Factorization of a 768-bit RSA modulus

Paul Zimmermann
(joint work with T. Kleinjung. K. Aoki, J. Franke, A. Lenstra, E. Thomé, J. Bos, P. Gaudry, A. Kruppa, P. Montgomery, D. A. Osvik, H. te Riele and A. Timofeev)

CARAMEL

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
INRIA
centre de recherche NANCY - GRAND-EST

Workshop on Tools for Cryptanalysis, 23 June 2010


Ninth Algorithmic Number Theory Symposium
ANTS IX
Nancy, France
July 19-23, 2010
http://ants9.org/
LOCAL ORGANIZATION
Anne-Lise Charbonnier
Jérémie Detrey
Pierrick Gaudry
Emmanuel Thomé
Paul Zimmermann
INVITED SPEAKERS
Henri Darmon, McGill University
Fritz Grunewald, HHU Düsseldorf
Jean-François Mestre, Univ. Paris 7
Gabriele Nebe, RWTH Aachen
Carl Pomerance, Dartmouth College
Oded Regev, Tel-Aviv University
PROGRAM COMMITTEE
Nigel Boston
John Cremona
Claus Fieker
Guillaume Hanrot, chair
Kevin Hare
Thorsten Kleinjung
Kamal Khuri-Makdisi
François Morain, chair
Takakazu Satoh
Igor Shparlinski
Alice Silverberg
Frederik Vercauteren
Loria
INRIA
cnrs
Nancy-Université

Summary of the talk

On December 12, 2009 we factored RSA-768.

"Wall-clock time" 2+ years.

Total about 1700 (single core) cpu years.

13 times more than breaking Trivium (cf Paul Stankovski's talk).

10 times less than (expected time) to break ECC2K-130 (cf Junfeng Fan's talk).

The RSA Factoring Challenge

Started in last millenium (1991), ended in 2007.

Encourage research in integer factoring.

Give an idea of which key size are still safe, and for how long.

First series (decimal): **RSA-100**, ..., **RSA-200**, ..., RSA-500

Second series (binary): **RSA-576, 640**, 704, **768**, 896, 1024, 1536, 2048.

(50,000USD were offered for **RSA-768**, 200,000USD for RSA-2048.)

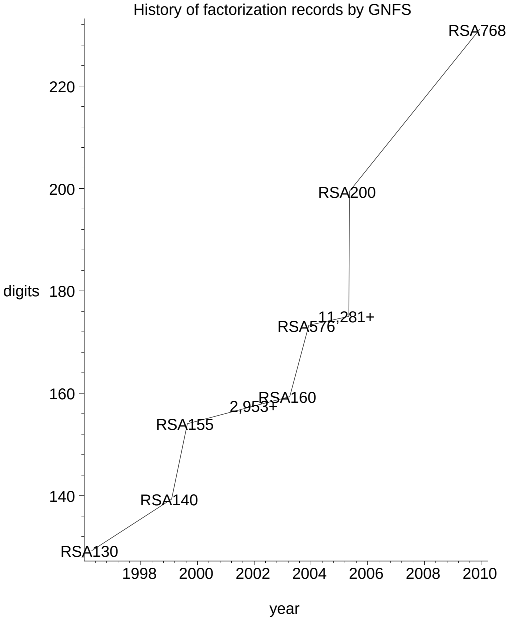
The Number Field Sieve (NFS)

Invented by Pollard in 1988

Algorithm of choice to factor RSA numbers $n = pq$, with p and q of the same size

Complexity $e^{c(\log n)^{1/3}(\log\log n)^{2/3}}$

Previous record: RSA-200, 200 digits (Bahr, Böhm, Franke, Kleinjung, 2005).


History of factorization records by GNFS
RSA768
220
200
RSA200
digits
180
11,281+
RSA576
160
RSA160
2,953+
RSA155
140
RSA140
RSA130
1998
2000
2002
2004
2006
2008
2010
year

The Number Field Sieve

- polynomial selection (40 cpu years)
- sieving (1500 cpu years)
- filtering (duplicates, singletons, cliques)
- merging
- linear algebra (155 cpu years)
- characters
- square root
- gcd

Tiny NFS factorization

Let us factor $n = 5105929$ by NFS.

Polynomial selection

$$F(x,y) = 173x^2 - 70xy - 63y^2, \quad G(x,y) = x - 172y$$

$f(x) = F(x,1)$ and $g(x) = G(x,1)$ have a common root $\mu = 172 \bmod n$

$$\mathrm{Res}(f(x), g(x)) = 5105929$$

Notations

All record NFS factorizations so far used a linear polynomial $g(x)$.

