Lattices

Maria Eichlseder

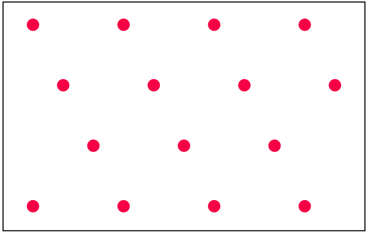
Partially based on slides by Mario Lamberger

Applied Cryptography 2 – ST 2020

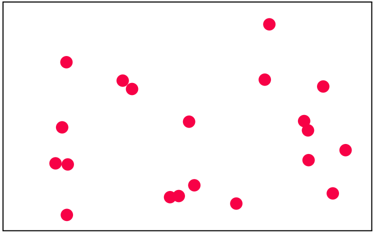
SCIENCE
PASSION
TECHNOLOGY

Outline

Introduction to Lattices



A lattice



Not a lattice

Lattices

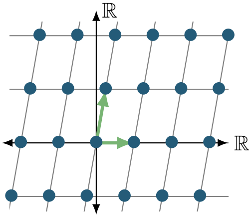
Definition:

A subset $\Lambda \subseteq \mathbb{R}^n$ is called lattice if there exist $\mathbb{R}$-linearly independent basis vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{R}^n$ such that

$$\Lambda = \mathbb{Z}\boldsymbol{b}_1 + \cdots + \mathbb{Z}\boldsymbol{b}_d = \left\{ \sum_{i=1}^{d} z_i \boldsymbol{b}_i \mid z_i \in \mathbb{Z} \right\}.$$

Example:

$$\boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \boldsymbol{b}_2 = \begin{pmatrix} 1/4 \\ \sqrt{2} \end{pmatrix}$$



Lattices in the wild: Examples I

Lattices in the wild: Examples II

Solutions of homogeneous integer equations

Let $A \in \mathbb{Z}^{d \times n}$. Consider the system of linear equations

$$A \cdot \boldsymbol{x} = \boldsymbol{0}.$$

The set of integer solutions $\{\boldsymbol{x} \in \mathbb{Z}^n \mid A \cdot \boldsymbol{x} = \boldsymbol{0}\}$ forms a lattice.

Solutions of modular equations in several variables

Let $\gcd(a_1 \ldots a_d) = 1$, $N \in \mathbb{N}$. The solutions $(x_1 \ldots x_d) \in \mathbb{Z}^d$ to

$$a_1 x_1 + \cdots + a_d x_d \equiv 0 \bmod N$$

form a d-dimensional lattice.

Lattice bases

- Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ be a basis of a lattice $\Lambda \subseteq \mathbb{R}^n$.
  Represent basis by matrix B of row vectors:

$$B = \begin{pmatrix} \boldsymbol{b}_1 \\ \vdots \\ \boldsymbol{b}_d \end{pmatrix} = \begin{pmatrix} b_{11} & \ldots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{d1} & \ldots & b_{dn} \end{pmatrix}$$

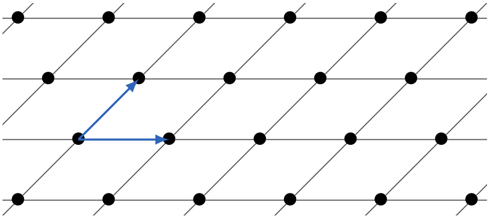
- In general, there is an infinite number of bases for a lattice.
- If $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ is a basis, another basis is

$$\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1}, \boldsymbol{b}_i + k\boldsymbol{b}_j, \boldsymbol{b}_{i+1} \ldots, \boldsymbol{b}_d \qquad i \neq j, \quad k \in \mathbb{Z}.$$

- Transition from one basis to another in general:
  multiply B with unimodular matrix M over $\mathbb{Z}$ $\quad(\det(M) = \pm 1)$

