Approx-SVP in Ideal lattices with Pre-Processing

Alice Pellet-Mary, Guillaume Hanrot and Damien Stehlé

LIP, ENS de Lyon

Journées C2 2018, October 8

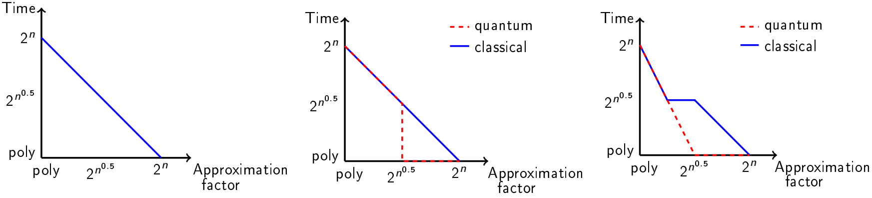
ENS DE LYON

European Research Council

Established by the European Commission

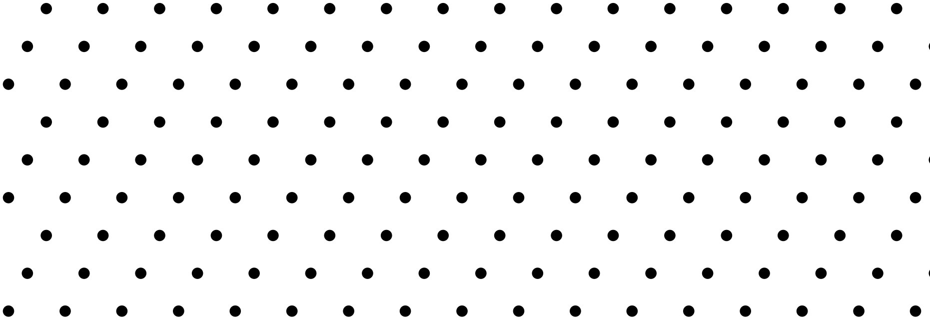
What is this talk about

Time/Approximation factor trade-off for SVP in ideal lattices:



BKZ algorithm | [CDPR16,CDW17] | This work (with $2^{O(n)}$ pre-processing)

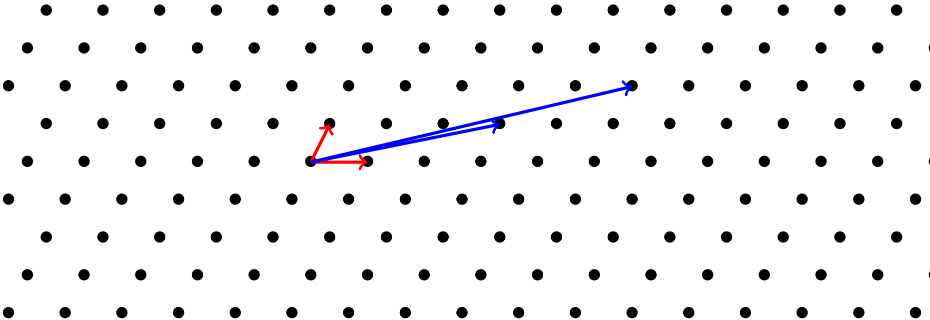
Lattices



Lattice

A lattice L is a discrete 'vector space' over $\mathbb{Z}$.

Lattices

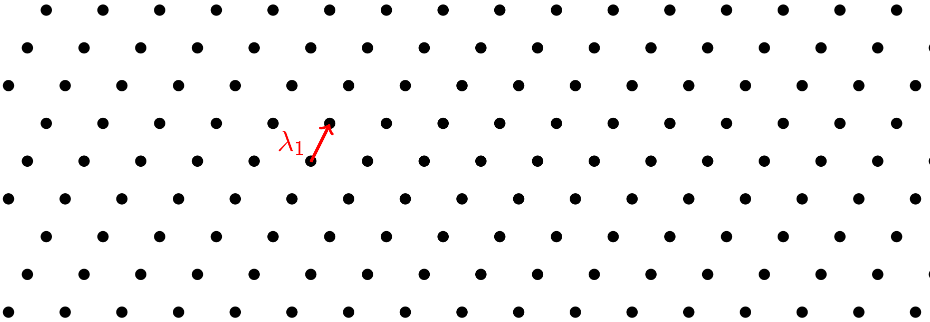


Lattice

A lattice L is a discrete 'vector space' over $\mathbb{Z}$.
A basis of L is an invertible matrix B such that $L = \{Bx \mid x \in \mathbb{Z}^n\}$.

