Théorie Algorithmique des Nombres

Problèmes difficiles en cryptographie

David Kohel

12/10/2020

12/10/2020

Problèmes difficiles en cryptographie

Un problème difficile est un problème computationnel dont sa résolution n'admet pas d'algorithme (connu) avec complexité polynôme dans la taille d'entrée.

Factorisation des entiers.

On peut multiplier des entiers $(\mathbb{Z} \times \mathbb{Z} \longrightarrow \mathbb{Z})$ en temps polynôme, en effet quadratique : $(p,q) \longmapsto pq$;
taille d'entrée $x = \log_2 p + \log_2 q$
complexité : $O((\log p)(\log q)) \subseteq O(x^2)$.
On a égalité pour $\log p \sim \log q$,
i.e. données d'entrées

$$\lambda \log p \leq \log q \leq \mu \log p,$$

λ, μ constants.

Il semble qu'il est difficile de décomposer un produit $n = pq$ en premiers (pire cas : p et q premiers).

$$\mathbb{Z} \times \mathbb{Z} \longrightarrow \mathbb{Z}$$
$$? \quad (p, q) \longmapsto n \ (= pq)$$
$$\notin \{1\} \times \mathbb{Z}, \ \mathbb{Z} \times \{1\}$$

On va étudier ce problème et présenter les approches algo. à sa résolution.

La difficulté de ce problème est nécessaire pour la sécurité de RSA.

Attention. Si on trouve une factorisation on peut vérifier que la décomposition est correcte.

Le logarithme discret

Soit $\mathbb{F}_q$ un corps fini ($q = p^r$, une puissance d'un premier p), et $\alpha \in \mathbb{F}_q^*$ un élément primitif (= générateur du groupe).

Alors on a une bijection :

$$(\mathbb{Z}/(q-1)\mathbb{Z}, +) \longrightarrow (\mathbb{F}_q, \cdot)$$
$$k \longmapsto \alpha^k$$

En particulier $0 \longmapsto 1$ (élément neutre), $1 \longmapsto \alpha$ (générateur du groupe), etc.

Le problème d'exponentiation est polynôme dans la taille d'entrée : $x = \log_2(q) + \log_2(k)$

Complexité :

$$O(M(\log q)\log(k)) \subseteq O(\log(q)^2 \log(k)) \subseteq O(x^3).$$

Il semble qu'il est difficile d'inverser l'application :

$$\exp_\alpha : \mathbb{Z}/(q-1)\mathbb{Z} \longrightarrow \mathbb{F}_q^*$$
$$k \longmapsto \alpha^k$$

Son inverse est l'application

$$\log_\alpha : \mathbb{F}_q^* \longrightarrow \mathbb{Z}/(q-1)\mathbb{Z}$$

telle que

$$\log_\alpha \circ \exp_\alpha = id_{\mathbb{Z}/(q-1)\mathbb{Z}}.$$

On va également étudier les approches algo. à ce problème.

<u>Rappel</u> La difficulté du problème du log. discret est nécessaire pour la sécurité du protocole de Diffie-Hellman et le cryptosystème ElGamal.

Factorisation

On a vu qu'on peut factiser n en temps $O(\sqrt{n})$ mults en utilisant division naïve (mod 2, mod 3, etc.). (Exponentiel)

Log discret:

Il existe un algorithme aussi avec complexité $O(\sqrt{p})$ opérations.

Rappel: L'algo naïf est dans la complexité $O(p)$. Données:

$\alpha, \beta \in \mathbb{F}_p^*$ ($p = q =$ premier).

On calcule:

α^k: $1, \alpha, \alpha^2, \alpha^3$, etc, $\beta = \alpha^k$

k 0, 1, 2, 3, k $O(k) \leq O(p)$ (pire cas)

alors $k = \log_\alpha(\beta) \in \mathbb{Z}/(p-1)\mathbb{Z}$

— trouvé après $O(p)$ mults.

Complexité: $O((\log p)^2 p)$.

N.B. Si $k \in \mathbb{Z}/(p-1)\mathbb{Z}$ avec dist. uniforme, l'espérance math de k est $\sim \frac{p-1}{2}$ et donc l'esperance de $O(k)$ est $O(p)$.

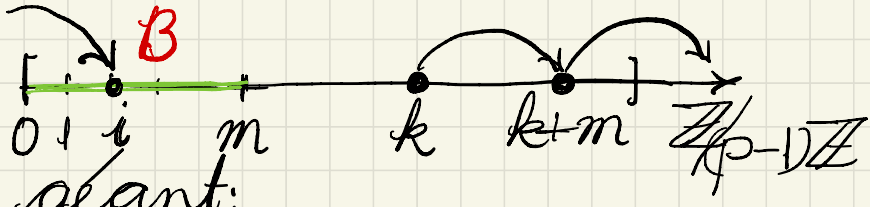
<u>Pas de bébé, pas de géant</u>

On pose $m = \lfloor \sqrt{p} \rfloor$. On calcule:

$1, \alpha, \dots, \alpha^m \in \mathbb{F}_p^*$ $\quad \mathcal{B} = \{1, \alpha, \dots, \alpha^m\}$

$0, 1, \dots, m \in \mathbb{Z}/(p-1)\mathbb{Z}$

Cette suite s'appelle les pas de bébé:



Pas de géant:

Étant donné α^m et β, on calcule $\beta, \alpha^m\beta, \alpha^{2m}\beta, \dots, \alpha^{dm}\beta = \alpha^i$ avec $\in \mathcal{B}$. $0 \leq i \leq m$.

