On Ideal Lattices and the GGH13 Multilinear Map

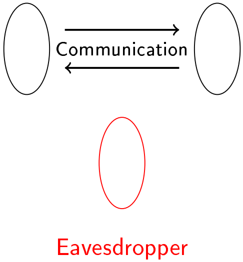
Alice Pellet-Mary

Under the supervision of Damien Stehlé

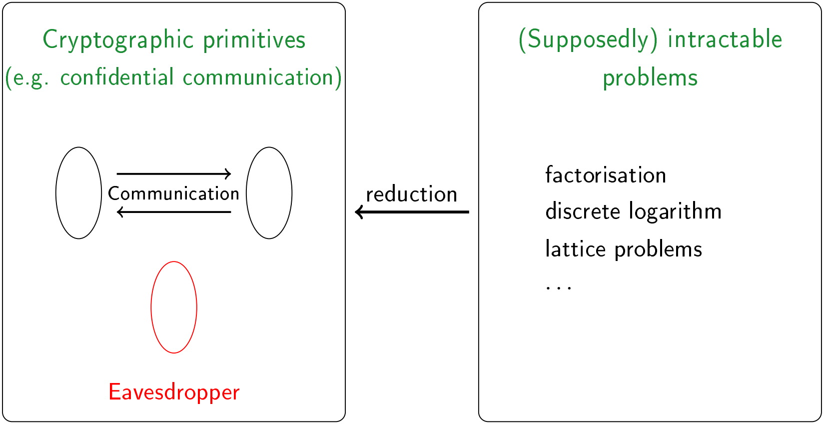
October 16, 2019

Cryptography and hard problems

Cryptographic primitives
(e.g. confidential communication)

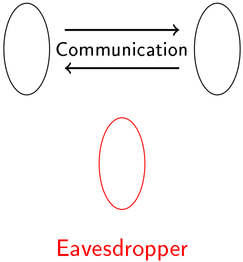


Cryptography and hard problems



Cryptography and hard problems

Cryptographic primitives
(e.g. confidential communication)



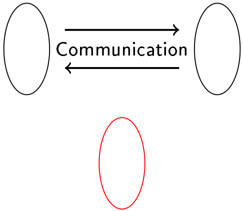
reduction ←

(Supposedly) intractable problems

- factorisation
- discrete logarithm
- lattice problems
- ...

Cryptography and hard problems

Cryptographic primitives
(e.g. confidential communication)



Eavesdropper

← reduction

(Supposedly) intractable problems
in a quantum world

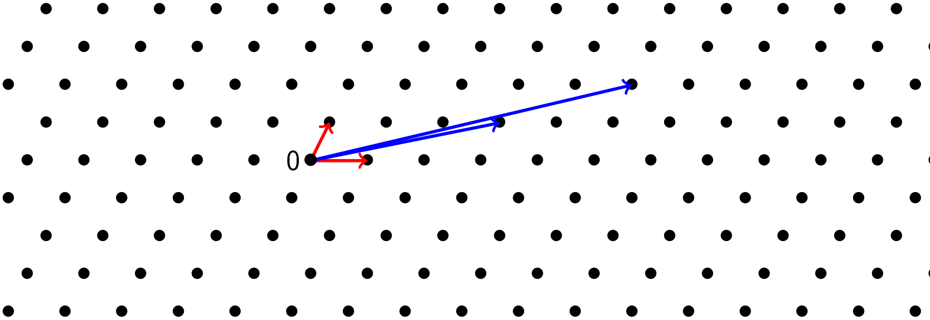
~~factorisation~~

~~discrete logarithm~~

lattice problems

...

Lattices

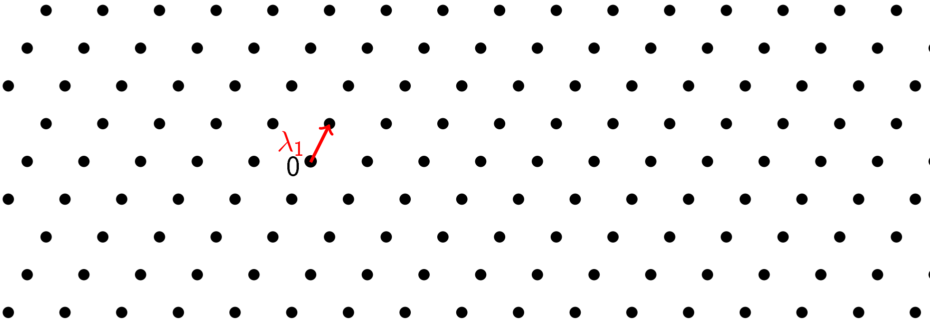


Lattice

A (full-rank) lattice L is a subset of $\mathbb{R}^n$ of the form $L = \{Bx \mid x \in \mathbb{Z}^n\}$, with $B \in \mathbb{R}^{n \times n}$ invertible. B is a basis of L.

$\begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix}$ and $\begin{pmatrix} 17 & 10 \\ 4 & 2 \end{pmatrix}$ are two bases of the above lattice.

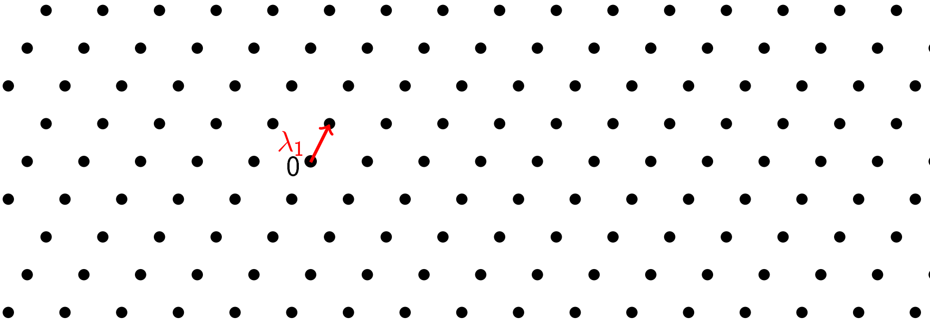
Lattice problems



Shortest Vector Problem (SVP)

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

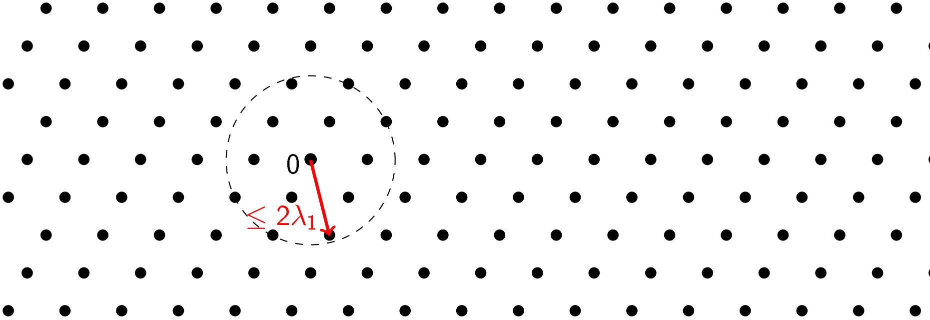
Lattice problems



Shortest Vector Problem (SVP)

Find a shortest (in Euclidean norm) non-zero vector.
Its Euclidean norm is denoted λ_1.
SIVP (Shortest Independent Vectors Problem): Find n linearly independent short vectors.

