Schnorr's Approach to Factoring via Lattices

Léo Ducas

CWI, Amsterdam, The Netherlands

RISC Seminar, May 20th, 2021

History

- *Factoring integers and computing discrete logarithms via diophantine approximation* [Schnorr 1991]
- *Factoring and Lattice Reduction* [Adleman 1995]
- *Average Time Fast SVP and CVP Algorithms: Factoring Integers in Polynomial Time* [Schnorr 2009]
- *A note on integer factorization using lattices* [Vera 2010]
- *Fast Factoring Integers by SVP Algorithms* [Schnorr 2021]

History

- *Factoring integers and computing discrete logarithms via diophantine approximation* [Schnorr 1991]
- *Factoring and Lattice Reduction* [Adleman 1995]
- *Average Time Fast SVP and CVP Algorithms: Factoring Integers in Polynomial Time* [Schnorr 2009]
- *A note on integer factorization using lattices* [Vera 2010]
- *Fast Factoring Integers by SVP Algorithms* [Schnorr 2021]

This talk

Not about [Schnorr 2021], but about the general approach.

Reviews of [Schnorr 2021]

- `https://github.com/lducas/SchnorrGate`
- `https://crypto.stackexchange.com/questions/88582`
- `https://twitter.com/inf_0_/status/1367376526300172288`

Factoring: the Quadratic Sieve [Pomerance, 1981]

Notation : $\equiv$ for congruence modulo N

Goal: Find a non-trivial[1] solution to $X^2 \equiv Y^2$

$\Rightarrow (X - Y)(X + Y) \equiv 0$

$\Rightarrow \gcd(X \pm Y, N)$ is a non-trivial factor of N

[1] $X \not\equiv \pm Y \mod N$

Factoring: the Quadratic Sieve [Pomerance, 1981]

Notation : $\equiv$ for congruence modulo N

Goal: Find a non-trivial[1] solution to $X^2 \equiv Y^2$

$\Rightarrow (X - Y)(X + Y) \equiv 0$
$\Rightarrow \gcd(X \pm Y, N)$ is a non-trivial factor of N

A two-steps process:

- Collect Relations
- Linear Algebra

[1] $X \not\equiv \pm Y \mod N$

Step 1: Relation Collection

- Define a **factor basis**: $\mathcal{F} = \{p | p \text{ is primes}, p \leq B\}$
- Repeat:

Step 1: Relation Collection

- Define a **factor basis**: $\mathcal{F} = \{p | p \text{ is primes}, p \leq B\}$
- Repeat:
  - Pick random X, compute $Z = X^2 \bmod N$
  - Use **trial division** to write $Z = \prod p_i^{e_i}$ $\qquad (p_i \in \mathcal{F})$
  - If successful, store the **relation** $X^2 \equiv \prod p_i^{e_i}$
- Until B relations are collected

The complexity trade-off

- Increasing B improves the success probability of each trial
- But more relations are needed
- The optimum is at $B = \exp(\tilde{O}(\sqrt{\log N}))$ $\qquad = L_N(1/2)$

Step 2: Linear Algebra

- We have collected relations:

$$\begin{array}{cccccc} X_1^2 & \equiv & {p_1}^{e_{1,1}} & {p_2}^{e_{1,2}} & {p_3}^{e_{1,3}} & \cdots \\ X_2^2 & \equiv & {p_1}^{e_{2,1}} & {p_2}^{e_{2,2}} & {p_3}^{e_{2,3}} & \cdots \\ X_3^2 & \equiv & {p_1}^{e_{3,1}} & {p_2}^{e_{3,2}} & {p_3}^{e_{3,3}} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}$$

- Combine the above to make all exponents even integers

Step 2: Linear Algebra

- We have collected relations:

$$\begin{array}{cccccc}
X_1^2 & \equiv & {p_1}^{e_{1,1}} & {p_2}^{e_{1,2}} & {p_3}^{e_{1,3}} & \cdots \\
X_2^2 & \equiv & {p_1}^{e_{2,1}} & {p_2}^{e_{2,2}} & {p_3}^{e_{2,3}} & \cdots \\
X_3^2 & \equiv & {p_1}^{e_{3,1}} & {p_2}^{e_{3,2}} & {p_3}^{e_{3,3}} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{array}$$

- Combine the above to make all exponents even integers
- Done by solving a linear system over $\mathbb{F}_2$