The $f(x)$ side is called the *algebraic side*. Let d be the degree of $f(x)$.

The $g(x)$ side is called the *rational side*.

Sieving for $n = 5105929$

$$F(x,y) = 173x^2 - 70xy - 63y^2, \quad G(x,y) = x - 172y$$

Find $F(a,b)$ and $G(a,b)$ smooth enough for a, b coprime

a,b	F(a,b)	G(a,b)
-573, 1213	$2^2 \cdot 3 \cdot 5^2 \cdot 23 \cdot 43^2$	$-1 \cdot 7 \cdot 11^2 \cdot 13 \cdot 19$
-108, 247	$3^2 \cdot 5^3 \cdot 37$	$-1 \cdot 2^5 \cdot 11^3$
-19, 39	$2^2 \cdot 5^3 \cdot 37$	$-1 \cdot 7 \cdot 31^2$
-9, 4	$3^3 \cdot 5^2 \cdot 23$	$-1 \cdot 17 \cdot 41$
7, 8	$3 \cdot 5^2 \cdot 7$	$-1 \cdot 37^2$
7, 12	$-1 \cdot 5^2 \cdot 7 \cdot 37$	$-1 \cdot 11^2 \cdot 17$
108, 127	$3^2 \cdot 5^3 \cdot 37$	$-1 \cdot 2^3 \cdot 11 \cdot 13 \cdot 19$
419, 529	$-1 \cdot 2^2 \cdot 3 \cdot 5^3 \cdot 43^2$	$-1 \cdot 41 \cdot 47^2$

Concept of ideal

Rational side: p divides $G(a, b) = bg(a/b)$ when a/b is a root of $g(x)$ mod p. Exactly one root for each p.

Algebraic side: p divides $F(a, b) = b^d f(a/b)$ when a/b is root of $f(x)$ mod p.

$f(x)$ might have from 0 to d roots mod p. Let r be such a root, we denote (p, r) the corresponding ideal to identify it uniquely.

Filtering

1. keep only one copy of duplicate relations (same a, b)
2. delete singletons (ideal (p, r) occurring in only one relation)
3. repeat Step 2 until no singleton remains
4. if (many) more relations than ideals, delete "cliques"
5. merge relations with common ideal (p, r), if this ideal occurs a few times

Linear algebra

Build a sparse matrix containing for each relation, the exponents of ideals occurring in that relation, reduced mod 2

$>$ *m* relations for *m* ideals $\longrightarrow$ a linear dependency exists

Lanczos and Wiedemann black-box algorithms: perform only matrix-vectors multiplications Mx, where M is the initial matrix. "Block" versions.

In our tiny example, just multiply the 8 relations:

$$\prod F(a,b) = (-1)^2 \cdot 2^6 \cdot 3^{10} \cdot 5^{20} \cdot 7^2 \cdot 23^2 \cdot 37^4 \cdot 43^4$$

$$\prod G(a,b) = (-1)^8 \cdot 2^8 \cdot 7^2 \cdot 11^8 \cdot 13^2 \cdot 17^2 \cdot 19^2 \cdot 31^2 \cdot 37^2 \cdot 41^2 \cdot 47^2$$

Square root

For each dependency (usually we find $20-30$):

Rational side: multiply together $a-\mu b$ for each (a,b) in the dependency, μ being the common root of $f(x)$ and $g(x)$ mod n:

$$\prod_{(a,b)\in S} a-\mu b = u^2 \quad \text{with } u = 15218777599577552.$$

Algebraic side: multiply together $a-xb$ mod $f(x)$:

$$\prod_{(a,b)\in S} a-xb = \frac{20009231592883459891453125000}{173^8}x + \frac{66414502509679015109578125000}{173^8} \bmod f,$$

whose square root mod f is:

$$v(x) = \frac{1}{173^3}(-759208295625x + 109567198125).$$

Square root (cont'd)

$$n = 5105929$$

$$u = 15218777599577552 \equiv 701937 \bmod n$$

$$v(x) = \frac{1}{173^3}(-759208295625x + 109567198125).$$

which gives:

$$v(\mu) = 4220991 \bmod n.$$

$$\gcd(u + v(\mu), n) = \gcd(701937 + 4220991, 5105929) = 2011$$

Factorisation of RSA-768

NTT: Kazumaro Aoki

EPFL: Joppe Bos, Thorsten Kleinjung, Arjen Lenstra, Dag Arne Osvik

Bonn: Jens Franke

CWI: Peter Montgomery, Herman te Riele, Andrey Timofeev

INRIA: Pierrick Gaudry, Alexander Kruppa, Emmanuel Thomé, PZ

Polynomial selection

Total cpu time: about 40 cpu-years (about 2% of total time).

