Calculateur quantique: factorisation des entiers

Plan

- Introduction
- Difficulté de la factorisation des entiers
- Cryptographie et la factorisation
- Exemple RSA
- L'informatique quantique
- L'algorithme quantique de Shor
- Conclusion

Introduction

- **Factorisation des entiers problème difficile**
  - Par essence ?
  - Ou seulement pas encore de méthode habile depuis l'antiquité?

- **Cryptographie et la factorisation des entiers**
  - Cryptographie classique (exemple permutation) fragilisée par l'évolution technologique et mathématique.
  - Apparition de la cryptographie moderne basée sur la difficulté de la factorisation des entiers. (problème temps exponentiel)
  - L'avènement de l'informatique quantique, fragilise la cryptographie moderne
  - Factorisation en temps polynomial

Difficulté de la factorisation

- Fonction trape
  - Multiplication de grands nombre entiers n'est pas compliquée.
  - Par exemple multiplier deux nombre de 65 chiffres est immédiat
  - Mais l'opération inverse est difficile
  - Euclide : chaque nombre naturel est un produit de nombres premiers et sa décomposition en facteur premiers est unique à l'ordre près
  - Exemple 12= 3x2x2
  - 1977: Quelle est la décomposition du nombre :
  - 114 381 625 757 888 867 669 235 779 976 146 612 010 218 296 721 242 362 562 561 842 935 706 935 245 733 897 830 597 123 563 958 705 058 989 075 147 599 290 026 879 543 541 ?

    Réponse 16 ans plus tard à l'université d' Oxford avec distribution des tâches à 600 volontaires connectés sur internet

    Nombre de Fermat: $F_n = 2^{2n} + 1$

    $F_0 = 3,\ F_1 = 5,\ F2 = 17,\ F_3 = 257,\ F_4 = 65\,537$ *mais* F_5 *pas premier*
  - Test de primalité: si N premier (sauf 2) alors N est diviseur de

$$2^{n-1} - 1$$

Difficulté de la factorisation

- **Exemple de factorisation par divisions successives**
  - Soit N
  - Factoriser N c'est recherché tous les nombres premiers n diviseurs de N avec 1<n<N
  - Si le plus petit n devient grand la liste des nombres premiers devient de plus en plus grands même pour un ordinateur classique
  - L'algorithme NFS (le plus efficace) temps exponentiel

- **Difficulté utilisée par la cryptographie moderne**

RSA

- 1977, auteurs: Ron **R**ivest, Adi **S**hamir et Len **A**dleman
- Cryptographie clé publique et clé privée
- Protocole de cryptage fondé sur :
  Si p et q sont deux nombres premiers distincts, n = pq, h = (p - 1)(q - 1).
  Si e est un entier, e E]1, h[, et e premier avec h (pas de diviseur commun entre e et h)
  Alors, il existe un unique entier d E]1, h[tel que ed º 1 [mod h]

Fontionnement RSA

Alice, choisit deux nombres premiers p et q, calcule leur produit n = pq, calcul h = (p - 1)(q - 1) et choisit un entier e Î]1, h[et premier avec h.

Puis elle rend publique, dans un annuaire, l'information (Alice : n, e).

a) Comment Bob envoie-t-il un message à Alice ?

Alice à choisi p = 13, q = 17 et e = 5, elle communique alors dans l'annuaire ses deux clés publiques n = pq = 221 et e = 5.

Bob veut envoyer à Alice le message "JE T'AIME"

Il l'écrit d'abord sous la forme (10, 5, 0, 20, 27, 1, 9, 13, 5), en utilisant la position des lettres dans l'alphabet (A est la 1ère lettre, ...) et en posant 0 correspondant à l'espace et 27 à l'apostrophe.

Puis Bob élève chaque nombre de la liste à la puissance e = 5 et en prend le reste dans la division par

n = 221.

Message initial		**Message crypté**
(10, 5, 0, 20, 27, 1, 9, 13, 5)	**Þ**	**(108, 31, 0, 141, 40, 1, 42, 13, 31)**

Bob transmet alors son "message" crypté à Alice.

b) Comment Alice déchiffre-t-elle le message de Bob ?

Alice, qui est la seule à connaître p et q, calcule le nombre dÎ]1, h[tel que

ed º 1 [mod h]. Ici h = 192 et d = 77.