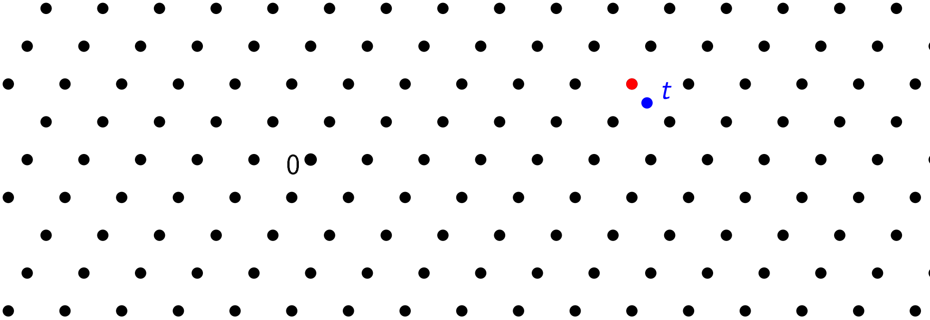
Lattice problems



Approximate Shortest Vector Problem (approx-SVP)

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

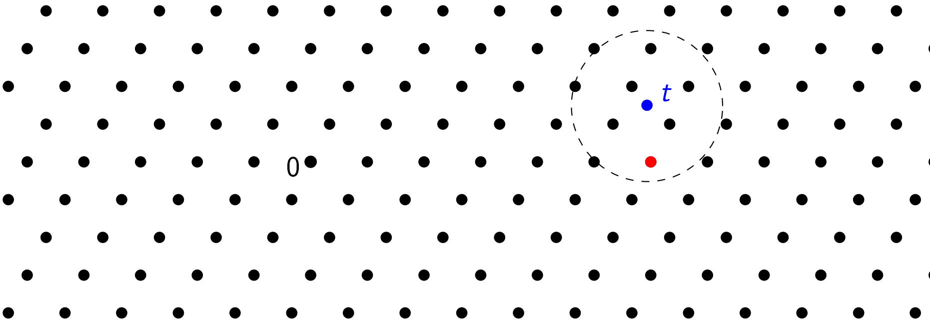
Lattice problems



Closest Vector Problem (CVP)

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

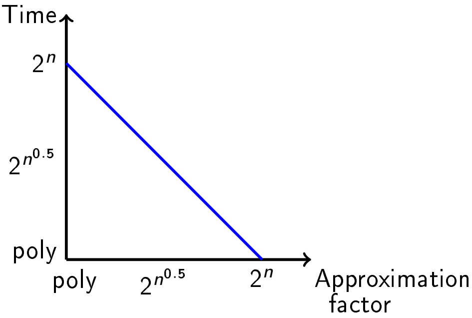
Lattice problems



Approximate Closest Vector Problem (approx-CVP)

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

Best Time/Approximation trade-off for SVP, CVP, SIVP (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.

Structured lattices

Motivation

Schemes using lattices are usually not efficient
(storage: n^2, matrix-vector mult: n^2)
$\Rightarrow$ improve efficiency using structured lattices

Structured lattices

Motivation

Schemes using lattices are usually not efficient
(storage: n^2, matrix-vector mult: n^2)
$\Rightarrow$ improve efficiency using structured lattices

Example: NIST post-quantum standardization process

- 26 candidates (2nd round)
- 12 lattice-based
- 11 using structured lattices

Structured lattices

Motivation

Schemes using lattices are usually not efficient
(storage: n^2, matrix-vector mult: n^2)
$\Rightarrow$ improve efficiency using structured lattices

Example: NIST post-quantum standardization process

- 26 candidates (2nd round)
- 12 lattice-based
- 11 using structured lattices

	Frodo (lvl 1) (unstructured lattices)	Kyber (lvl 1) (structured lattices)
secret key size (in Bytes)	19 888	1 632
public key size (in Bytes)	9 616	800

Structured lattices: example

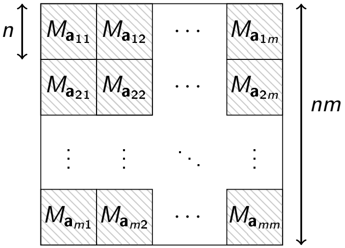
$$M_{\mathbf{a}} = \begin{pmatrix} a_1 & -a_n & \cdots & -a_2 \\ a_2 & a_1 & \cdots & -a_3 \\ \vdots & \ddots & \ddots & \vdots \\ a_n & a_{n-1} & \dots & a_1 \end{pmatrix}$$

basis of a special case of
ideal lattice

Structured lattices: example

$$M_{\mathbf{a}} = \begin{pmatrix} a_1 & -a_n & \cdots & -a_2 \\ a_2 & a_1 & \cdots & -a_3 \\ \vdots & \ddots & \ddots & \vdots \\ a_n & a_{n-1} & \cdots & a_1 \end{pmatrix}$$

basis of a special case of
ideal lattice

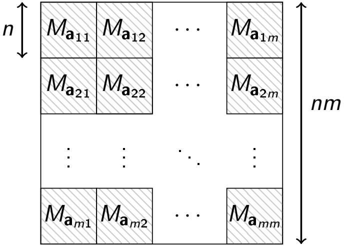


basis of a special case of
module lattice
of rank m

Structured lattices: example

$$M_{\mathbf{a}} = \begin{pmatrix} a_1 & -a_n & \cdots & -a_2 \\ a_2 & a_1 & \cdots & -a_3 \\ \vdots & \ddots & \ddots & \vdots \\ a_n & a_{n-1} & \cdots & a_1 \end{pmatrix}$$

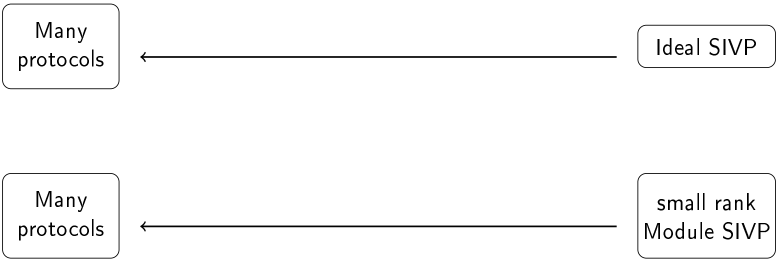
basis of a special case of
ideal lattice



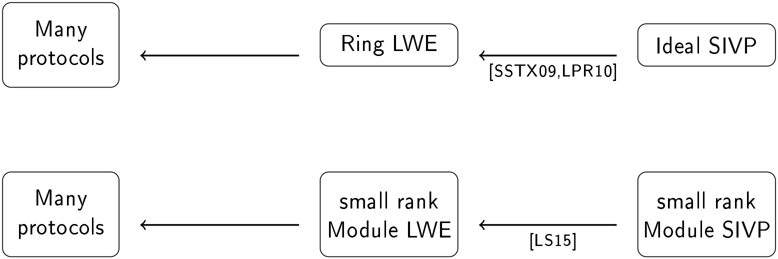
basis of a special case of
module lattice
of rank m

Is SVP still hard when restricted to ideal/module lattices?

Relations between problems and constructions



Relations between problems and constructions

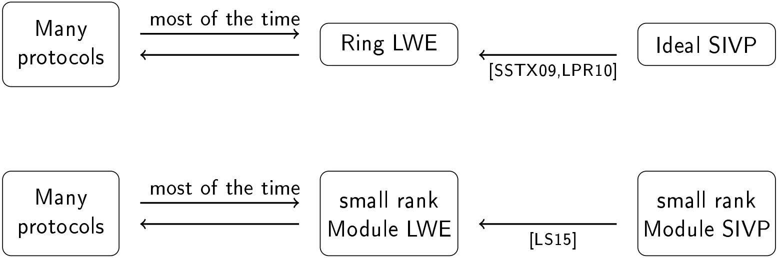


[SSTX09] D. Stehlé, R. Steinfeld, K. Tanaka, K. Xagawa. Efficient public key encryption based on ideal lattices. Asiacrypt.

[SSTX09] V. Lyubashevsky, C. Peikert, O. Regev. On ideal lattices and learning with errors over rings. Eurocrypt.

[LS15] A. Langlois, D. Stehlé. Worst-case to average-case reductions for module lattices. DCC.

Relations between problems and constructions

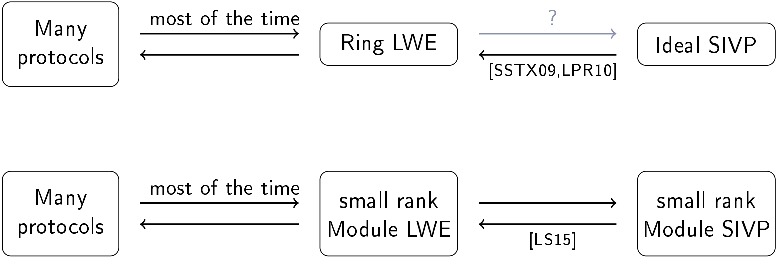


[SSTX09] D. Stehlé, R. Steinfeld, K. Tanaka, K. Xagawa. Efficient public key encryption based on ideal lattices. Asiacrypt.

