Séminaire ALGO, 5 février 2001

30 ans de factorisation d'entiers

F. Morain

ÉCOLE POLYTECHNIQUE

LIX

CNRS

morain@lix.polytechnique.fr

http ://www.lix.polytechnique.fr/Labo/Francois.Morain

Planche 1

Plan

Planche 2

I. Introduction

Pourquoi factorise-t-on ?

● problème naturel en théorie (algorithmique) des nombres : permet de se restreindre aux propriétés des nombres premiers ;

● relié à la sécurité de beaucoup d'algorithmes utilisés en cryptographie publique.

Ex. le protocole RSA :

$$N = p\,q, \quad \lambda(N) = \mathrm{ppcm}(p-1, q-1)$$

$$\mathrm{pgcd}(e, \lambda(N)) = 1 \;\Rightarrow\; \exists\, d,\; e\,d \equiv 1 \bmod \lambda(N)$$

clair	:	x
chiffrement	:	$y \equiv x^{e_B} \bmod N$
déchiffrement	:	$y^{d_B} \bmod N \equiv x^{e_B d_B} \equiv x$
signature	:	$x^{d_A} \bmod N$
vérification	:	$(x^{d_A})^{e_A} \equiv x \bmod N$

Fabrication : construire de grands nombres premiers est facile.

Sécurité : casser RSA $\Leftarrow$ factoriser N ; réciproque peut-être fausse (cf. Boneh & Venkatesan).

Planche 3

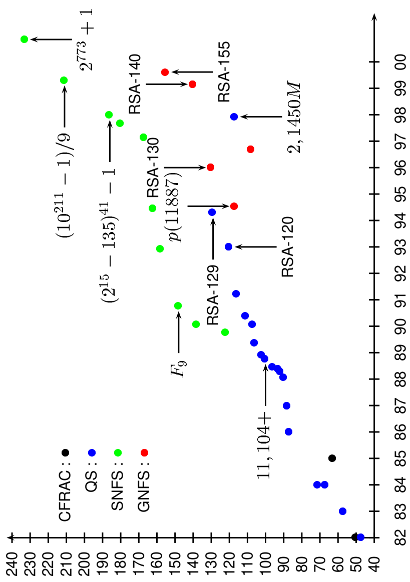

Planche 4

Quel est le problème ?

```
From: xxx@zzz (yyy)
Subject: Factoring public keys attack?
Newsgroups: sci.crypt
Date: 02 Oct 1999 22:12:54 GMT

Instead of trying to factor a prime based
public key after somebody has used it, why
not have a lookup table of all the keys.
It is quicker to create the keys than to
factor a key.

[...]

The government could have just been making
keys for the past 20 years to put on its
lookup table.  Then if you use one of the
keys of the standard lengths, they already
know the primes.
```

Réponse : $\pi(2^{256}) > 6 \times 10^{74}$.

Planche 5

Factorisation et complexité

Thm. La primalité est dans NP ∩ co-NP (et même dans RP, et même dans P sous GRH).

Dém.

co-NP : si N n'est pas premier, il existe un certificat vérifiable en temps polynomial (les facteurs de N).

NP : N possède un certificat de primalité (e.g., il existe a d'ordre $N-1$: $a^{N-1} \equiv 1 \bmod N$ et $a^{(N-1)/p} \not\equiv 1 \bmod N$ pour tout diviseur premier p de $N-1$). □

Thm. La factorisation des entiers est dans NP.

Dém. étant donné un facteur f de N, on vérifie en temps polynomial que $f \mid N$.

Thm. (Shor) La factorisation se fait en temps quantique-polynomial.

Dans la pratique : la primalité se fait rapidement (sommes de Jacobi ou courbes elliptiques, 10 secondes pour 512 bits) ; la factorisation un peu moins.

