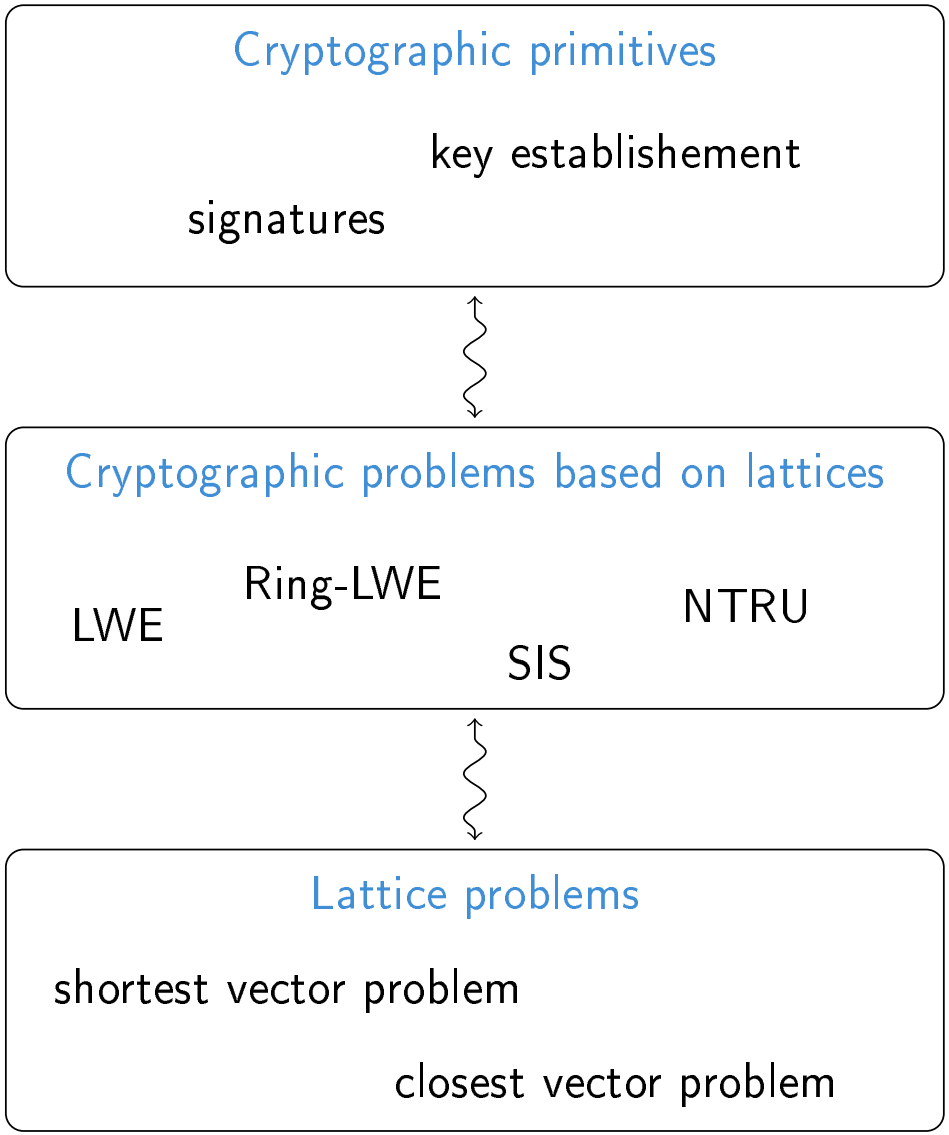
Lattice-based crypto, part 1:
Algorithmic problems over lattices