$$\begin{aligned}
f(x) &= 265482057982680\, x^6 \\
&+ 1276509360768321888\, x^5 \\
&- 5006815697800138351796828\, x^4 \\
&- 46477854471727854271772677450\, x^3 \\
&+ 6525437261935989397109667371894785\, x^2 \\
&- 18185779352088594356726018862434803054\, x \\
&- 277565266791543881995216199713801103343120,
\end{aligned}$$

$$\begin{aligned}
g(x) &= 34661003550492501851445829\, x \\
&- 1291187456580021223163547791574810881.
\end{aligned}$$

$$\mathrm{Res}(f(x), g(x)) = \mathrm{RSA768}$$

Polynomial selection

We used Kleinjung's 2006 algorithm (*On polynomial selection for the general number field sieve*, Mathematics of Computation).

$$g_1 = 13 \cdot 37 \cdot 79 \cdot 97 \cdot 103 \cdot 331 \cdot 601 \cdot 619 \cdot 769 \cdot 907 \cdot 1063$$

L2-norm of $f(x)$ is about $2.4 \cdot 10^{28}$ (log 65.35)

Classical Murphy base-m selection with $m \approx N^{1/7}$:

$$\begin{aligned}
f(x) &= 1030037421788922756904983440625677\, x^6\\
&+ 443133715614801195536692200584752\, x^5\\
&+ 496634096511982595472923847619463\, x^4\\
&- 281287624344966232333638462668051\, x^3\\
&- 321240386490830757300657164979031\, x^2\\
&+ 599011620087225409088255778576617\, x\\
&+ 444014458592954914308588024454438,
\end{aligned}$$

$$g(x) = x - 1030037421788922756904983440625675$$

L2-norm: $6.12 \cdot 10^{32}$ (log 75.49)

Root properties

The α-value of $f(x)$ is about -7.3.

Compared to a “random” polynomial, once we divide by all primes less than 2000, the remaining cofactor is smaller by about $\exp(-\alpha) \approx 1500$.

Altogether, we saved a factor about 25000 with the small norm, and about 1500 with the small α, thus about $37 \cdot 10^6$ in total!

Sieving

We used only *lattice sieving*, with *special-q* between 110M and 11100M.

Trivially parallel (split special-q range)

Total 64G relations (5Tb compressed), 1500 cpu-years.

INRIA 38%, EPFL 30%, NTT 15%, Bonn 8%, CWI 3%

A (q, ρ) pair produced on average 134 relations.

On average 4 relations every 3 seconds.

Sieving by vectors

Reference: *Continued Fractions and Lattice Sieving*, Jens Franke and Thorsten Kleinjung, SHARCS 2005.

One of the 64G relations

$F(104262663807, 271220)$ has 81 digits:

301114673492631466171967912486669486315616012885653409138028100146264068435983640

$2^3 \cdot 3^2 \cdot 5 \cdot 1429 \cdot 51827 \cdot 211373 \cdot 46625959 \cdot 51507481$
$\cdot 3418293469 \cdot 4159253327 \cdot 10999998887 \cdot 11744488037 \cdot 12112730947$

$G(104262663807, 271220)$ (42 digits):

$-350192248125072957913347620409394307733817$

$-1 \cdot 11 \cdot 1109 \cdot 93893 \cdot 787123 \cdot 9478097 \cdot 2934172201 \cdot 13966890601$

Sieving by vectors: the q-lattice

Consider a special-q, say $q = 10999998887$, and a root ρ of $f(x)$ mod q, say $\rho = 4941866850$.

Pairs (a, b) such that $F(a, b) = 0$ mod q reduce to $a/b = \rho$ mod q.

Lattice generated by $(q, 0)$ and $(\rho, 1)$ (cf talk of Marc Joye):

$$\begin{pmatrix} a \\ b \end{pmatrix} = u \begin{pmatrix} q \\ 0 \end{pmatrix} + v \begin{pmatrix} \rho \\ 1 \end{pmatrix}$$

Sieving by vectors: the q-lattice

We reduce the skew lattice (s is the skewness):

$$\begin{pmatrix} q & \rho \\ 0 & s \end{pmatrix}$$

On our example, with $s = 44205$, we find the reduced lattice:

$$\begin{pmatrix} -11152847 & 6513125 \\ 69\,s & 946\,s \end{pmatrix}$$

Thus the lattice of (a, b) pairs such that q divides $F(a, b)$ is:

$$\binom{a}{b} = i \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + j \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} := i \begin{pmatrix} -11152847 \\ 69 \end{pmatrix} + j \begin{pmatrix} 6513125 \\ 946 \end{pmatrix}$$

Sieving by vectors

Recall we need $\gcd(a, b) = 1$. How does it relate to $\gcd(i, j)$?

