

# Algorithmes pour l'arithmétique II

## Cours 8

Julien Lavauzelle

Université Paris 8

Master 2 ACC – Algorithmes pour l'arithmétique

21/11/2023

**Questions?**

1. Introduction
2. Le crible quadratique
3. Analyse et extensions

Nous avons déjà vu des **méthodes de factorisation d'entiers** de complexité exponentielle : divisions successives, algorithme de Fermat,  $\rho$  de Pollard, méthodes  $p - 1$  et  $p + 1$ , etc.

Nous avons déjà vu des **méthodes de factorisation d'entiers** de complexité exponentielle : divisions successives, algorithme de Fermat,  $\rho$  de Pollard, méthodes  $p - 1$  et  $p + 1$ , etc.

Nous allons maintenant étudier l'algorithme de **crible quadratique** qui permet de factoriser n'importe quel entier  $N$  en temps

$$O\left(\exp(\sqrt{\log N \log \log N})\right)$$

Nous avons déjà vu des **méthodes de factorisation d'entiers** de complexité exponentielle : divisions successives, algorithme de Fermat,  $\rho$  de Pollard, méthodes  $p - 1$  et  $p + 1$ , etc.

Nous allons maintenant étudier l'algorithme de **crible quadratique** qui permet de factoriser n'importe quel entier  $N$  en temps

$$O\left(\exp(\sqrt{\log N \log \log N})\right)$$

Son extension, l'algorithme de **crible algébrique** (ou crible par corps de nombres généralisé) (*general number field sieve*, NFS) atteint une complexité encore meilleure :

$$O\left(\exp\left(\left(\frac{64}{9} \log N\right)^{1/3} (\log \log N)^{2/3}\right)\right).$$

Nous avons déjà vu des **méthodes de factorisation d'entiers** de complexité exponentielle : divisions successives, algorithme de Fermat,  $\rho$  de Pollard, méthodes  $p - 1$  et  $p + 1$ , etc.

Nous allons maintenant étudier l'algorithme de **crible quadratique** qui permet de factoriser n'importe quel entier  $N$  en temps

$$O\left(\exp(\sqrt{\log N \log \log N})\right)$$

Son extension, l'algorithme de **crible algébrique** (ou crible par corps de nombres généralisé) (*general number field sieve*, NFS) atteint une complexité encore meilleure :

$$O\left(\exp\left(\left(\frac{64}{9} \log N\right)^{1/3} (\log \log N)^{2/3}\right)\right).$$

**Remarque.** Ces complexités ne sont ni exponentielles, ni polynomiales en  $\log_2(N)$ .

**Objectif :** représenter des grandeurs sous-exponentielles  $n^\alpha \ll f(n) \ll 2^{\beta n}$ .



**Objectif :** représenter des grandeurs sous-exponentielles  $n^a \ll f(n) \ll 2^{\beta n}$ .

**Définition.** Notation  $L$  :

$$L_n[a, b] := \exp \left( b n^a (\log n)^{1-a} \right).$$

On prendra souvent exp et log en base 2.

**Objectif :** représenter des grandeurs sous-exponentielles  $n^\alpha \ll f(n) \ll 2^{\beta n}$ .

**Définition.** Notation  $L$  :

$$L_n[a, b] := \exp \left( b n^a (\log n)^{1-a} \right).$$

On prendra souvent exp et log en base 2.

**Exemples :**

- ▶ Les fonctions exponentielles  $2^{\beta n}$  sont des  $L_n[1, \beta]$ .
- ▶ Les fonctions polynomiales  $n^\alpha$  sont des  $L_n[0, \alpha]$ .

**Objectif :** représenter des grandeurs sous-exponentielles  $n^\alpha \ll f(n) \ll 2^{\beta n}$ .

**Définition.** Notation  $L$  :

$$L_n[a, b] := \exp \left( b n^a (\log n)^{1-a} \right).$$

On prendra souvent exp et log en base 2.

**Exemples :**

- ▶ Les fonctions exponentielles  $2^{\beta n}$  sont des  $L_n[1, \beta]$ .
- ▶ Les fonctions polynomiales  $n^\alpha$  sont des  $L_n[0, \alpha]$ .

On va utiliser cette notation pour les algorithmes de factorisation. On s'intéresse donc à leur complexité en fonction de  $\log_2 N$  ou de  $\log_2 p$ . Ainsi :

- ▶ La complexité de la méthode  $\rho$  est en  $O(\sqrt{p}) =$

**Objectif :** représenter des grandeurs sous-exponentielles  $n^\alpha \ll f(n) \ll 2^{\beta n}$ .

**Définition.** Notation  $L$  :

$$L_n[a, b] := \exp \left( b n^a (\log n)^{1-a} \right).$$

On prendra souvent exp et log en base 2.

**Exemples :**

- ▶ Les fonctions exponentielles  $2^{\beta n}$  sont des  $L_n[1, \beta]$ .
- ▶ Les fonctions polynomiales  $n^\alpha$  sont des  $L_n[0, \alpha]$ .

On va utiliser cette notation pour les algorithmes de factorisation. On s'intéresse donc à leur complexité en fonction de  $\log_2 N$  ou de  $\log_2 p$ . Ainsi :

- ▶ La complexité de la méthode  $\rho$  est en  $O(\sqrt{p}) = O(L_{\log p}[1, \frac{1}{2}])$ .

**Objectif :** représenter des grandeurs sous-exponentielles  $n^\alpha \ll f(n) \ll 2^{\beta n}$ .

**Définition.** Notation  $L$  :

$$L_n[a, b] := \exp\left(b n^a (\log n)^{1-a}\right).$$

On prendra souvent exp et log en base 2.

**Exemples :**

- ▶ Les fonctions exponentielles  $2^{\beta n}$  sont des  $L_n[1, \beta]$ .
- ▶ Les fonctions polynomiales  $n^\alpha$  sont des  $L_n[0, \alpha]$ .

On va utiliser cette notation pour les algorithmes de factorisation. On s'intéresse donc à leur complexité en fonction de  $\log_2 N$  ou de  $\log_2 p$ . Ainsi :

- ▶ La complexité de la méthode  $\rho$  est en  $O(\sqrt{p}) = O(L_{\log p}[1, \frac{1}{2}])$ .
- ▶ La complexité de ECM est en  $O(\exp(\sqrt{2 \log p \log \log p})) =$

**Objectif :** représenter des grandeurs sous-exponentielles  $n^\alpha \ll f(n) \ll 2^{\beta n}$ .

**Définition.** Notation  $L$  :

$$L_n[a, b] := \exp\left(b n^a (\log n)^{1-a}\right).$$

On prendra souvent exp et log en base 2.

**Exemples :**

- ▶ Les fonctions exponentielles  $2^{\beta n}$  sont des  $L_n[1, \beta]$ .
- ▶ Les fonctions polynomiales  $n^\alpha$  sont des  $L_n[0, \alpha]$ .

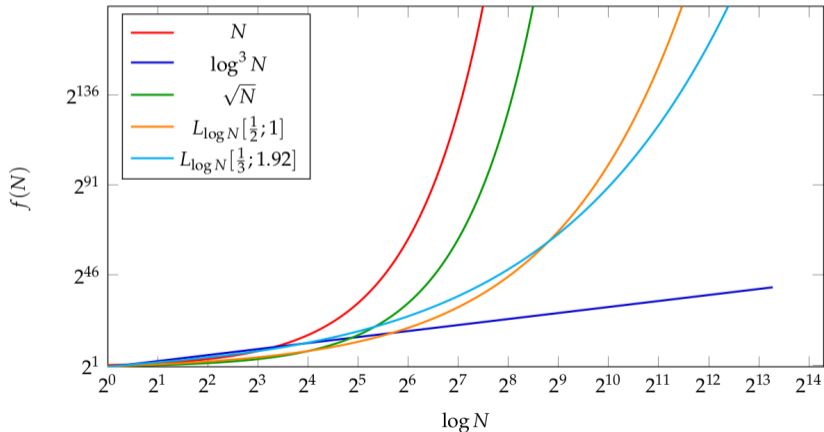
On va utiliser cette notation pour les algorithmes de factorisation. On s'intéresse donc à leur complexité en fonction de  $\log_2 N$  ou de  $\log_2 p$ . Ainsi :

- ▶ La complexité de la méthode  $\rho$  est en  $O(\sqrt{p}) = O(L_{\log p}[1, \frac{1}{2}])$ .
- ▶ La complexité de ECM est en  $O(\exp(\sqrt{2 \log p \log \log p})) = O(L_{\log p}[\frac{1}{2}, \sqrt{2}])$ .

Quelques **exemples** de comportement de fonctions (échelle « log log »).

Rappel :

- $2^{80}$  : très, très difficile à calculer, pour  $\simeq 2^{30}$  op./s sur un processeur
- $2^{128}$  : supposé inatteignable



1. Introduction
2. Le crible quadratique
3. Analyse et extensions



**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Exemple.** Soit  $N = 91$ .

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Exemple.** Soit  $N = 91$ .

On a :

- $4^2 = 16$
- $17^2 = 289 \equiv 16 \pmod{91}$ .

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Exemple.** Soit  $N = 91$ .

On a :

-  $4^2 = 16$

-  $17^2 = 289 \equiv 16 \pmod{91}$ .

Ainsi,  $\text{pgcd}(4 + 17, N)$  donne le diviseur 7 de  $N = 91$ .

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Exemple.** Soit  $N = 91$ .

On a :

-  $4^2 = 16$

-  $17^2 = 289 \equiv 16 \pmod{91}$ .

Ainsi,  $\text{pgcd}(4 + 17, N)$  donne le diviseur 7 de  $N = 91$ .

Et on a également  $\text{pgcd}(17 - 4, N) = 13$ .

**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Exemple.** Soit  $N = 91$ .

On a :

-  $4^2 = 16$

-  $17^2 = 289 \equiv 16 \pmod{91}$ .

Ainsi,  $\text{pgcd}(4 + 17, N)$  donne le diviseur 7 de  $N = 91$ .