Step 2: Linear Algebra

- We have collected relations:

$$\begin{array}{cccccc} X_1^2 & \equiv & {p_1}^{e_{1,1}} & {p_2}^{e_{1,2}} & {p_3}^{e_{1,3}} & \cdots \\ X_2^2 & \equiv & {p_1}^{e_{2,1}} & {p_2}^{e_{2,2}} & {p_3}^{e_{2,3}} & \cdots \\ X_3^2 & \equiv & {p_1}^{e_{3,1}} & {p_2}^{e_{3,2}} & {p_3}^{e_{3,3}} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}$$

- Combine the above to make all exponents even integers
- Done by solving a linear system over $\mathbb{F}_2$
- Obtain a solution to

$$X^2 \equiv Y^2 \bmod N$$

Optimizing Relation Collection

$X^2 \bmod N$ is as large as N for random X

Making it smaller would improve the success of trial division

Optimizing Relation Collection

$X^2 \bmod N$ is as large as N for random X

Making it smaller would improve the success of trial division

Could we aim for $X^2 \bmod N$ that are significantly smaller ?

Choose $X \approx \sqrt{N}$, so that $X^2 \approx N$

If $X = \sqrt{N} + \epsilon$, with $\epsilon \ll \sqrt{N}$, then:

$$X^2 \equiv 2\epsilon\sqrt{N} + \epsilon^2$$

Optimizing Relation Collection

$X^2 \bmod N$ is as large as N for random X

Making it smaller would improve the success of trial division

Could we aim for $X^2 \bmod N$ that are significantly smaller ?

Choose $X \approx \sqrt{N}$, so that $X^2 \approx N$

If $X = \sqrt{N} + \epsilon$, with $\epsilon \ll \sqrt{N}$, then:

$$X^2 \equiv 2\epsilon\sqrt{N} + \epsilon^2$$

The complexity gain

Improves the hidden constant in $\exp(\tilde{O}(\sqrt{\log N}))$ $= L_N(1/2)$

Aiming Better [Schnorr 1991]

A Relaxation

The left-hand-side needs not be square, B-smooth can do as well:

$$p_1{}^{e'_1} p_2{}^{e'_2} p_3{}^{e'_3} \cdots \equiv p_1{}^{e_1} p_2{}^{e_2} p_3{}^{e_3} \cdots$$

$$1 \equiv p_1{}^{e_1 - e'_1} p_2{}^{e_2 - e'_2} p_3{}^{e_3 - e'_3} \ldots$$

Our New Goal

Find positive exponents $(e'_1, e'_2, e'_3, \ldots)$ such that

$$p_1{}^{e'_1} p_2{}^{e'_2} p_3{}^{e'_3} \cdots \approx N$$

A Relaxation

The left-hand-side needs not be square, B-smooth can do as well:

$$\begin{aligned} {p_1}^{e_1'} {p_2}^{e_2'} {p_3}^{e_3'} \cdots &\equiv {p_1}^{e_1} {p_2}^{e_2} {p_3}^{e_3} \cdots \\ 1 &\equiv {p_1}^{e_1 - e_1'} {p_2}^{e_2 - e_2'} {p_3}^{e_3 - e_3'} \cdots \end{aligned}$$

Our New Goal

Find positive exponents $(e_1', e_2', e_3', \ldots)$ such that

$${p_1}^{e_1'} {p_2}^{e_2'} {p_3}^{e_3'} \cdots \approx N$$

This is an (approximate) knapsack problem !

$$e_1' \ln p_1 + e_2' \ln p_2 + e_3' \ln p_3 + \cdots \approx \ln N$$

Aiming with lattices [Schnorr 1991]

Choose a constant C to rewrite the knapsack as a lattice CVP

$$\begin{bmatrix} \ln p_1 & & & & \\ & \ln p_2 & & & \\ & & \ln p_3 & & \\ & & & \ddots & \\ & & & & \ln p_n \\ C \ln p_1 & C \ln p_2 & C \ln p_3 & \cdots & C \ln p_n \end{bmatrix} \cdot \begin{bmatrix} e'_1 \\ e'_2 \\ e'_3 \\ \vdots \\ e'_n \end{bmatrix} \approx \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ C \ln N \end{bmatrix}$$

Knapsack $\neq$ CVP

The lattice solution $(e'_1, e'_2, e'_3, \ldots)$ may not have positive exponents

Aiming with lattices [Schnorr 1991]

Choose a constant C to rewrite the knapsack as a lattice CVP

$$\begin{bmatrix} \ln p_1 & & & & \\ & \ln p_2 & & & \\ & & \ln p_3 & & \\ & & & \ddots & \\ & & & & \ln p_n \\ C\ln p_1 & C\ln p_2 & C\ln p_3 & \cdots & C\ln p_n \end{bmatrix} \cdot \begin{bmatrix} e'_1 \\ e'_2 \\ e'_3 \\ \vdots \\ e'_n \end{bmatrix} \approx \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ C\ln N \end{bmatrix}$$

Knapsack $\neq$ CVP

The lattice solution $(e'_1, e'_2, e'_3, \ldots)$ may not have positive exponents

But that might be OK !