$\gcd(a, b)$ divides $q \cdot \gcd(i, j)$: $\gcd(i, j) = 1 \Rightarrow \gcd(a, b) = 1$ or q.

We can rewrite $F(a, b)/q$ as $F'(i, j)$:

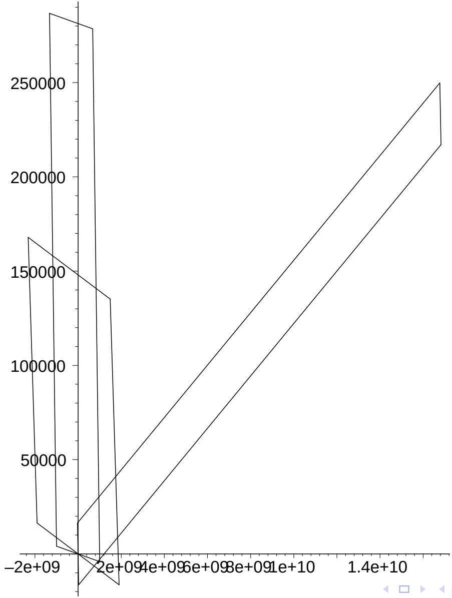
$$
\begin{aligned}
F'(i,j) &= 1516130325658467600476714461447852711499772547\,i^6 \\
&- 851354685226959013191256908958430909537139447150\,i^5 j \\
&- 1160865422044679275000842279069374339393281708219\,i^4 j^2 \\
&+ 31401533065035430877729182304741709354795306270110\,i^3 j^3 \\
&+ 23170460860723950600905984337555229255316340470000\,i^2 j^4 \\
&- 50317384600166516120833406579130302039250187402616\,i j^5 \\
&- 7807280760059800102689268443930905909245906201960\,j^6
\end{aligned}
$$

Now we sieve over a square region (in fact $0 \le |i|, j \le I/2$).

For RSA-768 we used $I = 2^{16}$, i.e., 2^{31} points per (q, ρ) pair.

We sieved over 480M (q, ρ) pairs, i.e., over about 10^{18} pairs (a, b) (among which about 60% coprime).

Different (q, ρ) pairs give different regions in the (a, b) plane:



Sieving by vectors: the p-sublattice

For (q, ρ) fixed, we want to find (i, j) coprime such that $F'(i, j)$ is smooth.

For a prime p, which locations $-I/2 \leq i < I/2, 0 < j < I/2$ are divisible by p?

- small p: use "line sieving": p divides at $i = i_0(j) + \lambda p$
- large p ($p \geq I$): there is 0 or 1 hit per line. Initialization cost (computing $i_0(j)$) dominates.

Lemma (Franke, Kleinjung, 2005)

If $p \geq I$ divides at location (i, j), the next location is given by:

$$(i', j') = (i, j) + \begin{cases} (\alpha, \beta) & \text{if } i + \alpha \geq -I/2 \\ (\gamma, \delta) & \text{if } i + \gamma < I/2 \\ (\alpha, \beta) + (\gamma, \delta) & \text{if } i + \alpha < -I/2 \text{ and } I/2 \leq i + \gamma \end{cases}$$

where

$$\beta, \delta > 0, \quad -I < \alpha \leq 0 \leq \gamma < I, \quad \gamma - \alpha \geq I.$$

Consider for example $p = 46625959 \approx 711l$.

f has two roots modulo p: 41898922 and 38600568.

Consider the root $R = 41898922$.

We need to convert it to the (i, j) plane:

$$r = -\frac{a_1 - Rb_1}{a_0 - Rb_0}$$

This gives $r = 25345641$.