Planche 6

Les différents type de méthode

Méthodes dépendant de $p \mid N$:

- crible, ρ ;
- Méthodes de type $p-1$:

  a- calculer $g = (a^{k!}-1, N)$ pour a premier avec N. Si $p \mid N$ et $p-1 \mid k!$, alors $g > 1$.

  b- autres groupes : $p+1$ (suites de Lucas) ; formes quadratiques ; ECM (courbes elliptiques), etc.

taille	N facteur de	auteur	date
48	$24^{121}+1$	Brent	08/10/97
49	$2^{1071}+1$	Zimmermann	24/06/98
53	$2^{677}-1$	Curry	14/09/98
54	$(6^{43}-1)^{42}+1$	Lygeros & Mizony	26/12/99

Méthodes générales : crible quadratique, algébrique.

Planche 7

Quelques complexités

$$L_x[\alpha, c] = \exp\left(c(\log x)^{\alpha}(\log\log x)^{1-\alpha}\right)$$

$$L_x[0,c] = (\log x)^c, \quad L_x[1,c] = x^c$$

méthode	temps de calcul
crible	$p = L_p[1,1]$
ρ	$\sqrt{p} = L_p[1,1/2]$
ECM	$L_p[1/2, \sqrt{2}]$
QS	$L_N[1/2, c]$
NFS	$L_N[1/3, c]$

N	$\sqrt{N}$	$L_N[1/2,1]$	$L_N[1/3,1]$
2^{128}	1.84×10^{19}	4.61×10^{8}	1.85×10^{5}
2^{256}	3.40×10^{38}	1.46×10^{13}	2.02×10^{7}
2^{512}	1.16×10^{77}	6.69×10^{19}	1.02×10^{10}
2^{768}	3.94×10^{115}	1.27×10^{25}	9.49×10^{11}
2^{1024}	1.34×10^{154}	4.42×10^{29}	3.82×10^{13}

Planche 8

II. Méthodes à combinaisons de congruences

Kraitchik : trouver x tq $x^2 \equiv 1 \bmod N$, $x \not\equiv \pm 1 \bmod N$, pour lequel $\mathrm{pgcd}(x-1, N)$ donne un facteur de N.

étape 1 : trouver des couples $\{(u_i, v_i)\}_{i \in I}$ tels que

$$u_i^2 \equiv v_i \bmod N, \quad u_i^2 \neq \pm v_i.$$

étape 2 : trouver $J \subset I$,

$$\prod_{j \in J} v_j = V_J^2$$

étape 3 :

$$U_J = \prod_{j \in J} u_j, \quad U_J^2 \equiv V_J^2 \bmod N.$$

étape 4 : $x = U_J / V_J \bmod N$ est une racine carrée de 1 et avec proba $\geqslant 1/2$, elle est non triviale et factorise N.

Planche 9

Comment tester qu'un nombre est un carré ?

$$v_i = \prod_{p \in \mathbb{P}} p^{\alpha(i,p)}$$

$$Z = \prod_{j \in J} v_j = \prod_{p \in \mathbb{P}} p^{\sum_J \alpha(j,p)}$$

$$Z = \square \Leftrightarrow \forall\, p, \sum_J \alpha(j,p) \equiv 0 \bmod 2$$

$\Rightarrow$ problème d'algèbre linéaire : on doit chercher les relations de dépendance de la matrice $\mathcal{M} = (\alpha(i,p) \bmod 2)$.

Pb. $\#\mathbb{P}$ est gigantesque.

Idée : on remplace $\mathbb{P}$ par une base de nombres premiers $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$:

$$v_i = \prod_{r=1}^{k} p_r^{\alpha(i,r)}$$

$$Z = \prod_{j \in J} v_j = \prod_{r=1}^{k} p_r^{\sum_J \alpha(j,r)}$$

Planche 10

Algorithme type

étape 0 : fixer $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$;

étape 1 : trouver des couples $\{(u_i, v_i)\}_{i \in I}$ tels que

$$u_i^2 \equiv v_i \bmod N, \quad u_i^2 \neq \pm v_i$$

et

$$v_i = \prod_{r=1}^{k} p_r^{\alpha(i,r)}.$$

étape 2 : trouver $J \subset I$, $\prod_{j \in J} v_j = V_J^2$ par algèbre linéaire.