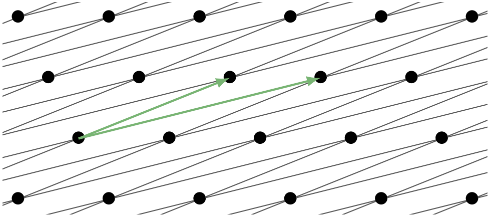
Lattice bases: Example I

$\Lambda \subseteq \mathbb{R}^2$ is generated by $\boldsymbol{b}_1 = (3, 0)$ and $\boldsymbol{b}_2 = (2, 2)$:



Lattice bases: Example II

Λ is also generated by $\boldsymbol{b}_1' = (8, 2)$ and $\boldsymbol{b}_2' = (5, 2)$:



Observation:

$$\begin{pmatrix} 3 & 0 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} 8 & 2 \\ 5 & 2 \end{pmatrix}$$

Lattice volume

Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{R}^n$ be a lattice basis. The Gram matrix $G \in \mathbb{R}^{d \times d}$ is defined as

$$G = B \cdot B^t = \begin{pmatrix} \langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle & \ldots & \langle \boldsymbol{b}_1, \boldsymbol{b}_d \rangle \\ \vdots & \ddots & \vdots \\ \langle \boldsymbol{b}_d, \boldsymbol{b}_1 \rangle & \ldots & \langle \boldsymbol{b}_d, \boldsymbol{b}_d \rangle \end{pmatrix}$$

Lattice volume

The volume or determinant of a lattice is

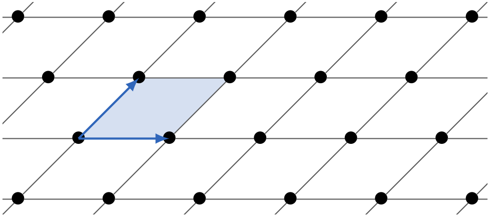
$$\mathrm{vol}(\Lambda) = \sqrt{\det(G)},$$

with G the Gram matrix for an arbitrary basis of Λ.

If B is already a square matrix ($d = n$), then this is simply $\mathrm{vol}(\Lambda) = \det(B)$.

Lattice volume: Example I

$\Lambda = (3,0)\mathbb{Z} + (2,2)\mathbb{Z} \subseteq \mathbb{Z}^2$:

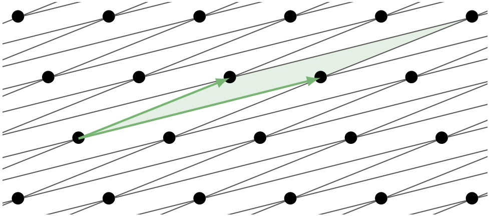


$$G = \begin{pmatrix} 3 & 0 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} 3 & 2 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 9 & 6 \\ 6 & 8 \end{pmatrix}.$$

Thus, $\mathrm{vol}(\Lambda) = \sqrt{\det(G)} = \sqrt{36} = 6 \quad (= \det(B))$.

Lattice volume: Example II

The same lattice is generated as $\Lambda' = (8,2)\mathbb{Z} + (5,2)\mathbb{Z} \subseteq \mathbb{Z}^2$:



$$G' = \begin{pmatrix} 8 & 2 \\ 5 & 2 \end{pmatrix} \begin{pmatrix} 8 & 5 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 68 & 44 \\ 44 & 29 \end{pmatrix}.$$

G' has the same determinant: $\mathrm{vol}(\Lambda') = \sqrt{\det(G')} = \sqrt{36} = 6 \quad (= \det(B'))$.

Lattices in cryptography

Applications of lattices in cryptography

Design of new cryptosystems from lattice problems SVP, CVP, BDD, LWE:

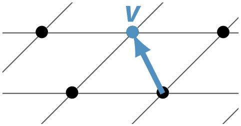
- Post-quantum public-key crypto[Seminar]: GGH ☠, NTRU, NIST PQC candidates, ...
- Provably "secure" hash functions: SWIFFT ☠
- Fully homomorphic encryption[Seminar]

Analysis of other cryptosystems such as ECDSA, RSA, ...:

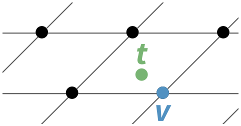
- Proof ingredient: factoring $N = pq$ is equivalent to knowing d
- Coppersmith's attack on RSA
- Bleichenbacher's attack[Lecture] on PKCS#1 v1.5

NP-hard problems: Given a lattice Λ, ...

