Lattices & Factoring
Invited Talk

Léo Ducas

CWI, Amsterdam, The Netherlands

PKC, May 10th, 2021
Cryptography is getting old.
Cryptography is getting old.
Cryptography has reached a non-negligible age.
Cryptography is getting old.
Cryptography has reached a non-negligible age.
Let’s write our history before it gets lost.
Typical narrative on Knapsack-based cryptography

- An embarrassment to forget
- Ajtai single-handedly put an end to that dark Era
Typical narrative on Knapsack-based cryptography

- An embarrassment to forget
- Ajtai single-handedly put an end to that dark Era

I do not subscribe to that narrative.
Typical narrative on Knapsack-based cryptography

- An embarrassment to forget
- Ajtai single-handedly put an end to that dark Era

I do not subscribe to that narrative.

If I have seen further, it is by standing on the shoulders of Giants.

- *Isaac Newton*

Ajtai is a Giant of Lattice-based Cryptography. Let’s enjoy the the view he had from the shoulders his own Giants.
Today’s Giants

Factoring with Lattices Short Vectors

C-P. Schnorr

L. Adleman
Today’s Giants

Factoring with Lattices Short Vectors

C-P. Schnorr  
L. Adleman

Decoding Lattices by Factorization

B. Chor  
R. Rivest
Part I:
Factoring with Lattice Short Vectors
Notation: \( \equiv \) for congruence modulo \( N \)

Goal: Find a non-trivial\(^1\) solution to \( X^2 \equiv Y^2 \)

\[ \Rightarrow (X - Y)(X + Y) \equiv 0 \]

\[ \Rightarrow \gcd(X \pm Y, N) \text{ is a non-trivial factor of } N \]

\(^1\)\( X \not\equiv \pm Y \mod N \)
Notation: $\equiv$ for congruence modulo $N$

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

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

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

A two-steps process:

- Collect Relations
- Linear Algebra

\(^1\) $X \not\equiv \pm Y \pmod{N}$
Step 1: Relation Collection

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

  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 \( B = \exp(\tilde{O}(\sqrt{\log N})) = L_N(1/2) \)
Step 1: Relation Collection

- Define a **factor basis**: \( \mathcal{F} = \{ p \mid p \text{ is primes, } p \leq B \} \)
- Repeat:
  - Pick random \( X \), compute \( Z = X^2 \mod N \)
  - Use **trial division** to write \( Z = \prod p_i^{e_i} \)
  - 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 \( B = \exp(\tilde{O}(\sqrt{\log N})) = L_N(1/2) \)
Step 2: Linear Algebra

- We have collected relations:

\[
\begin{align*}
X_1^2 & \equiv p_1^{e_{1,1}} p_2^{e_{1,2}} p_3^{e_{1,3}} \cdots \\
X_2^2 & \equiv p_1^{e_{2,1}} p_2^{e_{2,2}} p_3^{e_{2,3}} \cdots \\
X_3^2 & \equiv p_1^{e_{3,1}} p_2^{e_{3,2}} p_3^{e_{3,3}} \cdots \\
\vdots & \vdots \vdots \vdots \vdots \vdots \vdots \vdots
\end{align*}
\]

- Combine the above to make all exponents even integers
Step 2: Linear Algebra

- We have collected relations:

\[
\begin{align*}
X_1^2 & \equiv p_1^{e_{1,1}} p_2^{e_{1,2}} p_3^{e_{1,3}} \ldots \\
X_2^2 & \equiv p_1^{e_{2,1}} p_2^{e_{2,2}} p_3^{e_{2,3}} \ldots \\
X_3^2 & \equiv p_1^{e_{3,1}} p_2^{e_{3,2}} p_3^{e_{3,3}} \ldots \\
\vdots & \vdots \vdots \vdots \vdots \vdots \vdots \vdots 
\end{align*}
\]