étape 3 :

$$U_J = \prod_{j \in J} u_j, \quad U_J^2 \equiv V_J^2 \bmod N.$$

étape 4 : calculer $x = U_J / V_J \bmod N$ et $\mathrm{pgcd}(x-1, N)$.

Planche 11

La méthode de Dixon

On prend $u_i = i$ et $v_i \equiv i^2 \bmod N$.

Ex. $N = 2117$, $\mathcal{B} = \{-1, 2, 3, 5, 7, 11\}$:

rel	i	v_i
1	65	-1×3^2
2	74	$-1 \times 5^3 \times 7$
3	75	$-1 \times 2 \times 3 \times 11^2$
4	79	$-1 \times 2 \times 5 \times 11$
5	81	$2 \times 3 \times 5 \times 7$
6	92	-1×2^2
7	99	$-1 \times 2^4 \times 7^2$

$R_2 \times R_3 \times R_5$ donne :

$$(74 \times 75 \times 81)^2 \equiv (-5^3 \times 7)(-1 \times 2 \times 3 \times 11^2)(2 \times 3 \times 5 \times 7)$$

$$\equiv (2 \times 3 \times 5^2 \times 7 \times 11)^2 \bmod N$$

soit $746^2 \equiv 11550^2$ et $\mathrm{pgcd}(746 - 11550, N) = 73$.

Planche 12

Analyse de la méthode de Dixon

On utilise la fonction de De Bruijn :

$$\psi(x,y) = \#\{n \leqslant x, p \mid n \Rightarrow p \leqslant y\}.$$

Thm. Soit $L(x) = L_x[1/2,1] = \exp\left((\log x)^{1/2}(\log\log x)^{1/2}\right)$. $\forall\, \alpha > 0, \beta > 0$:

$$\psi(x^\alpha, L(x)^\beta) = \frac{x^\alpha}{L(x)^{\frac{\alpha}{2\beta}+o(1)}}.$$

$$\mathrm{Prob}(v_i \bmod N \text{ est } p_k \text{ friable}) = \psi(N,p_k)/N$$

Coût total :

$$C = O(\underbrace{k}_{\text{divisions}}\ \underbrace{(kN/\psi(N,p_k))}_{\#v_i}) + \underbrace{O(k^3)}_{\text{Gauss}}$$

Écrivons $p_k = L^b, k = \pi(p_k) = L^{b+o(1)}$ d'où

$\Rightarrow C = O(L^{2b+1/(2b)}) + O(L^{3b}) = O(L^{\max(2b+1/(2b),3b)})$

Thm. C minimal pour $b = 1/2 \Rightarrow$ coût $L(N)^{2+o(1)}$.

Thm. Si $v_i = O(N^\alpha)$ avec $\alpha \leqslant 1$, on trouve $b = \sqrt{\alpha}/2$ et $L_N[1/2, 2\sqrt{\alpha}]$.

Planche 13

Les différents algorithmes

- CFRAC : (Morrison & Brillhart, 1970) $\alpha = 1/2$
- QS et ses variantes : (Pomerance, Montgomery, Lenstra & Manasse) $\alpha = 1/2$.
- NFS : (Pollard, Lenstra, Buhler) $\alpha = 1/d$ avec d fonction de $N \Rightarrow$ changement dans la complexité.

Prop. Dixon, CFRAC, QS ont complexité $L_N[1/2, c]$; NFS a complexité $L_N[1/3, c]$.

Planche 14

III. Les problèmes d'algèbre linéaire

Rem. : $\mathcal{M}$ est creuse ($\Omega(N) \leqslant \log_2 N$).