- Shortest Vector Problem SVP: find <u>shortest</u> non-zero $\boldsymbol{v} \in \Lambda$



- Closest Vector Problem CVP: find $\boldsymbol{v} \in \Lambda$ <u>closest</u> to some given target $\boldsymbol{t}$



- Bounded Distance Decoding BDD: find <u>all</u> $\boldsymbol{v} \in \Lambda$ <u>close</u> to $\boldsymbol{t}$
- Learning with Errors LWE (many variants, related to BDD)

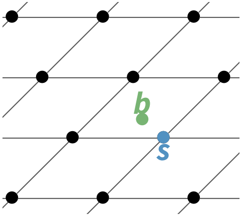
The Learning-with-Errors (LWE) problem (informally)

Solve a system of noisy linear equations with secret solution $\boldsymbol{s}$
(d unknowns, arbitrary number n of noisy equations mod p):

$$
\begin{aligned}
a_1^{(1)} \cdot s_1 + \ldots + a_d^{(1)} \cdot s_d &\approx b^{(1)} \quad \bmod p \\
a_1^{(2)} \cdot s_1 + \ldots + a_d^{(2)} \cdot s_d &\approx b^{(2)} \quad \bmod p \\
&\vdots \\
a_1^{(n)} \cdot s_1 + \ldots + a_d^{(n)} \cdot s_d &\approx b^{(n)} \quad \bmod p
\end{aligned}
$$

In lattice terms:

- Lattice Λ with basis $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d$
- Secret lattice point $\boldsymbol{s} = \sum s_i \boldsymbol{a}_i$
- Gaussian error vector $\boldsymbol{e}$
- Given Λ and $\boldsymbol{b} = \boldsymbol{s} + \boldsymbol{e}$, find $\boldsymbol{s}$.



Example: Simple LWE public-key encryption (Regev)

- Alice's keypair: private $\boldsymbol{s}$, public n noisy equations $(\boldsymbol{a}^{(i)}, b^{(i)})$:

$$a_1^{(i)} \cdot s_1 + \ldots + a_d^{(i)} \cdot s_d \approx b^{(i)} \mod p$$

1. Bob wants to send an encrypted plaintext bit x to Alice
2. He selects and sums a random subset σ of the n equations to produce a new noisy equation $(\boldsymbol{a}^{(\sigma)}, b^{(\sigma)})$:

$$a_1^{(\sigma)} \cdot s_1 + \ldots + a_d^{(\sigma)} \cdot s_d \approx b^{(\sigma)} \mod p$$

3. He sends $(\boldsymbol{a}^{(\sigma)}, c^{(\sigma)})$ to Alice, where

$$c^{(\sigma)} = \begin{cases} b^{(\sigma)} & \text{if bit } x = 0 \\ b^{(\sigma)} + \left\lfloor \frac{p}{2} \right\rfloor & \text{if bit } x = 1 \end{cases}$$

4. Alice uses $\boldsymbol{s}$ to check if $(\boldsymbol{a}^{(\sigma)}, c^{(\sigma)})$ is roughly correct ($x = 0$) or very wrong ($x = 1$)

Properties of lattice-based cryptosystems

- Use finite-field versions of lattices (instead of $\mathbb{R}^n$)

✓ Post-quantum security:
No known quantum algorithms faster than classical algorithms

✓ Worst-case hardness:
Breaking cryptosystem implies solving **any** problem instance

✗ Lattice-based designs often fall in one of two classes:

- Provably secure but not so efficient in practice (large keys)
- Very efficient but not provably secure / well-analyzed

✗ A history of (somewhat) broken schemes...