- 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:

\[
\begin{align*}
X_1^2 & \equiv p_1^{e_{1,1}} p_2^{e_{1,2}} p_3^{e_{1,3}} \ldots \\
X_2^2 & \equiv p_1^{e_{2,1}} p_2^{e_{2,2}} p_3^{e_{2,3}} \ldots \\
X_3^2 & \equiv p_1^{e_{3,1}} p_2^{e_{3,2}} p_3^{e_{3,3}} \ldots \\
\vdots & \vdots \vdots \vdots \vdots \\
\end{align*}
\]

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

\[
X^2 \equiv Y^2 \mod N
\]
$X^2 \mod N$ is as large as $N$ for random $X$

Making it smaller would improve the success of trial division
Optimizing Relation Collection

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

Making it smaller would improve the success of trial division

Could we aim for $X^2 \mod N$ 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

\(X^2 \mod N\) is as large as \(N\) for random \(X\)

Making it smaller would improve the success of trial division

Could we aim for \(X^2 \mod N\) 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 \(\exp(\tilde{O}(\sqrt{\log N})) = L_N(1/2)\)
A Relaxation

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

\[ p_1^{e_1'} p_2^{e_2'} p_3^{e_3'} \cdots \equiv p_1^{e_1} p_2^{e_2} p_3^{e_3} \cdots \]
\[ 1 \equiv p_1^{e_1-e_1'} p_2^{e_2-e_2'} p_3^{e_3-e_3'} \cdots \]

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:

\[ p_1^{e'_1} p_2^{e'_2} p_3^{e'_3} \cdots \equiv p_1^{e_1} p_2^{e_2} p_3^{e_3} \cdots \]
\[ 1 \equiv p_1^{e_1-e'_1} p_2^{e_2-e'_2} p_3^{e_3-e'_3} \cdots \]

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!

\[ e'_1 \ln p_1 + e'_2 \ln p_2 + e'_3 \ln p_3 + \cdots \approx \ln N \]
Choose a constant $C$ to rewrite the knapsack as a lattice CVP

\[
\begin{bmatrix}
\ln p_1 \\
\ln p_2 \\
\ln p_3 \\
\vdots \\
\ln p_n \\
C \ln p_1 \\
C \ln p_2 \\
C \ln p_3 \\
\vdots \\
C \ln p_n
\end{bmatrix}
\begin{bmatrix}
e'_1 \\
e'_2 \\
e'_3 \\
\vdots \\
e'_n
\end{bmatrix}
\approx
\begin{bmatrix}
0 \\
0 \\
0 \\
\vdots \\
0 \\
C \ln N
\end{bmatrix}
\]

**Knapsack $\neq$ CVP**

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

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

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

Knapsack $\ne$ 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
- Gaussian Heuristic seems invalid
- The $\ell_2$ norm is a bit inadequate
- Naive predictions of $\ell_2/\ell_1$ can also fail

Trial Division Pitfall

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

\[ p_i | u \lor p_i | v \implies p_i \not| S \]

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

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 $n \geq 50$.  

Léa Ducas (CWI)  

Lattices & Factoring
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 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 complexity analysis is still missing and appears quite challenging . . .
- It nevertheless found applications beyond factoring
  - An attempt at proving SVP $\geq$ Factoring [Adleman 1995]
  - A successful proof of NP-hardness for SVP [Ajtai 1998]
  - Idea reused for in relation to the abc-conjecture [Bright 2014]
  - Idea reused in a Module-LLL Algorithm [LPSW 2019]
Recall the gap between Knapsack and SVP

- Knapsack solutions $\in \{0, 1\}^n$, SVP solution $\mathbb{Z}^n$
- Knapsack was known to be NP-hard, but not SVP

The key Insight

Solutions in Schnorr-Adleman lattice are in correspondence with smooth and square-free integers.

We know how to count those!