Alice Pellet-Mary

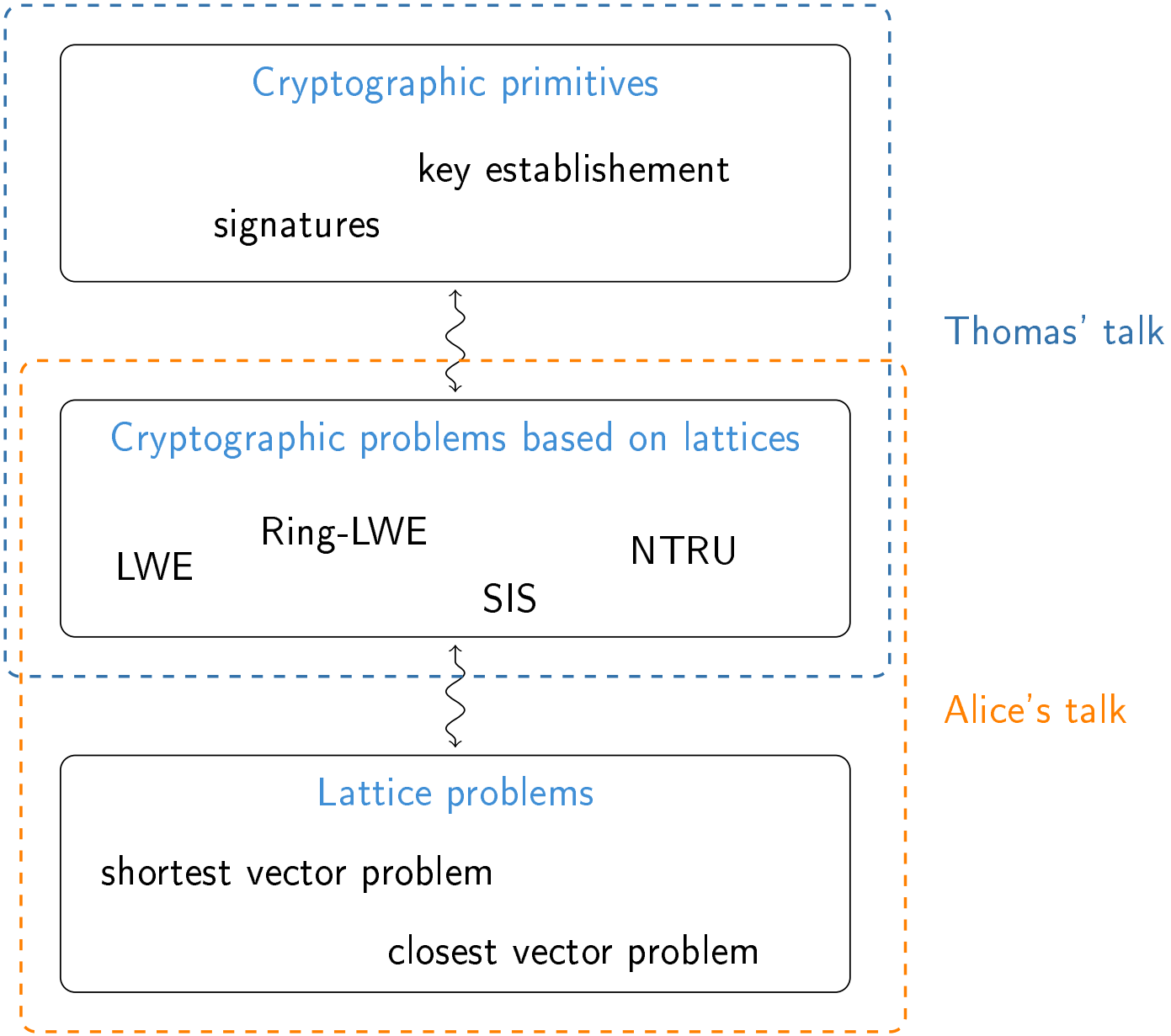
CNRS and university of Bordeaux, France

AsCrypto summer school
October 4-5, 2021

Plan of the talks



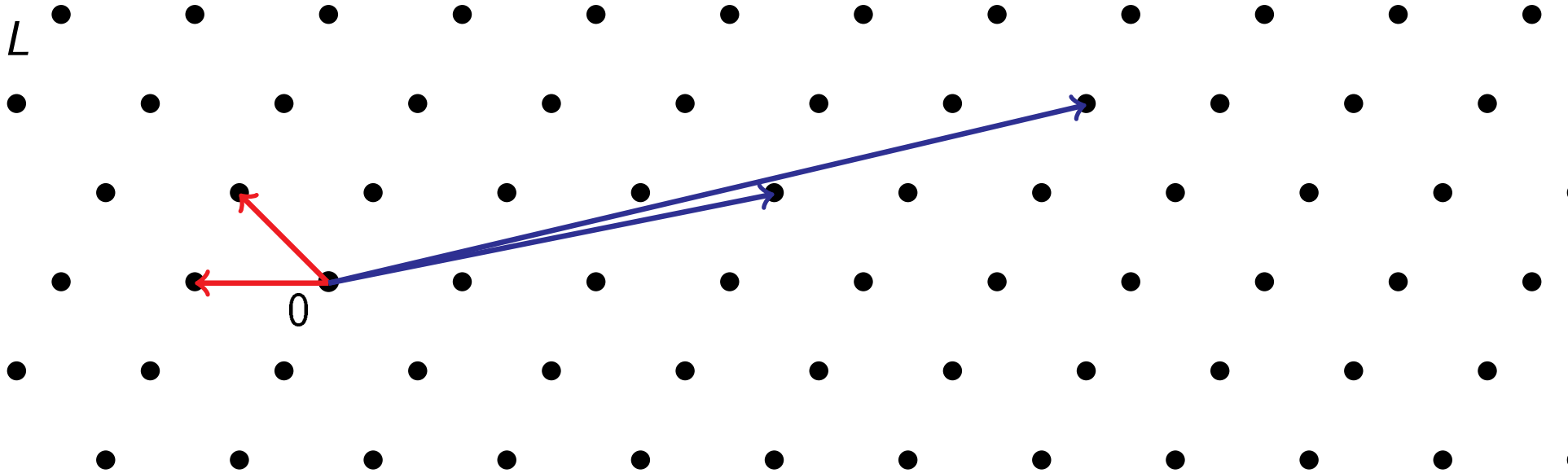
Plan of the talks



Outline of the talk

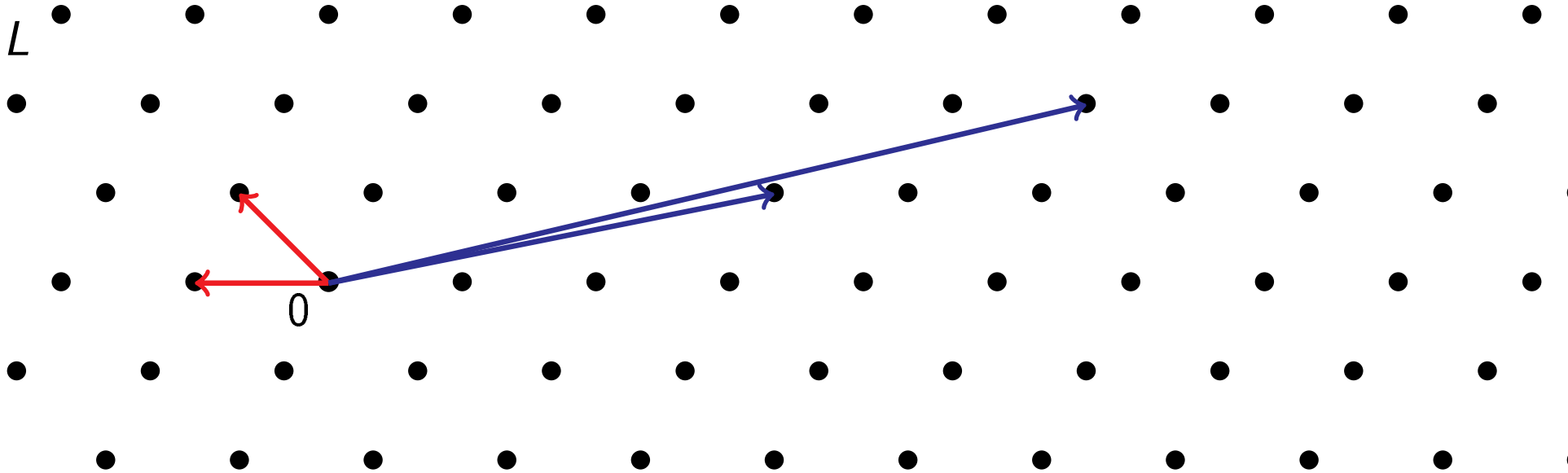
Outline of the talk

Lattices



- $L = \mathcal{L}(B) = \{Bx \mid x \in \mathbb{Z}^n\}$ is a lattice
- $B \in \mathrm{GL}_n(\mathbb{R})$ is a basis
- n is the dimension of L

Lattices



- $L = \mathcal{L}(B) = \{Bx \mid x \in \mathbb{Z}^n\}$ is a lattice
- $B \in \mathrm{GL}_n(\mathbb{R})$ is a basis
- n is the dimension of L

We represent a lattice by any of its basis

Algorithmic problems on lattices

Input: any basis of any lattice

Example of problems:

(1) Testing equality of lattices

(2) Testing inclusion of lattices

(3) Intersecting two lattices

(4) Computing a short vector of a lattice

(5) Computing a lattice vector close to a target

Algorithmic problems on lattices

Input: any basis of any lattice

Example of problems:

(1) Testing equality of lattices

(2) Testing inclusion of lattices

(3) Intersecting two lattices

(4) Computing a short vector of a lattice

(5) Computing a lattice vector close to a target

Quiz: which ones are easy or hard?

easy: polynomial time hard: no polynomial time algorithm known

Algorithmic problems on lattices

Input: any basis of any lattice

Example of problems:

(1) Testing equality of lattices $\Rightarrow$ easy

(2) Testing inclusion of lattices $\Rightarrow$ easy

(3) Intersecting two lattices $\Rightarrow$ easy

(4) Computing a short vector of a lattice $\Rightarrow$ hard

(5) Computing a lattice vector close to a target $\Rightarrow$ hard

Quiz: which ones are easy or hard?

easy: polynomial time hard: no polynomial time algorithm known

Testing inclusion / equality

Exercise

Given $B_1, B_2 \in \mathrm{GL}_n(\mathbb{R})$, how do you test if

1. $\mathcal{L}(B_1) \subseteq \mathcal{L}(B_2)$
2. $\mathcal{L}(B_1) = \mathcal{L}(B_2)$

Testing inclusion / equality

Exercise

Given $B_1, B_2 \in \mathrm{GL}_n(\mathbb{R})$, how do you test if

1. $\mathcal{L}(B_1) \subseteq \mathcal{L}(B_2)$
2. $\mathcal{L}(B_1) = \mathcal{L}(B_2)$

Solution:

1.

$$\mathcal{L}(B_1) \subseteq \mathcal{L}(B_2) \Leftrightarrow B_1 = B_2 \cdot X \quad \text{for some } X \in \mathbb{Z}^{n \times n}$$
$$\Leftrightarrow B_1 \cdot B_2^{-1} \in \mathbb{Z}^{n \times n}$$

Testing inclusion / equality

Exercise

Given $B_1, B_2 \in \mathrm{GL}_n(\mathbb{R})$, how do you test if

1. $\mathcal{L}(B_1) \subseteq \mathcal{L}(B_2)$
2. $\mathcal{L}(B_1) = \mathcal{L}(B_2)$

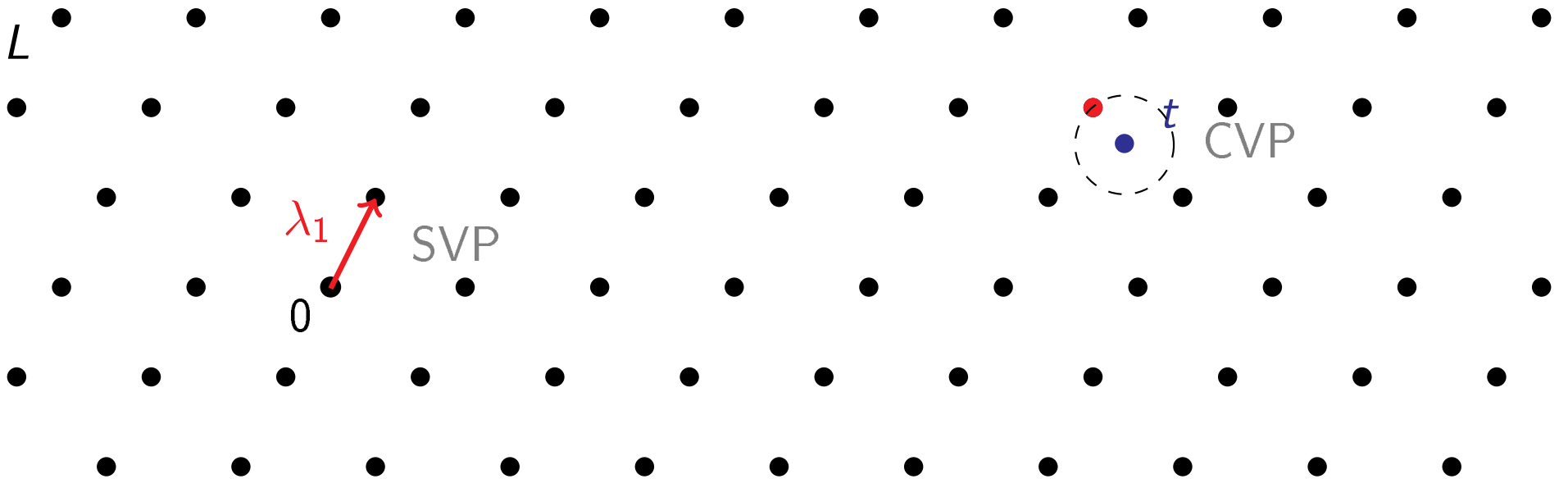
Solution:

1.

$$\mathcal{L}(B_1) \subseteq \mathcal{L}(B_2) \Leftrightarrow B_1 = B_2 \cdot X \quad \text{for some } X \in \mathbb{Z}^{n \times n}$$
$$\Leftrightarrow B_1 \cdot B_2^{-1} \in \mathbb{Z}^{n \times n}$$

2. $\mathcal{L}(B_1) = \mathcal{L}(B_2) \Leftrightarrow \mathcal{L}(B_1) \subseteq \mathcal{L}(B_2)$ and $\mathcal{L}(B_2) \subseteq \mathcal{L}(B_1)$

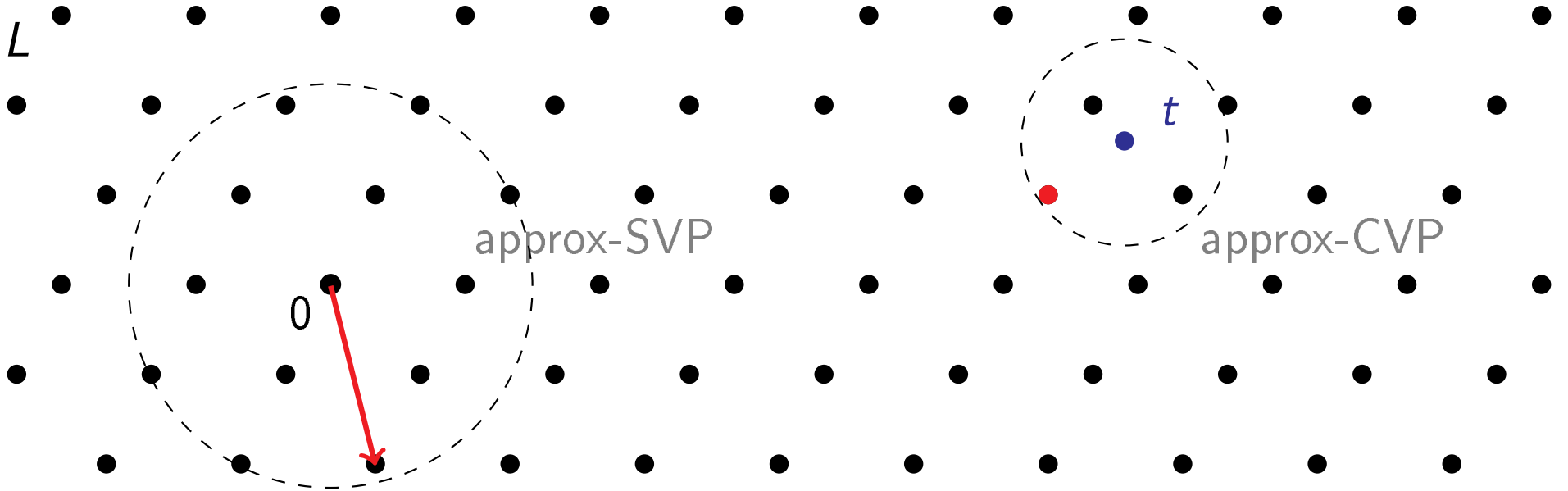
(Hard) Lattice problems



SVP : Shortest Vector Problem

CVP : Closest Vector Problem

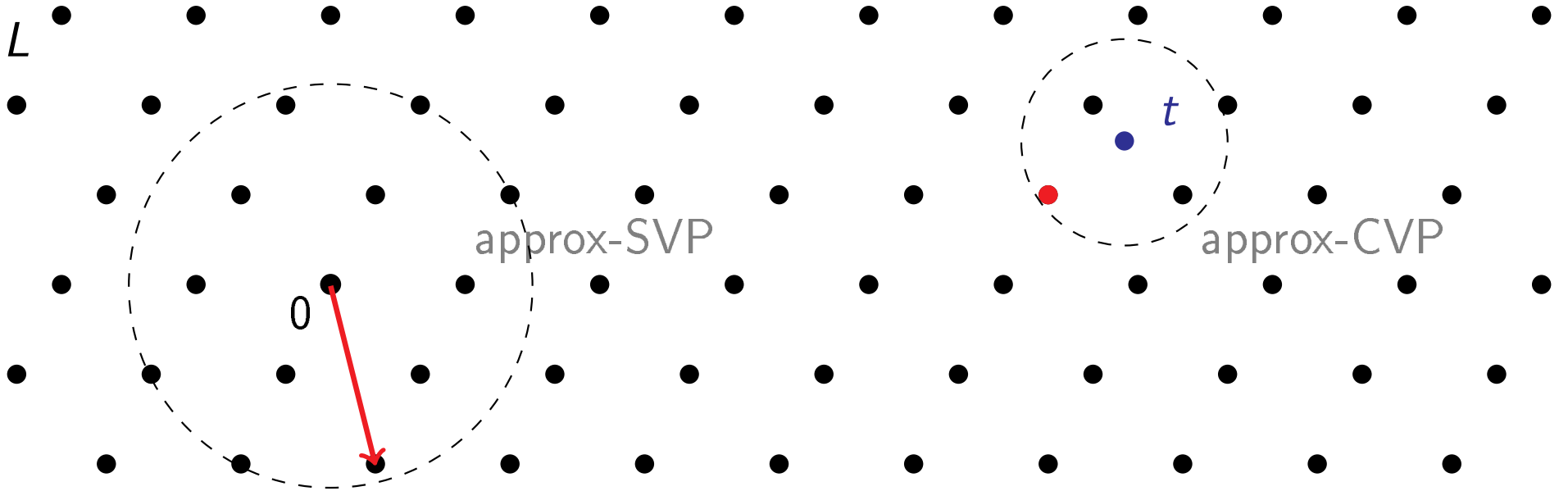
(Hard) Lattice problems



approx-SVP : Shortest Vector Problem

approx-CVP : Closest Vector Problem

(Hard) Lattice problems



approx-SVP : Shortest Vector Problem approx-CVP : Closest Vector Problem

Supposedly hard to solve when n is large (input: a bad basis of L)

- even with a quantum computer
- even with a small approximation factor $(\mathrm{poly}(n))$

SVP and CVP in dimension 2 – Exercise

Exercise

Let $B = \begin{pmatrix} 13 & 10 \\ 2 & 2 \end{pmatrix}$ and $t = \begin{pmatrix} 20 \\ -1 \end{pmatrix}$

1. Solve SVP in $\mathcal{L}(B)$

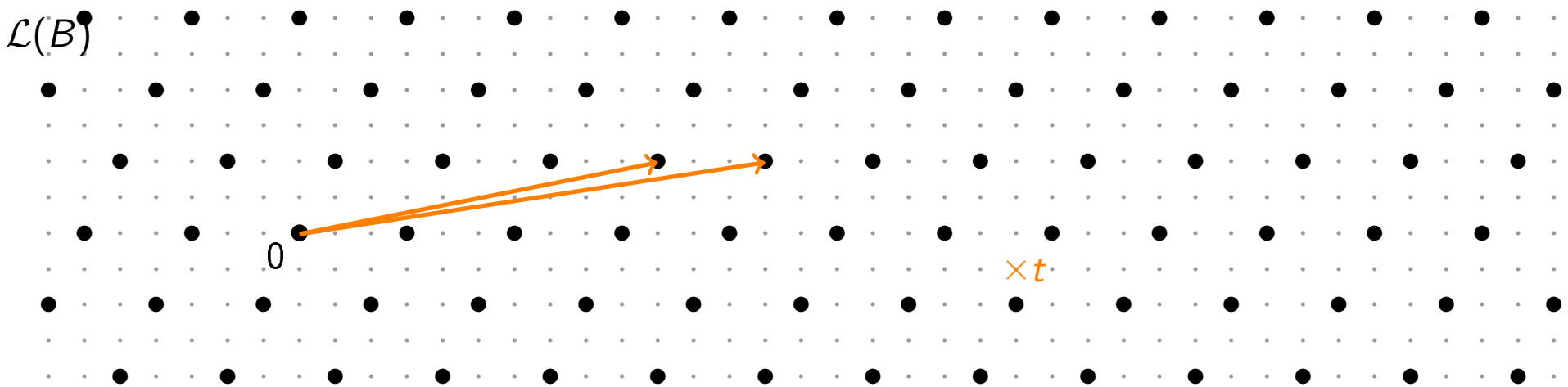
2. Solve CVP in $\mathcal{L}(B)$ with target t