NIST's Post-Quantum Crypto Competition (PQC) – Round ~~1~~ 2

Submissions	Signatures	KEM/Encryption	Total
Lattice-based	~~5~~ **3**	~~21~~ **9**	~~26~~ **12**
Code-based	~~3~~ 0	~~18~~ 7	~~21~~ 7
Multivariate	~~9~~ 4	~~4~~ 0	~~11~~ 4
Hash-based	~~2~~ 1		~~2~~ 1
Other (Isogeny, . . .)	~~1~~ 1	~~6~~ 1	~~8~~ 2
Total	~~22~~ 9	~~49~~ 17	~~68~~ 26

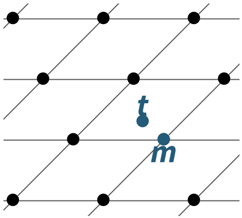
Lattice-based examples:
NTRU variants, NewHope, CRYSTALS (KYBER/DILITHIUM), Frodo. . .

https://csrc.nist.gov/Projects/Post-Quantum-Cryptography

https://www.safecrypto.eu/pqclounge/

Lattice problems & reduced bases

Babai's rounding technique to approximate CVP



Closest Vector Problem (CVP)

In lattice Λ of dimension d with basis matrix $\boldsymbol{B}$:

Given $\boldsymbol{t}$, find the closest $\boldsymbol{m} \in \Lambda$.

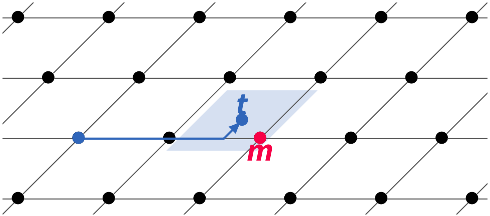
Babai's rounding technique can find an approximate solution ("reasonably close $\tilde{\boldsymbol{m}}$") with approximation factor $1 + 2d(\frac{9}{2})^{d/2}$:

1. Reduce the lattice basis $\boldsymbol{B}$ with LLL to get $\boldsymbol{B}'$
2. Solve $\boldsymbol{x} \cdot \boldsymbol{B}' = \boldsymbol{t}$ over $\mathbb{R}^d$
3. A lattice vector $\tilde{\boldsymbol{m}}$ close to $\boldsymbol{t}$ is obtained by rounding:

$$\tilde{\boldsymbol{m}} = \lfloor \boldsymbol{x} \rceil \cdot \boldsymbol{B}' = (\lfloor x_1 \rceil, \ldots, \lfloor x_d \rceil) \cdot \boldsymbol{B}'$$

Babai's rounding technique: Example I

$\Lambda = (3,0)\mathbb{Z} + (2,2)\mathbb{Z} \subseteq \mathbb{Z}^2$, target vector $\boldsymbol{t} = (5.4, 0.6)$:

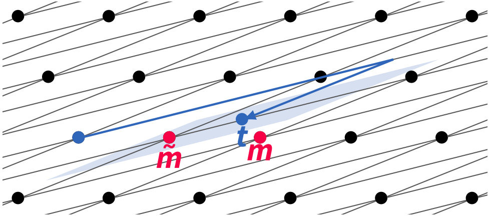


2 Find $\boldsymbol{x}$ such that $\boldsymbol{x} \cdot \boldsymbol{B} = \boldsymbol{t}$: $\quad (1.6, 0.3) \cdot \begin{pmatrix} 3 & 0 \\ 2 & 2 \end{pmatrix} = (5.4, 0.6)$

3 Approximate $\tilde{\boldsymbol{m}} = \lfloor \boldsymbol{x} \rceil \cdot \boldsymbol{B} = (2,0) \cdot \begin{pmatrix} 3 & 0 \\ 2 & 2 \end{pmatrix} = (6,0) = \boldsymbol{m}$

Babai's rounding technique: Example II

$\Lambda = (8,2)\mathbb{Z} + (5,2)\mathbb{Z} \subseteq \mathbb{Z}^2$, target vector $\boldsymbol{t} = (5.4, 0.6)$:



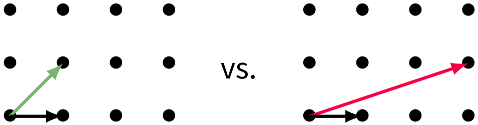
2 Find $\boldsymbol{x}$ such that $\boldsymbol{x} \cdot \boldsymbol{B} = \boldsymbol{t}$: $\quad (1.3, -1) \cdot \begin{pmatrix} 8 & 2 \\ 5 & 2 \end{pmatrix} = (5.4, 0.6)$

3 Approximate $\tilde{\boldsymbol{m}} = \lfloor \boldsymbol{x} \rceil \cdot \boldsymbol{B} = (1, -1) \cdot \begin{pmatrix} 8 & 2 \\ 5 & 2 \end{pmatrix} = (3, 0) \neq \boldsymbol{m}$

$\rightarrow$ *This only works well if **B** is a "nice" basis!*

How "nice" is a particular basis for a lattice?

- Short vectors are nice:

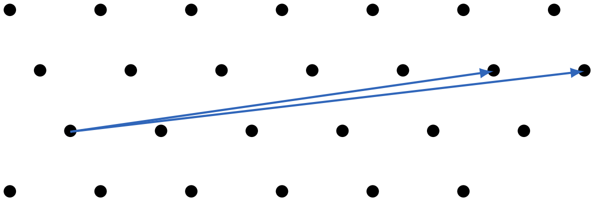


- (Near-)Orthogonality ($\langle \boldsymbol{x}, \boldsymbol{y} \rangle \approx 0$ for $\boldsymbol{x} \neq \boldsymbol{y}$) is nice:

Question: How to find a better basis?

Reduced lattice bases: Example

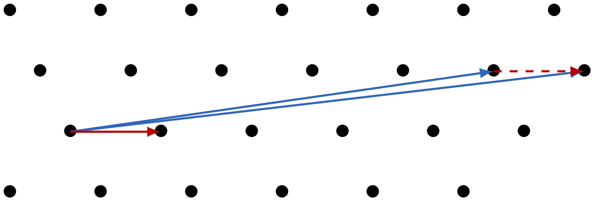
If $\dim(\Lambda) = 2$, Euclid does the trick:



Connection to continued fractions!

Reduced lattice bases: Example

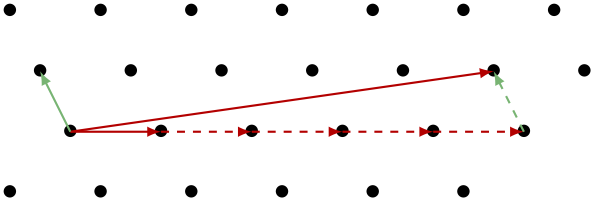
If $\dim(\Lambda) = 2$, Euclid does the trick:



Connection to continued fractions!

Reduced lattice bases: Example

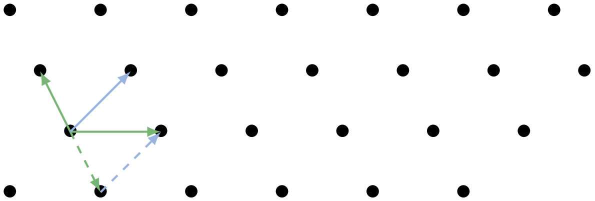
If $\dim(\Lambda) = 2$, Euclid does the trick:



Connection to continued fractions!

Reduced lattice bases: Example

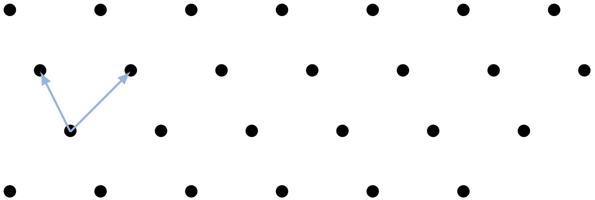
If $\dim(\Lambda) = 2$, Euclid does the trick:



Connection to continued fractions!

Reduced lattice bases: Example

If $\dim(\Lambda) = 2$, Euclid does the trick:



Connection to continued fractions!

Orthogonality

How to measure the quality of a basis in terms of orthogonality?