Nb	taille	#coeffs $\neq 0$ par ligne
RSA-100	$50,000 \times 50,000$	
RSA-110	$80,000 \times 80,000$	
RSA-120	$252,222 \times 245,810$ $(89,304 \times 89,088)$	
RSA-129	$569,466 \times 524,338$ $(188,614 \times 188,160)$	47
RSA-130	$3,504,823 \times 3,516,502$	39
RSA-140	$4,671,181 \times 4,704,451$	32
RSA-155	$6,699,191 \times 6,711,336$	62

Planche 15

A) Pivot de Gauss

$O(k^3)$ mais avec une faible constante (compactage des bits, machines vectorielles) ;

```
    do i=2, ni
        i1 = (piv-1)*nblocs
        i2 = (tabi(i)-1)*nblocs
CDEC$ INIT_DEP_FWD
        do k=1, nblocs
          M(i2+k) = M(i2+k).xor.M(i1+k)
        enddo
    enddo
```

Variantes prenant en compte les matrices creuses (élimination structurée).

Planche 16

B) Méthodes adaptées aux matrices creuses

– Wiedemann : on cherche le polynôme minimal de $\mathcal{M}$ par l'intermédiaire du polynôme minimal de la suite de bits $e_i = u \cdot (M^i b)$ avec l'algorithme de Berlekamp-Massey en temps $O(k^{2+\varepsilon})$; variante par blocs due à Coppersmith.

– Lanczos : adaptée de la méthode classique utilisée en analyse numérique, utilisée dans un corps fini (!), $O(k^{2+\varepsilon})$; meilleure constante que dans le cas de Wiedemann ; variante par blocs mise au point par P. L. Montgomery permet de trouver 64 relations de dépendance en même temps.

Planche 17

IV. Le crible quadratique et ses variantes

Version de base (Pomerance, 1981) :

$$u_i = i + \left\lfloor \sqrt{N} \right\rfloor, v_i = \left(i + \left\lfloor \sqrt{N} \right\rfloor\right)^2 - N.$$

Avantages :

○ $v_i \approx 2i\sqrt{N} \ll N$;

○ crible :

$$p \mid v_i \Leftrightarrow \left(i + \left\lfloor \sqrt{N} \right\rfloor\right)^2 \equiv N \bmod p$$

implique N carré modulo p et

$$p \mid v_i \Leftrightarrow i \equiv i_- \text{ ou } i \equiv i_+ \bmod p$$

Thm. L'algorithme a un temps de calcul $L_N[1/2, 3/\sqrt{8}]$, la place mémoire est $k = L_N[1/2, 1/\sqrt{8}]$.

Planche 18

Programmation du crible

procedure crible(L)

(* crible sur l'intervalle $[0, L[$ *)

1. $S[i] \leftarrow v_i$ pour $i \in [0, L[$;

2. **pour** $p \in \mathcal{B}$

 pour $i_0 = i_\pm(p)$

  $i \leftarrow i_0$;

  tant que $i < L$

   $S[i] \leftarrow S[i]/p$;

   $i \leftarrow i + p$;

3. **pour** les i tels que $S[i] = 1$, v_i est factorisé.

Rem. ∞ d'astuces pour accélérer (MPQS à la Montgomery, large prime variation, double large prime).

⇒ l'algorithme devient distribuable massivement : email (A. K. Lenstra & M. S. Manasse, 1990), INTERNET (RSA-129).

Planche 19

Large prime variation

$$v_{i_1} \equiv q \left(\prod p_j^{\alpha(i_1,j)}\right),$$
$$v_{i_2} \equiv q \left(\prod p_j^{\alpha(i_2,j)}\right),$$
$$\Rightarrow (u_{i_1} u_{i_2}/q)^2 \equiv \prod p_j^* \bmod N.$$

avantages : $\#\text{match} = O((\#\text{rel})^2)$ (paradoxe des anniversaires) ; accélération d'un facteur 2 à 3 ;

inconvénients : gestion des factorisations ; densification de la matrice $\mathcal{M}$.