[SSTX09] V. Lyubashevsky, C. Peikert, O. Regev. On ideal lattices and learning with errors over rings. Eurocrypt.

[LS15] A. Langlois, D. Stehlé. Worst-case to average-case reductions for module lattices. DCC.

Relations between problems and constructions

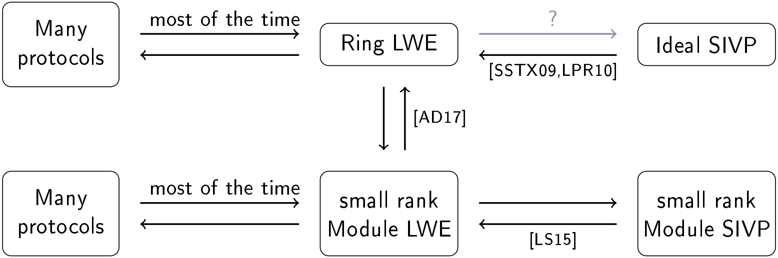


[SSTX09] D. Stehlé, R. Steinfeld, K. Tanaka, K. Xagawa. Efficient public key encryption based on ideal lattices. Asiacrypt.

[SSTX09] V. Lyubashevsky, C. Peikert, O. Regev. On ideal lattices and learning with errors over rings. Eurocrypt.

[LS15] A. Langlois, D. Stehlé. Worst-case to average-case reductions for module lattices. DCC.

Relations between problems and constructions



[AD17] M. Albrecht, A. Deo. Large modulus ring-LWE $\geq$ module-LWE. Asiacrypt.

Organization

Ideal-SVP with pre-processing

Eurocrypt 2019, with
G. Hanrot and D. Stehlé

Module-SVP with oracle

- rank 2
- arbitrary rank

Asiacrypt 2019, with
C. Lee, D. Stehlé and A. Wallet

Previous Works and Results

State-of-the-art: ideal-SVP

Solving SVP in ideal lattices:

- [RBV04]: algorithm for principal ideal lattices of small dimension

[RBV04] G. Rekaya, J.-C. Belfiore, E. Viterbo. A very efficient lattice reduction tool on fast fading channels. ISITA.

State-of-the-art: ideal-SVP

Solving SVP in ideal lattices:

[RBV04]: algorithm for principal ideal lattices of small dimension

[CGS14]: algorithm for principal ideal lattices in cyclotomic fields (without analysis)

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

State-of-the-art: ideal-SVP

Solving SVP in ideal lattices:

[RBV04]: algorithm for principal ideal lattices of small dimension

[CGS14]: algorithm for principal ideal lattices in cyclotomic fields (without analysis)

[CDPR16]: does the analysis of [CGS14]
$\Rightarrow 2^{O(\sqrt{n})}$ approximation factor in quantum poly time

[CDPR16] R. Cramer, L. Ducas, C. Peikert and O. Regev. Recovering short generators of principal ideals in cyclotomic rings. Eurocrypt.

State-of-the-art: ideal-SVP

Solving SVP in ideal lattices:

[RBV04]: algorithm for principal ideal lattices of small dimension

[CGS14]: algorithm for principal ideal lattices in cyclotomic fields (without analysis)

[CDPR16]: does the analysis of [CGS14]
$\Rightarrow 2^{O(\sqrt{n})}$ approximation factor in quantum poly time

[CDW17]: extends results of [CDPR16] to any ideal (in cyclotomic fields)

[CDW17] R. Cramer, L. Ducas, B. Wesolowski. Short stickelberger class relations and application to ideal-SVP. Eurocrypt.

State-of-the-art: ideal-SVP

Solving SVP in ideal lattices:

[RBV04]: algorithm for principal ideal lattices of small dimension

[CGS14]: algorithm for principal ideal lattices in cyclotomic fields (without analysis)

[CDPR16]: does the analysis of [CGS14]
$\Rightarrow 2^{O(\sqrt{n})}$ approximation factor in quantum poly time

[CDW17]: extends results of [CDPR16] to any ideal (in cyclotomic fields)

[PHS19]: extends [CDW17] to obtain more trade-offs (any number field, exponential pre-processing)

[PHS19] A. Pellet-Mary, G. Hanrot, D. Stehlé. Approx-SVP in ideal lattices with pre-processing. Eurocrypt.

State-of-the-art: module-SVP

Adapting LLL to module lattices:

[Nap96] LLL for some specific number fields
no bound on quality / run-time

[Nap96] H. Napias. A generalization of the LLL-algorithm over Euclidean rings or orders. Journal de théorie des nombres de Bordeaux.

State-of-the-art: module-SVP

Adapting LLL to module lattices:

[Nap96]	LLL for some specific number fields no bound on quality / run-time
[FP96]	LLL for any number fields no bound on quality / run-time bound on run-time for specific number fields

[FP96] C. Fieker, M. E. Pohst. Lattices over number fields. ANTS.

State-of-the-art: module-SVP

Adapting LLL to module lattices:

[Nap96]	LLL for some specific number fields no bound on quality / run-time
[FP96]	LLL for any number fields no bound on quality / run-time bound on run-time for specific number fields
[FS10]	forget about the module structure and do LLL in $\mathbb{Z}$

[FS10] C. Fieker, D. Stehlé. Short bases of lattices over number fields. ANTS.

State-of-the-art: module-SVP

Adapting LLL to module lattices:

[Nap96]	LLL for some specific number fields no bound on quality / run-time
[FP96]	LLL for any number fields no bound on quality / run-time bound on run-time for specific number fields
[FS10]	forget about the module structure and do LLL in $\mathbb{Z}$
[KL17]	LLL for norm-Euclidean fields bound on run-time but not on quality bound on quality for biquadratic fields

[KL17] T. Kim, C. Lee. Lattice reductions over euclidean rings with applications to cryptanalysis. IMACC.

State-of-the-art: module-SVP

Adapting LLL to module lattices:

[Nap96]	LLL for some specific number fields no bound on quality / run-time
[FP96]	LLL for any number fields no bound on quality / run-time bound on run-time for specific number fields
[FS10]	forget about the module structure and do LLL in $\mathbb{Z}$
[KL17]	LLL for norm-Euclidean fields bound on run-time but not on quality bound on quality for biquadratic fields
[LPSW19]	LLL for any number field bound on quality and run-time if oracle solving CVP in a fixed lattice

[LPSW19] C. Lee, A. Pellet-Mary, D. Stehlé, A. Wallet. An LLL algorithm for module lattices. To appear at Asiacrypt 2019.

Some mathematical background

First definitions

Notation

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

First definitions

Notation

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

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

First definitions

Notation

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

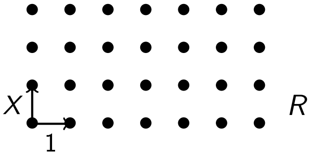
- Units: $R^\times = \{a \in R \mid \exists b \in R, ab = 1\}$
  - e.g. $\mathbb{Z}^\times = \{-1, 1\}$
- Principal ideals: $\langle g \rangle = \{gr \mid r \in R\}$ (i.e., all multiples of g)
  - e.g. $\langle 2 \rangle = \{\text{even numbers}\}$ in $\mathbb{Z}$
  - g is called a generator of $\langle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

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

R is a lattice

$$\begin{aligned} R = \mathbb{Z}[X]/(X^n+1) &\rightarrow \mathbb{C}^n \\ r(X) &\mapsto (r(\alpha_1), r(\alpha_2), \dots, r(\alpha_n)), \end{aligned}$$

where $\alpha_1, \dots, \alpha_n$ are the roots of $X^n + 1$ in $\mathbb{C}$



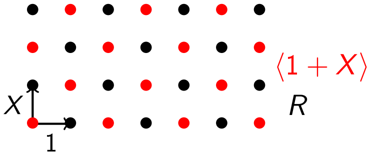
Why is $\langle g \rangle$ a lattice?