From $(p, r) = (46625959, 25345641)$ we find $(\alpha, \beta, \gamma, \delta)$ using Proposition 1 from the paper by Franke and Kleinjung (which amounts to computing a subtractive Euclidean sequence starting from p and r, and stopping as soon as the last two remainders are smaller than l).

$$
\begin{aligned}
(-p, r) &= (-46625959, 25345641) \to (-21280318, 25345641) \\
&\to (-21280318, 4065323) \to (-17214995, 4065323) \\
&\to (-13149672, 4065323) \to (-9084349, 4065323) \\
&\to (-5019026, 4065323) \to (-953703, 4065323) \\
&\to (-953703, 3111620) \to (-953703, 2157917) \\
&\to (-953703, 1204214) \to (-953703, 250511) \\
&\to (-703192, 250511) \to (-452681, 250511) \\
&\to (-202170, 250511) \to (-202170, 48341) \\
&\to (-153829, 48341) \to (-105488, 48341) \\
&\to (-57147, 48341) = (\alpha, \gamma)
\end{aligned}
$$

$$\alpha = -57147, \beta = 734, \gamma = 48341, \delta = 195$$

We start from $(i_0, j_0) = (0, 0)$.

Since $i + \alpha = -57147 < -I/2 = 32768$ and $i + \gamma = 48341 \geq I/2$, the next value is $(i_1, j_1) = (\alpha + \gamma, \beta + \delta) = (-8806, 929)$.

And indeed $F'(-8806, 929)$ is divisible by $p = 46625959$.

```
sage: for jj in range(1000):
....:     for ii in range(-2^15,2^15):
....:         if Fij(i=ii,j=jj) % 46625959 == 0:
....:             print ii, jj
....:             break
....:
0 0
-8806 929
```

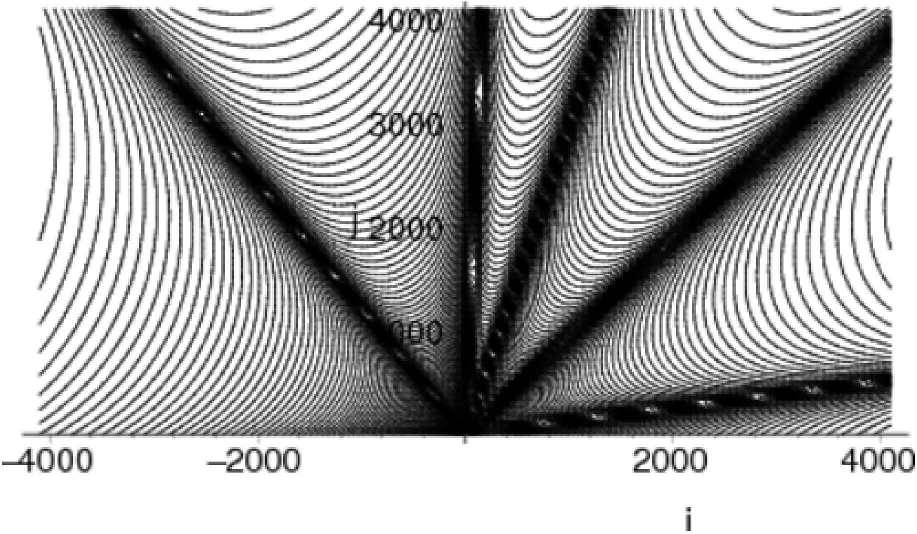
$$(0,0) \underbrace{\longrightarrow}_{\text{rule 3}} (-8806, 929) \underbrace{\longrightarrow}_{\text{rule 3}} (-17612, 1858) \underbrace{\longrightarrow}_{\text{rule 2}} (30729, 2053)$$

$$\underbrace{\longrightarrow}_{\text{rule 1}} (-26418, 2787) \underbrace{\longrightarrow}_{\text{rule 2}} (21923, 2982) \underbrace{\longrightarrow}_{\text{rule 3}} (13117, 3911)$$

$$\underbrace{\longrightarrow}_{\text{rule 3}} (4311, 4840) \underbrace{\longrightarrow}_{\text{rule 3}} (-4495, 5769) \longrightarrow \cdots$$

For $i = -8806$, $j = 929$, we get $a = a_0 i + a_1 j = 104262663807$, $b = b_0 i + b_1 j = 271220$, and the previous relation.

Impact of real roots (here RSA-512)



Sieving: technical details

Factor base bounds: 200M (rational side) and 1100M (algebraic side) on computer with 2Gb of RAM, otherwise 100M and 450M.

Large prime bounds: 2^{40} on both sides.

Cofactor bounds: 100-110 bits on rational side, 130-140 bits on algebraic side.

$\Longrightarrow$ up to 4 *large primes* in addition to *special-q*.