- $u/v \approx N \Rightarrow u \approx vN$, therefore $S = u - vN$ may be small
- Quality degrades as $v = \prod_{e'_i<0} p_i^{-e_i}$ gets larger

Attempting Average-Case Analysis

Lattice Pitfalls

- The lattice is not full dimensional — apply due projections
- Gaussian Heuristic seems invalid — for certain C
- The ℓ_2 norm is a bit inadequate — ℓ_1 more relevant
- Naive predictions of ℓ_2/ℓ_1 can also fail

Trial Division Pitfall

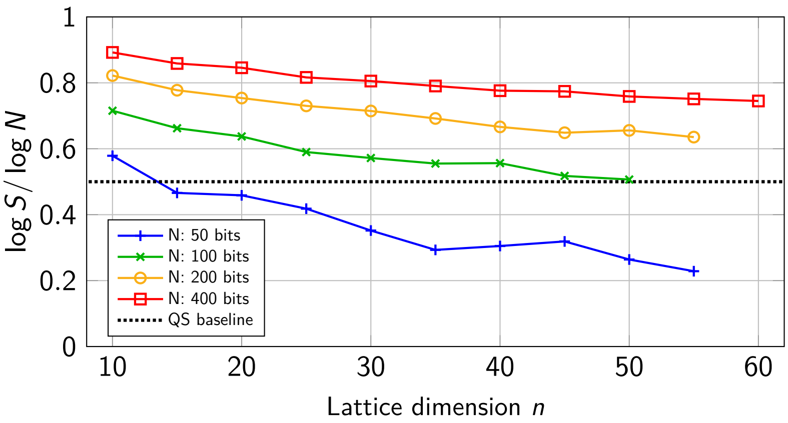
- B-Smoothness probability of $S = u - vN$ lower than expected

$$p_i | u \vee p_i | v \Rightarrow p_i \not| S$$

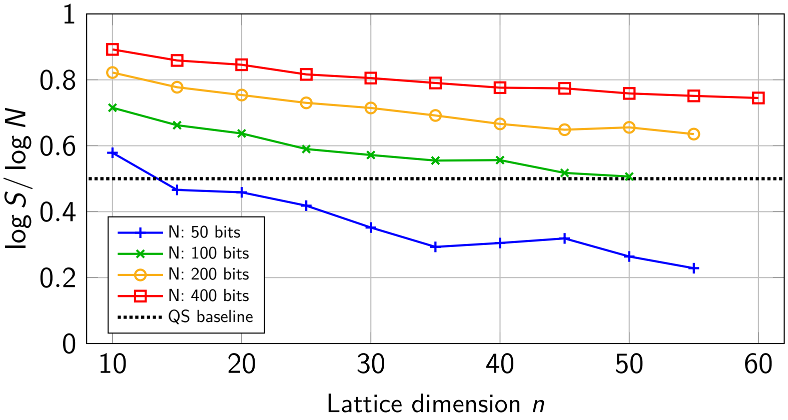
Mind the Variants

- Most papers force $B = p_n$ or $B = 1$. Here: B unconstrained.
- The diagonal part of the lattice may vary as well.

Experiments



Experiments



The size of S roughly dictates the cost of the non-lattice steps

For factoring a 100-bits N, to beat QS at the non-lattice steps, we should need a lattice dimension of at least $n \geq 50$.

My two Cents

- It's a deep and brilliant idea . . . that doesn't seem to work ☹
- A solid average-case complexity analysis is still missing

  and appears quite challenging . . .
- It nevertheless found applications beyond factoring

My two Cents

- It's a deep and brilliant idea . . . that doesn't seem to work ☹
- A solid average-case complexity analysis is still missing

  and appears quite challenging . . .
- It nevertheless found applications beyond factoring
  - An attempt at proving SVP $\geq$ Factoring [Adleman 1995]
  - Proof of NP-hardness of SVP [Ajtai 1998, Micciancio 1998]
  - Idea reused for in relation to the abc-conjecture [Bright 2014]
  - Idea reused in a Module-LLL Algorithm [LPSW 2019]