Orthogonality defect

The orthogonality defect of a basis $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d$ of a lattice Λ is

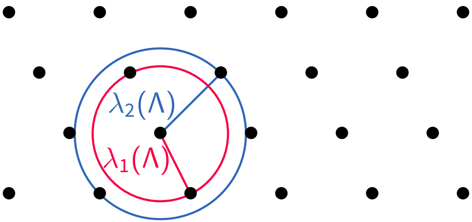
$$\operatorname{def}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d) = \frac{\|\boldsymbol{a}_1\| \cdots \|\boldsymbol{a}_d\|}{\operatorname{vol}(\Lambda)} \geq 1.$$

The larger the orthogonality defect, the less orthogonal the basis!

Short vectors

Radius $\lambda_i(\Lambda)$

Let $\Lambda \subseteq \mathbb{R}^n$ be a d-dimensional lattice. For $i \leq d$, $\lambda_i(\Lambda)$ is the minimum radius r such that $B(\mathbf{0}, r) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \|\boldsymbol{x}\| \leq r\}$ contains i linearly independent lattice vectors.



Reduced lattice bases

Different definitions:

- Informally: Basis of "short" and "nearly orthogonal" vectors
- Ideally: Basis vectors $\boldsymbol{a}_1 \ldots \boldsymbol{a}_d$ have lengths $\lambda_1 \ldots \lambda_d$

  Not clear how to compute for $d \geq 5$!
- Minkowski, HKZ: Very strong reduction of vector lengths.
- LLL, BKZ: Weaker definitions, better to calculate

LLL is implemented in most computer algebra systems!

In $\mathbb{R}^n$, we know how to get nice bases: Gram-Schmidt algorithm

Reminder: Gram-Schmidt orthogonalization in $\mathbb{R}^n$

Let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d \in \mathbb{R}^n$ be linearly independent vectors. Compute:

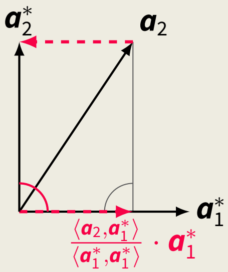
$$\boldsymbol{a}_1^* = \boldsymbol{a}_1$$

$$\boldsymbol{a}_2^* = \boldsymbol{a}_2 - \frac{\langle \boldsymbol{a}_2, \boldsymbol{a}_1^* \rangle}{\langle \boldsymbol{a}_1^*, \boldsymbol{a}_1^* \rangle} \cdot \boldsymbol{a}_1^*$$

$$\boldsymbol{a}_3^* = \boldsymbol{a}_3 - \frac{\langle \boldsymbol{a}_3, \boldsymbol{a}_1^* \rangle}{\langle \boldsymbol{a}_1^*, \boldsymbol{a}_1^* \rangle} \cdot \boldsymbol{a}_1^* - \frac{\langle \boldsymbol{a}_3, \boldsymbol{a}_2^* \rangle}{\langle \boldsymbol{a}_2^*, \boldsymbol{a}_2^* \rangle} \cdot \boldsymbol{a}_2^*$$

$$\ldots$$

$$\boldsymbol{a}_i^* = \boldsymbol{a}_i - \sum_{1 \le j < i} \frac{\langle \boldsymbol{a}_i, \boldsymbol{a}_j^* \rangle}{\langle \boldsymbol{a}_j^*, \boldsymbol{a}_j^* \rangle} \cdot \boldsymbol{a}_j^*$$



LLL algorithm (Lenstra, Lenstra, Lovász)

- Goal: "reduce" basis $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d$ of Λ
  - more orthogonal
  - shorter vectors
- Inspiration from $\mathbb{R}^n$: Gram-Schmidt orthogonalization
  - Get "as close as possible" to ideal GS values $\boldsymbol{a}_1^* \ldots \boldsymbol{a}_d^*$
  - $\|\boldsymbol{a}_i\|^2 = \|\boldsymbol{a}_i^*\|^2 + \sum_{j<i} \mu_{ij}^2 \|\boldsymbol{a}_j^*\|^2 \to$ change $\boldsymbol{a}_i$ to minimize μ_{ij}
- Idea: Investigate effect of standard basis changes:
  - Replacing $\boldsymbol{a}_i \leftarrow \boldsymbol{a}_i + m \cdot \boldsymbol{a}_j$
  - Swapping $\boldsymbol{a}_{i-1} \leftrightarrow \boldsymbol{a}_i$