$F(104262663807, 271220)$ has 81 digits:

301114673492631466171967912486669486315616012885653409138028100146264068435983640

$2^3 \cdot 3^2 \cdot 5 \cdot 1429 \cdot 51827 \cdot 211373 \cdot 46625959 \cdot 51507481$
$\cdot 3418293469 \cdot 4159253327 \cdot 10999998887 \cdot 11744488037 \cdot 12112730947$

$G(104262663807, 271220)$ (42 digits):

$-350192248125072957913347620409394307733817$

$-1 \cdot 11 \cdot 1109 \cdot 93893 \cdot 787123 \cdot 9478097 \cdot 2934172201 \cdot 13966890601$

Cofactorization

After removing all factor base primes for
$a = 104262663807, b = 271220$:

$$F(a,b)/q \rightarrow 2022557505660967144847336742419973936157$$

$$G(a,b) \rightarrow 40981262135862382801$$

1. decide whether $F(a,b)/q$ **and** $G(a,b)$ are 2^{40}-smooth
2. if so, find the corresponding factors

No need to be 100% correct: early abort strategy.

Algorithms of choice: P-1, P+1, ECM, MPQS.

Filtering

Duplicates: 27.4% (about 10 days).

Definition. Excess = # relations − # ideals.

Remains 48G relations for 35G ideals (excess 13G).

After one “singleton” pass: remains 29G relations for 14G ideals (excess 15G).

After several “singleton” passes: 25G relations for 10G ideals (excess 15G).

Clique removal: 2.5G relations for 1.7G ideals (excess 0.8G).

Total 10 days for singleton- and clique-removal.

Merge

Assume we have an initial matrix of dimension d.

Linear algebra cost: $\approx d$ matrix-vector products.

If total matrix weight is w, linear algebra cost depends on dw.

As long as we decrease dw, we can modify the matrix.

Example: 2-merge. If a prime p (resp. an ideal p, r) appears exactly two times, we can replace the corresponding two relations by one.

$$r_i = p_{17} p_{42} p_{83}, \quad r_j = p_7 p_{11} p_{42} p_{99} \Longrightarrow r_i r_j = p_7 p_{11} p_{17} p_{83} p_{99}$$

One less relation, one less ideal, excess unchanged.

d decreases by 1, w decreases by (at least) 2.

Merge: beginning of a Gaussian elimination.

We can do the same when a prime (resp. ideal) appears exactly 3 times: merge the three relations to obtain only two.

$$r_i = p_{17} p_{42} p_{83}, \quad r_j = p_7 p_{11} p_{42} p_{99}, \quad r_k = p_5 p_{42} p_{51} p_{52} p_{53}$$

$$r_i r_k = p_5 p_{17} p_{51} p_{52} p_{53} p_{83}, \quad r_j r_k = p_5 p_7 p_{11} p_{51} p_{52} p_{53} p_{99}$$

Here d decreases by 1, but w increases by 1.

At the end of the merge process: matrix of 193M rows/columns with 144 non-zero elements per row (105Gb).

2^{34}

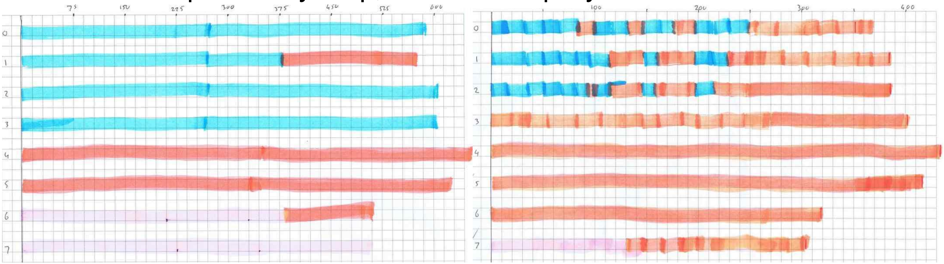
With relations having only ideals $< 2^{34}$ (instead of 2^{40}), we had enough relations to complete the factorization.

This represents 2% of relations, about 100Gb only.

$\Longrightarrow$ matrix of $253M$ rows/columns with 147 non-zero elements per row.

However: linear algebra would have been more difficult.

We used the block Wiedemann algorithm, using 8 sequences in parallel.
Distributed computation between 3 “sites”: INRIA (blue), EPFL (orange) and NTT (pink).
Wall clock time of 119 days, including 17h for the Berlekamp-Massey step, about 155 cpu-years.



(Cf appendices B and E of the Crypto paper.)