$\begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix}$ and $\begin{pmatrix} 17 & 11 \\ 4 & 2 \end{pmatrix}$ are two bases of the above lattice.

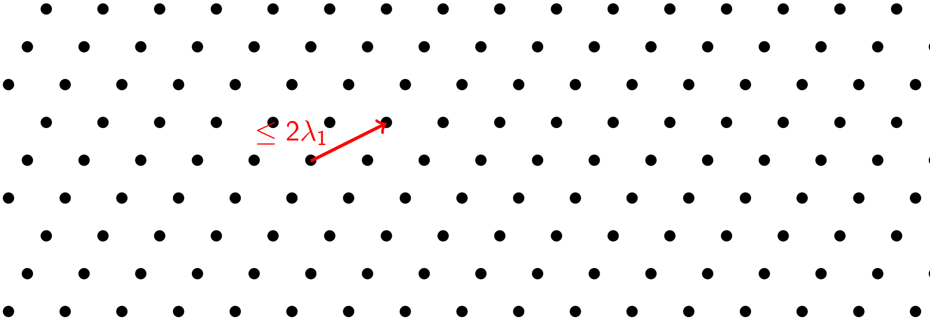
Lattices



Shortest Vector Problem (SVP)

Find a shortest (in Euclidean norm) non-zero vector.
Its Euclidean norm is denoted λ_1.

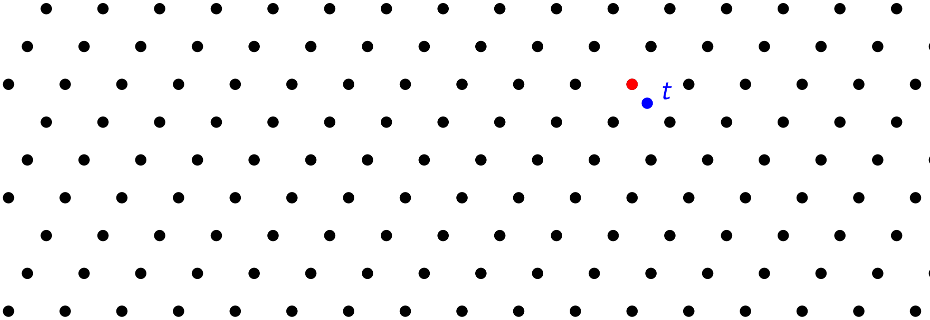
Lattices



Approximate Shortest Vector Problem (approx-SVP)

Find a short (in Euclidean norm) non-zero vector.
(e.g. of norm $\leq 2\lambda_1$).

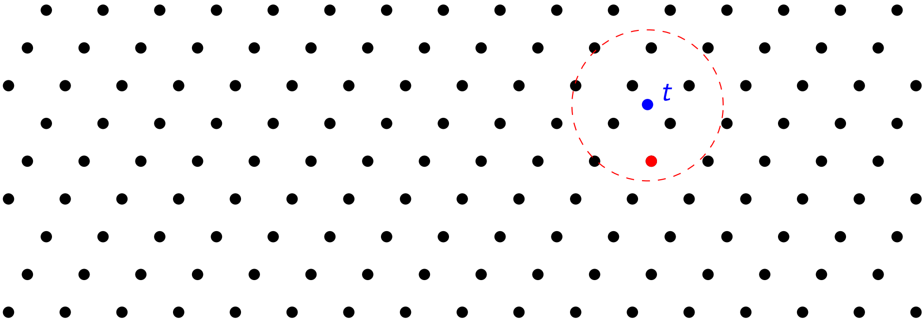
Lattices



Closest Vector Problem (CVP)

Given a target point t, find a point of the lattice closest to t.

Lattices



Approximate Closest Vector Problem (approx-CVP)

Given a target point t, find a point of the lattice close to t.