Et on a également  $\text{pgcd}(17 - 4, N) = 13$ .

**Remarque.** On obtient un facteur propre lorsque les  $a$  et  $b$  trouvés vérifient  $a \not\equiv \pm b \pmod{N}$ .



**Méthode de Fermat** : si on arrive à écrire  $N = a^2 - b^2$ , alors  $N = (a - b)(a + b)$  donne une factorisation de  $N$ .

Raffinement de l'idée (Kraitchik, puis Dixon) :

**Idée** : Si on obtient « seulement » une relation de divisibilité

$$N \mid a^2 - b^2 = (a - b)(a + b),$$

alors on peut tout de même espérer que  $\text{pgcd}(a - b, N)$  ou  $\text{pgcd}(a + b, N)$  donne un facteur propre de  $N$ .

**Exemple.** Soit  $N = 91$ .

On a :

-  $4^2 = 16$

-  $17^2 = 289 \equiv 16 \pmod{91}$ .

Ainsi,  $\text{pgcd}(4 + 17, N)$  donne le diviseur 7 de  $N = 91$ .

Et on a également  $\text{pgcd}(17 - 4, N) = 13$ .

**Remarque.** On obtient un facteur propre lorsque les  $a$  et  $b$  trouvés vérifient  $a \not\equiv \pm b \pmod{N}$ .

Peut-on trouver  $a^2 \equiv b^2 \pmod{N}$ ?

**Rappel.** On cherche  $a$  et  $b$  tels que

$$N \text{ divise } a^2 - b^2 = (a - b)(a + b).$$

**Est-ce fréquent?**

## Peut-on trouver $a^2 \equiv b^2 \pmod{N}$ ?

**Rappel.** On cherche  $a$  et  $b$  tels que

$$N \text{ divise } a^2 - b^2 = (a - b)(a + b).$$

**Est-ce fréquent?**

**Lemme.** Si  $N$  admet  $t$  diviseurs premiers distincts, alors l'équation  $x^2 \equiv 1 \pmod{N}$  admet  $2^t$  solutions. Parmi elles, il y a les deux solutions  $x = \pm 1$ .

## Peut-on trouver $a^2 \equiv b^2 \pmod{N}$ ?

**Rappel.** On cherche  $a$  et  $b$  tels que

$$N \text{ divise } a^2 - b^2 = (a - b)(a + b).$$

**Est-ce fréquent ?**

**Lemme.** Si  $N$  admet  $t$  diviseurs premiers distincts, alors l'équation  $x^2 \equiv 1 \pmod{N}$  admet  $2^t$  solutions. Parmi elles, il y a les deux solutions  $x = \pm 1$ .

**Conséquence.** Si  $N$  est composé ( $t \geq 2$ ) et si  $a^2 \equiv b^2 \pmod{N}$  ont été trouvés « aléatoirement », alors il y a plus d'une chance sur deux pour que  $a \not\equiv \pm b \pmod{N}$ .

## Peut-on trouver $a^2 \equiv b^2 \pmod{N}$ ?

**Rappel.** On cherche  $a$  et  $b$  tels que

$$N \text{ divise } a^2 - b^2 = (a - b)(a + b).$$

**Est-ce fréquent ?**

**Lemme.** Si  $N$  admet  $t$  diviseurs premiers distincts, alors l'équation  $x^2 \equiv 1 \pmod{N}$  admet  $2^t$  solutions. Parmi elles, il y a les deux solutions  $x = \pm 1$ .

**Conséquence.** Si  $N$  est composé ( $t \geq 2$ ) et si  $a^2 \equiv b^2 \pmod{N}$  ont été trouvés « aléatoirement », alors il y a plus d'une chance sur deux pour que  $a \not\equiv \pm b \pmod{N}$ .

**Question.** Comment trouver  $a, b$  tels que  $a^2 \equiv b^2 \pmod{N}$  ?

# Comment trouver $a^2 \equiv b^2 \pmod N$ ?

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'entiers de la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

**qui se décomposent sur  $\mathcal{P}$** , c'est-à-dire dont les diviseurs premiers sont dans  $\mathcal{P}$



**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'entiers de la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

**qui se décomposent sur  $\mathcal{P}$** , c'est-à-dire dont les diviseurs premiers sont dans  $\mathcal{P}$

3. on essaie de **combiner** certains des  $Q(x_i)$  collectés pour obtenir un carré modulo  $N$  :

$$Q(x_1) \cdots Q(x_k) \equiv b^2 \pmod N$$

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'entiers de la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

**qui se décomposent sur  $\mathcal{P}$** , c'est-à-dire dont les diviseurs premiers sont dans  $\mathcal{P}$

3. on essaie de **combiner** certains des  $Q(x_i)$  collectés pour obtenir un carré modulo  $N$  :

$$Q(x_1) \cdots Q(x_k) \equiv b^2 \pmod N$$

Alors, en posant

$$a := (x_1 + \lceil \sqrt{N} \rceil) \cdots (x_k + \lceil \sqrt{N} \rceil)$$

et

$b$  une racine carrée modulo  $N$  de  $Q(x_1) \cdots Q(x_k)$

on a obtenu  $a^2 \equiv b^2 \pmod N$ .

## Comment trouver $a^2 \equiv b^2 \pmod N$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

## Comment trouver $a^2 \equiv b^2 \pmod{N}$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

**Étape 1. Détermination de la base de facteurs.** On choisit la borne  $B = 6$ , donc par exemple  $\mathcal{P} = \{2, 3, 5\}$ .

# Comment trouver $a^2 \equiv b^2 \pmod N$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

**Étape 1. Détermination de la base de facteurs.** On choisit la borne  $B = 6$ , donc par exemple  $\mathcal{P} = \{2, 3, 5\}$ .

**Étape 2. Recherche de carrés se décomposant sur  $\mathcal{P}$ .** Modulo  $N$ , on obtient

$$\left\{ \begin{array}{llllll} (x = 0) & 41^2 = 1681 & \equiv 32 & \equiv 2^5 & \pmod N & \text{(ok)} \\ (x = 1) & 42^2 = 1764 & \equiv 115 & \equiv 5 \times 23 & \pmod N & \\ (x = 2) & 43^2 = 1849 & \equiv 200 & \equiv 2^3 \times 5^2 & \pmod N & \text{(ok)} \end{array} \right.$$

# Comment trouver $a^2 \equiv b^2 \pmod N$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

**Étape 1. Détermination de la base de facteurs.** On choisit la borne  $B = 6$ , donc par exemple  $\mathcal{P} = \{2, 3, 5\}$ .

**Étape 2. Recherche de carrés se décomposant sur  $\mathcal{P}$ .** Modulo  $N$ , on obtient

$$\left\{ \begin{array}{llllll} (x = 0) & 41^2 = 1681 & \equiv 32 & \equiv 2^5 & \pmod N & \text{(ok)} \\ (x = 1) & 42^2 = 1764 & \equiv 115 & \equiv 5 \times 23 & \pmod N & \\ (x = 2) & 43^2 = 1849 & \equiv 200 & \equiv 2^3 \times 5^2 & \pmod N & \text{(ok)} \end{array} \right.$$

**Étape 3. Recherche de relations.** Maintenant, on note que

$$41^2 \times 43^2 \equiv 2^8 \times 5^2 \equiv (2^4 \times 5)^2 \pmod N.$$

## Comment trouver $a^2 \equiv b^2 \pmod N$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

**Étape 1. Détermination de la base de facteurs.** On choisit la borne  $B = 6$ , donc par exemple  $\mathcal{P} = \{2, 3, 5\}$ .

**Étape 2. Recherche de carrés se décomposant sur  $\mathcal{P}$ .** Modulo  $N$ , on obtient

$$\begin{cases} (x = 0) & 41^2 = 1681 \equiv 32 \equiv 2^5 \pmod N & \text{(ok)} \\ (x = 1) & 42^2 = 1764 \equiv 115 \equiv 5 \times 23 \pmod N \\ (x = 2) & 43^2 = 1849 \equiv 200 \equiv 2^3 \times 5^2 \pmod N & \text{(ok)} \end{cases}$$

**Étape 3. Recherche de relations.** Maintenant, on note que

$$41^2 \times 43^2 \equiv 2^8 \times 5^2 \equiv (2^4 \times 5)^2 \pmod N.$$

**Conclusion.** Il résulte que

$$(41 \times 43)^2 \equiv 1763^2 \equiv (2^4 \times 5)^2 \equiv 80^2 \pmod N$$

et on a bien  $1763 \equiv 114 \not\equiv \pm 80 \pmod N$ .

## Comment trouver $a^2 \equiv b^2 \pmod N$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

**Étape 1. Détermination de la base de facteurs.** On choisit la borne  $B = 6$ , donc par exemple  $\mathcal{P} = \{2, 3, 5\}$ .

**Étape 2. Recherche de carrés se décomposant sur  $\mathcal{P}$ .** Modulo  $N$ , on obtient

$$\left\{ \begin{array}{llllll} (x = 0) & 41^2 = 1681 & \equiv 32 & \equiv 2^5 & \pmod N & \text{(ok)} \\ (x = 1) & 42^2 = 1764 & \equiv 115 & \equiv 5 \times 23 & \pmod N & \\ (x = 2) & 43^2 = 1849 & \equiv 200 & \equiv 2^3 \times 5^2 & \pmod N & \text{(ok)} \end{array} \right.$$

**Étape 3. Recherche de relations.** Maintenant, on note que

$$41^2 \times 43^2 \equiv 2^8 \times 5^2 \equiv (2^4 \times 5)^2 \pmod N.$$

**Conclusion.** Il résulte que

$$(41 \times 43)^2 \equiv 1763^2 \equiv (2^4 \times 5)^2 \equiv 80^2 \pmod N$$

et on a bien  $1763 \equiv 114 \not\equiv \pm 80 \pmod N$ .

Ainsi, on obtient  $\text{pgcd}(114 - 80, 1649) = 17$  un facteur propre de  $N = 1649$ .