Ex. factorisation de $11, 104 + \quad C100$ en 1988 par Lenstra, Manasse (+ email) : $\#\mathcal{B} = 50,000, \#$ relations $= 320,000, fu = 20,500, pa = 29,500.$

Planche 20

Double large prime

$$v_{i_1} \equiv q_1 q_2 \left(\prod p_j^*\right), v_{i_2} \equiv q_2 q_3 \left(\prod p_j^*\right),$$
$$v_{i_3} \equiv q_3 q_1 \left(\prod p_j^*\right),$$
$$\Rightarrow (u_{i_1} u_{i_2} u_{i_3} / q_1 q_2 q_3)^2 \equiv \prod p_j^* \bmod N.$$

Graphe $E = \{q_i\}$, $V = \{(q_i, q_j)\}$; une relation utile correspond à un cycle.

avantages : $\#$cycles $= O((\#\text{rel})^3)$ (?)

inconvénients : gérer un graphe gigantesque (union-find et recherche en largeur d'abord) ; densification.

Ex. $10^{142} + 1$ $C116$ (400 MIPS-années en 1991) : $\#\mathcal{B} = 120,000$ et

- $fu = 25,361, pa = 284,750, pp = 953,242$;
- graphe : $e = 1,237,992, v = 1,286,057$, $cc = 165,485$ d'où $117,420$ cycles ;
- matrice finale : $141,781 \times 120,000$ réduite à $44,971 \times 44,721$.

Planche 21

V. Le crible algébrique (NFS)

Déf. Soit $f(X) = a_d X^d + a_{d-1} X^{d-1} + \cdots a_0$ un polynôme irréductible sur $\mathbb{Q}$. On appelle corps de nombres l'ensemble $\mathbb{Q}[X]/(f(X)) = \{\sum_{i=0}^{d-1} b_i X^i, b_i \in \mathbb{Q}\}$. On le note aussi $\mathbb{Q}(\theta)$ où θ est une racine de f.

Opérations :

$$bc = \left(\sum_{i=0}^{d-1} b_i X^i\right)\left(\sum_{i=0}^{d-1} c_i X^i\right) \equiv \sum_{i=0}^{d-1} e_i X^i \bmod f(X).$$

Ex. $f(X) = X^2 + 1$, $\mathbb{Q}[X]/(X^2+1) = \mathbb{Q}(i)$

$$(b_1 X + b_0)(c_1 X + c_0)$$
$$= b_1 c_1 X^2 + (b_1 c_0 + b_0 c_1) X + b_0 c_0$$
$$\equiv (b_1 c_0 + b_0 c_1) X + b_0 c_0 - b_1 c_1.$$

Planche 22

Prop. Il existe une notion de nombre premier dans $\mathbb{Q}(\theta)$ ainsi qu'une notion de factorisation.

Ex. Dans $\mathbb{Q}(i)$, tous les nombres premiers $p \equiv 3 \bmod 4$ sont premiers, les $p \equiv 1 \bmod 4$ se factorisent en produit de deux nombres premiers, comme $13 = (3+2i)(3-2i)$.

Déf. Soient θ_i les conjugués de θ. On appelle norme du nombre $a - b\theta$ la quantité $\prod_i (a - b\theta_i) = b^d f(a/b)$.

Ex. $f(X) = X^2 + 1$,
$\mathrm{Norm}(a + bi) = (a+bi)(a-bi) = a^2 + b^2$.

Planche 23

L'algorithme

Pollard (1988) : soit $f(X) \in \mathbb{Z}[X]$ et un entier m tel que $f(m) \equiv 0 \bmod N$. Soit θ une racine de f dans $\mathbb{C}$ et $K = \mathbb{Q}[X]/(f(X)) = \mathbb{Q}(\theta)$. Soit ϕ l'homomorphisme d'anneaux :

$$\begin{array}{cccc} \phi : & \mathbb{Z}[\theta] & \to & \mathbb{Z}/N\mathbb{Z} \\ & \theta & \mapsto & m. \end{array}$$