Polynomial selection: Kleinjung 2008

Polynomial for RSA-768 found with CADO-NFS and msieve (J. Papadopoulos):

```
# norm 4.241918e-17 alpha -10.618674 e 4.326e-17 rroots 6
skew: 43219804.59
c0: -9538710351534297785929225273946092272863206970263l375
c1: -578246376783790448892047169792689836468010442O
c2:  32970259507636618884032016006025136056O9
c3: -37344650766830623857836813239461
c4: -647269506273086360484257O
c5: -17661930714618 3
c6:  21420000
Y0: -196422812801077330306834753201455359 94
Y1:  26077104631367
```

On a 2.83Ghz Core 2: 1.61 rel/sec (79% of the yield for the polynomial we used)

Was degree 6 optimal?

After a very limited search, we found the following degree-5 polynomial with Kleinjung's 2008 algorithm, with yield 0.91rel/sec (45% of the yield for the polynomial we used).

```
skew: 24796530.127
c5: 1649100
c4: 538215767802882789 1
c3: -27060957333063249379828846817 02
c2: -174374144862973725603010705072173062 7
c1: 906790115489398511265521204319982027499477901
c0: -283930489081575447500491838905199102738262115268940
Y1: 755592461363 9
Y0: -94307190292641194876155994530912726708510146 7
```

Naive square root

Rational side: accumulate the product of $a - b\mu$, and take its square root.

339,965,199 (a, b) pairs

Product side: 47.966.524.207 bits (6Gb)

With GMP 5.0.0 and FFT patch (Kruppa, Gaudry, PZ) on a 32Gb computer:

Accumulation: 2 hours.

Square root: 30 minutes.

Naive square root: Algebraic side

1. accumulate the product of $a - bx$ while reducing mod $f(x)$ in $\mathbb{Q}[x]$
2. choose an “inert” prime p
3. compute the square root mod p and lift mod p^k

Works well (this is what we use in CADO-NFS) but would require a 64Gb machine (at least) for RSA-768.

Naive square root: Algebraic side

Memory-cheap algorithm designed by E. Thomé, and implemented in CADO-NFS.

Idea: reduce modulo several p_i and reconstruct via CRT.

RSA-768: *wall clock time* 6h 30min on 18 nodes with 32Gb each (very first try, several possible optimizations).

Coppersmith "factorization factory"

Idea: to factor several numbers of the same size, use the same linear polynomial $g(x) = \ell x - m$.

Save in memory the pairs (a, b) such that $G(a, b)$ is smooth.

Can we reuse the 64G relations of RSA-768 ?

$g(x) = 34661003550492501851445829x - 1291187456580021223163547791574810881$

Not trivial because of the leading coefficient ℓ of $g(x)$.
Necessary condition: $n \equiv a_d m^d \bmod \ell$ for an algebraic polynomial $f(x) = a_d x^d + \cdots$
A priori a_d is of same size as ℓ.
Is the *factorization factory* still interesting with progress in polynomial selection?

Tools for NFS

- GGNFS (Chris Monico): includes the *lattice siever* from Franke and Kleinjung
- msieve (Jason Papadopoulos): very efficient for polynomial selection and filtering
- CADO-NFS: very efficient for polynomial selection, cofactorization during sieving (ECM), and linear algebra (block Wiedemann)

CADO-NFS

Developed in the Caramel and Tanc teams since 2007 with grant from ANR (*Agence Nationale de la Recherche*)

LGPL license, available from
`http://cado.gforge.inria.fr/`

Used by Shi Bai (ANU, Canberra) to (re)factor RSA-180.
What's new in CADO-NFS:

- polynomial selection from Kleinjung 2006 and 2008 (work in progress)
- independent implementation of *sieving by vectors*
- linear algebra with block Wiedemann (MPI + threads)
- naive but efficient square root

What next?

RSA-1024.

About 1000 times more difficult.

Should be factored around 2020.

Current open problems:

- polynomial selection with non-linear polynomials
- improve the cofactorization (use GPUs?)
- memory usage of the Berlekamp-Massey step (up to 1Tb for RSA-768)

The factorization

RSA768 =
1230186684530117755130494958384962720772853569595334792197
3224521517264005072636575187452021997864693899564749427740
6384592519255732630345373154826850791702612214291346167042
9214311602221240479274737794080665351419597459856902143413
=
3347807169895689878604416984821269081770479498371376856891
2431388982883793878002287614711652531743087737814467999489
*
3674604366679959042824463379962795263227915816434308764267
6032283815739666511279233373417143396810270092798736308917