## Comment trouver $a^2 \equiv b^2 \pmod N$ ? Exemple.

**Exemple :**  $N = 1649$  donne  $\lceil \sqrt{N} \rceil = 41$ .

**Étape 1. Détermination de la base de facteurs.** On choisit la borne  $B = 6$ , donc par exemple  $\mathcal{P} = \{2, 3, 5\}$ .

**Étape 2. Recherche de carrés se décomposant sur  $\mathcal{P}$ .** Modulo  $N$ , on obtient

$$\left\{ \begin{array}{llllll} (x = 0) & 41^2 = 1681 & \equiv 32 & \equiv 2^5 & \pmod N & \text{(ok)} \\ (x = 1) & 42^2 = 1764 & \equiv 115 & \equiv 5 \times 23 & \pmod N & \\ (x = 2) & 43^2 = 1849 & \equiv 200 & \equiv 2^3 \times 5^2 & \pmod N & \text{(ok)} \end{array} \right.$$

**Étape 3. Recherche de relations.** Maintenant, on note que

$$41^2 \times 43^2 \equiv 2^8 \times 5^2 \equiv (2^4 \times 5)^2 \pmod N.$$

**Conclusion.** Il résulte que

$$(41 \times 43)^2 \equiv 1763^2 \equiv (2^4 \times 5)^2 \equiv 80^2 \pmod N$$

et on a bien  $1763 \equiv 114 \not\equiv \pm 80 \pmod N$ .

Ainsi, on obtient  $\text{pgcd}(114 - 80, 1649) = 17$  un facteur propre de  $N = 1649$ .

**Remarque.** Avec l'algorithme de Fermat, on aurait dû aller jusqu'à  $57^2$  pour factoriser.

## Comment trouver $a^2 \equiv b^2 \pmod N$ ?

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'éléments **qui se décomposent sur**  $\mathcal{P}$ , et qui s'écrivent sous la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

3. on essaie de **combiner** certains  $Q(x_i)$  pour obtenir un carré modulo  $N$  :

$$Q(x_1) \cdots Q(x_k) \equiv b^2 \pmod N$$

Alors, on aura obtenu  $a^2 \equiv b^2 \pmod N$  où  $a := (x_1 + \lceil \sqrt{N} \rceil) \cdots (x_k + \lceil \sqrt{N} \rceil)$ .

**Trois questions :**

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'éléments **qui se décomposent sur**  $\mathcal{P}$ , et qui s'écrivent sous la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

3. on essaie de **combiner** certains  $Q(x_i)$  pour obtenir un carré modulo  $N$  :

$$Q(x_1) \cdots Q(x_k) \equiv b^2 \pmod N$$

Alors, on aura obtenu  $a^2 \equiv b^2 \pmod N$  où  $a := (x_1 + \lceil \sqrt{N} \rceil) \cdots (x_k + \lceil \sqrt{N} \rceil)$ .

**Trois questions :**

1. quelle base de facteurs premiers  $\mathcal{P}$  choisir ?

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'éléments **qui se décomposent sur  $\mathcal{P}$** , et qui s'écrivent sous la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

3. on essaie de **combiner** certains  $Q(x_i)$  pour obtenir un carré modulo  $N$  :

$$Q(x_1) \cdots Q(x_k) \equiv b^2 \pmod N$$

Alors, on aura obtenu  $a^2 \equiv b^2 \pmod N$  où  $a := (x_1 + \lceil \sqrt{N} \rceil) \cdots (x_k + \lceil \sqrt{N} \rceil)$ .

**Trois questions :**

1. quelle base de facteurs premiers  $\mathcal{P}$  choisir ?
2. comment obtient-on ces éléments  $B$ -friables de la forme  $Q(x)$  ?  
→ technique de crible (ici, quadratique)

**Idée.** Étant donnée une borne de lissité  $B \geq 2$  :

1. on crée une base de facteurs premiers  $\mathcal{P} = \{p_1, \dots, p_s\}$ , tous inférieurs à  $B$
2. on collecte une quantité importante d'éléments **qui se décomposent sur  $\mathcal{P}$** , et qui s'écrivent sous la forme

$$Q(x) := (x + \lceil \sqrt{N} \rceil)^2 - N$$

3. on essaie de **combiner** certains  $Q(x_i)$  pour obtenir un carré modulo  $N$  :

$$Q(x_1) \cdots Q(x_k) \equiv b^2 \pmod N$$

Alors, on aura obtenu  $a^2 \equiv b^2 \pmod N$  où  $a := (x_1 + \lceil \sqrt{N} \rceil) \cdots (x_k + \lceil \sqrt{N} \rceil)$ .

## Trois questions :

1. quelle base de facteurs premiers  $\mathcal{P}$  choisir ?
2. comment obtient-on ces éléments  $B$ -friables de la forme  $Q(x)$  ?  
→ technique de crible (ici, quadratique)
3. comment savoir quels  $Q(x_i)$  multiplier pour obtenir un carré modulo  $N$  ?  
→ algèbre linéaire dans  $\mathbb{F}_2$

Première étape : construction de la **base de facteurs**.

Première étape : construction de la **base de facteurs**.

Si  $x < \sqrt{N}/3$ , alors on a

$$Q(x) = (x + \lceil \sqrt{N} \rceil)^2 - N \simeq x^2 + 2\sqrt{N}x < N.$$

Première étape : construction de la **base de facteurs**.

Si  $x < \sqrt{N}/3$ , alors on a

$$Q(x) = (x + \lceil \sqrt{N} \rceil)^2 - N \simeq x^2 + 2\sqrt{N}x < N.$$

Donc  $(Q(x) \bmod N)$  vaut  $Q(x)$ , et pour tout premier  $p \geq 2$ , on a

$$p \mid Q(x) \implies N \text{ est un carré modulo } p$$



Première étape : construction de la **base de facteurs**.

Si  $x < \sqrt{N}/3$ , alors on a

$$Q(x) = (x + \lceil \sqrt{N} \rceil)^2 - N \simeq x^2 + 2\sqrt{N}x < N.$$

Donc  $(Q(x) \bmod N)$  vaut  $Q(x)$ , et pour tout premier  $p \geq 2$ , on a

$$p \mid Q(x) \implies N \text{ est un carré modulo } p$$

Pour constituer la base de facteurs, on peut donc considérer **uniquement** les premiers  $p$  tel que  $\left(\frac{N}{p}\right) = 1$  et  $p \leq B$ .

Première étape : construction de la **base de facteurs**.

Si  $x < \sqrt{N}/3$ , alors on a

$$Q(x) = (x + \lceil \sqrt{N} \rceil)^2 - N \simeq x^2 + 2\sqrt{N}x < N.$$

Donc  $(Q(x) \bmod N)$  vaut  $Q(x)$ , et pour tout premier  $p \geq 2$ , on a

$$p \mid Q(x) \implies N \text{ est un carré modulo } p$$

Pour constituer la base de facteurs, on peut donc considérer **uniquement** les premiers  $p$  tel que  $\left(\frac{N}{p}\right) = 1$  et  $p \leq B$ .

**Exemple :**  $N = 369713 = 457 \times 809$ . On choisit ici  $B = 21$ . On a alors

$p$	2	3	5	7	11	13	17	19
$\left(\frac{N}{p}\right)$	1	-1	-1	1	1	-1	-1	1

Première étape : construction de la **base de facteurs**.

Si  $x < \sqrt{N}/3$ , alors on a

$$Q(x) = (x + \lceil \sqrt{N} \rceil)^2 - N \simeq x^2 + 2\sqrt{N}x < N.$$

Donc  $(Q(x) \bmod N)$  vaut  $Q(x)$ , et pour tout premier  $p \geq 2$ , on a

$$p \mid Q(x) \implies N \text{ est un carré modulo } p$$

Pour constituer la base de facteurs, on peut donc considérer **uniquement** les premiers  $p$  tel que  $\left(\frac{N}{p}\right) = 1$  et  $p \leq B$ .

**Exemple :**  $N = 369713 = 457 \times 809$ . On choisit ici  $B = 21$ . On a alors

$p$	2	3	5	7	11	13	17	19
$\left(\frac{N}{p}\right)$	1	-1	-1	1	1	-1	-1	1

La base de facteurs est donc  $\{2, 7, 11, 19\}$ .

Troisième étape : l'**algèbre linéaire**.

Troisième étape : l'**algèbre linéaire**.

Soit  $\mathcal{P} = \{p_1, \dots, p_s\}$  la base de facteurs construite précédemment.

Troisième étape : l'**algèbre linéaire**.

Soit  $\mathcal{P} = \{p_1, \dots, p_s\}$  la base de facteurs construite précédemment.

Supposons que l'on ait collecté  $t$  éléments  $Q(x_1), \dots, Q(x_t)$  tels que les  $Q(x_i) \bmod N$  se décomposent dans  $\mathcal{P}$  (étape II).

### Troisième étape : l'algèbre linéaire.

Soit  $\mathcal{P} = \{p_1, \dots, p_s\}$  la base de facteurs construite précédemment.

Supposons que l'on ait collecté  $t$  éléments  $Q(x_1), \dots, Q(x_t)$  tels que les  $Q(x_i) \bmod N$  se décomposent dans  $\mathcal{P}$  (étape II). On peut alors écrire  $Q(x_i) \bmod N$  sous la forme

$$p_1^{e_1^{(i)}} \times p_2^{e_2^{(i)}} \times \dots \times p_s^{e_s^{(i)}}$$

et on note  $e(x_i) = (e_1^{(i)}, \dots, e_s^{(i)})$ .

### Troisième étape : l'algèbre linéaire.

Soit  $\mathcal{P} = \{p_1, \dots, p_s\}$  la base de facteurs construite précédemment.

Supposons que l'on ait collecté  $t$  éléments  $Q(x_1), \dots, Q(x_t)$  tels que les  $Q(x_i) \bmod N$  se décomposent dans  $\mathcal{P}$  (étape II). On peut alors écrire  $Q(x_i) \bmod N$  sous la forme

$$p_1^{e_1^{(i)}} \times p_2^{e_2^{(i)}} \times \dots \times p_s^{e_s^{(i)}}$$

et on note  $\mathbf{e}(x_i) = (e_1^{(i)}, \dots, e_s^{(i)})$ .