A proof that $SVP \geq Knapsack$

Therefore SVP is NP-hard

Learn more from Daniele's talk next week at the RISC seminar.
Recall the gap between Knapsack and SVP

- Knapsack solutions $\in \{0, 1\}^n$, SVP solution $\mathbb{Z}^n$
- Knapsack was known to be NP-hard, but not SVP

The key Insight

- $\{0, 1\}^n$ solutions in Schnorr-Adleman lattice are in correspondence with smooth and square-free integers
- We know how to count those!
A Surprising Twist

Recall the gap between Knapsack and SVP

- Knapsack solutions $\in \{0, 1\}^n$, SVP solution $\mathbb{Z}^n$
- Knapsack was known to be NP-hard, but not SVP

The key Insight

- $\{0, 1\}^n$ solutions in Schnorr-Adleman lattice are in correspondence with smooth and square-free integers
- We know how to count those!

A proof that SVP $\geq$ Knapsack

- Therefore SVP is NP-hard
- Learn more from Daniele’s talk next week at the RISC seminar
Part II: Decoding Lattices by Factorization
In this whole section we work with the $\ell_1$ norm!

**Bounded Distance Decoding with radius $r$**

- Given $t = v + e$ where $v \in \mathcal{L}$ and $\|e\| \leq r$
- Recover $v$ and $e$

Unique solution guaranteed for $r \leq \lambda_1(\mathcal{L})/2$.

**Minkowsky’s bound**

$$\frac{\lambda_1(\mathcal{L})}{\det(\mathcal{L})^{1/n}} \leq O(n)$$

We want a lattice and decoding alg. close to this bound.
The Key Idea

- Subset-sums is hard
- Subset-product is easy (trial divisions)
- Take logarithm, disguise the later as the former, get crypto.
The Key Idea

- Subset-sums is hard
- Subset-product is easy (trial divisions)
- Take logarithm, disguise the later as the former, get crypto.

Variants/Follow-ups

- Originally over $\mathbb{F}_p[X]$; variants over $\mathbb{Z}$:
  - [Naccache Stern '97, Okamoto Tanaka Uchiyama '00].
The Key Idea

- Subset-sums is hard
- Subset-product is easy (trial divisions)
- Take logarithm, disguise the later as the former, get crypto.

Variants/Follow-ups

- [Lenstra ’90, Li Ling Xing Yeo ’17].
- Originally over $\mathbb{F}_p[X]$; variants over $\mathbb{Z}$:
  - [Naccache Stern ’97, Okamoto Tanaka Uchiyama ’00].

A Coding Gem Hidden Inside

- [Brier et al. ’15]: Remove crypto from [NS’97], hides a good decodable binary code.
- [D. Pierrot ’18]: [CR88, OTU00], hides a good decodable lattice.
Choose a modulus $M = 3^k$ 
And a factor basis $\mathcal{F} = \{2, 5, 7, 11, 13, \ldots, p_n\}$ 

Define the morphism $\psi : \mathbb{Z}_n \to (\mathbb{Z}/M\mathbb{Z})^*$: 

$$
\psi : x \mapsto \prod p_i^{x_i} \mod M
$$

And finally define the kernel lattice 

$$
\mathcal{L} := \ker \psi = \left\{ v \in \mathbb{Z}^n \mid \prod p_i^{\gamma_i} = 1 \mod M \right\}
$$
Chor-Rivest Lattice (over the integers)

- Choose a modulus $M = 3^k$
- And a factor basis $\mathcal{F} = \{2, 5, 7, 11, 13, \ldots, p_n\}$

$$B := p_n \sim n \ln n$$

- Define the morphism $\psi : \mathbb{Z}^n \rightarrow (\mathbb{Z}/M\mathbb{Z})^*$:

$$\psi : x \mapsto \prod p_i^{x_i} \mod M$$

- And finally define the kernel lattice

$$\mathcal{L} := \ker \psi = \left\{ v \in \mathbb{Z}^n \mid \prod p_i^{v_i} = 1 \mod M \right\}$$

The lattice can be computed efficiently!