Non-linear polynomial selection

Consider two degree-3 polynomials.

```
sage: R.<a0,a1,a2,a3,b0,b1,b2,b3,x> = PolynomialRing(QQ)
sage: f = a3*x^3+a2*x^2+a1*x+a0
sage: g = b3*x^3+b2*x^2+b1*x+b0
sage: f.resultant(g,x)
a3^3*b0^3 - a2*a3^2*b0^2*b1 + a1*a3^2*b0*b1^2 - a0*a3^2*b1^3
+ a2^2*a3*b0^2*b2 - 2*a1*a3^2*b0^2*b2 - a1*a2*a3*b0*b1*b2
+ 3*a0*a3^2*b0*b1*b2 + a0*a2*a3*b1^2*b2 + a1^2*a3*b0*b2^2
- 2*a0*a2*a3*b0*b2^2 - a0*a1*a3*b1*b2^2 + a0^2*a3*b2^3
- a2^3*b0^2*b3 + 3*a1*a2*a3*b0^2*b3 - 3*a0*a3^2*b0^2*b3
+ a1*a2^2*b0*b1*b3 - 2*a1^2*a3*b0*b1*b3 - a0*a2*a3*b0*b1*b3
- a0*a2^2*b1^2*b3 + 2*a0*a1*a3*b1^2*b3 - a1^2*a2*b0*b2*b3
+ 2*a0*a2^2*b0*b2*b3 + a0*a1*a3*b0*b2*b3 + a0*a1*a2*b1*b2*b3
- 3*a0^2*a3*b1*b2*b3 - a0^2*a2*b2^2*b3 + a1^3*b0*b3^2
- 3*a0*a1*a2*b0*b3^2 + 3*a0^2*a3*b0*b3^2 - a0*a1^2*b1*b3^2
+ 2*a0^2*a2*b1*b3^2 + a0^2*a1*b2*b3^2 - a0^3*b3^3
```

If $|a_0|, \ldots, |b_3| < n^{1/6}$, we get $\approx n^{8/6}$ resultants: we expect $\approx n^{1/3}$ of them to equal n.

Montgomery's geometric progression idea

(Work in progress with Peter Montgomery and Thomas Prest.)
Assume we search two polynomials of degree $d = 3$.

Assume we know a "small" geometric progression mod n:

$$c_0, c_1 = c_0 m \bmod n, c_2 = c_1 m \bmod n, c_3 = c_2 m \bmod n$$

LLL-reduce the matrix, where K is an integer:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ Kc_0 & Kc_1 & Kc_2 & Kc_3 \end{pmatrix}$$

Assume K is large enough so that we get two short vectors of the form:

$$\begin{pmatrix} a_0 & b_0 \\ a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \\ 0 & 0 \end{pmatrix}$$

Then we have both $a_0c_0 + a_1c_1 + a_2c_2 + a_3c_3 = 0$ and $b_0c_0 + b_1c_1 + b_2c_2 + b_3c_3 = 0$.

Since c_i is a geometric progression mod n, this implies:

$$c_0(a_0 + a_1m + a_2m^2 + a_3m^3) = 0 \bmod n$$

$$c_0(b_0 + b_1m + b_2m^2 + b_3m^3) = 0 \bmod n$$

m is a common root of $f(x) = a_3x^3 + a_2x^2 + a_1x + a_0$ and $g(x) = b_3x^3 + b_2x^2 + b_1x + b_0$.

Q1: how large should we choose K?

Q2: how large is $\mathrm{Res}(f, g)$ wrt n?

Q3: how to find a “small” geometric progression mod n?

How to find a small geometric progression?

Take a random m modulo n.
Reduce the lattice

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ m & n & 0 & 0 \\ m^2 & 0 & n & 0 \\ m^3 & 0 & 0 & n \end{pmatrix}$$

Let (c_0, c_1, c_2, c_3) be a short vector.
We have $c_1 = c_0 m \bmod n$, $c_2 = c_0 m^2 \bmod n$, $c_3 = c_0 m^3 \bmod n$

```
sage: n=716415207617514354551336164756670904340633322282478717954 29
sage: m=408032887941196215928681238939838458855834978853637675053 05
sage: L=matrix([[1,0,0,0],[m,n,0,0],[m^2,0,n,0],[m^3,0,0,n]])
sage: L = L.transpose().LLL()
sage: c = L.row(0); c
(-205424878029429476494656408346637779556002 31,
-490733114170605505303143065129890962904108 80,
461208924866760555036444217140196835587791 5,
-806519320499212957640276650321104071586148 04)
sage: [(c[0]*m^i - c[i]) % n for i in range(4)]
[0, 0, 0, 0]
```