Alors, on a  $Q(x_{i_1})Q(x_{i_2}) \dots Q(x_{i_k}) \equiv p_1^{e_1} \dots p_s^{e_s} \bmod N$  où

$$(e_1, \dots, e_s) = \mathbf{e}(x_{i_1}) + \mathbf{e}(x_{i_2}) + \dots + \mathbf{e}(x_{i_k}).$$



### Troisième étape : l'algèbre linéaire.

Soit  $\mathcal{P} = \{p_1, \dots, p_s\}$  la base de facteurs construite précédemment.

Supposons que l'on ait collecté  $t$  éléments  $Q(x_1), \dots, Q(x_t)$  tels que les  $Q(x_i) \bmod N$  se décomposent dans  $\mathcal{P}$  (étape II). On peut alors écrire  $Q(x_i) \bmod N$  sous la forme

$$p_1^{e_1^{(i)}} \times p_2^{e_2^{(i)}} \times \dots \times p_s^{e_s^{(i)}}$$

et on note  $e(x_i) = (e_1^{(i)}, \dots, e_s^{(i)})$ .

Alors, on a  $Q(x_{i_1})Q(x_{i_2}) \dots Q(x_{i_k}) \equiv p_1^{e_1} \dots p_s^{e_s} \pmod N$  où

$$(e_1, \dots, e_s) = e(x_{i_1}) + e(x_{i_2}) + \dots + e(x_{i_k}).$$

**Lemme.** L'élément  $Q(x_{i_1}) \dots Q(x_{i_k})$  est un carré modulo  $N$  si et seulement si

$$\sum_{j=1}^k e(x_{i_j}) = \mathbf{0} \pmod 2.$$

**Lemme.** L'élément  $Q(x_{i_1}) \dots Q(x_{i_k})$  est un carré modulo  $N$  si et seulement si

$$\sum_{j=1}^k e(x_{i_j}) = 0 \pmod{2}.$$

**Lemme.** L'élément  $Q(x_{i_1}) \dots Q(x_{i_k})$  est un carré modulo  $N$  si et seulement si

$$\sum_{j=1}^k e(x_{i_j}) = \mathbf{0} \pmod{2}.$$

On va alors construire une matrice  $M$  entière de taille  $(t \times s)$  telle la  $i$ -ème ligne de  $M$  est le vecteur ligne  $e(x_i)$  :

$$M = \begin{bmatrix} e_1^{(1)} & e_2^{(1)} & \dots & e_s^{(1)} \\ e_1^{(2)} & \dots & \dots & e_s^{(2)} \\ \vdots & & & \\ e_1^{(t)} & \dots & \dots & e_s^{(t)} \end{bmatrix} \in \mathbb{N}^{t \times s}$$

**Lemme.** L'élément  $Q(x_{i_1}) \dots Q(x_{i_k})$  est un carré modulo  $N$  si et seulement si

$$\sum_{j=1}^k e(x_{i_j}) = \mathbf{0} \pmod{2}.$$

On va alors construire une matrice  $M$  entière de taille  $(t \times s)$  telle la  $i$ -ème ligne de  $M$  est le vecteur ligne  $e(x_i)$  :

$$M = \begin{bmatrix} e_1^{(1)} & e_2^{(1)} & \dots & e_s^{(1)} \\ e_1^{(2)} & \dots & \dots & e_s^{(2)} \\ \vdots & & & \\ e_1^{(t)} & \dots & \dots & e_s^{(t)} \end{bmatrix} \in \mathbb{N}^{t \times s}$$

Pour obtenir un carré modulo  $N$  de la forme  $Q(x_1) \dots Q(x_k)$ , il suffit donc de chercher un élément du noyau à gauche de la matrice  $(M \pmod{2})$ , car

$$\mathbf{u} = (u_1, \dots, u_t) \in \mathbb{F}_2^t \text{ vérifie } \mathbf{u}M = \mathbf{0} \implies \sum u_i e(x_i) = \mathbf{0}.$$

**Lemme.** L'élément  $Q(x_{i_1}) \dots Q(x_{i_k})$  est un carré modulo  $N$  si et seulement si

$$\sum_{j=1}^k e(x_{i_j}) = \mathbf{0} \pmod{2}.$$

On va alors construire une matrice  $M$  entière de taille  $(t \times s)$  telle la  $i$ -ème ligne de  $M$  est le vecteur ligne  $e(x_i)$  :

$$M = \begin{bmatrix} e_1^{(1)} & e_2^{(1)} & \dots & e_s^{(1)} \\ e_1^{(2)} & \dots & \dots & e_s^{(2)} \\ \vdots & & & \\ e_1^{(t)} & \dots & \dots & e_s^{(t)} \end{bmatrix} \in \mathbb{N}^{t \times s}$$

Pour obtenir un carré modulo  $N$  de la forme  $Q(x_1) \dots Q(x_k)$ , il suffit donc de chercher un élément du noyau à gauche de la matrice  $(M \pmod{2})$ , car

$$u = (u_1, \dots, u_t) \in \mathbb{F}_2^t \text{ vérifie } uM = \mathbf{0} \implies \sum u_i e(x_i) = \mathbf{0}.$$

**Remarque.** Pour que la matrice  $(M \pmod{2})$  ait un noyau (à gauche) non-nul, il est suffisant que le nombre de lignes non-nulles de  $(M \pmod{2})$  soit plus grand que le nombre de ses colonnes. En pratique, on souhaite donc que  $t$  soit sensiblement plus grand que  $s$ .

## Étape III : exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ , supposons que l'on ait obtenu les éléments suivants :

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728
décomposition sur $\mathcal{P}$	$2^6 \times 7 \times 19$	$2^5 \times 7^3$	$2^8 \times 11^2$	$2^3 \times 7^2 \times 19^2$	$2^6 \times 7 \times 19^2$

## Étape III : exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ , supposons que l'on ait obtenu les éléments suivants :

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728
décomposition sur $\mathcal{P}$	$2^6 \times 7 \times 19$	$2^5 \times 7^3$	$2^8 \times 11^2$	$2^3 \times 7^2 \times 19^2$	$2^6 \times 7 \times 19^2$

La matrice  $M$  est alors :

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

## Étape III : exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ , supposons que l'on ait obtenu les éléments suivants :

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728
décomposition sur $\mathcal{P}$	$2^6 \times 7 \times 19$	$2^5 \times 7^3$	$2^8 \times 11^2$	$2^3 \times 7^2 \times 19^2$	$2^6 \times 7 \times 19^2$

La matrice  $M$  est alors :

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

Modulo 2, on obtient :

$$(M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$



## Étape III : exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ , supposons que l'on ait obtenu les éléments suivants :

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728
décomposition sur $\mathcal{P}$	$2^6 \times 7 \times 19$	$2^5 \times 7^3$	$2^8 \times 11^2$	$2^3 \times 7^2 \times 19^2$	$2^6 \times 7 \times 19^2$

La matrice  $M$  est alors :

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

Modulo 2, on obtient :

$$(M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Puis, on trouve un élément  $u$  tel que  $uM = \mathbf{0}$ , par exemple

$$u = (0, 1, 1, 1, 1) \text{ ou encore } u = (0, 0, 1, 0, 0)$$

## Étape III : exemple

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728

$$(M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$u = (0, 1, 1, 1, 1)$$

## Étape III : exemple

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

$$(M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$u = (0, 1, 1, 1, 1)$$

On calcule  $u \cdot M = (22, 6, 2, 4)$ , donc on va construire

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

$$(M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$u = (0, 1, 1, 1, 1)$$

On calcule  $u \cdot M = (22, 6, 2, 4)$ , donc on va construire

1.  $b$  une racine carrée de  $Q(x_2)Q(x_3)Q(x_4)Q(x_5) = p_1^{22} p_2^6 p_3^2 p_4^4$ , c'est-à-dire

$$b = p_1^{11} p_2^3 p_3 p_4^2 \equiv 369672 \pmod{N}$$

## Étape III : exemple

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

$$(M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$u = (0, 1, 1, 1, 1)$$

On calcule  $u \cdot M = (22, 6, 2, 4)$ , donc on va construire

1.  $b$  une racine carrée de  $Q(x_2)Q(x_3)Q(x_4)Q(x_5) = p_1^{22} p_2^6 p_3^2 p_4^4$ , c'est-à-dire

$$b = p_1^{11} p_2^3 p_3 p_4^2 \equiv 369672 \pmod{N}$$

2.  $a$  le produit des  $x_i + \lceil \sqrt{N} \rceil$  correspondant, donc

$$a = (x_2 + \lceil \sqrt{N} \rceil)(x_3 + \lceil \sqrt{N} \rceil)(x_4 + \lceil \sqrt{N} \rceil)(x_5 + \lceil \sqrt{N} \rceil) \equiv 102784 \pmod{N}$$

## Étape III : exemple

$x_i$	6	8	24	106	120
$Q(x_i)$	8512	10976	30976	141512	161728

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix} \qquad (M \bmod 2) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \qquad \mathbf{u} = (0, 1, 1, 1, 1)$$

On calcule  $\mathbf{u} \cdot M = (22, 6, 2, 4)$ , donc on va construire

1.  $b$  une racine carrée de  $Q(x_2)Q(x_3)Q(x_4)Q(x_5) = p_1^{22} p_2^6 p_3^2 p_4^4$ , c'est-à-dire

$$b = p_1^{11} p_2^3 p_3 p_4^2 \equiv 369672 \pmod{N}$$

2.  $a$  le produit des  $x_i + \lceil \sqrt{N} \rceil$  correspondant, donc

$$a = (x_2 + \lceil \sqrt{N} \rceil)(x_3 + \lceil \sqrt{N} \rceil)(x_4 + \lceil \sqrt{N} \rceil)(x_5 + \lceil \sqrt{N} \rceil) \equiv 102784 \pmod{N}$$

Puis on a (finalement) :

$$\text{pgcd}(a - b, N) = 457 \quad \text{et} \quad \text{pgcd}(a + b, N) = 809.$$

## Étape II : effritement (méthode naïve)

Deuxième étape : **effritement**, ou phase de collection

**But :** pour  $t \geq s$ , trouver  $t$  éléments de la forme  $Q(x)$  qui se décomposent dans la base de facteurs  $\mathcal{P} = \{p_1, \dots, p_s\}$

## Étape II : effritement (méthode naïve)

Deuxième étape : **effritement**, ou phase de collection

**But** : pour  $t \geq s$ , trouver  $t$  éléments de la forme  $Q(x)$  qui se décomposent dans la base de facteurs  $\mathcal{P} = \{p_1, \dots, p_s\}$

Une première idée est de chercher ces éléments de manière itérative ( $x$  croissant de 1 en 1).