Complexity of SVP/CVP

Applications

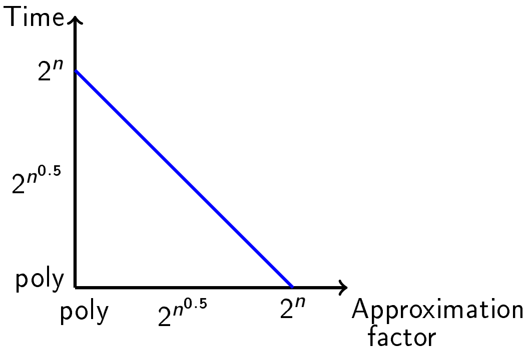
SVP and CVP in general lattices are conjectured to be hard to solve both quantumly and classically $\Rightarrow$ used in cryptography

Complexity of SVP/CVP

Applications

SVP and CVP in general lattices are conjectured to be hard to solve both quantumly and classically $\Rightarrow$ used in cryptography

Best Time/Approximation trade-off for general lattices: BKZ algorithm



Structured lattices

Improve efficiency of lattice-based crypto using structured lattices.
$\Rightarrow$ E.g. ideal lattices

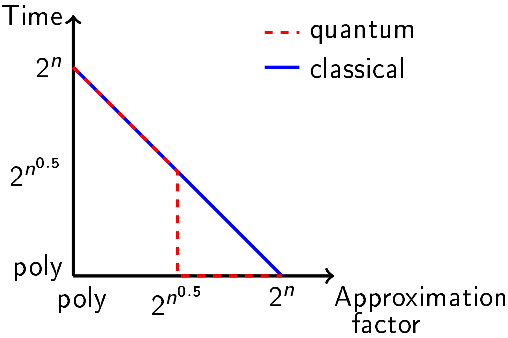
Structured lattices

Improve efficiency of lattice-based crypto using structured lattices.
$\Rightarrow$ E.g. ideal lattices

Is approx-SVP still hard when restricted to ideal lattices?

SVP in ideal lattices

[CDPR16,CDW17]: Better than BKZ in the quantum setting

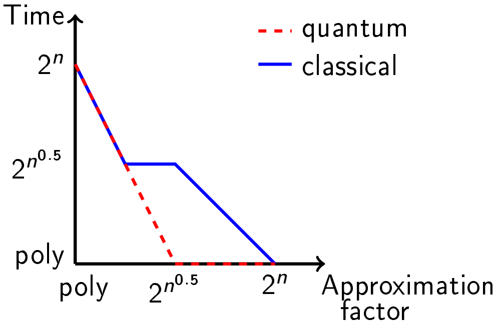


- Heuristic
- For prime power cyclotomic fields

[CDPR16] R. Cramer, L. Ducas, C. Peikert and O. Regev. Recovering Short Generators of Principal Ideals in Cyclotomic Rings, Eurocrypt.

[CDW17] R. Cramer, L. Ducas, B. Wesolowski. Short Stickelberger Class Relations and Application to Ideal-SVP, Eurocrypt.

This work



- Heuristic
- Pre-processing $2^{O(n)}$, independent of the choice of the ideal (non-uniform algorithm).

Outline of the talk

1. Definitions and objective
2. The CDPR algorithm
3. This work

First definitions

Notation

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

First definitions

Notation

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

- Units: $R^\times = \{a \in R \mid \exists b \in R, ab = 1\}$
  - e.g. $\mathbb{Z}^\times = \{-1, 1\}$

First definitions

Notation

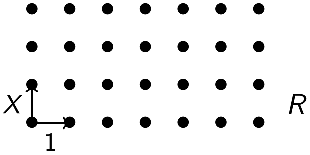
$R = \mathbb{Z}[X]/(X^n + 1)$ for $n = 2^k$

- Units: $R^\times = \{a \in R \mid \exists b \in R, ab = 1\}$
  - e.g. $\mathbb{Z}^\times = \{-1, 1\}$
- Principal ideals: $\langle g \rangle = \{gr \mid r \in R\}$ (i.e. all multiples of g)
  - e.g. $\langle 2 \rangle = \{\text{even numbers}\}$ in $\mathbb{Z}$
  - g is called a generator of $\langle g \rangle$
  - The generators of $\langle g \rangle$ are exactly the ug for $u \in R^\times$

Why is $\langle g \rangle$ a lattice?