Alice élève alors chaque nombre du message crypté à la puissance d = 77, en prend le reste dans la division par n = 221 et reconstitue le texte original, grâce au théorème énoncé plus haut et à son corollaire.

Message crypté		**Message initial**
(108, 31, 0, 141, 40, 1, 42, 13, 31)	**Þ**	**(10, 5, 0, 20, 27, 1, 9, 13, 5)**

La sécurité de RSA

- **La décomposition de N**
  - RSA s'appuie sur la difficulté de la factorisation des grands nombres
  - Donc prendre un N grand pour sécuriser le protocole
  - Concours laboratoires RSA pour sonder la sécurité
  - Records RSA- 640(193 chiffres décimaux) établi avec un réseau de 80 processeurs cadencés à 2,2 GH pendant quatre mois.
  - Il y a 25 ans il faudrait des millions d'années.

- **Mais apparition informatique quantique :**
  - Promesse de factorisation en temps polynomial .
  - Sécurité RSA menacée

Eléments de l'informatique quantique

- **Fusion sciences de l'information et physique quantiques**
  - 1982 Richard Feynmann : utilisation de la physique quantique à la place de la physique classique comme support matériel de l'information et du calcul
  - conséquences :
    - Traiter des problèmes hors de portée de l'informatique actuelle.
  - 1990 premiers résultats, algorithmiques et théoriques puis expérimentaux
  - Eléments de base:

Le qubit (quantum + bit), état quantique qui représente la plus petite entité de stockage d'information quantique.

Il se compose d'une superposition de deux états de base, par convention nommés |0> et |1> prononcés: ket 0 et ket 1.

Un qubit peut être soit dans l'état 0 soit dans l'état 1, ou soit dans une superposition de l'état 0 et 1.

En général, le qubit est dans l'état , les coefficients étant des nombres complexes vérifiant

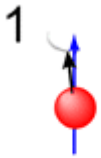
$|\alpha|^2 + |\beta|^2 = 1$. Mais le résultat final est soit 0 ou 1.

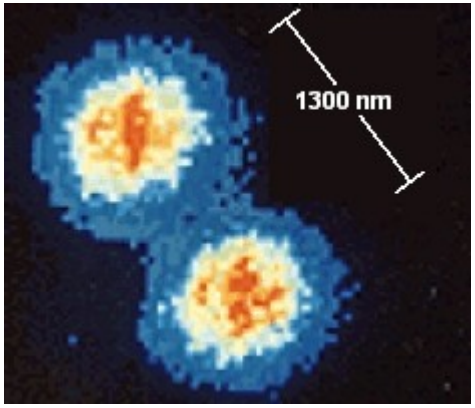
La probabilité de mesurer l'état 0 vaut $|\alpha|^2$, tandis que celle de mesurer l'état 1 vaut $|\beta|^2$,, après mesure le qubit se trouve donc dans l'état mesuré

Mais finit-on quand-même par obtenir ainsi le résultat que l'on cherche? Oui, si l'algorithme quantique a été bien conçu

Eléments de l'informatique quantique

Conséquence: tout le jeu de l'algorithmique quantique a pour but d'amener aussi près de 1 que possible la probabilité d'obtenir un résultat pertinent, et cela en effectuant aussi peu d'opérations que possible.





A gauche, les Drs Isaac Chuang et Costantino Yannoni du MIT manipulant des éprouvettes contenant des molécules organiques. Aussi étonnant que cela soit, elles constituent le coeur de leur ordinateur quantique qui peut se réduire à une seule molécule et ses qubits représentés par les noyaux atomiques contenus dans la molécule ! A l'arrière-plan le cylindre métallisé du système RMN constitué par un aimant à supra-conducteur plongé dans un dewar contenant de l'hélium liquide à -269°C. Au centre, les deux spins d'une particule. A droite, deux ions de baryum au repos

Eléments de l'informatique quantique

- ## Superposition quantique du qubit

L'état d'un registre de n qubits pourra être une superposition d'un ensemble quelconque des 2^n valeurs possibles sur n bits

un calcul quantique pourra agir simultanément sur 2^n valeurs différentes, ceci apporte une importante parallélisation

Ainsi si une fonction peut être calculée avec 2^n arguments différents, on calculera toutes ses valeurs simultanément

Dans un programme on ne pourra pas affecter la valeur d'une variable quantique à une autre, ni utiliser cette valeur plus d'une fois

Donc il faut une algorithmique radicalement nouvelle, et des langages de programmation qui respectent ces lois du monde quantique, en particulier téléportation quantique.