SVP and CVP in dimension 2 – Exercise

Exercise

Let $B = \begin{pmatrix} 13 & 10 \\ 2 & 2 \end{pmatrix}$ and $t = \begin{pmatrix} 20 \\ -1 \end{pmatrix}$

1. Solve SVP in $\mathcal{L}(B)$
2. Solve CVP in $\mathcal{L}(B)$ with target t

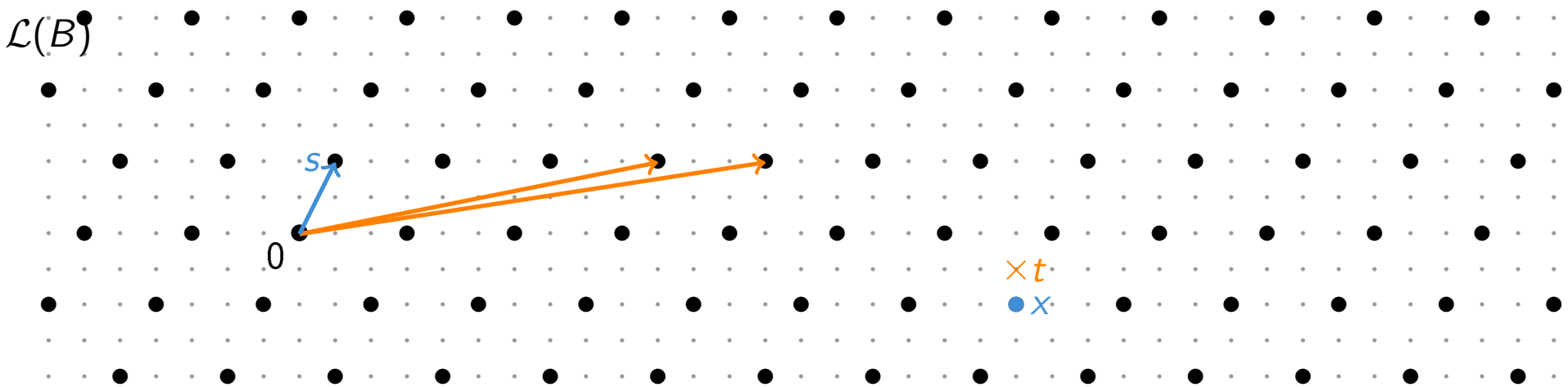


SVP and CVP in dimension 2 – Exercise

Exercise

Let $B = \begin{pmatrix} 13 & 10 \\ 2 & 2 \end{pmatrix}$ and $t = \begin{pmatrix} 20 \\ -1 \end{pmatrix}$

1. Solve SVP in $\mathcal{L}(B) \Rightarrow s = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$
2. Solve CVP in $\mathcal{L}(B)$ with target $t \Rightarrow x = \begin{pmatrix} 20 \\ -2 \end{pmatrix}$



Algorithm for solving SVP in dimension 2

The Lagrange-Gauss algorithm:

- For lattices of rank $n = 2$ only
- Solves exact SVP
- Polynomial time

Algorithm for solving SVP in dimension 2

The Lagrange-Gauss algorithm:

- For lattices of rank $n = 2$ only
- Solves exact SVP
- Polynomial time

video

Algorithm for solving SVP in dimension 2

The Lagrange-Gauss algorithm:

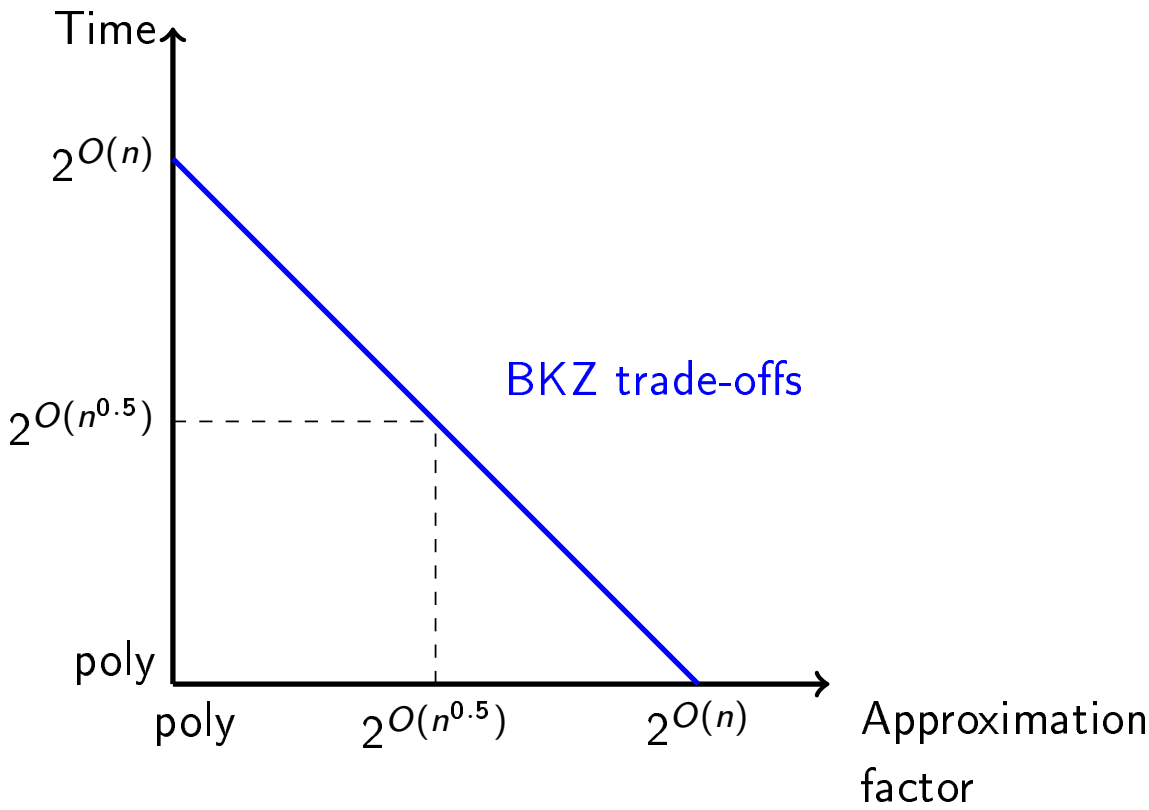
- For lattices of rank $n = 2$ only
- Solves exact SVP
- Polynomial time

video

But remember: when n is large, solving exact SVP is hard

Asymptotic hardness of SVP and CVP

Best Time/Approximation trade-off for SVP, CVP (even quantumly):
BKZ algorithm [Sch87,SE94]

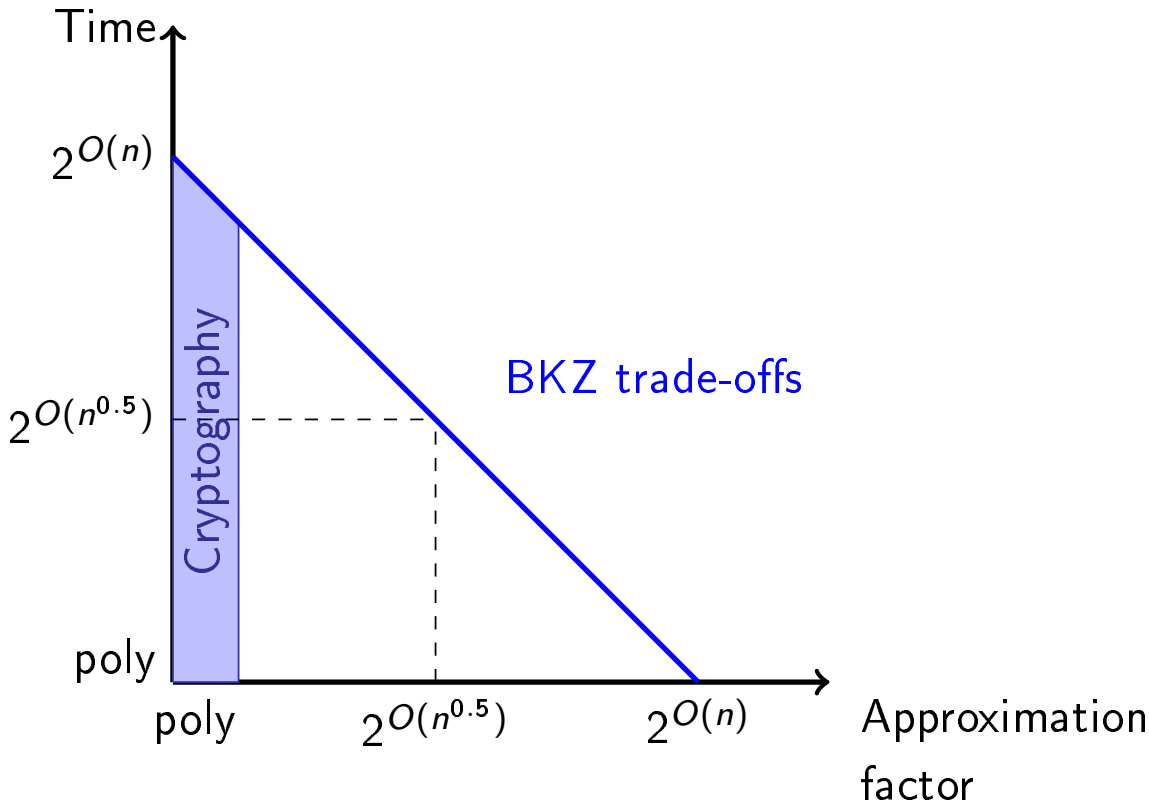


[Sch87] C.-P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. TCS.