$R \simeq \mathbb{Z}^n$

$$R = \mathbb{Z}[X]/(X^n + 1) \to \mathbb{Z}^n$$
$$r = r_0 + r_1 X + \cdots + r_{n-1} X^{n-1} \mapsto (r_0, r_1, \ldots, r_{n-1})$$

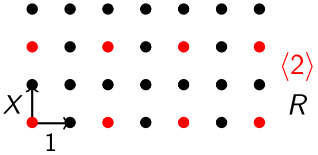


Why is $\langle g \rangle$ a lattice?

$R \simeq \mathbb{Z}^n$

$$R = \mathbb{Z}[X]/(X^n + 1) \to \mathbb{Z}^n$$
$$r = r_0 + r_1 X + \cdots + r_{n-1} X^{n-1} \mapsto (r_0, r_1, \ldots, r_{n-1})$$

$\langle g \rangle \subseteq R \simeq \mathbb{Z}^n$ + stable by '+' and '-' $\Rightarrow$ lattice



Objective of this talk

Objective

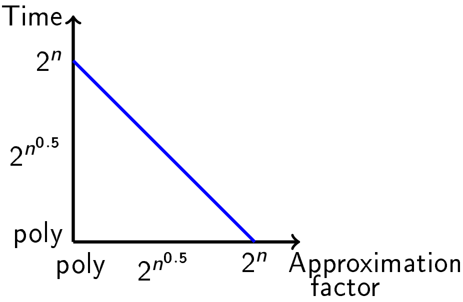
Given a basis of a principal ideal $\langle g \rangle$ and $\alpha \in (0,1]$,
Find $r \in \langle g \rangle$ such that $\|r\| \leq 2^{\widetilde{O}(n^\alpha)} \cdot \lambda_1$.

Objective of this talk

Objective

Given a basis of a principal ideal $\langle g \rangle$ and $\alpha \in (0, 1]$,
Find $r \in \langle g \rangle$ such that $\|r\| \leq 2^{\widetilde{O}(n^\alpha)} \cdot \lambda_1$.

BKZ algorithm can do it in time $2^{O(n^{1-\alpha})}$, can we do better?



Outline of the talk

Main idea of the CDPR algorithm (on an idea of [CGS14])

Idea

Maybe g is a somehow small element of $\langle g \rangle$

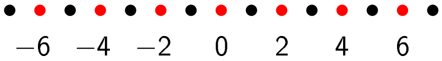
[CGS14]: P. Campbell, M. Groves, and D. Shepherd. Soliloquy: A cautionary tale.

Main idea of the CDPR algorithm (on an idea of [CGS14])

Idea

Maybe g is a somehow small element of $\langle g \rangle$

If $\mathbf{n = 1}$: e.g. $\langle 2 \rangle \Rightarrow 2$ and -2 are the smallest elements.



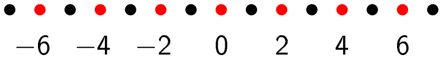
[CGS14]: P. Campbell, M. Groves, and D. Shepherd. Soliloquy: A cautionary tale.

Main idea of the CDPR algorithm (on an idea of [CGS14])

Idea

Maybe g is a somehow small element of $\langle g \rangle$

If n = 1: e.g. $\langle 2 \rangle \Rightarrow 2$ and -2 are the smallest elements.



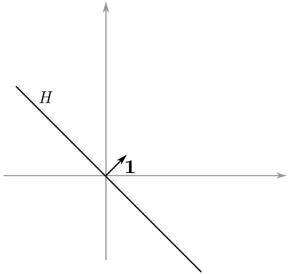
For larger n: one of the generators is somehow small

[CGS14]: P. Campbell, M. Groves, and D. Shepherd. Soliloquy: A cautionary tale.

The Log space

Log : $R \to \mathbb{R}^n$ (somehow generalising log to R)

Let $\mathbf{1} = (1, \cdots, 1)$ and $H = \mathbf{1}^{\perp}$.



