué.

Perspectives

- Creuser encore la représentation de Zeckendorf,
- mixer les représentations de Zeckendorf et binaires classiques,
- améliorer la recherche de chaînes euclidiennes,
- trouver une "*grande*" classe de nombres pour lesquels une chaîne courte est facile à trouver.