On factorise :

$$(a - b\theta) = \prod_{\pi \in \mathcal{B}_K} \pi^{v_\pi(a-b\theta)}$$

$$\phi(a - b\theta) = a - bm = \prod_{p \in \mathcal{B}} p^{w_p(a-bm)}$$

On cherche alors un ensemble $\mathcal{A}$ de couples (a, b) tq simultanément

$$\prod_{(a,b)\in\mathcal{A}} (a - b\theta) = (A - B\theta)^2, \quad \prod_{(a,b)\in\mathcal{A}} (a - bm) = Z^2$$

d'où

$$\phi((A - B\theta)^2) \equiv (A - Bm)^2 \equiv Z^2 \bmod N$$

Planche 24

Choix de f : le plus simple est de choisir d et d'écrire N en base d : $m = \lfloor N^{1/d} \rfloor$,
$N = m^d + a_{d-1}m^{d-1} + \cdots + a_0$, d'où
$f(X) = X^d + a_{d-1}X^{d-1} + \cdots + a_0$.

Thm. Si $f(X) = X^d + \cdots$, avec

$$d = K(\log N)^{1/3}(\log \log N)^{2/3}$$

le temps de calcul de NFS est : $L_N[1/3, c]$.

Planche 25

Comment factoriser dans $\mathbb{Q}(\theta)$?

On factorise la norme de $a - b\theta$:

$$\mathrm{Norm}(a - b\theta) = \pm \prod_p p^{\nu_p(a,b)}$$

qui conduit à :

$$(a - b\theta) = \prod_p [p, r]^{\nu_p(a,b)}$$

où $[p, r]$ est un idéal de $\mathbb{Z}[\theta]$ tq $f(r) \equiv 0 \bmod p$.

Rem. bien sûr, on crible sur a et b (crible linéaire ou par réseau).

Comment trouver un carré dans $\mathbb{Q}(\theta)$? Si z est un carré dans $\mathbb{Q}(\theta)$, alors $\mathrm{Norm}(z)$ est un carré dans $\mathbb{Q}$. Pour rendre cette condition suffisante, on utilise des caractères (à la Adleman).

Planche 26

Exemple (simple)

$N = 2117 = 46^2 + 1$, $m = 46$, $f(X) = X^2 + 1$. La base est constituée de $[2, 1] = 1 + i$, $[5, 2] = 1 + 2i$, $[13, 8] = 2 + 3i$ et des nombres premiers plus petits que 13. On crible pour $(a, b) \in [-50, 50] \times [1, 6]$.

$-18 - \theta$ a norme $325 = 5^2 \times 13$ et on détermine que : $-18 - \theta = [5, 2]^2 \times [13, 5]$ car $-18 - 2 \equiv 0 \bmod 5$ et $-18 - 5 \equiv 0 \bmod 13$.

D'où une relation avec $R_3 \times R_6 \times R_8 \times R_{13}$:

$$\phi(-2368 - 2226\theta) \equiv 11088^2 \equiv 503^2 \bmod N.$$

Or $-2368 - 2226\theta = (21 - 53\theta)^2$ d'où

$$(\phi(21 - 53\theta))^2 \equiv 503^2 \bmod 2117$$

et $\mathrm{pgcd}((21 - 53 \times 46) - 503, 2117) = 73$.

Planche 27

rel	(a,b)	alg	rat
1	$(-18,1)$	$[5,2]^2 \times [13,8]$	$2^2 \times 7$
2	$(-7,1)$	$[2,1] \times [5,3]^2$	3×13
3	$(-2,1)$	$[5,3]$	$2^2 \times 11$
4	$(-1,1)$	$[2,1]$	$3^2 \times 5$
5	$(2,1)$	$[5,2]$	$2^4 \times 3$
6	$(3,1)$	$[2,1] \times [5,3]$	7^2
7	$(8,1)$	$[5,3] \times [13,8]$	2×3^3
8	$(18,1)$	$[5,3]^2 \times [13,5]$	2^6
9	$(-29,2)$	$[5,3] \times [13,5]^2$	$3^2 \times 7$
10	$(2,3)$	$[13,5]$	$2^2 \times 5 \times 7$
11	$(1,5)$	$[2,1] \times [13,8]$	$3 \times 7 \times 11$
12	$(-2,11)$	$[5,3]^3$	$2^3 \times 3^2 \times 7$
13	$(17,19)$	$[2,1] \times [5,3]^2 \times [13,5]$	$3^4 \times 11$
14	$(-38,41)$	$[5,2]^5$	$2^3 \times 3 \times 7 \times 11$

Planche 28

Dans la réalité

– La taille des coefficients de f influe sur l'algorithme : SNFS : permet d'attaquer $b^n \pm 1$ facilement ; GNFS : tous les nombres.