The Log space

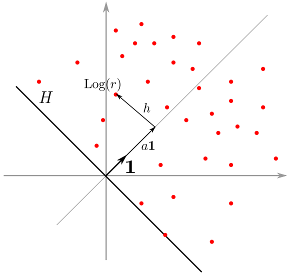
$\text{Log} : R \to \mathbb{R}^n$ (somehow generalising log to R)

Let $\mathbf{1} = (1, \cdots, 1)$ and $H = \mathbf{1}^\perp$.

Properties

$\text{Log}\, r = h + a\mathbf{1}$, with $h \in H$

- $a \geq 0$



The Log space

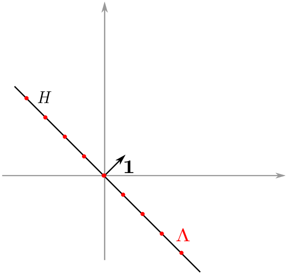
Log : $R \to \mathbb{R}^n$ (somehow generalising log to R)

Let $\mathbf{1} = (1, \cdots, 1)$ and $H = \mathbf{1}^\perp$.

Properties

$\mathrm{Log}\, r = h + a\mathbf{1}$, with $h \in H$

- $a \geq 0$
- $a = 0$ iff r is a unit
- $\Lambda := \mathrm{Log}(R^\times)$ is a lattice



The Log space

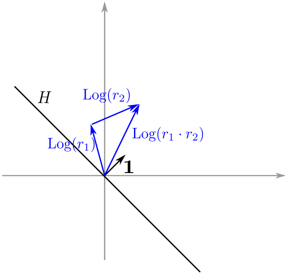
Log : $R \to \mathbb{R}^n$ (somehow generalising log to R)

Let $\mathbf{1} = (1, \cdots, 1)$ and $H = \mathbf{1}^\perp$.

Properties

$\text{Log}\, r = h + a\mathbf{1}$, with $h \in H$

- $a \geq 0$
- $a = 0$ iff r is a unit
- $\Lambda := \text{Log}(R^\times)$ is a lattice
- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$



The Log space

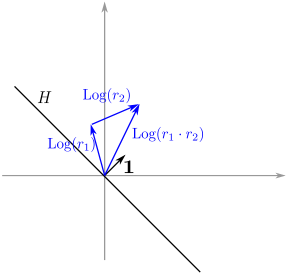
Log : $R \to \mathbb{R}^n$ (somehow generalising log to R)

Let $\mathbf{1} = (1, \cdots, 1)$ and $H = \mathbf{1}^{\perp}$.

Properties

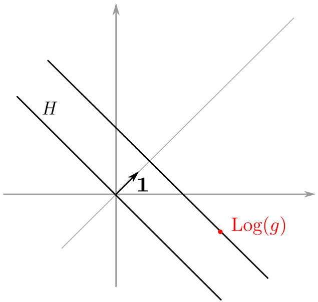
$\text{Log}\, r = h + a\mathbf{1}$, with $h \in H$

- $a \geq 0$
- $a = 0$ iff r is a unit
- $\Lambda := \text{Log}(R^{\times})$ is a lattice
- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$
- $\|r\| \simeq 2^{\|\text{Log}\, r\|_\infty}$



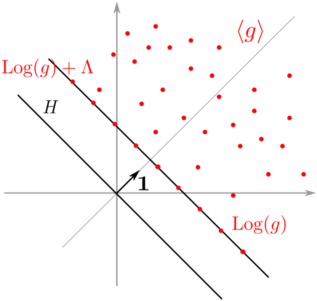
The CDPR algorithm

What does $\mathrm{Log}\langle g \rangle$ look like?



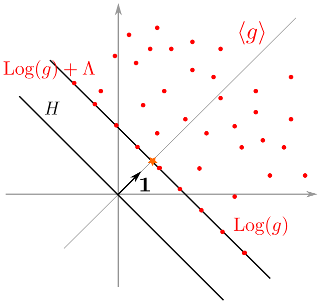
The CDPR algorithm

What does $\mathrm{Log}\langle g\rangle$ look like?



The CDPR algorithm

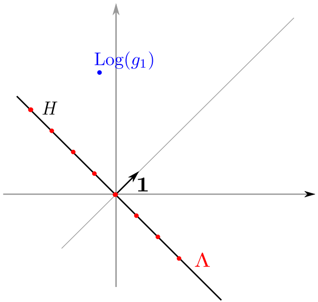
What does $\mathrm{Log}\langle g\rangle$ look like?