[SE94] C.-P. Schnorr and M. Euchner. Lattice basis reduction: improved practical algorithms and solving subset sum problems. Mathematical programming.

Asymptotic hardness of SVP and CVP

Best Time/Approximation trade-off for SVP, CVP (even quantumly):
BKZ algorithm [Sch87,SE94]



[Sch87] C.-P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. TCS.

[SE94] C.-P. Schnorr and M. Euchner. Lattice basis reduction: improved practical algorithms and solving subset sum problems. Mathematical programming.

Exact SVP in practice

- $n = 2 \rightsquigarrow$ easy, very efficient in practice

Exact SVP in practice

- $n = 2 \rightsquigarrow$ easy, very efficient in practice
- up to $n = 80$ or $n = 100 \rightsquigarrow$ a few minutes on a personal laptop

Exact SVP in practice

- $n = 2 \rightsquigarrow$ easy, very efficient in practice
- up to $n = 80$ or $n = 100 \rightsquigarrow$ a few minutes on a personal laptop
- up to $n = 170 \rightsquigarrow$ a few days on a big computer with optimized code

Exact SVP in practice

- $n = 2 \rightsquigarrow$ easy, very efficient in practice
- up to $n = 80$ or $n = 100 \rightsquigarrow$ a few minutes on a personal laptop
- up to $n = 170 \rightsquigarrow$ a few days on a big computer with optimized code
- from $n = 500$ to $n = 1000 \rightsquigarrow$ cryptography

Exact SVP in practice

- $n = 2 \rightsquigarrow$ easy, very efficient in practice
- up to $n = 80$ or $n = 100 \rightsquigarrow$ a few minutes on a personal laptop
- up to $n = 170 \rightsquigarrow$ a few days on a big computer with optimized code
- from $n = 500$ to $n = 1000 \rightsquigarrow$ cryptography

Exercise

Do exercise 1 of the exercises sheet (`https://apelletm.pages.math.cnrs.fr/page-perso/documents/presentations/AsCrypto_exercises.pdf`).

Exact SVP in practice

- $n = 2 \rightsquigarrow$ easy, very efficient in practice
- up to $n = 80$ or $n = 100 \rightsquigarrow$ a few minutes on a personal laptop
- up to $n = 170 \rightsquigarrow$ a few days on a big computer with optimized code
- from $n = 500$ to $n = 1000 \rightsquigarrow$ cryptography

Exercise

Do exercise 1 of the exercises sheet (`https://apelletm.pages.math.cnrs.fr/page-perso/documents/presentations/AsCrypto_exercises.pdf`).

Solution: in my case, it was $\texttt{dim} \approx 57$

Outline of the talk

Limitations of SVP (and CVP)

SVP and CVP are hard in the worst case

Limitations of SVP (and CVP)

SVP and CVP are hard in the worst case

- no efficient algorithm that works for any lattice

Limitations of SVP (and CVP)

SVP and CVP are hard in the worst case

- no efficient algorithm that works for any lattice
- but for some lattice (or some basis of a lattice) it might be easy
  - Exercise 2

Limitations of SVP (and CVP)

SVP and CVP are hard in the worst case

- no efficient algorithm that works for any lattice
- but for some lattice (or some basis of a lattice) it might be easy
  - Exercise 2
  - Solution: $\texttt{dim} \approx 170$

Limitations of SVP (and CVP)

SVP and CVP are hard in the worst case

- no efficient algorithm that works for any lattice
- but for some lattice (or some basis of a lattice) it might be easy
  - Exercise 2
  - Solution: $\texttt{dim} \approx 170$

For crypto, we need problems that are hard on average

(i.e., for a random instance, the problem is hard with overwhelming probability)

The SIS problem

Notations: q, B integers, $1 \leq B \ll q$, $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$

SIS (Short Integer Solution) [Ajt96]

Given $A \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ (with $n \log q < m$)

Find $x \in \{-B, \cdots, B\}^m \setminus \{0\}$ s.t. $x \cdot A = 0 \bmod q$

[Ajt96] M. Ajtai. Generating hard instances of lattice problems. STOC.

The SIS problem

Notations: q, B integers, $1 \leq B \ll q$, $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$

SIS (Short Integer Solution) [Ajt96]

Given $\mathbf{A} \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ (with $n \log q < m$)

Find $x \in \{-B, \cdots, B\}^m \setminus \{0\}$ s.t. $\mathbf{x} \cdot \mathbf{A} = 0 \bmod q$

Solving SIS with non-negligible probability (e.g., $\geq 2^{-80}$) $\overset{\sim}{\iff}$ Solving SVP in any lattice of rank n

[Ajt96] M. Ajtai. Generating hard instances of lattice problems. STOC.

SIS is a lattice problem

SIS (Short Integer Solution)

Given $A \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ (with $n \log q < m$)

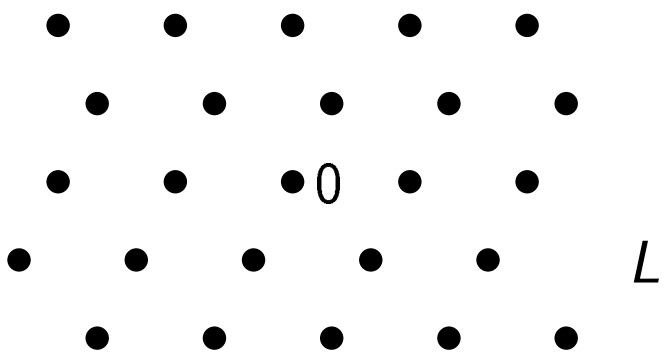
Find $x \in \{-B, \cdots, B\}^m \setminus \{0\}$ s.t. $x \cdot A = 0 \bmod q$

SIS is a lattice problem

SIS (Short Integer Solution)

Given $A \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ (with $n \log q < m$)

Find $x \in \{-B, \cdots, B\}^m \setminus \{0\}$ s.t. $x \cdot A = 0 \bmod q$



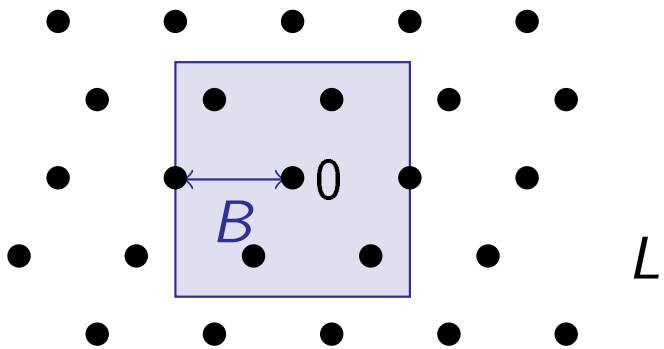
$$L = \{x \in \mathbb{Z}^m \mid xA = 0 \bmod q\}$$

SIS is a lattice problem

SIS (Short Integer Solution)

Given $A \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ (with $n \log q < m$)

Find $x \in \{-B, \cdots, B\}^m \setminus \{0\}$ s.t. $x \cdot A = 0 \bmod q$



$$L = \{x \in \mathbb{Z}^m \mid xA = 0 \bmod q\}$$

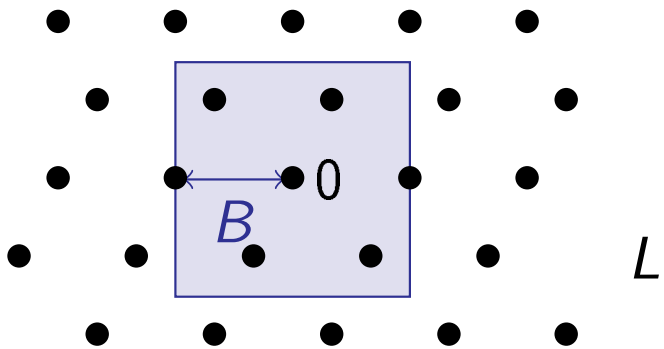
SIS $\approx$ SVP in L

SIS is a lattice problem

SIS (Short Integer Solution)