- Discrete logarithms modulo $M = 3^k$ is easy
- Rewrites as a subset-sum lattice

$$\mathcal{L} = \left\{ v \in \mathbb{Z}^n \mid \sum v_i \text{dlog} \ p_i = 0 \mod \varphi(M) \right\}$$
Lattice Parameters

Lattice parameters

- \( \text{dim } \mathcal{L} = n \)
- \( \text{det } \mathcal{L} \leq \varphi(M) \leq M \)

Claim: \( \lambda_1(\mathcal{L}) \geq \log M / \log B \) (Not exactly true . . .)

- Recall that \( \mathcal{L} = \{ \mathbf{v} \in \mathbb{Z}^n \mid \prod p_i^{v_i} = 1 \mod M \} \).
- For \( \mathbf{v} \neq 0 \) to be in \( \mathcal{L} \), \( \prod p_i^{v_i} \) must wrap around \( \mod M \)
- In particular \( B \| \mathbf{v} \|_1 \geq M \) (This proof is a bit bogus !)
### Lattice Parameters

<table>
<thead>
<tr>
<th>dim $\mathcal{L} = n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>det $\mathcal{L} \leq \varphi(M) \leq M$</td>
</tr>
</tbody>
</table>

**Claim:** $\lambda_1(\mathcal{L}) \geq \log M / \log B$  
(Not exactly true . . .)

- Recall that $\mathcal{L} = \{v \in \mathbb{Z}^n \mid \prod p_i^{v_i} = 1 \mod M\}$.
- For $v \neq 0$ to be in $\mathcal{L}$, $\prod p_i^{v_i}$ must wrap around mod $M$.
- In particular $B^{\|v\|_1} \geq M$  
  (This proof is a bit bogus !)

**Instantiate with** $k = n$, i.e. $M = 3^n$

$$\frac{\lambda_1(\mathcal{L})}{\det(\mathcal{L})^{1/n}} \geq O\left(\frac{n}{\log n}\right)$$

That is only $O(\log n)$ factor away from Minkowsky bound.
Decoding Chor-Rivest Lattice

Bounded Distance Decoding with radius $r = \log M / \log B$

- Given $t = v + e$ where $v \in \mathcal{L}$ and $\|e\| \leq r$
- Recover $v$ and $e$

- Compute

$$f = \prod p_i^{t_i} \mod M = \prod p_i^{v_i} \prod p_i^{e_i} \mod M = \prod p_i^{e_i} \mod M$$
Decoding Chor-Rivest Lattice

Bounded Distance Decoding with radius \( r = \log M / \log B \)

- Given \( t = v + e \) where \( v \in \mathcal{L} \) and \( \|e\| \leq r \)
- Recover \( v \) and \( e \)

- Compute

\[
f = \prod p_i^{t_i} \mod M = \prod p_i^{v_i} \prod p_i^{e_i} \mod M = \prod p_i^{e_i} \mod M
\]

- Note \( \prod p_i^{e_i} \leq B^r \leq M \): we know it over \( \mathbb{Z} \) not just mod \( M \)
- Factorize it by trial division: recover \( e \)
Dealing with Negative Errors

Now assume $2 \cdot B^r < \sqrt{M}$.

$$f = \prod_{e_i > 0} p_i^{e_i} \cdot \prod_{e_i < 0} p_i^{e_i} = u/v \mod M.$$ 

Lemma (Recovering $u, v$ given $f$ and $M$)

Let $u, v, M$ be coprime s.t. $u, v < \sqrt{M}/2$, and let $f = u/v \mod M$. Then, $\pm (u, v)$ are the shortest vectors of the 2-dimensional lattice

$$L = \{(x, y) \in \mathbb{Z}^2 | x - fy = 0 \mod M\}.$$ 

In particular, given $f$ and $M$, one can recover $(u, v)$ in poly-time.
The last mile?

We are still $O(\log n)$ away from Minkowsky’s bound...
The issue is that we do not have enough small primes.
To get down to $O(1)$ away from Minkowsky’s bound, we need

$$n$$ primes of ‘size’ $O(1)$.