LLL algorithm

LLL Algorithm

Let $\frac{1}{4} < c < 1$ be a constant (usually $c = \frac{3}{4}$).
Let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d$ be the basis of a lattice $\Lambda \subseteq \mathbb{R}^n$.
Let $\boldsymbol{a}_1^*, \ldots, \boldsymbol{a}_d^*$ be the GS vectors, $\mu_{ij} = \frac{\langle \boldsymbol{a}_i, \boldsymbol{a}_j^* \rangle}{\langle \boldsymbol{a}_j^*, \boldsymbol{a}_j^* \rangle}$ ($\star$ update here!)

$i \leftarrow 2$
while $i < d$ **do**
  for $j = i - 1$ **to** 1 **do**
    if $|\mu_{ij}| > \frac{1}{2}$: Replace $\boldsymbol{a}_i \leftarrow \boldsymbol{a}_i - \lfloor \mu_{ij} \rceil \boldsymbol{a}_j$ $\star$
  if $\|\mu_{i,i-1}\boldsymbol{a}_{i-1}^* + \boldsymbol{a}_i^*\|^2 < c\|\boldsymbol{a}_{i-1}^*\|^2$: Swap $\boldsymbol{a}_{i-1} \leftrightarrow \boldsymbol{a}_i$ $\star$, $i \leftarrow i - 1$
  else: $i \leftarrow i + 1$

LLL algorithm: Properties

A lattice basis is LLL-reduced if the algorithm leaves it unchanged:

Theorem

Let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d$ be an LLL-reduced lattice basis in $\mathbb{R}^n$. Then:

1. Orthogonality defect: $1 \leq \operatorname{def}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_d) \leq 2^{\frac{d(d-1)}{4}}$.
2. Short vectors: $\|\boldsymbol{a}_1\| \leq 2^{\frac{d-1}{2}} \lambda_1(\Lambda)$.

Worst-case running time: $\mathcal{O}(d^5 n \log^3(B))$ if $\|\boldsymbol{a}_i\|^2 \leq B$ for all i.

Bleichenbacher’s attack

PKCS1-V1_5 Encryption

PKCS#1 v1.5 padding for RSA

1. Generate $(|n| - |m|)/8 - 3 \geq 8$ non-zero `random bytes`
2. | 00 | 02 | `random bytes` | 00 | `message` m |
   |---|---|---|---|---|
3. Convert to $\mathbb{Z}_n$ and encrypt with RSA

- Intuitive "ad hoc" design, no proof
- Rationale: Randomness required for semantic security
- Decryption: Error if $m = c^d \bmod n$ is not of this format, i.e., c is not "PKCS-conforming (PKCSc)"

Bleichenbacher's Attack

- Goal: Recover message m from $c = m^e \mod n$
- Adaptive chosen-ciphertext attack:
  - Many SSLv3.0 servers behave like "PKCSc oracle"
  - Attacker adapts queries based on c and previous answers
  - Based on high-dimension lattices
- Practical setting, practical complexity

Bleichenbacher's Attack: Lattice Version

- Goal: find m, given $c = m^e \mod n$.
- Generate modified ciphertexts $c'_i = cs_i^e \mod n$
  - Compute $c' = cs^e \mod n$ (ciphertext of $m' = ms$) for random s
  - c' is accepted by "PKCSc oracle" with probability $\approx 2^{-16}$
  - Repeat to get N accepted values with $s_1, \dots, s_N$
- Approximate plaintexts $m'_i = ms_i \mod n$:
  - If we know m'_i for some i, we can recover m.
  - However, we only know "approximations" of $m'_i = 00\ 02\ ?\dots ?$:

$$2A \leq m'_i < 3A \qquad \text{or} \qquad |m'_i - 2.5A| \leq 0.5A,$$

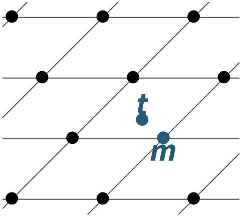
where $A = \boxed{00}\boxed{01}\boxed{00}\boxed{\cdots}\boxed{00} = 2^{|n|-16} \approx n \cdot 2^{-16}$.
  - "Hidden Number Problem"