– Problèmes :

  – factorisation dans les corps de nombre (euclidéanité, contourné par Adleman) ;

  – calcul de la racine carrée (Couveignes, Montgomery, Nguyen) ;

  – choix du polynôme (résultant de deux polynômes, choix de f par échantillonage astucieux – beaucoup de racines modulo les p_i^e, idées de Murphy & Montgomery).

– Supplante PPMPQS dans la zone 120dd–130dd.

– Nouvelles idées : utiliser deux large primes de chaque côté (4 large primes ! !) $\Rightarrow$ provoque une explosion combinatoire accélérant le crible, mais pourquoi ?

Planche 29

VI. Quelques records

```
http://www.rsa.com/rsalabs/html/factoring.html
           RSA Honor Roll
           --------------
```

dd	qui	avec	date	temps
100	Manasse & A. K. Lenstra	1991	MPQS	7 MIPS-années
110	AKL	1992	MPQS	un mois sur $5/8$ d'une MasPar 16K
120	AKL, Dodson, Denny, Manasse, Lioen, te Riele	1993	PPMPQS	835 MIPS-années
129	Atkins, Graff, AKL, Leyland + INTERNET	1994	PPMPQS	5000 MIPS-années
130	Dodson, Montgomery, Elkenbracht-Huizing, AKL, WWW, Fante, Leyland, Weber, Zayer	1996	NFS	500 MIPS-années
140	te Riele, Cavallar, Lioen, Montgomery, Dodson, AKL, Leyland, Murphy, Zimmermann	1999	NFS	1500 MIPS-années
155	CABAL	1999	NFS	8000 MIPS-années

Planche 30


```
RSA-155 =
1094173864157052742180970732204\
0357612003732945449205990913842\
1314763499842889347847179972578\
9126733249762575289978183379707\
6537244027146743531593354333897
=
                1026395928297411\
0577205419657399167590071656780\
8038066803341933521790711307779
*
                1066034883801684\
5482092722036001287867920795857\
5989291522270608237193062808643
```

Planche 31

RSA-140 vs. RSA-155

	RSA-140	RSA-155
base rat.	$250,000$ ($p_i \leqslant 3,497,867$)	$2,661,384$ ($p_i \leqslant 44,000,000$)
base alg.	$800,000$ ($\mathrm{Norm} \leqslant 12,174,433$)	$6,304,167$ ($\mathrm{Norm} \leqslant 110,000,000$)
large prime	rat : $500,000,000$ alg : $1,000,000,000$	$1,000,000,000$
# rel	$66,933,395$ ($10,327,897$ doubles)	$130.8M$ ($6.1M$ doubles)
date	$\leqslant$ noël 1998 1 mois	fin avril 1999 3.7 mois
matrice	$4,671,181 \times 4,704,451$	$6,699,191 \times 6,711,336$
alg. lin.	810 Mo, 100 heures Cray C916 (SARA)	2 Go, 224 heures Cray C916 (SARA)

Planche 32

Planche 33

Comparison of 2,773+ with 10,211-, RSA155, and RSA140 :

	2,773+	10,211-	RSA155	RSA140
Digits	227(C)	211	155	140
Algorithm	SNFS	SNFS	GNFS	GNFS
Completion	Nov 00	Apr 99	Aug 99	Feb 99
Sieving CPU years	57.4	10.9	35.7	8.9
Sieving calendar weeks	22	8	15	4
Line-siever relations	36.4M	23.5M	36.0M	30.0M
Lattice-siever relations	83.2M	32.9M	94.8M	38.0M
Factor base bound	2*10^7	2^24	see rept.	see rept.
Large prime bound for f	10^9	5*10^8	10^9	10^9
for g	10^9	6*10^8	10^9	5*10^8
Cray matrix size	6.8M	4.8M	6.7M	4.7M
matrix weight in 1's per row	65	49	62	32
Cray linear algebra CPU hours	250	121	224	100