Given $A \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ (with $n \log q < m$)

Find $x \in \{-B, \cdots, B\}^m \setminus \{0\}$ s.t. $x \cdot A = 0 \bmod q$



$$L = \{x \in \mathbb{Z}^m \mid xA = 0 \bmod q\}$$

SIS $\approx$ SVP in L

Exercise 3 (worksheet)

Solution of exercise 3

```
set_random_seed(42)
A = random_matrix(Integers(127),5,10)

# kernel of A modulo q
B1 = Matrix(A.right_kernel().basis())
# all the (0,q,0,...,0) vectors
B2 = 127*identity_matrix(10)
# concatenation of the rows
B = Matrix(ZZ, list(B1)+list(B2))
print(B)

x = IntegerLattice(B).shortest_vector()
print(x)
print(A*x)
```

$$x = (1, 0, -1, 5, 5, 4, -2, 3, -3, -1)$$

The LWE problem

Notations: q, B integers, $1 \leq B \ll q$, $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$

LWE (Learning With Errors) [Reg05]

Sample $\mathbf{A} \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \text{Uniform}(\{-B, \cdots, B\}^n)$

Given $\mathbf{A}$ and $\mathbf{b}$, where $\mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$

Recover s or e

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. STOC.

The LWE problem

Notations: q, B integers, $1 \leq B \ll q$, $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$

LWE (Learning With Errors) [Reg05]

Sample $\mathbf{A} \leftarrow \text{Uniform}(\mathbb{Z}_q^{m \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \text{Uniform}(\{-B, \cdots, B\}^n)$

Given $\mathbf{A}$ and $\mathbf{b}$, where $\mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$

Recover s or e

Remark. Sometimes $\mathbf{s}$ is uniform in $\mathbb{Z}_q$ (not small)

- this is (almost) equivalent
- prove it (*hint:* you are allowed to change m)

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. STOC.

The LWE problem

Notations: q, B integers, $1 \leq B \ll q$, $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$

LWE (Learning With Errors) [Reg05]

Sample $\mathbf{A} \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^{m \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \mathrm{Uniform}(\{-B, \cdots, B\}^n)$

Given $\mathbf{A}$ and $\mathbf{b}$, where $\mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$

Recover $\mathbf{s}$ or $\mathbf{e}$

Solving LWE with non-negligible probability (e.g., $\geq 2^{-80}$) $\overset{\sim}{\Longleftrightarrow}$ Solving SVP in any lattice of rank n

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. STOC.

LWE is a lattice problem

LWE (Learning With Errors)

Sample $\mathbf{A} \leftarrow \text{Uniform}(\mathbb{Z}_q^{n \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \text{Uniform}(\{-B, \cdots, B\}^n)$

Given $\mathbf{A}$ and $\mathbf{b}$, where $\mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$

Recover s or e

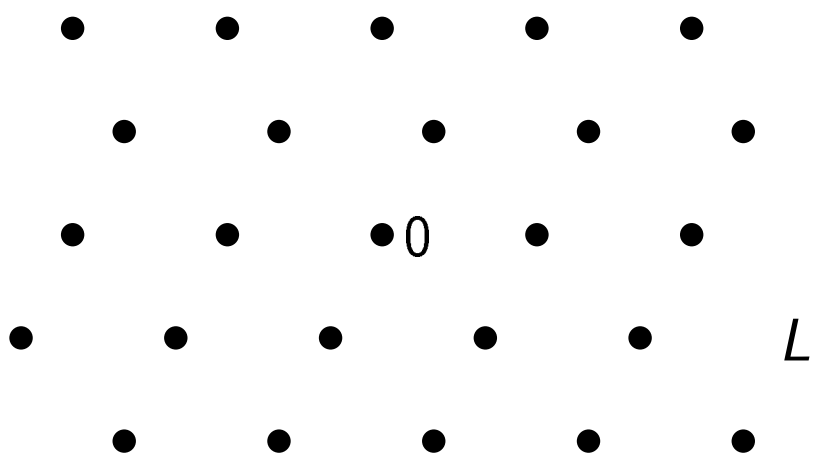
LWE is a lattice problem

LWE (Learning With Errors)

Sample $A \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^{n \times n})$ and $s, e \leftarrow \mathrm{Uniform}(\{-B, \cdots, B\}^n)$

Given A and b, where $b := As + e \bmod q$

Recover s or e



$$L = \{x \in \mathbb{Z}^n \mid \exists s \in \mathbb{Z}^n, As = x \bmod q\}$$

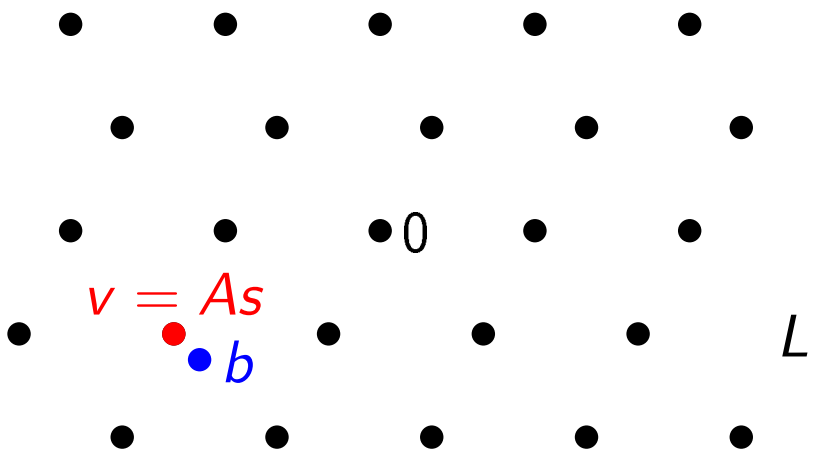
LWE is a lattice problem

LWE (Learning With Errors)

Sample $\mathbf{A} \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^{n \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \mathrm{Uniform}(\{-B, \cdots, B\}^n)$

Given $\mathbf{A}$ and $\mathbf{b}$, where $\mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$

Recover s or e



$$L = \{x \in \mathbb{Z}^n \mid \exists s \in \mathbb{Z}^n, As = x \bmod q\}$$

$$b = v + e,$$

where $v \in L$ and e small

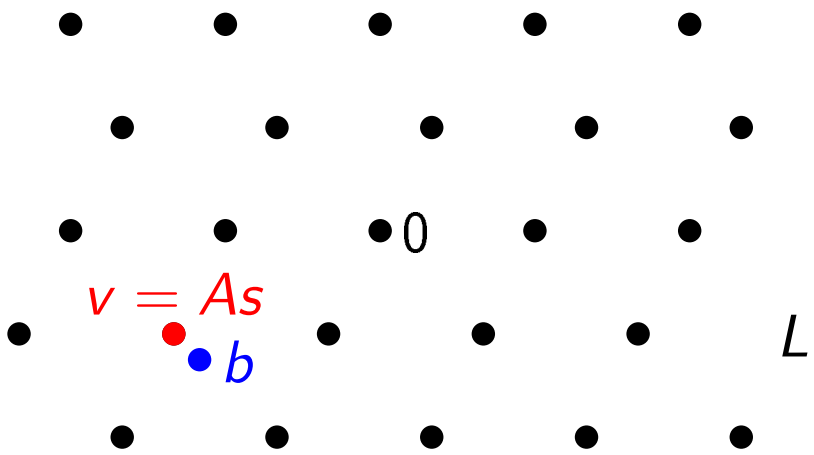
LWE is a lattice problem

LWE (Learning With Errors)

Sample $A \leftarrow \text{Uniform}(\mathbb{Z}_q^{n \times n})$ and $s, e \leftarrow \text{Uniform}(\{-B, \cdots, B\}^n)$

Given A and b, where $b := As + e \bmod q$

Recover s or e



$$L = \{x \in \mathbb{Z}^n \mid \exists s \in \mathbb{Z}^n, As = x \bmod q\}$$

$$b = v + e,$$
where $v \in L$ and e small

LWE $\approx$ CVP in L

Summary on SIS and LWE

SIS and LWE are average-case problems

Summary on SIS and LWE

SIS and LWE are average-case problems

$\Rightarrow$ Good for crypto

(negligible probability to sample a weak key)

Summary on SIS and LWE

SIS and LWE are average-case problems
$\Rightarrow$ Good for crypto
(negligible probability to sample a weak key)

SIS $\overset{\sim}{\longleftrightarrow}$ average case SVP

LWE $\overset{\sim}{\longleftrightarrow}$ average case CVP

Decision variant of LWE

decision-LWE

Sample $A \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^{n \times n})$ and $s, e \leftarrow \mathrm{Uniform}(\{-B, \cdots, B\}^n)$

Given A and b, where

$$b := A s + e \bmod q \quad \text{or} \quad b \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^n)$$

Guess whether b is uniform or not.