R is a lattice

$$\begin{aligned} R = \mathbb{Z}[X]/(X^n+1) &\to \mathbb{C}^n \\ r(X) &\mapsto (r(\alpha_1), r(\alpha_2), \ldots, r(\alpha_n)), \end{aligned}$$

where $\alpha_1, \ldots, \alpha_n$ are the roots of $X^n + 1$ in $\mathbb{C}$

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

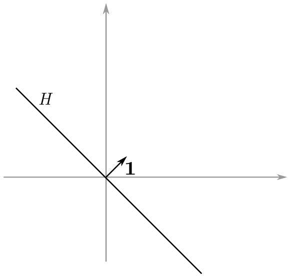


The Log Unit Lattice and Previous Works on ideal-SVP

The Log space

Log : $R \to \mathbb{R}^n$ (take the log of every coordinate)

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



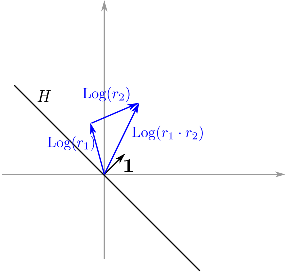
Log : $R \to \mathbb{R}^n$ (take the log of every coordinate)

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

Properties

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

- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$



The Log space

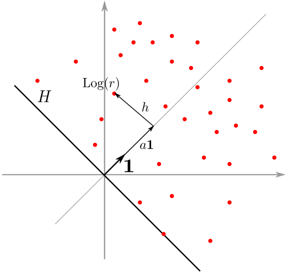
$\text{Log} : R \to \mathbb{R}^n$ (take the log of every coordinate)

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

Properties

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

- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$
- $a \geq 0$



The Log space

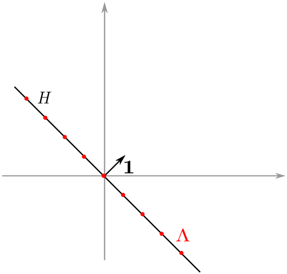
$\text{Log} : R \to \mathbb{R}^n$ (take the log of every coordinate)

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

Properties

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

- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$
- $a \geq 0$
- $a = 0$ iff r is a unit



The Log space

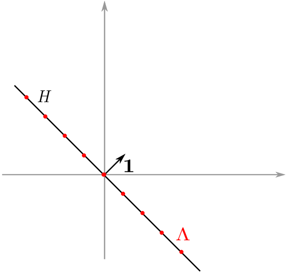
$\text{Log} : R \to \mathbb{R}^n$ (take the log of every coordinate)

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

Properties

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

- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$
- $a \geq 0$
- $a = 0$ iff r is a unit



The Log unit lattice

$\Lambda := \text{Log}(R^\times)$ is a lattice in H.

The Log space

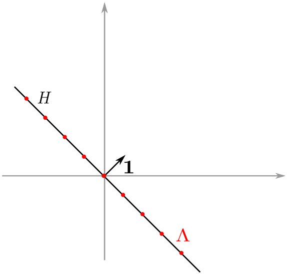
$\text{Log} : R \to \mathbb{R}^n$ (take the log of every coordinate)

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

Properties

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

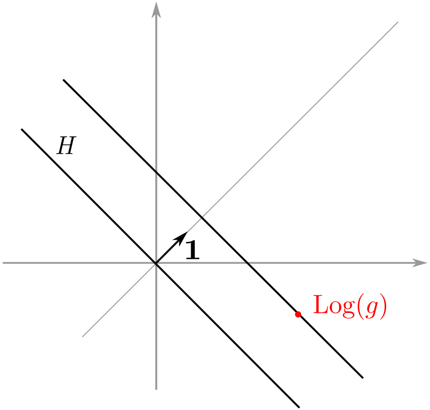
- $\text{Log}(r_1 \cdot r_2) = \text{Log}(r_1) + \text{Log}(r_2)$
- $a \geq 0$
- $a = 0$ iff r is a unit
- $\|r\| \simeq 2^{\|\text{Log}\, r\|_\infty}$



The Log unit lattice

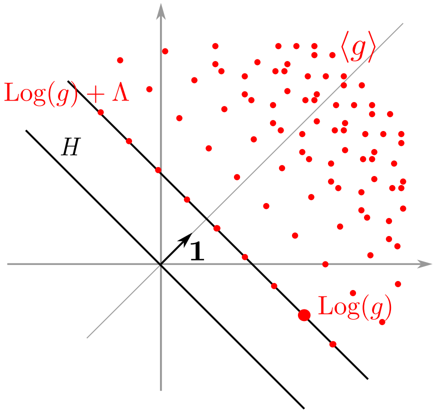
$\Lambda := \text{Log}(R^\times)$ is a lattice in H.

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

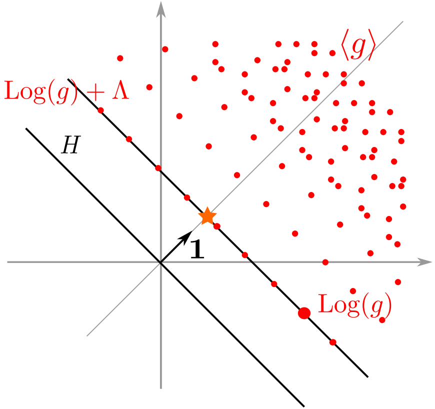


Previous algorithms

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



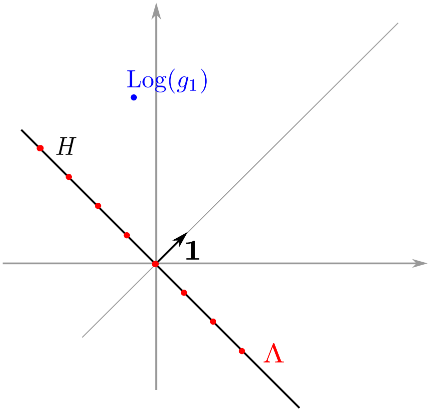
What does $\text{Log}\langle g \rangle$ look like?



Previous algorithms

[CGS14,CDPR16]:

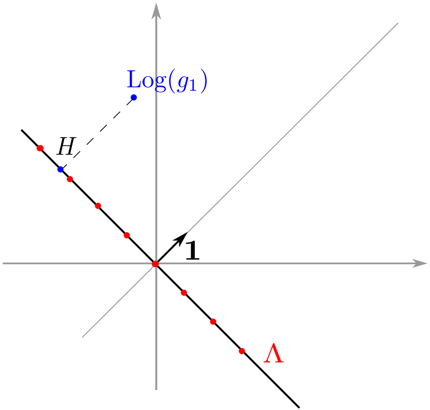
- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum poly time



[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.

[CGS14,CDPR16]:

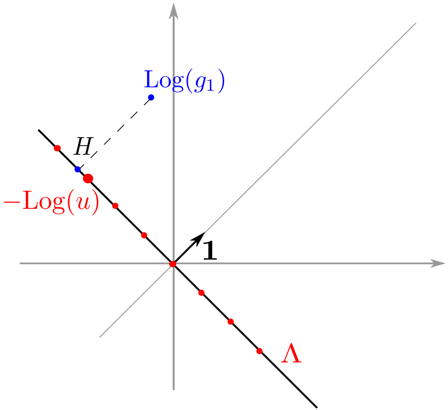
- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum poly time



[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.

[CGS14,CDPR16]:

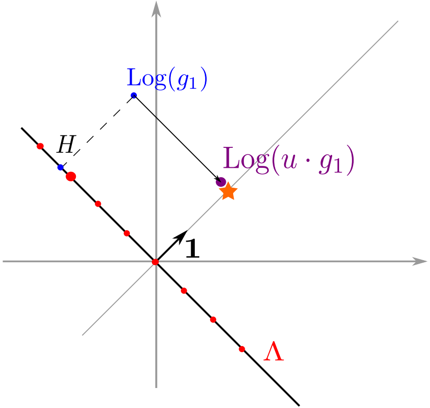
- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum poly time
- Solve CVP in Λ



[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.