Eléments de l'informatique quantique

La téléportation quantique

protocole de communications quantiques consistant à transférer l’état quantique d’un système vers un autre système similaire et séparé spatialement du premier en utilisant l'intrication quantique.

Le principe du transfert de l'information quantique consiste à utiliser une paire auxiliaire de particules intriqués, et de les positionner dans des conditions bien particulières, de part et d'autre du système, c'est à dire à l'émetteur et au récepteur. Ces particules étant intriquées, ne formant qu'un tout, vont pouvoir interagir ensemble

Eléments de l'informatique quantique

Principe d'un calculateur quantique

Quand un qubit est polarisé ça va leur est déterminée pas sa phase : P comprise entre 0 et 180 degrés

Si P n'est pas comprise entre 0 et 90 degrés il est dans un état de superposition.

Donc il faut un algorithme qui amène P systématiquement soit à 0 ou 90 degrés (o ou 1)

Puissance des calculateurs quantiques

Pour factoriser un nombre N de 300 chiffres, le meilleur algorithme fait de l'ordre 10^{26}opérations

Un ordinateur cassique dont les performances sont de l'ordre de 100 teraflops (10^{14}opérations par seconde)

Il faudra 10^{12}secondes c'est à dire 30000 ans.

1994 Peter Shor : avec un calculateur quantique il suffira de 10 secondes

Algorithme de Shor

Algorithme quantique

- Pour factoriser un nombre *N* en temps $O((logN)^3)$ et en espace $O(logN)$

- Conséquence RSA peut être déchiffré par factorisation de sa clé publique *N*

    - *Procédure*

Soit un entier donné N, nous essayons de trouver un autre entier p compris entre 1 et N qui divise N

L'algorithme de Shor consiste en deux parties :

- Une réduction du problème de factorisation en un problème de recherche d'ordre, qui peut être effectué sur un ordinateur classique (algorithme d'Euclide)
- Un algorithme quantique pour résoudre le problème de recherche d'ordre.

Algorithme de Shor

Partie classique

1. Prendre un nombre pseudo-aléatoire a < N
2. Calculer le pgcd(a, N). Ceci peut être effectué par l'utilisation de l'algorithme d'Euclide.
3. Si pgcd(a, N) ≠ 1, alors c'est un facteur non-trivial de N, donc effectué.
4. Autrement, utiliser la sous-routine de recherche de période (ci-dessous) pour trouver r, la période de la fonction suivante :

   c.a.d. le plus petit entier r pour lequel f(x + r) = f(x).
5. Si r est impair, retourner à l'étape 1.
6. Si a r/2 ≡ -1 (mod N), retourner à l'étape 1.
7. Les facteurs de N sont pgcd(ar/2 ± 1, N). Effectué.

Algorithme de Shor

Partie classique

1. Prendre un nombre pseudo-aléatoire a < N
2. Calculer le pgcd(a, N). Ceci peut être effectué par l'utilisation de l'algorithme d'Euclide.
3. Si pgcd(a, N) ≠ 1, alors c'est un facteur non-trivial de N, donc effectué.
4. Autrement, utiliser la sous-routine de recherche de période (ci-dessous) pour trouver r, la période de la fonction suivante :

   c.a.d. le plus petit entier r pour lequel f(x + r) = f(x).
5. Si r est impair, retourner à l'étape 1.
6. Si a r/2 ≡ -1 (mod N), retourner à l'étape 1.
7. Les facteurs de N sont pgcd(ar/2 ± 1, N). Effectué.

