PA Cybersecurity/IPP.CS.M1/MSc&T-CTD

Introduction to Cryptology (INF558)

F. MORAIN

How to factor an integer

How to factor an integer

Very very old problem, but few algorithms known:

- division by small primes;
- $p-1$ (and variants like ECM – Elliptic Curve Method);
- RHO (random mappings);
- combining congruences.

Ref. Crandal / Pomerance, *Prime numbers – A Computational Perspective*. See also my lectures notes for INF558.

Kraitchik (1920): find x s.t. $N \mid x^2 - 1, x \neq \pm 1$.

Ex. If $N = 143$, there exist 4 solutions ± 1, ± 12 and $\gcd(12-1, 143) = 11$.

Rem. We prefer to work on integers modulo N, i.e. $x \equiv y \bmod N \iff N \mid x - y$; $x \bmod N =$ remainder of the euclidean division of x by N.

The general approach: approximating squares

Step 0: build a prime basis $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$.

Step 1: find at least $k+1$ relations $(R_i)_{i \in I}$:
$R_i = \prod_{j=1}^{k} p_j^{a_{i,j}} \equiv 1 \bmod N$

Step 2: find $I' \subset I$ s.t.

$$\prod_{i \in I'} R_i = x^2$$

over $\mathbb{Z}$, which is equivalent to

$$\forall\, j, \sum_{i \in I'} a_{i,j} \equiv 0 \bmod 2,$$

which is a classical linear algebra problem.

Step 3: x is a square-root of 1 and with probability $\geq 1/2$, $\gcd(x-1, N)$ is non-trivial.

The general approach: approximating squares

Step 0: build a prime basis $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$.

Step 1: find at least $k+1$ relations $(R_i)_{i \in I}$:
$R_i = \prod_{j=1}^{k} p_j^{a_{i,j}} \equiv 1 \bmod N$

Step 2: find $I' \subset I$ s.t.

$$\prod_{i \in I'} R_i = x^2$$

over $\mathbb{Z}$, which is equivalent to

$$\forall j, \sum_{i \in I'} a_{i,j} \equiv 0 \bmod 2,$$

which is a classical linear algebra problem.

Step 3: x is a square-root of 1 and with probability $\geq 1/2$, $\gcd(x-1, N)$ is non-trivial.

The general approach: approximating squares

Step 0: build a prime basis $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$.

Step 1: find at least $k+1$ relations $(R_i)_{i \in I}$:
$R_i = \prod_{j=1}^{k} p_j^{a_{i,j}} \equiv 1 \bmod N$

Step 2: find $I' \subset I$ s.t.

$$\prod_{i \in I'} R_i = x^2$$

over $\mathbb{Z}$, which is equivalent to

$$\forall j, \sum_{i \in I'} a_{i,j} \equiv 0 \bmod 2,$$

which is a classical linear algebra problem.

Step 3: x is a square-root of 1 and with probability $\geq 1/2$, $\gcd(x-1, N)$ is non-trivial.

The general approach: approximating squares

Step 0: build a prime basis $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$.

Step 1: find at least $k+1$ relations $(R_i)_{i \in I}$:
$R_i = \prod_{j=1}^{k} p_j^{a_{i,j}} \equiv 1 \bmod N$

Step 2: find $I' \subset I$ s.t.

$$\prod_{i \in I'} R_i = x^2$$

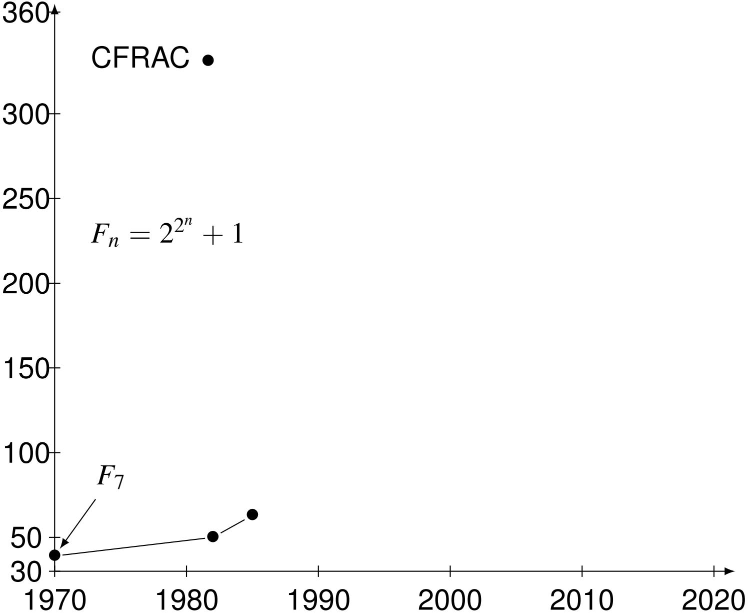
over $\mathbb{Z}$, which is equivalent to

$$\forall j, \sum_{i \in I'} a_{i,j} \equiv 0 \bmod 2,$$

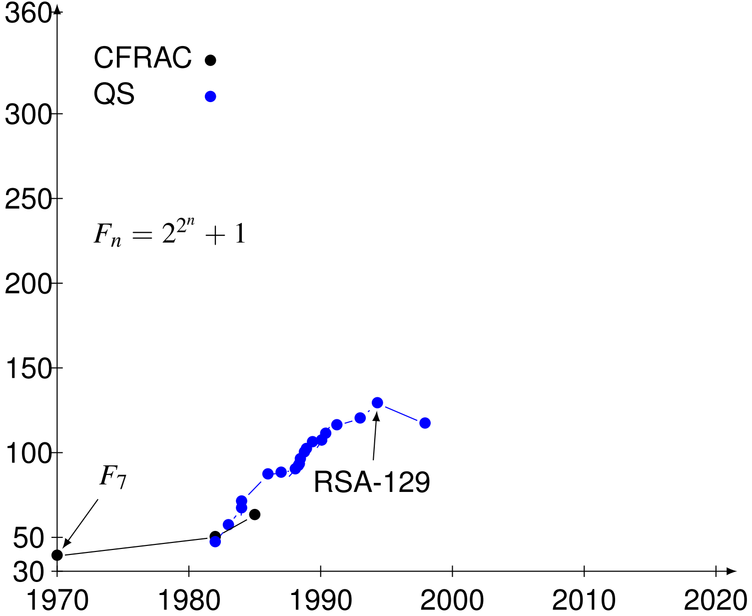
which is a classical linear algebra problem.