- Switching back from $\mathbb{Z}$ to $\mathbb{F}_p[X]$ doesn’t improve asymptotics
- Elliptic curves could?
- And what about Mordell-Weil lattices? [Shioda ’91, Elkies ’94]
The last mile?

We are still $O(\log n)$ away from Minkowsky’s bound...
The issue is that we do not have enough small primes.
To get down to $O(1)$ away from Minkowsky’s bound, we need

\[ n \text{ primes of ‘size’ } O(1). \]

- Switching back from $\mathbb{Z}$ to $\mathbb{F}_p[X]$ doesn’t improve asymptotics
- Elliptic curves could?
- And what about Mordell-Weil lattices? [Shioda ’91, Elkies ’94]

A Recent Result

Using a completely different approach (construction $D$ lattice over BCH codes), we are now $O(\sqrt{\log n})$ away from Minkowsky’s bound

[Mook Peikert 2020]
Chor-Rivest Knapsack Cryptosystem is *not* Broken

- And offers very short ciphertexts!
- The underlying assumption is intriguing, especially quantumly
  Some kind of reverse of discrete logarithm problem

Chor-Rivest Decoding can be practical

- Better decoding in a pure LWE-based scheme?

And for Something Completely Different

- VBB Obfuscation of "near-equality" tests!
Part III: A Critique of Research in Lattice-Based Cryptography
Due credits

SIS/LWE formalism have achieved impressive feats, and the foundational work from TCS experts was exceptionally thorough.

---

Not a critique of the contributions, but of what we have done of them.
Due credits

SIS/LWE formalism have achieved impressive feats, and the foundational work from TCS experts was exceptionally thorough.

But ...

- Worst-case hardness is not a silver bullet and does not dispense us from cryptanalysis
- We have locked ourselves in subspace of designs and current designs likely far from optimal
- Some very interesting ideas have been buried if not demoted to cryptographic sins

\(^2\)Not a critique of the contributions, but of what we have done of them.
$\mathbb{Z}^n$, the Saddest of all Lattices

All algorithmic tasks (encode, decode, sample) in lattice-based cryptography are reduced to $\mathbb{Z}$ or $\mathbb{Z}^n$. Yet, geometrically (packing, covering, ...) $\mathbb{Z}^n$ is the **worst** lattice.

---

$^3$If you ever deal with prime cyclotomics rings, please read
\( \mathbb{Z}^n \), the Saddest of all Lattices

All algorithmic tasks (encode, decode, sample) in lattice-based cryptography are reduced to \( \mathbb{Z} \) or \( \mathbb{Z}^n \).

Yet, geometrically (packing, covering, . . .) \( \mathbb{Z}^n \) is the worst lattice.

There are so many more!

Root lattices\(^3\)

---

\(^3\)If you ever deal with prime cyclotomics rings, please read https://www.math.leidenuniv.nl/scripties/BachVanWoerden.pdf
A Diversity of Lattices

$\mathbb{Z}^n$, the Saddest of all Lattices

All algorithmic tasks (encode, decode, sample) in lattice-based cryptography are reduced to $\mathbb{Z}$ or $\mathbb{Z}^n$. Yet, geometrically (packing, covering, ... ) $\mathbb{Z}^n$ is the worst lattice.

There are so many more !

Root lattices, Leech lattice, Construction D lattices, Barnes-Well lattices, Craig’s lattices, Schnorr-Adleman lattices, Chor-Rivest lattices, Mordell-Weil lattices, ...

http://www.math.rwth-aachen.de/~Gabriele.Nebe/LATTICES/

---

If you ever deal with prime cyclotomics rings, please read https://www.math.leidenuniv.nl/scripties/BachVanWoerden.pdf
Cryptography Strives in Diversity!

Lattice-based Cryptography needs:
- More diversity of Backgrounds
- More diversity of Point of View
- More diversity of Goals
- More diversity of People!
Lattice-based Cryptography needs:

- More diversity of Backgrounds
- More diversity of Point of View
- More diversity of Goals
- More diversity of People!

Thank You!