## Étape II : effritement (méthode naïve)

Deuxième étape : **effritement**, ou phase de collection

**But :** pour  $t \geq s$ , trouver  $t$  éléments de la forme  $Q(x)$  qui se décomposent dans la base de facteurs  $\mathcal{P} = \{p_1, \dots, p_s\}$

Une première idée est de chercher ces éléments de manière itérative ( $x$  croissant de 1 en 1).

### COLLECTION D'ÉLÉMENTS FRIABLES (MÉTHODE NAÏVE)

1.  $i \leftarrow 0$ ,  $x = 0$ ,  $r = \lceil \sqrt{N} \rceil$ ,  $q = r^2 - N$
2. `liste_elements` = []
3.  $M = []$
4. **Tant que**  $i < t$ :
  - 4.1 **Si**  $q$  se décompose sur  $\mathcal{P}$  comme  $q = p_1^{e_1} \dots p_s^{e_s}$ 
    - Ajouter le vecteur  $(e_1, \dots, e_s)$  comme dernière ligne de  $M$
    - Ajouter  $(x, q)$  à `liste_elements`
    - $i \leftarrow i + 1$
  - 4.2  $q \leftarrow q + 2(x + r) + 1$
  - 4.3  $x \leftarrow x + 1$
5. **Retourner**  $M$  et `liste_elements`

# Exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ .

$x$	$Q(x)$	$(e_1, \dots, e_4)$	"reste"
0	1168	[4, 0, 0, 0]	73
1	2387	[0, 1, 1, 0]	31
2	3608	[3, 0, 1, 0]	41
3	4831	[0, 0, 0, 0]	4831
4	6056	[3, 0, 0, 0]	757
5	7283	[0, 0, 0, 0]	7283
<b>6</b>	<b>8512</b>	<b>[6, 1, 0, 1]</b>	<b>1</b>
7	9743	[0, 0, 0, 0]	9743
<b>8</b>	<b>10976</b>	<b>[5, 3, 0, 0]</b>	<b>1</b>
9	12211	[0, 0, 0, 0]	12211
⋮	⋮	⋮	⋮
23	29711	[0, 0, 1, 0]	2701
<b>24</b>	<b>30976</b>	<b>[8, 0, 2, 0]</b>	<b>1</b>
25	32243	[0, 0, 0, 1]	1697
⋮	⋮	⋮	⋮
<b>106</b>	<b>141512</b>	<b>[3, 2, 0, 2]</b>	<b>1</b>
⋮	⋮	⋮	⋮
<b>120</b>	<b>161728</b>	<b>[6, 1, 0, 2]</b>	<b>1</b>

# Exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ .

$x$	$Q(x)$	$(e_1, \dots, e_4)$	"reste"
0	1168	[4, 0, 0, 0]	73
1	2387	[0, 1, 1, 0]	31
2	3608	[3, 0, 1, 0]	41
3	4831	[0, 0, 0, 0]	4831
4	6056	[3, 0, 0, 0]	757
5	7283	[0, 0, 0, 0]	7283
<b>6</b>	<b>8512</b>	<b>[6, 1, 0, 1]</b>	<b>1</b>
7	9743	[0, 0, 0, 0]	9743
<b>8</b>	<b>10976</b>	<b>[5, 3, 0, 0]</b>	<b>1</b>
9	12211	[0, 0, 0, 0]	12211
⋮	⋮	⋮	⋮
23	29711	[0, 0, 1, 0]	2701
<b>24</b>	<b>30976</b>	<b>[8, 0, 2, 0]</b>	<b>1</b>
25	32243	[0, 0, 0, 1]	1697
⋮	⋮	⋮	⋮
<b>106</b>	<b>141512</b>	<b>[3, 2, 0, 2]</b>	<b>1</b>
⋮	⋮	⋮	⋮
<b>120</b>	<b>161728</b>	<b>[6, 1, 0, 2]</b>	<b>1</b>

On obtient la liste

$\{(6, 8512), (8, 10976), (24, 30976),$   
 $(106, 141512), (120, 161728)\}$

# Exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ .

$x$	$Q(x)$	$(e_1, \dots, e_4)$	"reste"
0	1168	[4, 0, 0, 0]	73
1	2387	[0, 1, 1, 0]	31
2	3608	[3, 0, 1, 0]	41
3	4831	[0, 0, 0, 0]	4831
4	6056	[3, 0, 0, 0]	757
5	7283	[0, 0, 0, 0]	7283
<b>6</b>	<b>8512</b>	<b>[6, 1, 0, 1]</b>	<b>1</b>
7	9743	[0, 0, 0, 0]	9743
<b>8</b>	<b>10976</b>	<b>[5, 3, 0, 0]</b>	<b>1</b>
9	12211	[0, 0, 0, 0]	12211
$\vdots$	$\vdots$	$\vdots$	$\vdots$
23	29711	[0, 0, 1, 0]	2701
<b>24</b>	<b>30976</b>	<b>[8, 0, 2, 0]</b>	<b>1</b>
25	32243	[0, 0, 0, 1]	1697
$\vdots$	$\vdots$	$\vdots$	$\vdots$
<b>106</b>	<b>141512</b>	<b>[3, 2, 0, 2]</b>	<b>1</b>
$\vdots$	$\vdots$	$\vdots$	$\vdots$
<b>120</b>	<b>161728</b>	<b>[6, 1, 0, 2]</b>	<b>1</b>

On obtient la liste

$\{(6, 8512), (8, 10976), (24, 30976),$   
 $(106, 141512), (120, 161728)\}$

Cela donne la matrice

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

# Exemple

Pour  $N = 369713$  et  $\mathcal{P} = \{2, 7, 11, 19\}$ .

$x$	$Q(x)$	$(e_1, \dots, e_4)$	"reste"
0	1168	[4, 0, 0, 0]	73
1	2387	[0, 1, 1, 0]	31
2	3608	[3, 0, 1, 0]	41
3	4831	[0, 0, 0, 0]	4831
4	6056	[3, 0, 0, 0]	757
5	7283	[0, 0, 0, 0]	7283
<b>6</b>	<b>8512</b>	<b>[6, 1, 0, 1]</b>	<b>1</b>
7	9743	[0, 0, 0, 0]	9743
<b>8</b>	<b>10976</b>	<b>[5, 3, 0, 0]</b>	<b>1</b>
9	12211	[0, 0, 0, 0]	12211
⋮	⋮	⋮	⋮
23	29711	[0, 0, 1, 0]	2701
<b>24</b>	<b>30976</b>	<b>[8, 0, 2, 0]</b>	<b>1</b>
25	32243	[0, 0, 0, 1]	1697
⋮	⋮	⋮	⋮
<b>106</b>	<b>141512</b>	<b>[3, 2, 0, 2]</b>	<b>1</b>
⋮	⋮	⋮	⋮
<b>120</b>	<b>161728</b>	<b>[6, 1, 0, 2]</b>	<b>1</b>

On obtient la liste

$$\{(6, 8512), (8, 10976), (24, 30976), (106, 141512), (120, 161728)\}$$

Cela donne la matrice

$$M = \begin{bmatrix} 6 & 1 & 0 & 1 \\ 5 & 3 & 0 & 0 \\ 8 & 0 & 2 & 0 \\ 3 & 2 & 0 & 2 \\ 6 & 1 & 0 & 2 \end{bmatrix}$$

**Problème.** Pour chaque ligne calculée, on fait beaucoup de tests de divisibilité qui mènent à un échec. Autrement dit, il y a **beaucoup** de zéros dans les  $(e_1, \dots, e_s)$  calculés.

## Étape II : effritement (méthode de criblage)

**Idée** : pour être plus efficace, on va collecter ces éléments par une méthode de **crible**.

## Étape II : effritement (méthode de criblage)

**Idée** : pour être plus efficace, on va collecter ces éléments par une méthode de **crible**.

**Remarque.** La notion de crible a déjà été vue pour établir une liste de nombres premiers : c'est le **crible d'Eratostène**.

## Étape II : effritement (méthode de criblage)

**Idée** : pour être plus efficace, on va collecter ces éléments par une méthode de **crible**.

**Remarque.** La notion de crible a déjà été vue pour établir une liste de nombres premiers : c'est le **crible d'Eratostène**.

Pour  $A \geq s$  un entier dépendant de  $B$  qu'on déterminera plus tard :

### MÉTHODE DE CRIBLE QUADRATIQUE (VERSION PÉDAGOGIQUE)

1. On initialise un tableau  $T = [Q(0), Q(1), \dots, Q(A)]$
2. Pour chaque premier  $p \in \mathcal{P}$  :
  - 2.1  $e = 1$
  - 2.2 Tant que  $p^e \leq A$  :
    - On cherche tous les  $0 \leq y < p^e$  tels que  $p^e$  divise  $T[y]$
    - On divise par  $p$  tous les  $T[x]$  où  $x$  est de la forme  $y + kp^e$  (criblage)
    - On incrémente  $e \leftarrow e + 1$
3. **Retourner** tous les  $(x, Q(x))$  tels que  $T[x] = 1$ .