Decision variant of LWE

decision-LWE

Sample $A \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^{n \times n})$ and $s, e \leftarrow \mathrm{Uniform}(\{-B, \cdots, B\}^n)$

Given A and b, where

$$b := A s + e \bmod q \quad \text{or} \quad b \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^n)$$

Guess whether b is uniform or not.

decision LWE $\overset{\sim}{\Longleftrightarrow}$ (search) LWE

Decision variant of LWE

decision-LWE

Sample $\mathbf{A} \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^{n \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \mathrm{Uniform}(\{-B, \cdots, B\}^n)$

Given $\mathbf{A}$ and $\mathbf{b}$, where

$$\mathbf{b} := \mathbf{A}\,\mathbf{s} + \mathbf{e} \bmod q \quad \text{or} \quad \mathbf{b} \leftarrow \mathrm{Uniform}(\mathbb{Z}_q^n)$$

Guess whether $\mathbf{b}$ is uniform or not.

decision LWE $\overset{\sim}{\Longleftrightarrow}$ (search) LWE

$\Rightarrow$ decision problems can be easier to use for crypto

LWE vs SIS

$$\text{decision-LWE} \overset{\sim}{\Longleftrightarrow} \text{(search) LWE} \overset{\sim}{\Longleftrightarrow} \text{SIS}$$

LWE vs SIS

$$\text{decision-LWE} \overset{\sim}{\Longleftrightarrow} \text{(search) LWE} \overset{\sim}{\Longleftrightarrow} \text{SIS}$$

Exercise

Prove that decision-LWE $\leq$ SIS

Hint: Assume that we know $\mathbf{x}$ small such that $\mathbf{x}\,\mathbf{A} = 0 \bmod q$

How to distinguish

$$\mathbf{b} := \mathbf{A}\,\mathbf{s} + \mathbf{e} \bmod q \quad \text{vs} \quad \mathbf{b} \leftarrow \text{Uniform}(\mathbb{Z}_q^n)$$

LWE vs SIS

$$\text{decision-LWE} \overset{\sim}{\Longleftrightarrow} \text{(search) LWE} \overset{\sim}{\Longleftrightarrow} \text{SIS}$$

Exercise

Prove that decision-LWE $\leq$ SIS

Hint: Assume that we know $\mathbf{x}$ small such that $\mathbf{x}\,\mathbf{A} = 0 \bmod q$

How to distinguish

$$\mathbf{b} := \mathbf{A}\,\mathbf{s} + \mathbf{e} \bmod q \quad \text{vs} \quad \mathbf{b} \leftarrow \text{Uniform}(\mathbb{Z}_q^n)$$

Solution: Compute $\mathbf{x}\,\mathbf{b}$, this gives

$$\underbrace{\mathbf{x}\,\mathbf{A}}_{0}\,\mathbf{s} + \mathbf{x}\,\mathbf{e} = \underbrace{\mathbf{x}\,\mathbf{e}}_{\text{small}} \bmod q \quad \text{vs} \quad \underbrace{\mathbf{x}\,\mathbf{b}}_{\text{uniform}}$$

Outline of the talk

The ring R

$$R = \mathbb{Z}[X]/(X^n + 1) \qquad \text{with } n = 2^k$$

The ring R

$$R = \mathbb{Z}[X]/(X^n + 1) \qquad \text{with } n = 2^k$$

$r \in R$ can be seen as:

- $r = \sum_{i=0}^{n-1} r_i X^i \rightsquigarrow$ polynomial of degree $n-1 \pmod{X^n+1}$
- $\vec{r} = (r_0, \cdots, r_{n-1}) \in \mathbb{Z}^n \rightsquigarrow$ vector of dim n

The ring R

$$R = \mathbb{Z}[X]/(X^n + 1) \qquad \text{with } n = 2^k$$

$r \in R$ can be seen as:

- $r = \sum_{i=0}^{n-1} r_i X^i \rightsquigarrow$ polynomial of degree $n-1 \pmod{X^n+1}$
- $\vec{r} = (r_0, \cdots, r_{n-1}) \in \mathbb{Z}^n \rightsquigarrow$ vector of dim n

Operations:

- addition $\rightsquigarrow$ coordinate-wise
- multiplication $\rightsquigarrow$ more complex

The ring R

$$R = \mathbb{Z}[X]/(X^n + 1) \qquad \text{with } n = 2^k$$

$r \in R$ can be seen as:

- $r = \sum_{i=0}^{n-1} r_i X^i \rightsquigarrow$ polynomial of degree $n-1 \pmod{X^n+1}$
- $\vec{r} = (r_0, \cdots, r_{n-1}) \in \mathbb{Z}^n \rightsquigarrow$ vector of dim n

Operations:

- addition $\rightsquigarrow$ coordinate-wise
- multiplication $\rightsquigarrow$ more complex

Exercise

$n = 4$, $r_1 = 1 + 3X - X^3$ and $r_2 = 2X + 4X^2 + X^3$

$r_1 + r_2 = ??$ $\qquad$ $r_1 \cdot r_2 = ??$

The ring R

$$R = \mathbb{Z}[X]/(X^n + 1) \qquad \text{with } n = 2^k$$

$r \in R$ can be seen as:

- $r = \sum_{i=0}^{n-1} r_i X^i \rightsquigarrow$ polynomial of degree $n-1$ $(\text{mod } X^n + 1)$
- $\vec{r} = (r_0, \cdots, r_{n-1}) \in \mathbb{Z}^n \rightsquigarrow$ vector of dim n

Operations:

- addition $\rightsquigarrow$ coordinate-wise
- multiplication $\rightsquigarrow$ more complex

Exercise

$n = 4$, $r_1 = 1 + 3X - X^3$ and $r_2 = 2X + 4X^2 + X^3$

$r_1 + r_2 = 1 + 5X + 4X^2 \qquad r_1 \cdot r_2 = -1 + 6X + 11X^2 + 13X^3$

Ring-LWE (RLWE)

$R_q := R/(qR) = \mathbb{Z}_q[X]/(X^n + 1)$

decision-RLWE [SSTX09, LPR10]

Sample

- $a \leftarrow \text{Uniform}(R_q)$
- $s, e \in R$ with coefficients in $\{-B, \cdots, B\}$

Distinguish between

$$b = a \cdot s + e \bmod q \quad \text{and} \quad b \leftarrow \text{Uniform}(R_q)$$

[SSTX09] Stehlé, Steinfeld, Tanaka, and Xagawa. Efficient public key encryption based on ideal lattices. Asiacrypt.

[LPR10] Lyubashevsky, Peikert, and Regev. On ideal lattices and learning with errors over rings. Eurocrypt.

Ring-LWE (RLWE)

$R_q := R/(qR) = \mathbb{Z}_q[X]/(X^n + 1)$

decision-RLWE [SSTX09, LPR10]

Sample

- $a \leftarrow \text{Uniform}(R_q)$
- $s, e \in R$ with coefficients in $\{-B, \cdots, B\}$

Distinguish between

$$b = a \cdot s + e \bmod q \quad \text{and} \quad b \leftarrow \text{Uniform}(R_q)$$

Exercise: Find the matrix M_a such that $\overrightarrow{as + e}$ $= M_a \vec{s} + \vec{e}$

[SSTX09] Stehlé, Steinfeld, Tanaka, and Xagawa. Efficient public key encryption based on ideal lattices. Asiacrypt.

[LPR10] Lyubashevsky, Peikert, and Regev. On ideal lattices and learning with errors over rings. Eurocrypt.

Ring-LWE (RLWE)

$R_q := R/(qR) = \mathbb{Z}_q[X]/(X^n + 1)$

decision-RLWE [SSTX09, LPR10]

Sample

- $a \leftarrow \text{Uniform}(R_q)$
- $s, e \in R$ with coefficients in $\{-B, \cdots, B\}$

Distinguish between

$$b = a \cdot s + e \bmod q \quad \text{and} \quad b \leftarrow \text{Uniform}(R_q)$$

Exercise: Find the matrix M_a such that $\overrightarrow{as + e}$ $= M_a \ \vec{s} + \vec{e}$

Solution: $M_a = \begin{pmatrix} a_0 & -a_{n-1} & \cdots & -a_2 \\ a_1 & a_0 & \cdots & -a_3 \\ \vdots & \vdots & \ddots & \vdots \\ a_n & a_{n-1} & \cdots & a_0 \end{pmatrix}$

Hardness of Ring-LWE

Ring-LWE = LWE with structured matrices

Hardness of Ring-LWE

Ring-LWE = LWE with structured matrices

Advantage of RLWE: more efficient protocols, but what about security?