The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$



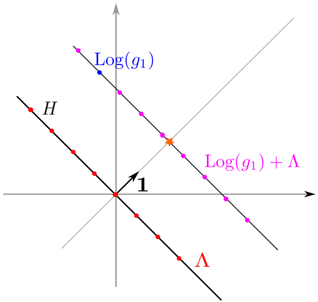
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$



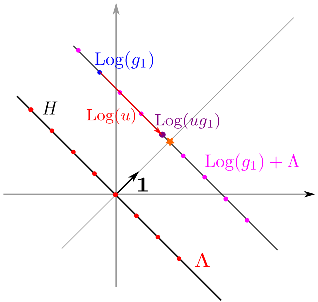
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$



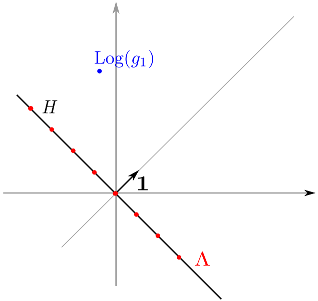
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$



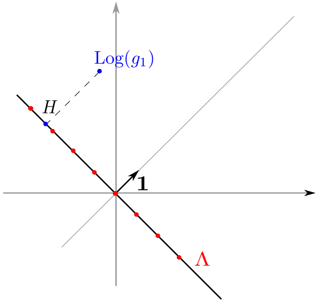
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$
- Solve CVP in Λ



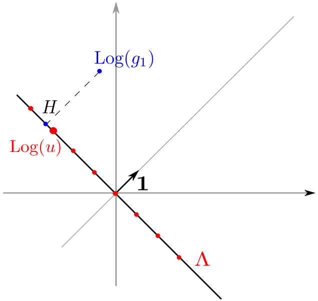
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$
- Solve CVP in Λ



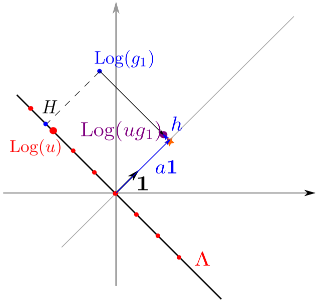
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$
- Solve CVP in Λ



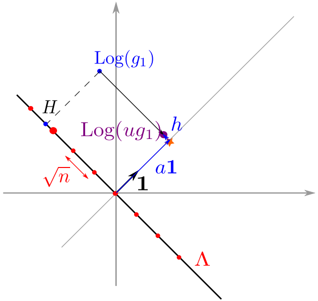
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$
- Solve CVP in Λ
  - Good basis of Λ
    $\Rightarrow$ CVP in poly time
    $\Rightarrow \|h\| \leq \widetilde{O}(\sqrt{n})$



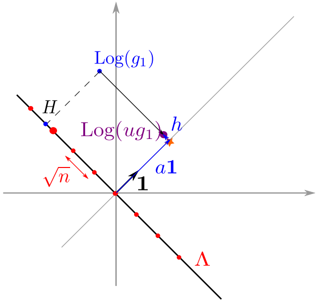
[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

The CDPR algorithm

The CDPR Algorithm:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum time $\mathrm{poly}(n)$
  - [BEFGK17]: classical time $2^{\widetilde{O}(\sqrt{n})}$
- Solve CVP in Λ
  - Good basis of Λ
    $\Rightarrow$ CVP in poly time
    $\Rightarrow \|h\| \leq \widetilde{O}(\sqrt{n})$



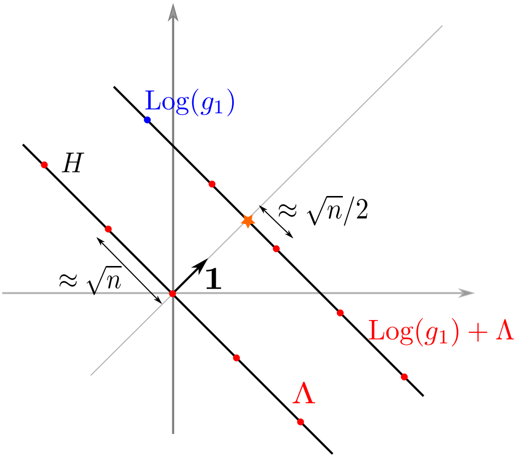
$$\|ug_1\| \leq 2^{\widetilde{O}(\sqrt{n})} \cdot \lambda_1$$

[BS16]: J.F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, SODA.