## Étape II : effritement (méthode de criblage)

**Idée** : pour être plus efficace, on va collecter ces éléments par une méthode de **crible**.

**Remarque.** La notion de crible a déjà été vue pour établir une liste de nombres premiers : c'est le **crible d'Eratostène**.  
Pour  $A \geq s$  un entier dépendant de  $B$  qu'on déterminera plus tard :

### MÉTHODE DE CRIBLE QUADRATIQUE (VERSION PÉDAGOGIQUE)

1. On initialise un tableau  $T = [Q(0), Q(1), \dots, Q(A)]$
2. Pour chaque premier  $p \in \mathcal{P}$  :
  - 2.1  $e = 1$
  - 2.2 Tant que  $p^e \leq A$  :
    - On cherche tous les  $0 \leq y < p^e$  tels que  $p^e$  divise  $T[y]$
    - On divise par  $p$  tous les  $T[x]$  où  $x$  est de la forme  $y + kp^e$  (criblage)
    - On incrémente  $e \leftarrow e + 1$
3. **Retourner** tous les  $(x, Q(x))$  tels que  $T[x] = 1$ .

### Remarques :

- la recherche de solution de  $Q(y) \equiv N \pmod{p}$  est essentiellement une recherche de racine carrée ( $\rightarrow$  Tonelli-Shanks, ou Cipolla)
- on peut déduire très efficacement les solutions modulo  $p^e$  des solutions modulo  $p^{e-1}$  (relèvement de Hensel).

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

## Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$

## Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579



# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$		
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579		
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579		
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579		
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579		
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579		
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579		
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579		

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579
	2	[7]	1	767	41	1859	43	2959	439	83	289	5183	359	901	859	7439	143	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579
	2	[7]	1	767	41	1859	43	2959	439	83	289	5183	359	901	859	7439	143	8579
11	1	[5, 3]	1	767	41	169	43	269	439	83	289	5183	359	901	859	7439	13	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579
	2	[7]	1	767	41	1859	43	2959	439	83	289	5183	359	901	859	7439	143	8579
11	1	[5, 3]	1	767	41	169	43	269	439	83	289	5183	359	901	859	7439	13	8579
13	1	[3, 1]	1	59	41	13	43	269	439	83	289	5183	359	901	859	7439	1	8579

# Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579
	2	[7]	1	767	41	1859	43	2959	439	83	289	5183	359	901	859	7439	143	8579
11	1	[5, 3]	1	767	41	169	43	269	439	83	289	5183	359	901	859	7439	13	8579
13	1	[3, 1]	1	59	41	13	43	269	439	83	289	5183	359	901	859	7439	1	8579
	2	[3]	1	59	41	1	43	269	439	83	289	5183	359	901	859	7439	1	8579

## Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579
	2	[7]	1	767	41	1859	43	2959	439	83	289	5183	359	901	859	7439	143	8579
11	1	[5, 3]	1	767	41	169	43	269	439	83	289	5183	359	901	859	7439	13	8579
13	1	[3, 1]	1	59	41	13	43	269	439	83	289	5183	359	901	859	7439	1	8579
	2	[3]	1	59	41	1	43	269	439	83	289	5183	359	901	859	7439	1	8579

Et on obtient la liste  $\{(0, 224), (3, 1859), (14, 8008)\}$

## Exemple de crible

Pour  $N = 73217 = 211 \times 347$  (nouveau  $N$ ) avec la base de facteurs  $\mathcal{P} = \{2, 7, 11, 13\}$ .

Les étapes successives de l'algorithme sont :

$p$	$e$	solutions	$T_0$	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	$T_6$	$T_7$	$T_8$	$T_9$	$T_{10}$	$T_{11}$	$T_{12}$	$T_{13}$	$T_{14}$	$T_{15}$
			224	767	1312	1859	2408	2959	3512	4067	4624	5183	5744	6307	6872	7439	8008	8579
2	1	[0]	112	767	656	1859	1204	2959	1756	4067	2312	5183	2872	6307	3436	7439	4004	8579
	2	[2, 0]	56	767	328	1859	602	2959	878	4067	1156	5183	1436	6307	1718	7439	2002	8579
	3	[2, 4, 6, 0]	28	767	164	1859	301	2959	439	4067	578	5183	718	6307	859	7439	1001	8579
	4	[2, 8, 10, 0]	14	767	82	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
	5	[0, 2]	7	767	41	1859	301	2959	439	4067	289	5183	359	6307	859	7439	1001	8579
7	1	[4, 0]	1	767	41	1859	43	2959	439	581	289	5183	359	901	859	7439	143	8579
	2	[7]	1	767	41	1859	43	2959	439	83	289	5183	359	901	859	7439	143	8579
11	1	[5, 3]	1	767	41	169	43	269	439	83	289	5183	359	901	859	7439	13	8579
13	1	[3, 1]	1	59	41	13	43	269	439	83	289	5183	359	901	859	7439	1	8579
	2	[3]	1	59	41	1	43	269	439	83	289	5183	359	901	859	7439	1	8579

Et on obtient la liste  $\{(0, 224), (3, 1859), (14, 8008)\}$

La matrice associée est

$$M = \begin{bmatrix} 5 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 3 & 1 & 1 & 1 \end{bmatrix}$$



**Observation 1.** Les termes du tableau peuvent être initialement grands, et leur division par des premiers est légèrement coûteuse en pratique.

**Observation 1.** Les termes du tableau peuvent être initialement grands, et leur division par des premiers est légèrement coûteuse en pratique. Pour éviter cela, on peut

- remplacer les valeurs exactes  $Q(x)$  par  $T[x] = \lfloor \log Q(x) \rfloor$
- soustraire  $\lfloor \log p \rfloor$  (précalculé) à  $T[x]$  au lieu de diviser  $T[x]$  par  $p$
- en fin d'algorithme, plutôt que de tester si  $T[x] = 1$ , on vérifie si  $|T[x]| \leq \log B^2$

**Observation 1.** Les termes du tableau peuvent être initialement grands, et leur division par des premiers est légèrement coûteuse en pratique. Pour éviter cela, on peut

- remplacer les valeurs exactes  $Q(x)$  par  $T[x] = \lfloor \log Q(x) \rfloor$
- soustraire  $\lfloor \log p \rfloor$  (précalculé) à  $T[x]$  au lieu de diviser  $T[x]$  par  $p$
- en fin d'algorithme, plutôt que de tester si  $T[x] = 1$ , on vérifie si  $|T[x]| \leq \log B^2$

**Observation 2.** Si  $A$  est grand, il devient peu probable que  $Q(x)$  soit  $B$ -friable lorsque  $x$  s'approche de  $A$ .

**Observation 1.** Les termes du tableau peuvent être initialement grands, et leur division par des premiers est légèrement coûteuse en pratique. Pour éviter cela, on peut

- remplacer les valeurs exactes  $Q(x)$  par  $T[x] = \lfloor \log Q(x) \rfloor$
- soustraire  $\lfloor \log p \rfloor$  (précalculé) à  $T[x]$  au lieu de diviser  $T[x]$  par  $p$
- en fin d’algorithme, plutôt que de tester si  $T[x] = 1$ , on vérifie si  $|T[x]| \leq \log B^2$

**Observation 2.** Si  $A$  est grand, il devient peu probable que  $Q(x)$  soit  $B$ -friable lorsque  $x$  s’approche de  $A$ .

**Idée :** on remplace  $Q(x)$  par des  $Q_{u,v}(x) = Q(ux + v)$  où  $u$  et  $v$  sont choisis de telle sorte que  $Q_{u,v}(x)$  donne des nombres « petits » modulo  $N$ , lorsque  $x \in [0, A]$ .

C’est la variante dite « à **polynômes multiples** » ;

- cela permet de réduire sensiblement la taille de  $A$ ,
- on peut choisir avantageusement  $u$  et  $v$  pour produire beaucoup de relations.

## FACTORISATION PAR CRIBLE QUADRATIQUE (PÉDAGOGIQUE)

**Entrée :**  $N$  un entier à factoriser

**Sortie :** un facteur propre de  $N$

1. **Initialisation :** calculer  $B \simeq 2^{0.5\sqrt{\log N \log \log N}}$  (on verra pourquoi)
2. Calculer la **base de facteurs**  $\mathcal{P} = \{p_1, \dots, p_s\}$  tels que  $p_j \leq B$  et  $\left(\frac{N}{p_j}\right) = 1$ .
3. **Effritement :**
  - 3.1 Calculer la matrice  $M$  et les éléments  $\{(x, Q(x))\}$  associés par criblage.
  - 3.2 Si  $M$  a un noyau à gauche nul, revenir à 2. avec  $B \leftarrow 2B$ .
4. **Algèbre linéaire :**
  - 4.1 Calculer une solution aléatoire  $u$  de  $u \cdot (M \bmod 2) = \mathbf{0}$
  - 4.2 Calculer  $a = \prod_{j \in J} (x_j + \lceil \sqrt{N} \rceil)$  et  $b = \prod_{j \in J} Q(x_j)$  où  $J = \{j, u_j \neq 0\}$ .
5. **Si**  $\{\text{pgcd}(a - b, N), \text{pgcd}(a + b, N)\}$  ne contient pas de facteur propre de  $N$ , revenir à l'étape 4.1.
6. **Sinon**, retourner les facteurs propres obtenus.

Qu'en est-il de la **complexité** de l'algorithme ?

1. Introduction
2. Le crible quadratique
3. Analyse et extensions

D'abord, **quelques résultats de théorie des nombres.**

D'abord, **quelques résultats de théorie des nombres.**

Soit  $M$  un entier  $\geq 2$ .

**Théorème de Tchebychev.** Le nombre  $\pi(M)$  de nombres premiers compris entre 2 et  $M$  vérifie

$$\alpha \frac{M}{\log M} \leq \pi(M) \leq \beta \frac{M}{\log M} .$$



D'abord, **quelques résultats de théorie des nombres.**

Soit  $M$  un entier  $\geq 2$ .