[CGS14,CDPR16]:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum poly time
- Solve CVP in Λ



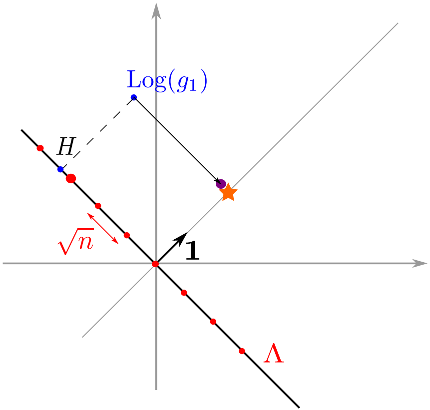
[BS16]: J.-F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields. SODA.

[CGS14,CDPR16]:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum poly time
- Solve CVP in Λ
  - Good basis of Λ (cyclotomic field)

    $\Rightarrow$ 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.

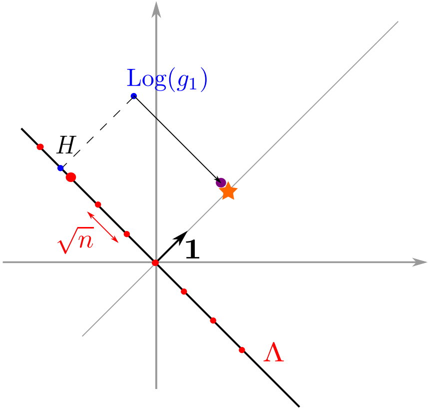
[CGS14,CDPR16]:

- Find a generator g_1 of $\langle g \rangle$.
  - [BS16]: quantum poly time
- Solve CVP in Λ
  - Good basis of Λ (cyclotomic field)

    $\Rightarrow$ CVP in poly time

    $\Rightarrow \|h\| \leq \widetilde{O}(\sqrt{n})$

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

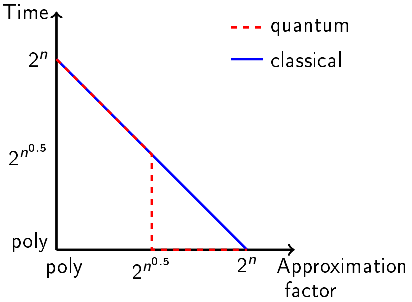


[BS16]: J.-F. Biasse, F. Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields. SODA.

[CGS14,CDPR16]:

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

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

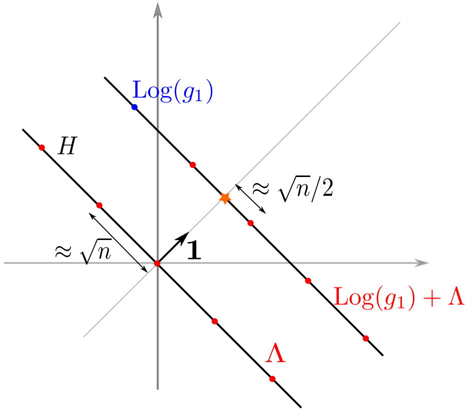


- Heuristic
- Cyclotomic fields

[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.

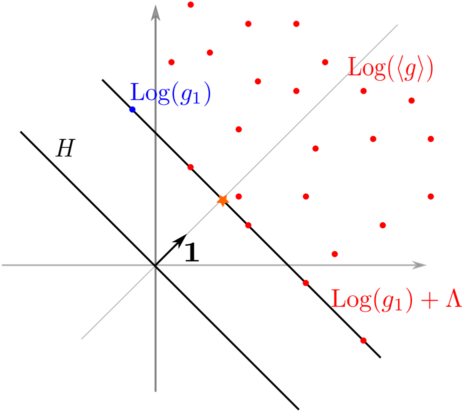
Getting Intermediate Trade-offs, with Pre-processing

Idea

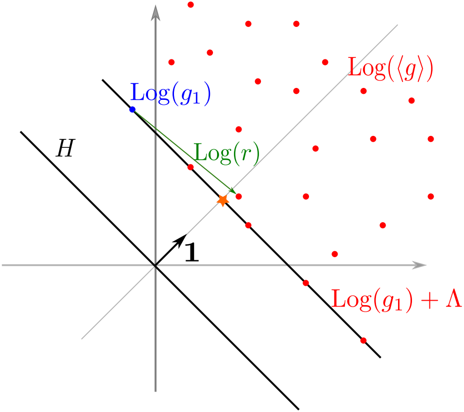


$\mathrm{Log}(g_1)$
H
$\approx \sqrt{n}/2$
$\approx \sqrt{n}$
$\mathbf{1}$
$\mathrm{Log}(g_1) + \Lambda$
Λ

Idea



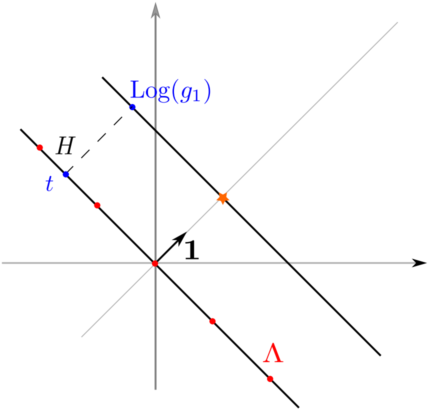
Idea



Important

$\text{Log}\, r = h + a\mathbf{1}$ with a small (and $h \in H$).

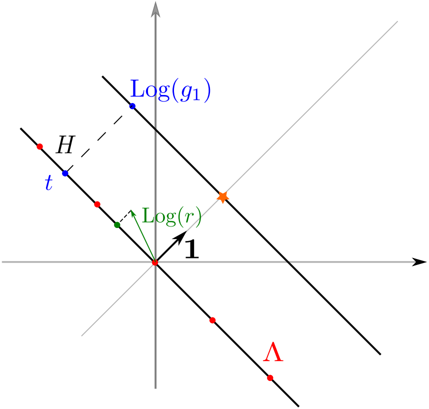
Idea



Important

$\text{Log}\, r = h + a\mathbf{1}$ with a small (and $h \in H$).

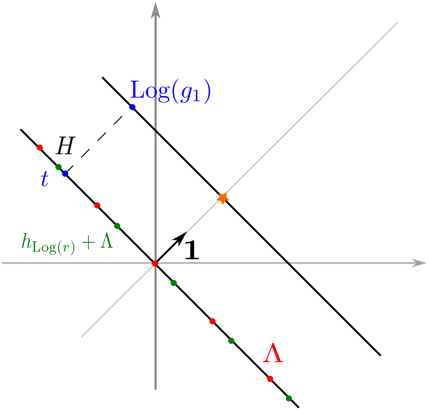
Idea



Important

$\operatorname{Log} r = h + a\mathbf{1}$ with a small (and $h \in H$).

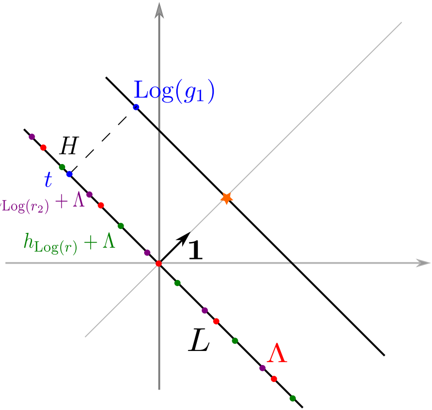
Idea



Important

$\mathrm{Log}\, r = h + a\mathbf{1}$ with a small (and $h \in H$).

Idea



Important

$\mathrm{Log}\, r = h + a\mathbf{1}$ with a small (and $h \in H$).