[BEFGK17]: J.F. Biasse, T. Espitau, P.A. Fouque, A. Gélin, P. Kirchner. Computing generator in cyclotomic integer rings, Eurocrypt.

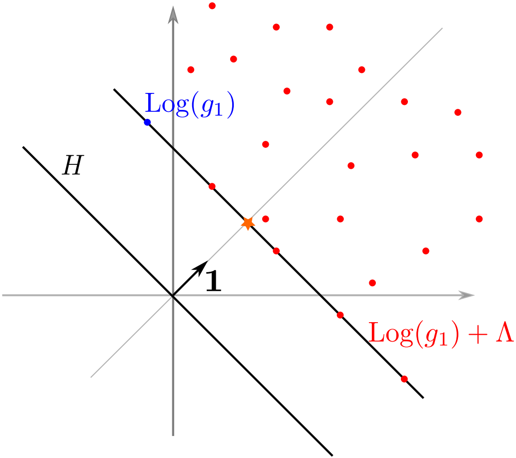
Outline of the talk

Idea

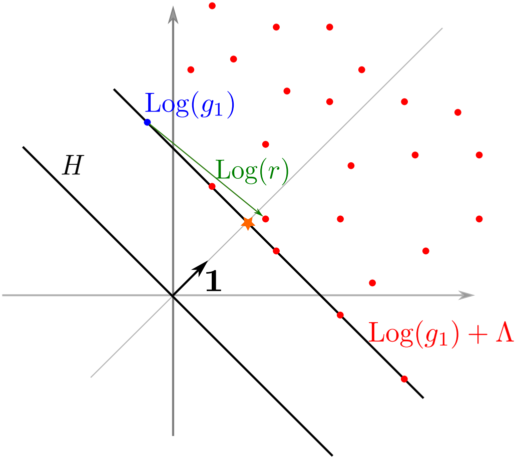


$\mathrm{Log}(g_1)$
H
$\approx \sqrt{n}/2$
$\approx \sqrt{n}$
$\mathbf{1}$
$\mathrm{Log}(g_1) + \Lambda$
Λ