**Théorème de Tchebychev.** Le nombre  $\pi(M)$  de nombres premiers compris entre 2 et  $M$  vérifie

$$\alpha \frac{M}{\log M} \leq \pi(M) \leq \beta \frac{M}{\log M}.$$

**Remarque.** Les théorèmes de Hadamard et de de la Vallée-Poussin assurent  $\pi(M) \sim \frac{M}{\ln M}$ .

D'abord, **quelques résultats de théorie des nombres.**

Soit  $M$  un entier  $\geq 2$ .

**Théorème de Tchebychev.** Le nombre  $\pi(M)$  de nombres premiers compris entre 2 et  $M$  vérifie

$$\alpha \frac{M}{\log M} \leq \pi(M) \leq \beta \frac{M}{\log M}.$$

**Remarque.** Les théorèmes de Hadamard et de de la Vallée-Poussin assurent  $\pi(M) \sim \frac{M}{\ln M}$ .

On définit la **fonction de de Bruijn**  $\psi(M, B)$  comme le nombre d'entiers  $\leq M$  qui sont  $B$ -friables.

D'abord, **quelques résultats de théorie des nombres.**

Soit  $M$  un entier  $\geq 2$ .

**Théorème de Tchebychev.** Le nombre  $\pi(M)$  de nombres premiers compris entre 2 et  $M$  vérifie

$$\alpha \frac{M}{\log M} \leq \pi(M) \leq \beta \frac{M}{\log M}.$$

**Remarque.** Les théorèmes de Hadamard et de de la Vallée-Poussin assurent  $\pi(M) \sim \frac{M}{\ln M}$ .

On définit la **fonction de de Bruijn**  $\psi(M, B)$  comme le nombre d'entiers  $\leq M$  qui sont  $B$ -friables.

**Proposition.** Si  $\log M \ll B \ll M$ , alors  $\psi(M, B)$  vérifie

$$\frac{\psi(M, B)}{M} \sim \left( \frac{\log M}{\log B} \right)^{-(\log M)/(\log B)}.$$

## Étape 1 : base de facteurs

- ▶ Il y a  $\pi(B)$  premiers pour lesquels on doit tester la résiduosit  quadratique de  $N$ 
  - complexit  en  $O(B \log B \log N)$
  - on a donc  $s \simeq \pi(B)/2$  premiers dans  $\mathcal{P}$

## Étape 1 : base de facteurs

- ▶ Il y a  $\pi(B)$  premiers pour lesquels on doit tester la résiduosit  quadratique de  $N$ 
  - complexit  en  $O(B \log B \log N)$
  - on a donc  $s \simeq \pi(B)/2$  premiers dans  $\mathcal{P}$

##  tape 2 : effritement par crible. Soit $M$ la taille maximale d'un entier $Q(x)$   traiter.

- ▶ Alors, il faudra traiter  $s \cdot \frac{M}{\psi(M, B)}$  entiers en moyenne

## Étape 1 : base de facteurs

- ▶ Il y a  $\pi(B)$  premiers pour lesquels on doit tester la résiduosit  quadratique de  $N$ 
  - complexit  en  $O(B \log B \log N)$
  - on a donc  $s \simeq \pi(B)/2$  premiers dans  $\mathcal{P}$

##  tape 2 : effritement par crible. Soit $M$ la taille maximale d'un entier $Q(x)$   traiter.

- ▶ Alors, il faudra traiter  $s \cdot \frac{M}{\psi(M,B)}$  entiers en moyenne
- ▶ On peut montrer que pour chaque entier du tableau, on fait  $O(\log \log B)$  op rations

## Étape 1 : base de facteurs

- ▶ Il y a  $\pi(B)$  premiers pour lesquels on doit tester la résiduosit  quadratique de  $N$ 
  - complexit  en  $O(B \log B \log N)$
  - on a donc  $s \simeq \pi(B)/2$  premiers dans  $\mathcal{P}$

##  tape 2 : effritement par crible. Soit $M$ la taille maximale d'un entier $Q(x)$   traiter.

- ▶ Alors, il faudra traiter  $s \cdot \frac{M}{\psi(M,B)}$  entiers en moyenne
- ▶ On peut montrer que pour chaque entier du tableau, on fait  $O(\log \log B)$  op rations
- ▶ La complexit  est donc  $C_2 = O(s \log \log B \cdot \frac{M}{\psi(M,B)}) = O(B \frac{\log \log B}{\log B} u^u)$  o   $u = \frac{\log M}{\log B}$

## Étape 1 : base de facteurs

- ▶ Il y a  $\pi(B)$  premiers pour lesquels on doit tester la résiduosit  quadratique de  $N$ 
  - complexit  en  $O(B \log B \log N)$
  - on a donc  $s \simeq \pi(B)/2$  premiers dans  $\mathcal{P}$

##  tape 2 : effritement par crible. Soit $M$ la taille maximale d'un entier $Q(x)$   traiter.

- ▶ Alors, il faudra traiter  $s \cdot \frac{M}{\psi(M,B)}$  entiers en moyenne
- ▶ On peut montrer que pour chaque entier du tableau, on fait  $O(\log \log B)$  op rations
- ▶ La complexit  est donc  $C_2 = O(s \log \log B \cdot \frac{M}{\psi(M,B)}) = O(B \frac{\log \log B}{\log B} u^u)$  o   $u = \frac{\log M}{\log B}$
- ▶ Si  $M \simeq \sqrt{N}$  et  $\log B \geq \sqrt{\log N}$ , alors on obtient :

$$\log C_2 \simeq \log B + \frac{\log N \log \log N}{4 \log B}$$



## Étape 1 : base de facteurs

- ▶ Il y a  $\pi(B)$  premiers pour lesquels on doit tester la résiduosit  quadratique de  $N$   
→ complexit  en  $O(B \log B \log N)$   
→ on a donc  $s \simeq \pi(B)/2$  premiers dans  $\mathcal{P}$

##  tape 2 : effritement par crible. Soit $M$ la taille maximale d'un entier $Q(x)$   traiter.

- ▶ Alors, il faudra traiter  $s \cdot \frac{M}{\psi(M,B)}$  entiers en moyenne
- ▶ On peut montrer que pour chaque entier du tableau, on fait  $O(\log \log B)$  op rations
- ▶ La complexit  est donc  $C_2 = O(s \log \log B \cdot \frac{M}{\psi(M,B)}) = O(B \frac{\log \log B}{\log B} u^u)$  o   $u = \frac{\log M}{\log B}$
- ▶ Si  $M \simeq \sqrt{N}$  et  $\log B \geq \sqrt{\log N}$ , alors on obtient :

$$\log C_2 \simeq \log B + \frac{\log N \log \log N}{4 \log B}$$

Cette derni re quantit  se maximise pour  $\log B \simeq \frac{1}{2} \sqrt{\log N \log \log N}$ , donc  $B \in L_{\log N}[\frac{1}{2}, \frac{1}{2}]$ , et donne

$$C_2 = \exp(\sqrt{\log N \log \log N}) = L_{\log N}[\frac{1}{2}, 1].$$

**Étape 3** : algèbre linéaire.

- ▶ On résout un système linéaire **creux** de taille  $t \times s$  sur  $\mathbb{F}_2$  où  $t \simeq s \in O\left(\frac{B}{\log B}\right)$

**Étape 3** : algèbre linéaire.

- ▶ On résout un système linéaire **creux** de taille  $t \times s$  sur  $\mathbb{F}_2$  où  $t \simeq s \in O(\frac{B}{\log B})$   
→ Par l'algorithme de Wiedemann (par exemple), on a une complexité en

$$O(ts) = O(B^2 / \log^2 B) = O(\exp(\sqrt{\log N \log \log N})) = O(L_{\log N}[\frac{1}{2}, 1])$$

Étape 3 : algèbre linéaire.

- ▶ On résout un système linéaire **creux** de taille  $t \times s$  sur  $\mathbb{F}_2$  où  $t \simeq s \in O(\frac{B}{\log B})$   
→ Par l'algorithme de Wiedemann (par exemple), on a une complexité en

$$O(ts) = O(B^2 / \log^2 B) = O(\exp(\sqrt{\log N \log \log N})) = O(L_{\log N}[\frac{1}{2}, 1])$$

- ▶ Calculs terminaux (produits, pgcd) en  $O(B \log^2 N)$

**Étape 3 :** algèbre linéaire.

- ▶ On résout un système linéaire **creux** de taille  $t \times s$  sur  $\mathbb{F}_2$  où  $t \simeq s \in O(\frac{B}{\log B})$   
→ Par l'algorithme de Wiedemann (par exemple), on a une complexité en

$$O(ts) = O(B^2 / \log^2 B) = O(\exp(\sqrt{\log N \log \log N})) = O(L_{\log N}[\frac{1}{2}, 1])$$

- ▶ Calculs terminaux (produits, pgcd) en  $O(B \log^2 N)$

**Conclusion.** L'algorithme de crible quadratique permet de factoriser un entier  $N$  en temps

$$O(\exp(\sqrt{\log N \log \log N})).$$

Le **crible algébrique** généralise l'idée du crible quadratique. L'idée est de chercher des  $a^2 \equiv b^2 \pmod N$  en cherchant des carrés dans des **anneaux d'entiers**.

Le **crible algébrique** généralise l'idée du crible quadratique. L'idée est de chercher des  $a^2 \equiv b^2 \pmod N$  en cherchant des carrés dans des **anneaux d'entiers**.

**Idée.** Soit  $f(X) \in \mathbb{Z}[X]$  unitaire et irréductible, et  $m \in \mathbb{Z}$  qui satisfait  $f(m) \equiv 0 \pmod N$ . En posant  $\alpha = \bar{X} \in \mathbb{Z}[X]/(f)$ , on a  $\mathbb{Z}[X]/(f) = \mathbb{Z}[\alpha]$ .

Le **crible algébrique** généralise l'idée du crible quadratique. L'idée est de chercher des  $a^2 \equiv b^2 \pmod N$  en cherchant des carrés dans des **anneaux d'entiers**.