On a donc $\alpha^{lm}\beta = \alpha^i$, ou

$$\beta = \alpha^{i-lm},$$

donc $k = i - lm \bmod (p-1)$.

Mais $m, l \in O(\sqrt{p})$ et donc on a calculé $O(\sqrt{p})$ mults.

La complexité est donc

$$O((\log p)^2 \sqrt{p}). \quad \text{(mult naïve)}$$

Attention:

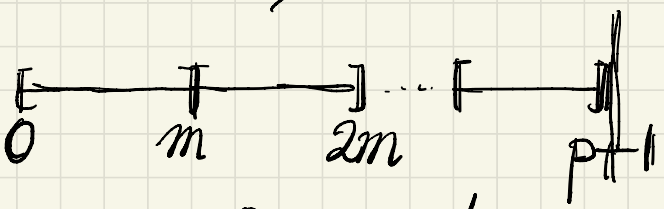
Si on teste $\alpha^{jm} \in \{1, \alpha, \ldots, \alpha^m\}$ en traversant toute la liste, on utilise m opérations pour chaque j dans $0 < j \leq \sqrt{p}$.

La complexité dévient $O(p)$.

On a besoin d'un abre binaire de recherche pour identifier α^{jm} dans $\{1, \alpha, \ldots, \alpha^m\}$.

N.B. On va trouver une collision $\alpha^{lm}\beta = \alpha^i$ avec $l \leq \frac{p-1}{m}$ car

$$\bigcup_{j=1}^{(p-1)/m} \left([\![1, \dots, m]\!] + jm \right) = \mathbb{Z}/(p-1)\mathbb{Z}.$$



Remarque On peut varier m par un constant. $\in O(\sqrt{p})$

Si m est plus petit, la borne $\frac{p-1}{m}$ sur le nombre de pas de géant est plus grand.

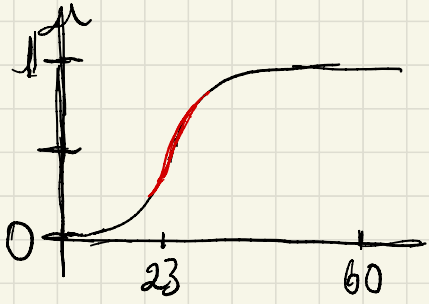
Or, on a besoin de garder en mémoire les m pas de bébé.

Autres algorithmes avec des complexités exponentielles, mais plus efficaces ou avec moins de mémoire.

Pollard ρ (algo. probabiliste)
$\Rightarrow$ temps espéré.

Paradoxe des anniversaires

Dans une salle de cours, quelle est la probabilité que deux personnes partage le même jour d'anniversaire (dans une année de 365 jours).
Quand le nombre de personnes est 23, la probabilité d'une collision est 50%.
En effet la proba :



En générale, pour un ensemble de N éléments, et un échantillon de k éléments au hazard (avec une distribution uniforme), la proba que les k sont distincts

$q(N,k)$ = probabilité que k éléments soient distincts

$$= \prod_{i=1}^{k} \frac{(N-i+1)}{N} = \frac{N!}{N^k (N-k)!}$$

$p(N,k) = 1 - q(N,k)$

= proba d'une collision.

<u>Approximations</u>

$$q(N,k) \approx \exp\left(-\binom{k}{2}/N\right) = \exp\left(\frac{-k(k-1)}{2N}\right)$$

en particulier pour $k = \sqrt{2N}$, $\approx \exp\left(\frac{-k^2}{2N}\right)$.

la prop descend à $\approx e^{-1} = 0{,}3678\ldots$,

et pour $k = 2\sqrt{N}$ $\approx e^{-2} = 0{,}13533\ldots$

Prochaine fois:

Algorithme de Floyd:

Algorithme pour déterminer un cycle dans une suite récurrente engendré par une fonction $f : S \longrightarrow S$:

$x_0, x_1, x_2, \ldots \in S$

définie par $f(x_i) = x_{i+1}$.

Défn: On dit que la suite est déterministe si x_{i+1} est déterminé par x_i.

Remarque. Si S est fini, de cardinal N, toute suite déterministe est récurrente.

Algorithme de Pollard est application au cas spécifique de l'algo de Floyd.