The lattice L

$$
L = \left(\begin{array}{c|cccc}
\Lambda & h_{\mathrm{Log}(r_1)}, & \cdots & , h_{\mathrm{Log}(r_\nu)} & \\
\hline
 & 1 & & & \\
 & & 1 & & \\
0 & & & \ddots & \\
 & & & & 1
\end{array}\right)
\qquad
t = \left(\begin{array}{c}
h_{Log(g_1)} \\
\hline
 \\
 \\
0 \\
 \\
\end{array}\right)
$$

The lattice L

$$L = \left[\begin{array}{c|cccc} \Lambda & \multicolumn{4}{c}{h_{\mathrm{Log}(r_1)}, \cdots, h_{\mathrm{Log}(r_\nu)}} \\ \hline & 1/\sqrt{n} & & & \\ 0 & & 1/\sqrt{n} & & \\ & & & \ddots & \\ & & & & 1/\sqrt{n} \end{array}\right] \qquad t = \left|\begin{array}{c} h_{Log(g_1)} \\ \hline \\ 0 \\ \\ \end{array}\right.$$

The lattice L

$$L = \left(\begin{array}{c|cccc} \Lambda & h_{\mathrm{Log}(r_1)}, & \cdots & , h_{\mathrm{Log}(r_\nu)} & \\ \hline & 1/\sqrt{n} & & & \\ 0 & & 1/\sqrt{n} & & \\ & & & \ddots & \\ & & & & 1/\sqrt{n} \end{array}\right) \qquad t = \left(\begin{array}{c} h_{Log(g_1)} \\ \hline \\ 0 \\ \\ \end{array}\right)$$

Heuristic

For some $\nu = \widetilde{O}(n)$, the covering radius of L satisfies $\mu(L) = O(1)$.

(i.e., for all target t, there exists $s \in L$ such that $\|t - s\| = O(1)$)

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 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 does not depend on $\langle g \rangle \quad \Rightarrow$ 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 does not depend on $\langle g \rangle$ $\Rightarrow$ Pre-processing on L

[Laa16,DLW19,Ste19]:

- 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)

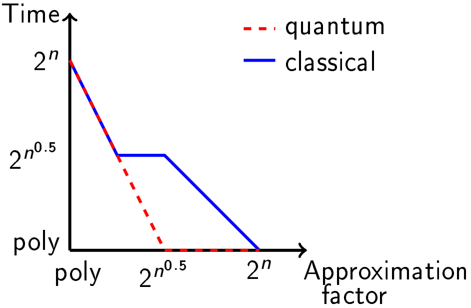
[Laa16] T. Laarhoven. Finding closest lattice vectors using approximate Voronoi cells. SAC.

[DLW19]: E. Doulgerakis, T. Laarhoven, and B. de Weger. Finding closest lattice vectors using approximate Voronoi cells. PQCRYPTO.

[Ste19]: N. Stephens-Davidowitz. A time-distance trade-off for GDD with preprocessing – instantiating the DLW heuristic. CCC.

Conclusion

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



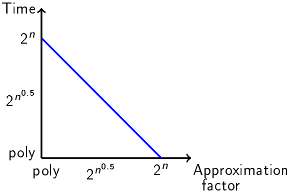
- $2^{O(n)}$ pre-processing
- heuristic
- any number field

Under the carpet

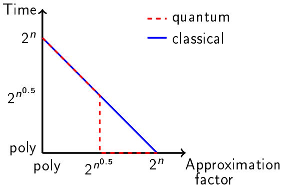
- Any ideal
  - unify units and class group (cf [Buc88])
- Any number field
  - the trade-offs may change with the discriminant
- Heuristics
  - maths justification
  - numerical experiments

[Buc88] J. Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. Séminaire de théorie des nombres.

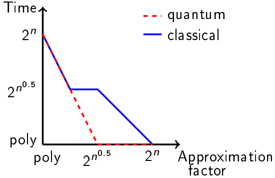
Time/Approximation trade-offs for SVP in ideal lattices:



BKZ algorithm



[CDW17]



[PHS19]
(with $2^{O(n)}$ pre-processing)

(Figures are for prime power cyclotomic fields)

Organization

Ideal-SVP with pre-processing

Eurocrypt 2019, with
G. Hanrot and D. Stehlé

Module-SVP with oracle

- rank 2
- arbitrary rank

Asiacrypt 2019, with
C. Lee, D. Stehlé and A. Wallet

Organization

Ideal-SVP with pre-processing

Eurocrypt 2019, with

G. Hanrot and D. Stehlé

Module-SVP with oracle

- rank 2
- arbitrary rank

Asiacrypt 2019, with

C. Lee, D. Stehlé and A. Wallet

Rank 2 modules

$$M = \begin{pmatrix} M_a & M_b \\ M_c & M_d \end{pmatrix}$$

M_a, M_b, M_c, M_d bases of ideals $\langle a \rangle$, $\langle b \rangle$, $\langle c \rangle$, $\langle d \rangle$ in $R = \mathbb{Z}[X]/(X^n + 1)$

Rank 2 modules

$$M = \begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline \end{array}$$

$$a, b, c, d \in R = \mathbb{Z}[X]/(X^n + 1)$$

Rank 2 modules

$$M = \begin{array}{|c|c|} \hline a & b \\ \hline c & d \\ \hline \end{array}$$

$a, b, c, d \in R = \mathbb{Z}[X]/(X^n + 1)$

$\Rightarrow$"R-lattice" of dimension 2

Rank 2 modules

$$M = \begin{array}{|c|c|} \hline a & b \\ \hline c & d \\ \hline \end{array}$$

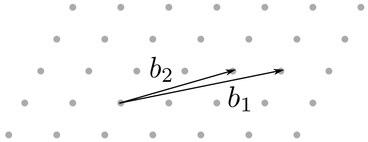
$a, b, c, d \in R = \mathbb{Z}[X]/(X^n + 1)$

$\Rightarrow$"R-lattice" of dimension 2

Can we extend Gauss' algorithm to matrices over R?

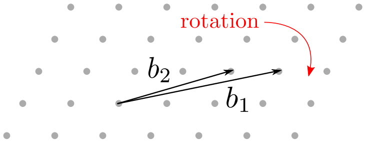
Gauss' Algorithm and Limitations

Gauss' algorithm (over $\mathbb{Z}$)



$$M = \begin{pmatrix} 10 & 7 \\ 2 & 2 \end{pmatrix}$$

Gauss' algorithm (over $\mathbb{Z}$)

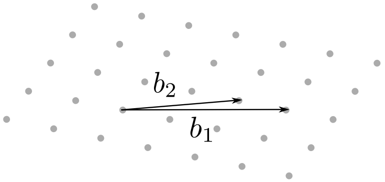


rotation

$$M = \begin{pmatrix} 10 & 7 \\ 2 & 2 \end{pmatrix}$$

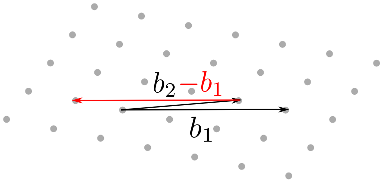
Compute QR factorization

Gauss' algorithm (over $\mathbb{Z}$)



$$M = \begin{pmatrix} 10.2 & 7.3 \\ 0 & 0.6 \end{pmatrix}$$

Gauss' algorithm (over $\mathbb{Z}$)

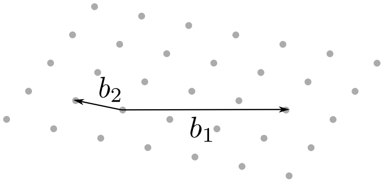


reduce b_2 with b_1

$$M = \begin{pmatrix} 10.2 & 7.3 \\ 0 & 0.6 \end{pmatrix}$$

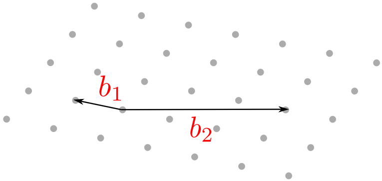
"Euclidean division" (over $\mathbb{R}$) of 7.3 by 10.2