Algorithme de Shor

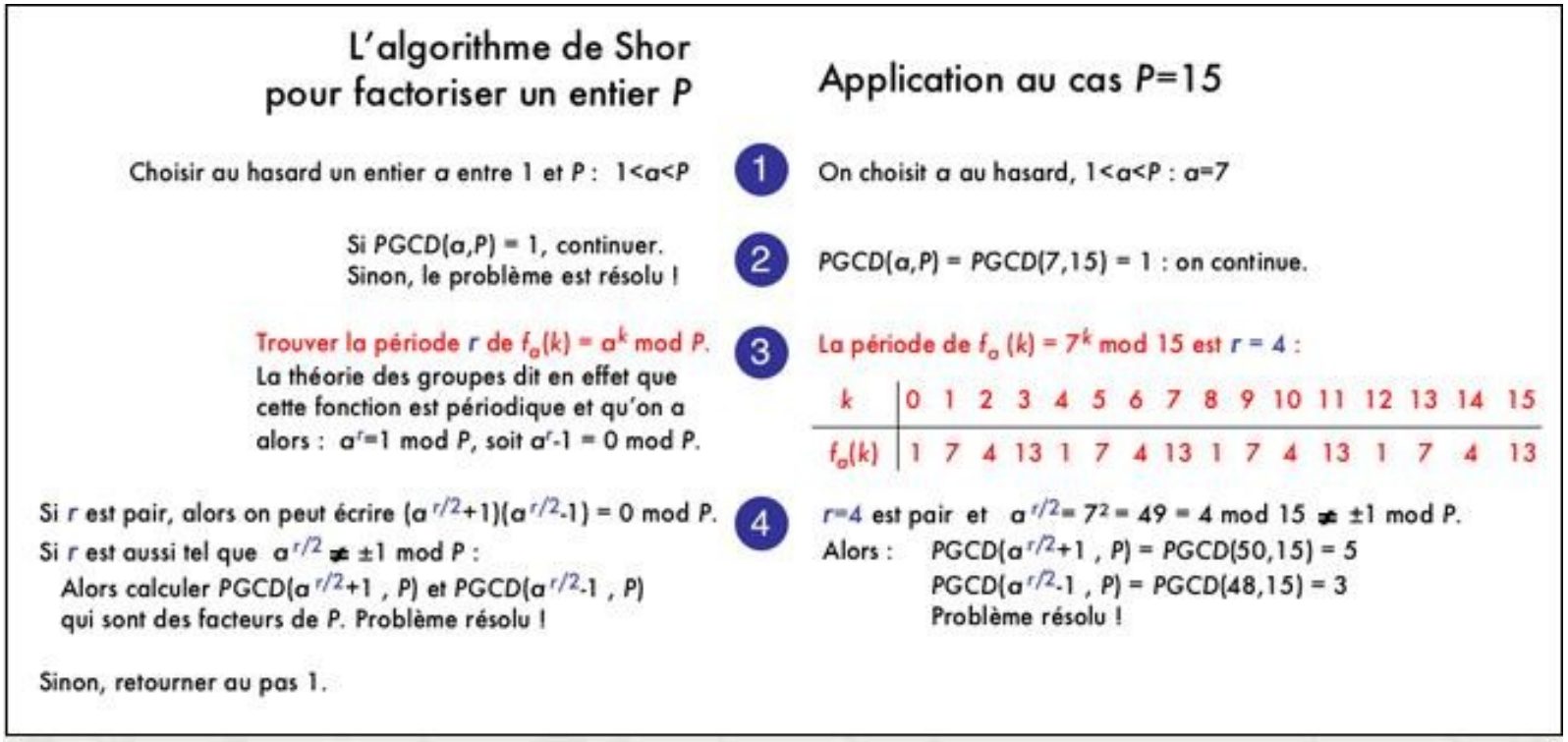
Partie quantique

Objectif : trouve la période en utilisant la transformée de Fourier quantique et est responsable de l'accélération quantique.

- Pour calculer la période d'une fonction f, nous évaluons la fonction en tous ses points simultanément
- 1. Créer une superposition d'états.
- 2. Implémenter la fonction f comme une transformation quantique.
- 3. Exécuter une transformation de Fourier quantique.

Algorithme de Shor

Exemple



conclusion

- Possibilité de factorisation de grands nombres entiers en temps polynomial
- Résoudre de nombreux problèmes difficiles pour l'informatique classiques
- Proposer une distribution de clés sécrétes plus sûre via la cryptographie quantique
- Construction d'ordinateurs quantiques possible d'ici 2015
- Conséquences: diplomatiques, politiques, financières .....