Solving the Hidden Number Problem via CVP I

Idea: write as Closest Vector Problem in a lattice Λ

Lattice Λ spanned by rows of the basis matrix $\boldsymbol{B}$:

$$\boldsymbol{B} = \begin{pmatrix} 2^{-16} & s_1 & s_2 & \dots & s_N \\ 0 & n & 0 & \dots & 0 \\ 0 & 0 & n & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & \dots & 0 & n \end{pmatrix}$$



Unknown lattice vector $\boldsymbol{m} = \begin{pmatrix} m \cdot 2^{-16} & m'_1 & m'_2 & \dots & m'_N \end{pmatrix}$

Known target vector $\boldsymbol{t} = \begin{pmatrix} 0.5A & 2.5A & 2.5A & \dots & 2.5A \end{pmatrix}$

Solving the Hidden Number Problem via CVP II

- The vectors' distance is

$$\|\boldsymbol{m} - \boldsymbol{t}\| \le \sqrt{N+1} \cdot 0.5A \le \sqrt{N+1} \cdot n \cdot 2^{-16}.$$

- An average lattice distance is about

$$d_{\mathrm{avg}} \approx \sqrt{d} \cdot \mathrm{vol}(\Lambda)^{1/d} = \sqrt{N+1}(n^N \cdot 2^{-16})^{\frac{1}{N+1}}.$$

- If $\|\boldsymbol{m} - \boldsymbol{t}\| \ll d_{\mathrm{avg}}$, we expect $\boldsymbol{m}$ is the closest vector to $\boldsymbol{t}$.

- Solve CVP for $\boldsymbol{t}$ to get $\boldsymbol{m}$ with Babai's Rounding Technique!

Attack in Practice

Complexity in practice:

- 1024-bit n: need $N+1 \gg 60$ or $80 \cdot 2^{16} \approx 5.2$ million queries

Application to SSL v3 handshake (client attacks server):

1. Client sends chosen ciphertext c' as `PreMasterSecret`
2. If c' is not PKCSc, server aborts the connection
   - Failure after `ClientKeyExchange`
3. If c' is PKCSc, server continues, but attacker's reply invalid
   - Failure after `Finished`

Other scenarios: detailed error messages, timing attack, …

Conclusion

- There are various complexity-theoretically hard problems related to lattices
- Solving those problems is much easier when knowing a good basis
- No fast quantum algorithm to solve them is known

- Applications in crypto:
  - Design of post-quantum secure signatures and public-key encryption
  - Cryptanalysis tool for solving “approximate equations”

Questions you should be able to answer

1. What is a lattice? What are basic and desirable properties of a lattice basis?
2. Define the Closest Vector Problem (CVP) in a lattice. Explain Babai's algorithm to solve the CVP, and illustrate why it requires a reduced basis to work well.
3. What is an LLL-reduced lattice basis? Briefly describe the two basic steps of the LLL algorithm.
4. Explain the basic idea of the lattice version of Bleichenbacher's attack on PKCS#1.5 padding.

Bibliography

[Bab86] László Babai. **On Lovász' lattice reduction and the nearest lattice point problem**. Combinatorica 6.1 (1986), pp. 1–13. DOI: 10.1007/BF02579403.

[Ble98] Daniel Bleichenbacher. **Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1**. Advances in Cryptology – CRYPTO 1998. Vol. 1462. LNCS. Springer, 1998, pp. 1–12. DOI: 10.1007/BFb0055716.

[LLL82] Arjen K. Lenstra, Hendrik Lenstra, and László Lovász. **Factoring polynomials with rational coefficients**. Mathematische Annalen 261.4 (1982), pp. 515–534. DOI: 10.1007/BF01457454.