Planche 34

Les équipes

20.1 % (3057 CPU days) Alec Muffett (Sun Microsystems Professional Services, UK)
17.5 % (2092 CPU days) Paul Leyland (Microsoft, Cambridge, UK)
14.6 % (1819) Peter L. Montgomery, Stefania Cavallar (CWI)
13.6 % (2222) Bruce Dodson (Lehigh University, Bethlehem, PA, USA)

13.0 % (1801) Francois Morain and Gerard Guillerm (Ecole Polytechnique)
6.4 % (576) Joel Marchand (Ecole Polytechnique/CNRS)

5.0 % (737) Arjen K. Lenstra (Citibank, USA)
4.5 % (252) Paul Zimmermann (Loria)
4.0 % (366) Jeff Gilchrist (Entrust Technologies Ltd., Ottawa, Canada)
0.65 % (62) Karen Aardal (Utrecht University, The Netherlands)
0.56 % (47) Chris and Craig Putnam

Planche 35

Pour RSA-155

Le polynôme trouvé après 100 MIPSY :

$$\begin{aligned} F_1(x,y) = & \quad 11\,93771\,38320 \; x^5 \\ & -80\,16893\,72849\,97582 \; x^4y \\ & -66269\,85223\,41185\,74445 \; x^3y^2 \\ & +1\,18168\,48430\,07952\,18803\,56852 \; x^2y^3 \\ & +745\,96615\,80071\,78644\,39197\,43056 \; x\,y^4 \\ & -40\,67984\,35423\,62159\,36191\,37084\,05064 \; y^5 \\ m = & \quad 3912\,30797\,21168\,00077\,13134\,49081 \end{aligned}$$

Les coefficients sont de taille 10^{23} au lieu de 10^{25}.

Les valeurs de F_1 ont des proba de friabilité équivalentes à celles de nombres plus petits d'un facteur 800.

Planche 36

2. Algèbre linéaire

Fabrication de la matrice : variante de l'élimination structurée appelée filtrage.

Matrice finale : $(M_0|M_1)$ avec M_0 matrice $n \times 100$ correspondant aux caractères et aux nombres premiers < 40 ; M_1 une matrice $6,699,191 \times 6,711,336$.

– calcul de 64 relations de dépendance dans M_1, soit $(r_i)_{1 \leqslant i \leqslant 64}$;

– résolution de

$$(l_1, l_2, \ldots, l_{64}) \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{64} \end{pmatrix} M_0 = 0$$

soit

$$(l_1, l_2, \ldots, l_{64}) M_0' = 0$$

où M_0' est de taille 64×100.

VII. Pour finir

A) Jouons à Madame Soleil

It is unwise to make predictions about the difficulty of factoring

Retour aux complexités :

$T(N)$	$N \mapsto N^2$
$\sqrt{N}$	T^2
$L_N[1/2]$	$T^{\sqrt{2}}$
$L_N[1/3]$	$T^{\sqrt[3]{2}}$

Ex. $N = 2^{512}$, $T(N) = 8000$ MIPSY, $T(2^{1024}) = 82715$ MIPSY, mais avec une matrice $(3 \times 10^8)^2$ (faisable en 2018 (Brent) ?).

Et la loi de Joy ? les ordinateurs doublent de vitesse tous les deux ans. Dans deux ans, on peut savoir faire 2^{512+35} dans le même temps.

Planche 37

B) Conclusions

– La factorisation est un défi mathématique (5 idées en 2000 ans, dont 3 en 20 ans) et bien souvent technologique.

– La programmation est délicate et doit être optimisée (machines, etc.).

– L'algèbre linéaire est souvent le facteur limitant.

– On n'est pas débarrassé de RSA pour le moment.

Planche 38