Gauss' algorithm (over $\mathbb{Z}$)



$$M = \begin{pmatrix} 10.2 & -2.9 \\ 0 & 0.6 \end{pmatrix}$$

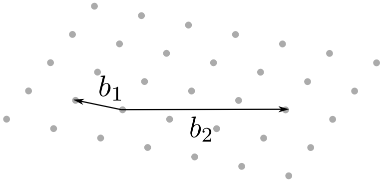
Gauss' algorithm (over $\mathbb{Z}$)



swap

$$M = \begin{pmatrix} -2.9 & 10.2 \\ 0.6 & 0 \end{pmatrix}$$

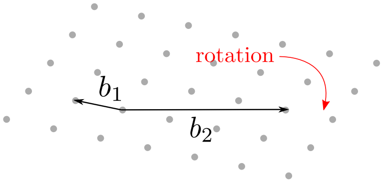
Gauss' algorithm (over $\mathbb{Z}$)



start again

$$M = \begin{pmatrix} -2.9 & 10.2 \\ 0.6 & 0 \end{pmatrix}$$

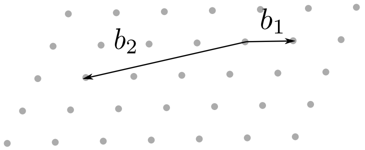
Gauss' algorithm (over $\mathbb{Z}$)



rotation

$$M = \begin{pmatrix} -2.9 & 10.2 \\ 0.6 & 0 \end{pmatrix}$$

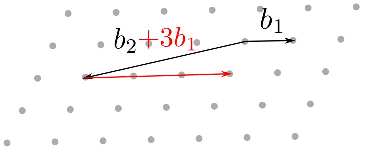
Gauss' algorithm (over $\mathbb{Z}$)



rotation

$$M = \begin{pmatrix} 3 & -10 \\ 0 & -2 \end{pmatrix}$$

Gauss' algorithm (over $\mathbb{Z}$)

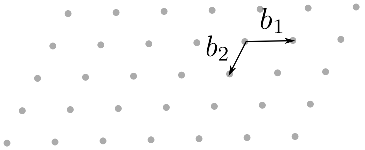


reduce b_2 with b_1

$$M = \begin{pmatrix} 3 & -10 \\ 0 & -2 \end{pmatrix}$$

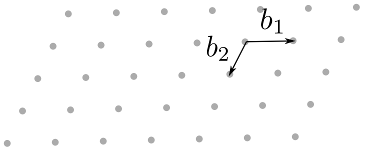
"Euclidean division" (over $\mathbb{R}$) of -10 by 3

Gauss' algorithm (over $\mathbb{Z}$)



$$M = \begin{pmatrix} 3 & -1 \\ 0 & -2 \end{pmatrix}$$

Gauss' algorithm (over $\mathbb{Z}$)

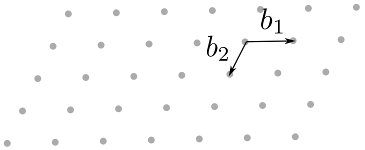


$$M = \begin{pmatrix} 3 & -1 \\ 0 & -2 \end{pmatrix}$$

For Gauss' algorithm over $K_{\mathbb{R}}$, we need

- rotation
- Euclidean division

Gauss' algorithm (over $\mathbb{Z}$)



$$M = \begin{pmatrix} 3 & -1 \\ 0 & -2 \end{pmatrix}$$

For Gauss' algorithm over $K_{\mathbb{R}}$, we need

- rotation $\Rightarrow$ ok
- Euclidean division $\Rightarrow$?

Euclidean division

Over $\mathbb{Z}$

Input: $a, b \in \mathbb{Z}$, $a \neq 0$
Output: $r \in \mathbb{Z}$
such that $|b + ra| \leq |a|/2$

Euclidean division

Over $\mathbb{Z}$

Input: $a, b \in \mathbb{Z}$, $a \neq 0$
Output: $r \in \mathbb{Z}$
such that $|b + ra| \leq |a|/2$

CVP in $\mathbb{Z}$ with target $-b/a$.

Euclidean division

Over $\mathbb{Z}$

Input: $a, b \in \mathbb{Z}$, $a \neq 0$
Output: $r \in \mathbb{Z}$
such that $|b + ra| \leq |a|/2$

CVP in $\mathbb{Z}$ with target $-b/a$.

Over R

CVP in R with target $-b/a$
$\Rightarrow$ output $r \in R$

Euclidean division

Over $\mathbb{Z}$

Input: $a, b \in \mathbb{Z}$, $a \neq 0$
Output: $r \in \mathbb{Z}$
such that $|b + ra| \leq |a|/2$

CVP in $\mathbb{Z}$ with target $-b/a$.

Over R

CVP in R with target $-b/a$
$\Rightarrow$ output $r \in R$

Difficulty: Typically
$\|b + ra\| \approx \sqrt{n} \cdot \|a\| \gg \|a\|$.

Euclidean division

Over $\mathbb{Z}$

Input: $a, b \in \mathbb{Z}$, $a \neq 0$
Output: $r \in \mathbb{Z}$
such that $|b + ra| \leq |a|/2$

CVP in $\mathbb{Z}$ with target $-b/a$.

Over R

CVP in R with target $-b/a$
$\Rightarrow$ output $r \in R$

Difficulty: Typically
$\|b + ra\| \approx \sqrt{n} \cdot \|a\| \gg \|a\|$.

Relax the requirement

Find $x, y \in R$ such that

- $\|xa + yb\| \leq \|a\|/2$
- $\|y\| \leq \mathrm{poly}(n)$

$\Rightarrow$ sufficient for Gauss' algo

Computing the Relaxed Euclidean Division

Using the Log space

Objective: find $x, y \in R$ such that

- $\|xa + yb\| \leq \|a\|/2$
- $\|y\| \leq \mathrm{poly}(n)$

Using the Log space

Objective: find $x, y \in R$ such that

- $\|xa + yb\| \leq \|a\|/2$
- $\|y\| \leq \mathrm{poly}(n)$

Difficulty: Log works well with $\times$, but not with $+$

Solution: If $\|\,\mathrm{Log}(u) - \mathrm{Log}(v)\| \leq \varepsilon$
then $\|u - v\| \lesssim \varepsilon \cdot \min(\|u\|, \|v\|)$
(requires to extend Log to take arguments into account)

Using the Log space

Objective: find $x, y \in R$ such that

- $\|xa + yb\| \leq \|a\|/2$
- $\|y\| \leq \mathrm{poly}(n)$

Difficulty: Log works well with $\times$, but not with $+$

Solution: If $\|\operatorname{Log}(u) - \operatorname{Log}(v)\| \leq \varepsilon$
then $\|u - v\| \lesssim \varepsilon \cdot \min(\|u\|, \|v\|)$
(requires to extend Log to take arguments into account)

New objective

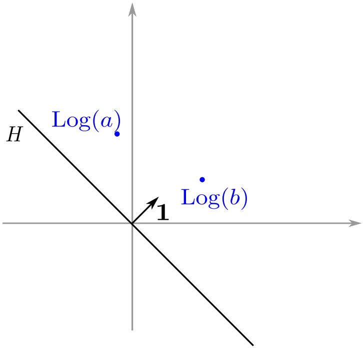
Find $x, y \in R$ such that

- $\|\operatorname{Log}(xa) - \operatorname{Log}(yb)\| \leq \varepsilon$
- $\|\operatorname{Log}(y)\|_\infty \leq O(\log n)$

Idea

Objective: find $x, y \in R$ s.t.

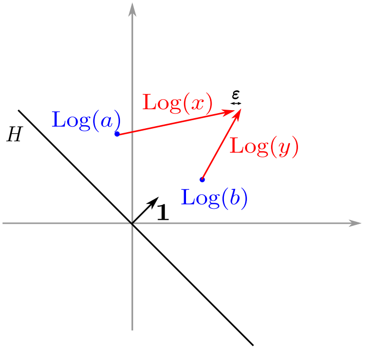
- $\| \mathrm{Log}(xa) - \mathrm{Log}(yb) \| \leq \varepsilon$
- $\| \mathrm{Log}(y) \|_\infty \leq O(\log n)$



Idea

Objective: find $x, y \in R$ s.t.

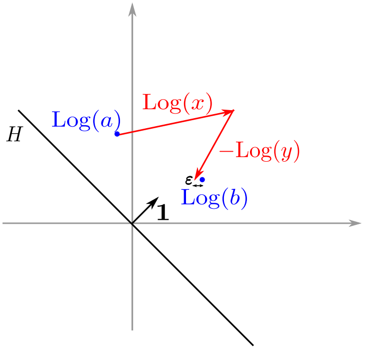
- $\| \text{Log}(xa) - \text{Log}(yb) \| \leq \varepsilon$
- $\| \text{Log}(y) \|_\infty \leq O(\log n)$



Idea

Objective: find $x, y \in R$ s.t.