Step 3: x is a square-root of 1 and with probability $\geq 1/2$, $\gcd(x-1, N)$ is non-trivial.

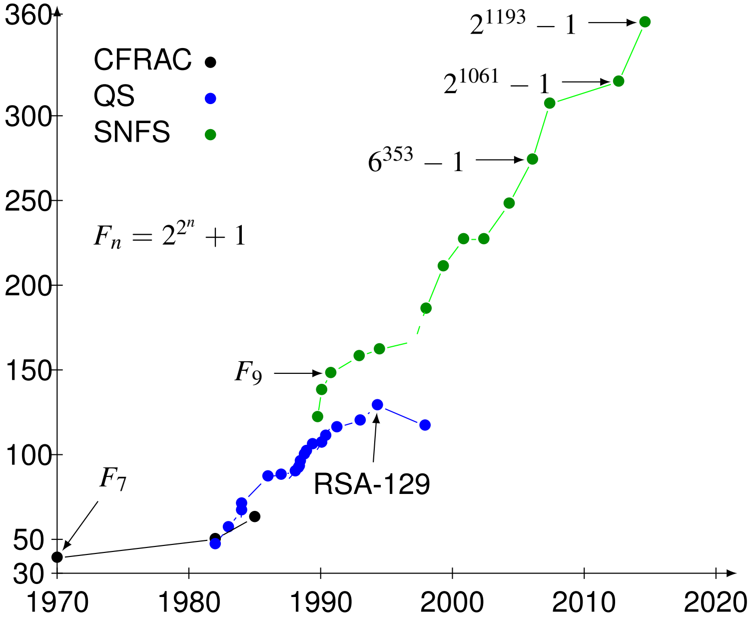
A brief history



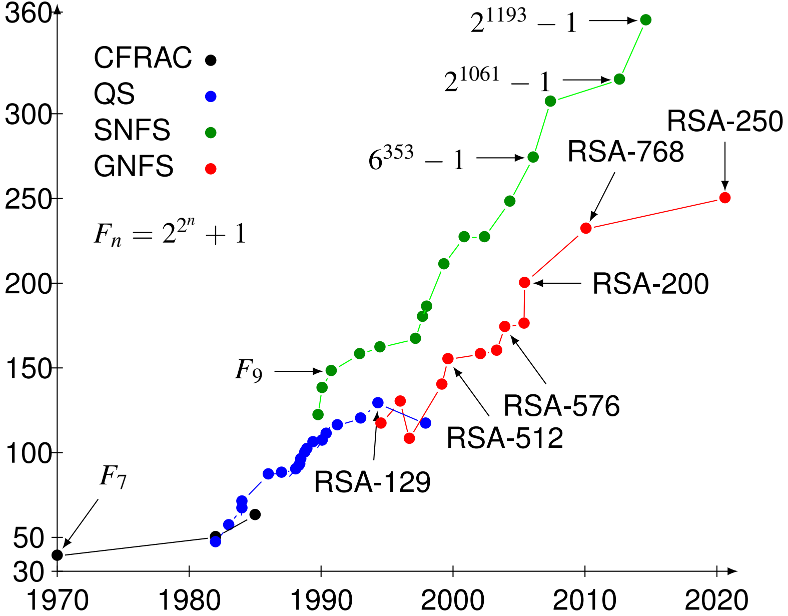
A brief history



A brief history



A brief history

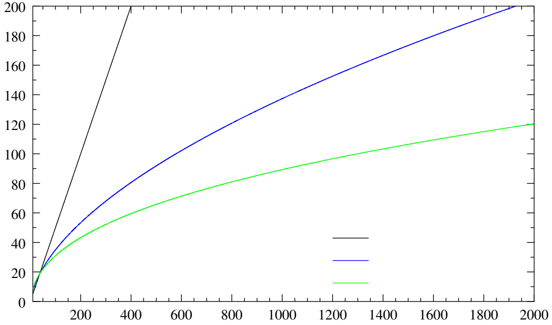


Complexities

$$L_N[\alpha, c] = \exp((c + o(1))(\log N)^{\alpha}(\log\log N)^{1-\alpha}).$$

CFRAC, QS: $L_N[1/2, c]$.

NFS: $L_N[1/3, c]$.



RSA-250/795b with CADO-NFS

- **Who?** Fabrice Boudot, Pierrick Gaudry, Aurore Guillevic, Nadia Heninger,Emmanuel Thomé, Paul Zimmermann. Announced during CRYPTO'2020. `https://gitlab.inria.fr/cado-nfs/records/`
- **Sieving:** (2019/10 - 2020/02) 2,450 core years; 6,132,671,469 unique relations.
- **Linear algebra** (after filtering): 404,711,409 rows; (2020/02) 250 core years (block Wiedemann in parallel).

Take home message

- Very old foundational problem in number theory.
- Heavy non-trivial implementations of algorithms using a lot of maths + distributed computations.
- Very advanced vs. quantum algorithms: 795b vs. (at most) 48b.