Hardness of Ring-LWE

Ring-LWE = LWE with structured matrices

Advantage of RLWE: more efficient protocols, but what about security?

- Ring-LWE $\leq$ LWE

Hardness of Ring-LWE

Ring-LWE = LWE with structured matrices

Advantage of RLWE: more efficient protocols, but what about security?

- Ring-LWE $\leq$ LWE
- but so far

  best attacks on Ring-LWE $\approx$ best attacks on LWE

  (the structure does not help)

Hardness of Ring-LWE

Ring-LWE = LWE with structured matrices

Advantage of RLWE: more efficient protocols, but what about security?

- Ring-LWE $\leq$ LWE
- but so far

  best attacks on Ring-LWE $\approx$ best attacks on LWE

  (the structure does not help)

Conclusion (so far): better efficiency for the same security guarantees

NTRU

$B \ll q$, $R_q = \mathbb{Z}_q[X]/(X^n + 1)$

NTRU [HPS98]

Sample $f, g \in R$ with coefficients in $\{-B, \cdots, B\}$.
Distinguish between

$$h = f \cdot g^{-1} \bmod q \quad \text{and} \quad h \leftarrow \text{Uniform}(R_q)$$

[HPS98] Hoffstein, Pipher, and Silverman. NTRU: a ring based public key cryptosystem. ANTS.

NTRU

$B \ll q$, $R_q = \mathbb{Z}_q[X]/(X^n + 1)$

NTRU [HPS98]

Sample $f, g \in R$ with coefficients in $\{-B, \cdots, B\}$.
Distinguish between

$$h = f \cdot g^{-1} \bmod q \quad \text{and} \quad h \leftarrow \text{Uniform}(R_q)$$

Example: $n = 1$ $(R = \mathbb{Z})$, $f = 3$, $g = 5$ and $q = 1031$

$$h = f \cdot g^{-1} = 413$$

[HPS98] Hoffstein, Pipher, and Silverman. NTRU: a ring based public key cryptosystem. ANTS.

NTRU

$B \ll q$, $R_q = \mathbb{Z}_q[X]/(X^n + 1)$

NTRU [HPS98]

Sample $f, g \in R$ with coefficients in $\{-B, \cdots, B\}$.
Distinguish between

$$h = f \cdot g^{-1} \bmod q \quad \text{and} \quad h \leftarrow \text{Uniform}(R_q)$$

Example: $n = 1$ $(R = \mathbb{Z})$, $f = 3$, $g = 5$ and $q = 1031$

$$h = f \cdot g^{-1} = 413$$

Exercises:

1. check that $f \cdot h = g \bmod q$
2. why is it unsafe to take $h = f$ or $h = g^{-1} \bmod q$?

[HPS98] Hoffstein, Pipher, and Silverman. NTRU: a ring based public key cryptosystem. ANTS.

NTRU

$B \ll q$, $R_q = \mathbb{Z}_q[X]/(X^n + 1)$

NTRU [HPS98]

Sample $f, g \in R$ with coefficients in $\{-B, \cdots, B\}$.
Distinguish between

$$h = f \cdot g^{-1} \bmod q \quad \text{and} \quad h \leftarrow \text{Uniform}(R_q)$$

Example: $n = 1$ $(R = \mathbb{Z})$, $f = 3$, $g = 5$ and $q = 1031$

$$h = f \cdot g^{-1} = 413$$

Exercises:

1. check that $f \cdot h = g \bmod q$
2. why is it unsafe to take $h = f$ or $h = g^{-1} \bmod q$?
   - f is small, easy to distinguish from Uniform(R_q) (which is likely $\approx q$)
   - if $h = g^{-1}$, one can compute $h^{-1} = g \bmod q$ and same situation as above

Hardness of NTRU

- NTRU is assumed to be post-quantumly hard when n is large and $q = \mathrm{poly}(n)$

[Pei16] Peikert. A decade of lattice cryptography. Foundations and Trends in TCS.

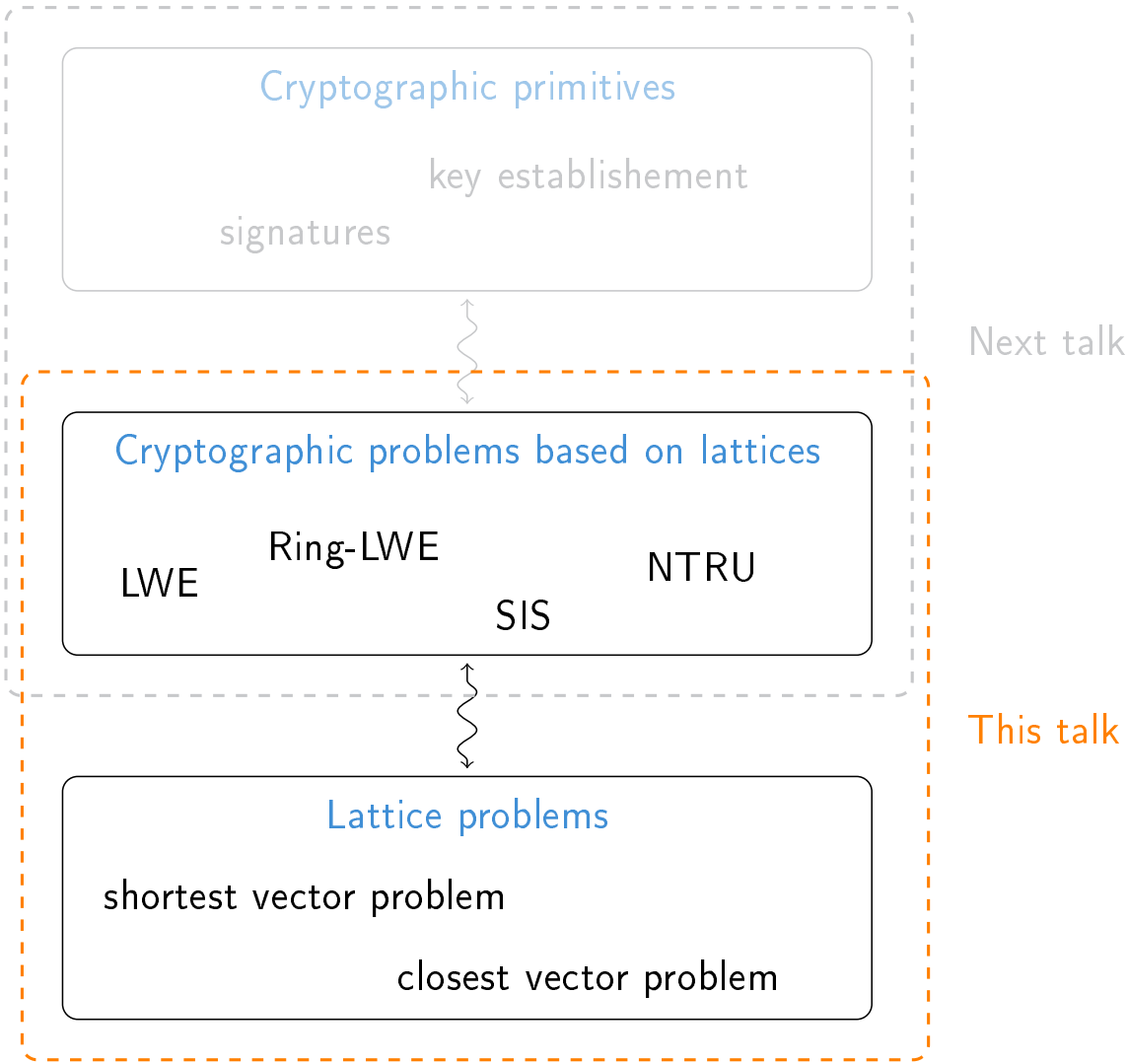
Hardness of NTRU

- NTRU is assumed to be post-quantumly hard when n is large and $q = \mathrm{poly}(n)$
- NTRU $\leq$ RLWE [Pei16]
- we do not know whether NTRU $\approx$ RLWE or NTRU $<$ RLWE

[Pei16] Peikert. A decade of lattice cryptography. Foundations and Trends in TCS.

Conclusion

What we have seen



Takeaway: all these problems are supposed to be post-quantum

(for a good choice of parameters)

After the break

Lattice-based crypto, part 2:

How to construct cryptographic primitives from all these problems?