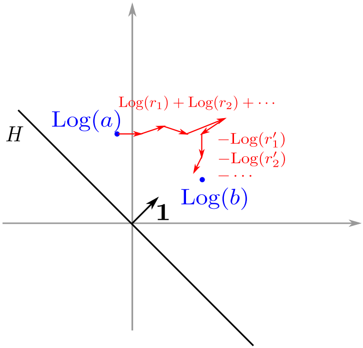
- $\|(\mathrm{Log}(x) - \mathrm{Log}(y)) - \mathrm{Log}(b/a)\| \leq \varepsilon$
- $\|\mathrm{Log}(y)\|_\infty \leq O(\log n)$



Idea

Objective: find $x, y \in R$ s.t.

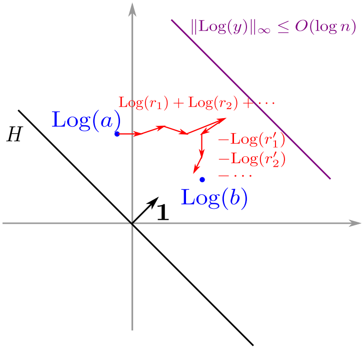
- $\|(\mathrm{Log}(x) - \mathrm{Log}(y)) - \mathrm{Log}(b/a)\| \leq \varepsilon$
- $\|\mathrm{Log}(y)\|_\infty \leq O(\log n)$



Idea

Objective: find $x, y \in R$ s.t.

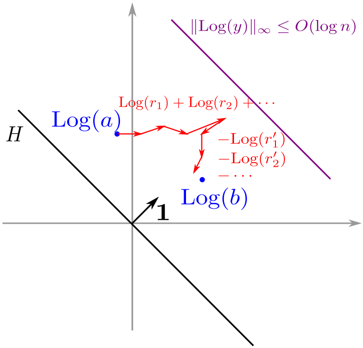
- $\|(\text{Log}(x) - \text{Log}(y)) - \text{Log}(b/a)\| \leq \varepsilon$
- $\|\text{Log}(y)\|_\infty \leq O(\log n)$



Idea

Objective: find $x, y \in R$ s.t.

- $\|(\mathrm{Log}(x) - \mathrm{Log}(y)) - \mathrm{Log}(b/a)\| \leq \varepsilon$
- $\|\,\mathrm{Log}(y)\|_\infty \leq O(\log n)$



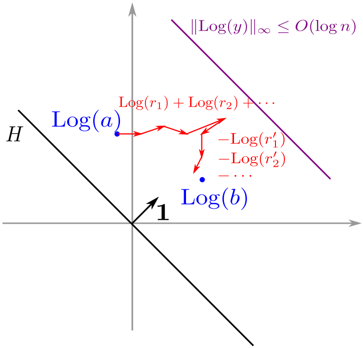
Solve **exact** CVP in L with target t

$$L = \begin{pmatrix} \Lambda & \mathrm{Log}\, r_1 & \cdots & \mathrm{Log}\, r_{n^2} \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}, \quad t = \begin{pmatrix} \mathrm{Log}(b/a) \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Idea

Objective: find $x, y \in R$ s.t.

- $\|(\mathrm{Log}(x) - \mathrm{Log}(y)) - \mathrm{Log}(b/a)\| \leq \varepsilon$
- $\|\mathrm{Log}(y)\|_\infty \leq O(\log n)$



Solve **exact** CVP in L with target t with an oracle

$$L = \begin{pmatrix} \Lambda & \mathrm{Log}\, r_1 & \cdots & \mathrm{Log}\, r_{n^2} \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}, \quad t = \begin{pmatrix} \mathrm{Log}(b/a) \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Applications of the extended division

Complexity of the extended division

Quantum $\mathrm{poly}(n)$ if we have an oracle solving CVP in L

Applications of the extended division

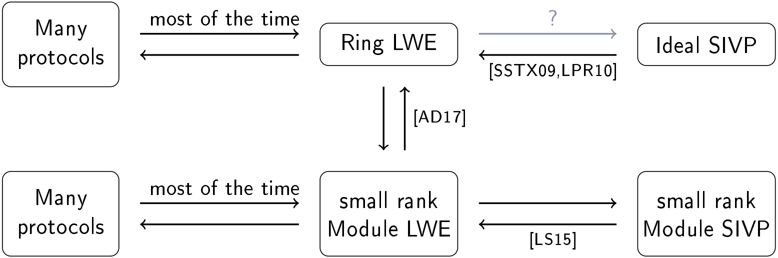
Complexity of the extended division

Quantum $\mathrm{poly}(n)$ if we have an oracle solving CVP in L

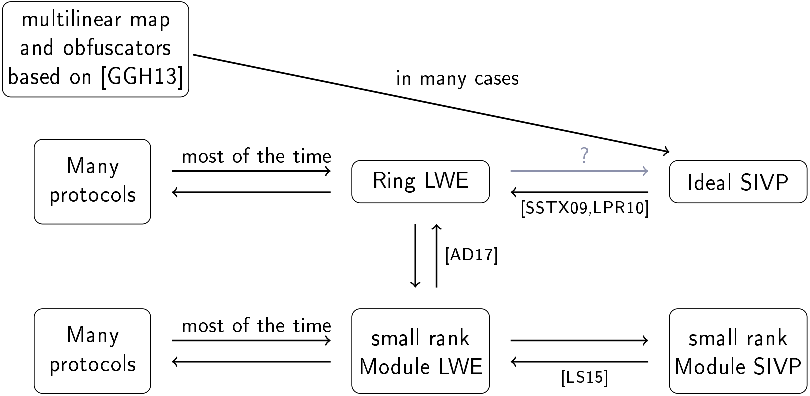
Applications:

- Mimic Gauss' algorithm with 2×2 matrices over R
  - approximation factor $\mathrm{poly}(n)$ for rank-2 modules
- Extend the LLL algorithm to modules of rank m
  - approximation factor $\mathrm{poly}(n)^{O(m)}$ for rank-m modules

Summary and other works



Summary and other works



[GGH13] S. Garg, C. Gentry, S. Halevi. Candidate multilinear maps from ideal lattices. Eurocrypt.

In the thesis

Ideal and Module SVP

Ideal-SVP with pre-processing

Eurocrypt 2019, with G. Hanrot and D. Stehlé

Module-SVP with oracle

- rank 2
- arbitrary rank

Asiacrypt 2019, with C. Lee, D. Stehlé and A. Wallet

GGH13 map and applications

Statistical leakage of GGH13

Asiacrypt 2018, with L. Ducas

Quantum attack on GGH13 based obfuscators

Crypto 2018

Conclusion

Main bottleneck of our algorithms: CVP in L
(one lattice L per number field)

Conclusion

Main bottleneck of our algorithms: CVP in L
(one lattice L per number field)

Perspectives:

- understand the lattice L
  - remove the heuristics?
  - efficient CVP solver for some number fields?

Conclusion

Main bottleneck of our algorithms: CVP in L
(one lattice L per number field)

Perspectives:

- understand the lattice L
  - remove the heuristics?
  - efficient CVP solver for some number fields?
- varying defining polynomial
  - same geometry but different algebraic properties?

Conclusion

Main bottleneck of our algorithms: CVP in L
(one lattice L per number field)

Perspectives:

- understand the lattice L
  - remove the heuristics?
  - efficient CVP solver for some number fields?
- varying defining polynomial
  - same geometry but different algebraic properties?
- remove pre-processing/oracle?
- enumeration/sieving in modules? ($\Rightarrow$ BKZ algorithm for modules)

Conclusion

Main bottleneck of our algorithms: CVP in L
(one lattice L per number field)

Perspectives:

- understand the lattice L
  - remove the heuristics?
  - efficient CVP solver for some number fields?
- varying defining polynomial
  - same geometry but different algebraic properties?
- remove pre-processing/oracle?
- enumeration/sieving in modules? ($\Rightarrow$ BKZ algorithm for modules)

Thank you