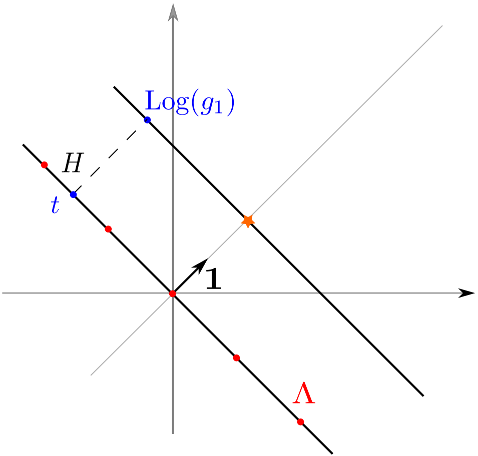
Idea



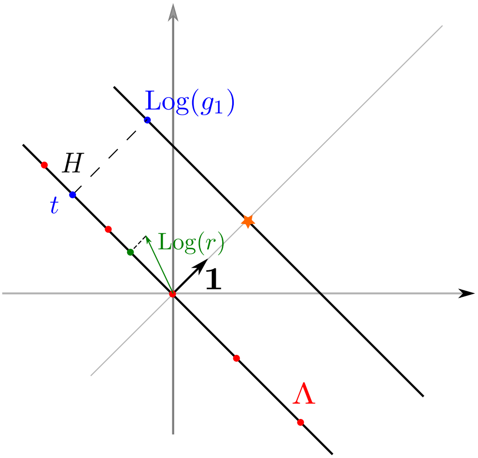
Idea



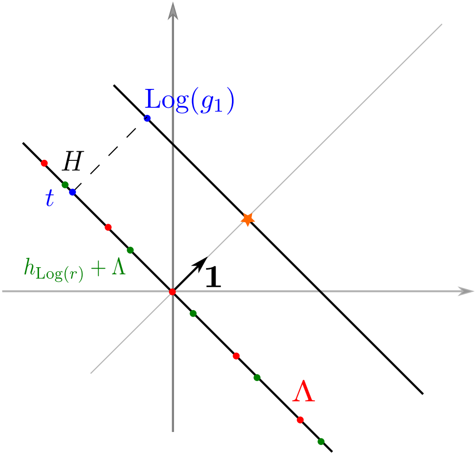
Idea



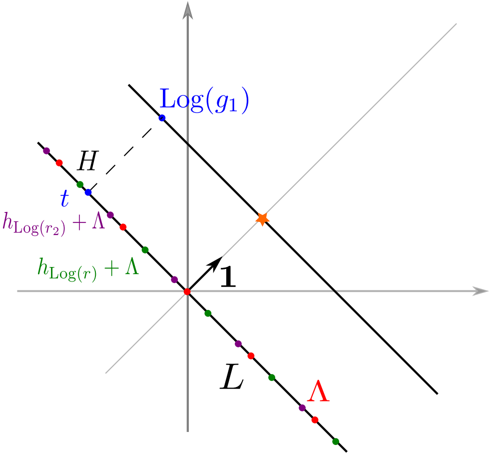
Idea



Idea



Idea



$\mathrm{Log}(g_1)$
H
t
$h_{\mathrm{Log}(r_2)} + \Lambda$
$h_{\mathrm{Log}(r)} + \Lambda$
$\mathbf{1}$
L
Λ

How to solve CVP in L?

CDPR	This work
Good basis of Λ	No good basis of L known

How to solve CVP in L?

CDPR	This work
Good basis of Λ	No good basis of L known

Key observation

$L = \Lambda \cup \bigcup_i (h_{\mathrm{Log}\, r_i} + \Lambda)$ does not depend on $\langle g \rangle$

How to solve CVP in L?

CDPR	This work
Good basis of Λ	No good basis of L known

Key observation

$L = \Lambda \cup \bigcup_i (h_{\text{Log}\, r_i} + \Lambda)$ does not depend on $\langle g \rangle$ $\Rightarrow$ Pre-processing on L

How to solve CVP in L?

CDPR	This work
Good basis of Λ	No good basis of L known

Key observation

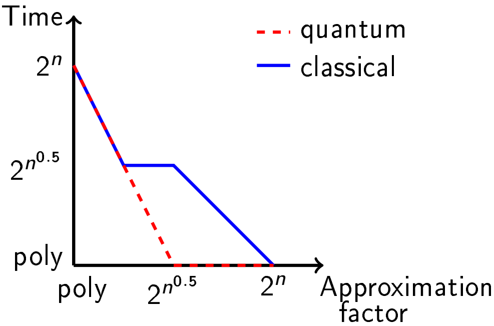
$L = \Lambda \cup \bigcup_i (h_{\text{Log}\, r_i} + \Lambda)$ does not depend on $\langle g \rangle$ $\Rightarrow$ Pre-processing on L

[Laa16]:
- Find $s \in L$ such that $\|s - t\| = \widetilde{O}(n^{\alpha})$
- Time: $2^{\widetilde{O}(n^{1-2\alpha})}$ (query) $+ 2^{O(n)}$ (pre-processing)

[Laa16] T. Laarhoven. Finding closest lattice vectors using approximate Voronoi cells. SAC.

Conclusion

Approximation	Query time	Pre-processing
$2^{\widetilde{O}(n^\alpha)}$	$2^{\widetilde{O}(n^{1-2\alpha})} + (\text{poly}(n)$ or $2^{\widetilde{O}(\sqrt{n})})$	$2^{O(n)}$



$+2^{O(n)}$ Pre-processing / Non-uniform algorithm

Extensions

- Non principal ideals ✓
- Generalization to other number fields ✓
- Removing the heuristics ?

Extensions

- Non principal ideals ✓
- Generalization to other number fields ✓
- Removing the heuristics ?

Questions?