**Idée.** Soit  $f(X) \in \mathbb{Z}[X]$  unitaire et irréductible, et  $m \in \mathbb{Z}$  qui satisfait  $f(m) \equiv 0 \pmod N$ . En posant  $\alpha = \bar{X} \in \mathbb{Z}[X]/(f)$ , on a  $\mathbb{Z}[X]/(f) = \mathbb{Z}[\alpha]$ .

On considère ensuite le morphisme d'anneaux

$$\begin{aligned} \phi &: \mathbb{Z}[\alpha] &\rightarrow & \mathbb{Z}/N\mathbb{Z} \\ &P(\alpha) &\mapsto & \bar{P}(m) \pmod N \end{aligned}$$

Alors on cherche  $P \in \mathbb{Z}[X]$  tel que  $P(\alpha)$  est un carré  $z^2 \in \mathbb{Z}[\alpha]$  et  $\bar{P}(m)$  est un carré  $b^2 \in \mathbb{Z}/N\mathbb{Z}$ . Si  $a = \phi(z)$ , alors  $a^2 \equiv b^2 \pmod N$ .

Ensuite, pour trouver les éléments  $P$ , on va cribler les **normes** d'éléments  $u + v\alpha \in \mathbb{Z}[\alpha]$  (analogues des petits  $p_i$  pour le crible quadratique).



Le **crible algébrique** généralise l'idée du crible quadratique. L'idée est de chercher des  $a^2 \equiv b^2 \pmod{N}$  en cherchant des carrés dans des **anneaux d'entiers**.

**Idée.** Soit  $f(X) \in \mathbb{Z}[X]$  unitaire et irréductible, et  $m \in \mathbb{Z}$  qui satisfait  $f(m) \equiv 0 \pmod{N}$ . En posant  $\alpha = \bar{X} \in \mathbb{Z}[X]/(f)$ , on a  $\mathbb{Z}[X]/(f) = \mathbb{Z}[\alpha]$ .

On considère ensuite le morphisme d'anneaux

$$\begin{aligned} \phi &: \mathbb{Z}[\alpha] &\rightarrow & \mathbb{Z}/N\mathbb{Z} \\ &P(\alpha) &\mapsto & \bar{P}(m) \pmod{N} \end{aligned}$$

Alors on cherche  $P \in \mathbb{Z}[X]$  tel que  $P(\alpha)$  est un carré  $z^2 \in \mathbb{Z}[\alpha]$  et  $\bar{P}(m)$  est un carré  $b^2 \in \mathbb{Z}/N\mathbb{Z}$ . Si  $a = \phi(z)$ , alors  $a^2 \equiv b^2 \pmod{N}$ .

Ensuite, pour trouver les éléments  $P$ , on va cribler les **normes** d'éléments  $u + v\alpha \in \mathbb{Z}[\alpha]$  (analogues des petits  $p_i$  pour le crible quadratique).

La phase d'algèbre linéaire est sensiblement identique à celle du crible quadratique

Le **crible algébrique** généralise l'idée du crible quadratique. L'idée est de chercher des  $a^2 \equiv b^2 \pmod{N}$  en cherchant des carrés dans des **anneaux d'entiers**.

**Idée.** Soit  $f(X) \in \mathbb{Z}[X]$  unitaire et irréductible, et  $m \in \mathbb{Z}$  qui satisfait  $f(m) \equiv 0 \pmod{N}$ . En posant  $\alpha = \bar{X} \in \mathbb{Z}[X]/(f)$ , on a  $\mathbb{Z}[X]/(f) = \mathbb{Z}[\alpha]$ .

On considère ensuite le morphisme d'anneaux

$$\begin{aligned} \phi &: \mathbb{Z}[\alpha] &\rightarrow & \mathbb{Z}/N\mathbb{Z} \\ P(\alpha) &\mapsto & \bar{P}(m) & \pmod{N} \end{aligned}$$

Alors on cherche  $P \in \mathbb{Z}[X]$  tel que  $P(\alpha)$  est un carré  $z^2 \in \mathbb{Z}[\alpha]$  et  $\bar{P}(m)$  est un carré  $b^2 \in \mathbb{Z}/N\mathbb{Z}$ . Si  $a = \phi(z)$ , alors  $a^2 \equiv b^2 \pmod{N}$ .

Ensuite, pour trouver les éléments  $P$ , on va cribler les **normes** d'éléments  $u + v\alpha \in \mathbb{Z}[\alpha]$  (analogues des petits  $p_i$  pour le crible quadratique).

La phase d'algèbre linéaire est sensiblement identique à celle du crible quadratique

Au final, on obtient une **complexité** du crible algébrique en

$$O\left(\exp\left(\left(\frac{64}{9} \log N\right)^{1/3} (\log \log N)^{2/3}\right)\right).$$

Le **crible algébrique** généralise l'idée du crible quadratique. L'idée est de chercher des  $a^2 \equiv b^2 \pmod{N}$  en cherchant des carrés dans des **anneaux d'entiers**.

**Idée.** Soit  $f(X) \in \mathbb{Z}[X]$  unitaire et irréductible, et  $m \in \mathbb{Z}$  qui satisfait  $f(m) \equiv 0 \pmod{N}$ . En posant  $\alpha = \bar{X} \in \mathbb{Z}[X]/(f)$ , on a  $\mathbb{Z}[X]/(f) = \mathbb{Z}[\alpha]$ .

On considère ensuite le morphisme d'anneaux

$$\begin{aligned} \phi &: \mathbb{Z}[\alpha] &\rightarrow & \mathbb{Z}/N\mathbb{Z} \\ &P(\alpha) &\mapsto & \bar{P}(m) \pmod{N} \end{aligned}$$


Alors on cherche  $P \in \mathbb{Z}[X]$  tel que  $P(\alpha)$  est un carré  $z^2 \in \mathbb{Z}[\alpha]$  et  $\bar{P}(m)$  est un carré  $b^2 \in \mathbb{Z}/N\mathbb{Z}$ . Si  $a = \phi(z)$ , alors  $a^2 \equiv b^2 \pmod{N}$ .

Ensuite, pour trouver les éléments  $P$ , on va cribler les **normes** d'éléments  $u + v\alpha \in \mathbb{Z}[\alpha]$  (analogues des petits  $p_i$  pour le crible quadratique).

La phase d'algèbre linéaire est sensiblement identique à celle du crible quadratique

Au final, on obtient une **complexité** du crible algébrique en

$$O\left(\exp\left(\left(\frac{64}{9} \log N\right)^{1/3} (\log \log N)^{2/3}\right)\right).$$

 *A Tale of Two Sieves*. C. Pomerance. Notices of the AMS. 1996. . [[lien](#)].

 *Prime Numbers, a Computational Perspective*. R. Crandall, C. Pomerance. Springer. 2001.

Pour extraire des facteurs de taille  $\leq 60$ -70 chiffres (« moyens »), on utilise la **méthode ECM**. Le record de factorisation d'ECM a produit un facteur de 83 chiffres.

Pour des facteurs de taille plus importante, on utilise le **crible algébrique**.

Pour extraire des facteurs de taille  $\leq 60$ -70 chiffres (« moyens »), on utilise la **méthode ECM**. Le record de factorisation d'ECM a produit un facteur de 83 chiffres.

Pour des facteurs de taille plus importante, on utilise le **crible algébrique**.

Le **record** de factorisation de modules RSA est RSA-250, effectuée en février 2020 :

```
214032465024074496126442307283933356300861471514475501779775492088141802344714013664
334551909580467961099285187247091458768739626192155736304745477052080511905649310668
7691590019759405693457452230589325976697471681738069364894699871578494975937497937

= 641352894770715802787901901705773890848250147429434472081168596320245323446302386235
98752668347708737661925585694639798853367

× 333720275949781565562260106053551142279407603447675546667845209870238417292100370802
57448673296881877565718986258036932062711
```

Pour extraire des facteurs de taille  $\leq 60$ -70 chiffres (« moyens »), on utilise la **méthode ECM**. Le record de factorisation d'ECM a produit un facteur de 83 chiffres.

Pour des facteurs de taille plus importante, on utilise le **crible algébrique**.

Le **record** de factorisation de modules RSA est RSA-250, effectuée en février 2020 :

```
214032465024074496126442307283933356300861471514475501779775492088141802344714013664
334551909580467961099285187247091458768739626192155736304745477052080511905649310668
7691590019759405693457452230589325976697471681738069364894699871578494975937497937

= 641352894770715802787901901705773890848250147429434472081168596320245323446302386235
98752668347708737661925585694639798853367

× 333720275949781565562260106053551142279407603447675546667845209870238417292100370802
57448673296881877565718986258036932062711
```

**Temps de factorisation** : équivalent 2700 années (oui !) de calcul sur 1 coeur.

Source :

<https://lists.gforge.inria.fr/pipermail/cado-nfs-discuss/2020-February/001166.html>

Une **implémentation** proche de l'état de l'art de la recherche : CADO-NFS

- majoritairement codé en C, C++, plus de l'assembleur pour accélérer certains calculs
- licence LGPL (libre), développé principalement en France (notamment une équipe Inria à Nancy)
- permet de factoriser sur un processeur standard (Intel(R) Xeon(R) CPU E5-2650 @2.00GHz) : (source : site web cado-nfs)

RSA-120	RSA-130	RSA-140	RSA-155
1,9 heure	7,5 heures	23 heures	5,3 jours


- utilisé pour la factorisation record

Une **implémentation** proche de l'état de l'art de la recherche : CADO-NFS

- majoritairement codé en C, C++, plus de l'assembleur pour accélérer certains calculs
- licence LGPL (libre), développé principalement en France (notamment une équipe Inria à Nancy)
- permet de factoriser sur un processeur standard (Intel(R) Xeon(R) CPU E5-2650 @2.00GHz) : (source : site web cado-nfs)

RSA-120	RSA-130	RSA-140	RSA-155
1,9 heure	7,5 heures	23 heures	5,3 jours

- utilisé pour la factorisation record

 *CADO-NFS, An Implementation of the Number Field Sieve Algorithm.* The CADO-NFS Development Team. . 2017.  
<http://cado-nfs.gforge.inria.fr>



**Questions?**