格的理论、算法和应用

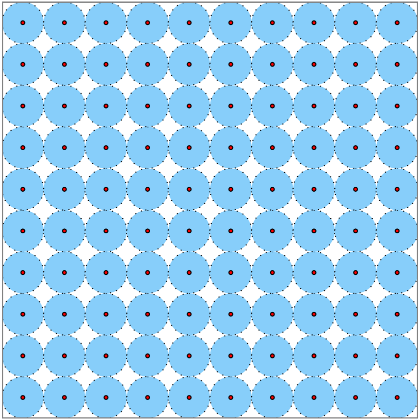
一个入门性的介绍

陈经纬

二〇二二年十一月四日

从一个趣味题开始

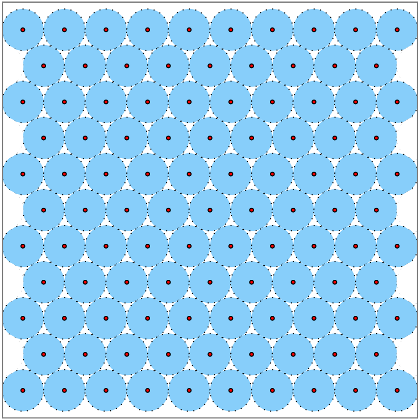
给定一个 10×10 的正方形, 最多可放入多少个直径为 1 的硬币?



可放入 100 个直径为 1 的硬币

从一个趣味题开始

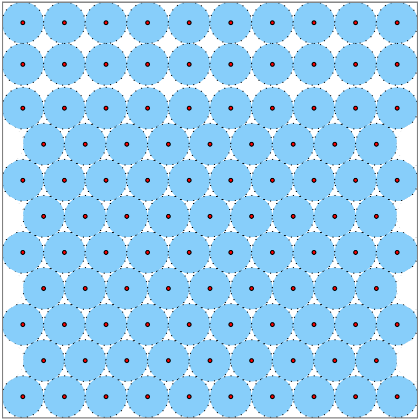
给定一个 10×10 的正方形，最多可放入多少个直径为 1 的硬币？



可放入 105 个直径为 1 的硬币

从一个趣味题开始

给定一个 10×10 的正方形, 最多可放入多少个直径为 1 的硬币?



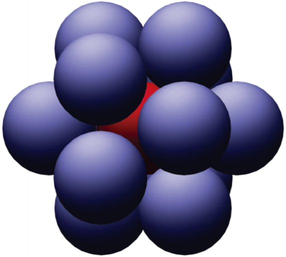
可放入 106 个直径为 1 的硬币

堆球问题

Sir Walter Raleigh (1552–1618)

- Raleigh 爵士: 如何让有限的炮弹仓尽量多地携带加农炮?

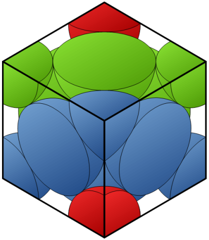
Thomas Harriot (1560–1621)



Harriot 给出的堆球方案

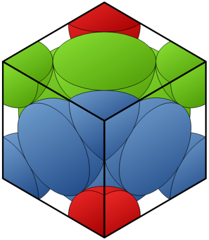
- Harriot: 每个球都恰好跟 12 个球相切.

Harriot 方案的堆球密度



面心立方填充 (face-centered cubic)

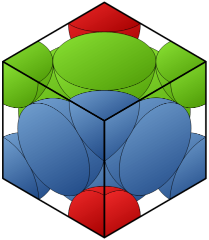
Harriot 方案的堆球密度



面心立方填充 (face-centered cubic)

- 4 个半径为 r 的球的体积: $4 \cdot \mathrm{vol}(\mathscr{B}(\mathbf{0}, r)) = \dfrac{16}{3}\pi r^3$.

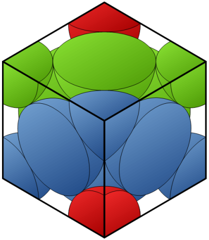
Harriot 方案的堆球密度



面心立方填充 (face-centered cubic)

- 4 个半径为 r 的球的体积: $4 \cdot \mathrm{vol}(\mathcal{B}(\mathbf{0}, r)) = \frac{16}{3}\pi r^3$.
- 边长为 $a = 2\sqrt{2}\, r$ 的立方体体积: $a^3 = 16\sqrt{2}\, r^3$.

Harriot 方案的堆球密度



面心立方填充 (face-centered cubic)

- 4 个半径为 r 的球的体积: $4 \cdot \mathrm{vol}(\mathscr{B}(\mathbf{0}, r)) = \dfrac{16}{3}\pi r^3$.
- 边长为 $a = 2\sqrt{2}\,r$ 的立方体体积: $a^3 = 16\sqrt{2}\,r^3$.
- 堆球密度: $\dfrac{\pi}{\sqrt{18}} \approx 0.74$.

Johannes Kepler (1571–1630)

猜想 (J. Kepler. *The Six-Cornered Snow Flake*, 1611)[1]

在一个容器中堆放同样的小球, 所能得到的最大密度是 $\pi/\sqrt{18}$.

[1] 1998 年, 被 Thomas Hales 用计算机程序证明; 2014 年完成形式化验证.

Gauß 的无心插柳

Carl Friedrich Gauß (1777–1855)

Gauß 的贡献

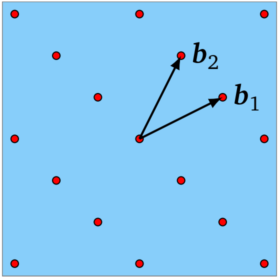
在三维空间中堆同样大小的球. 若它们的球心构成一个格 (或者格的一部分), 那么堆球的密度不会超过 $\pi/\sqrt{18}$.

格 (Lattice)

若 $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n \in \mathbb{R}^n$ 线性无关, 则称

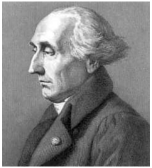
$$\Lambda = \left\{ \sum_{i=1}^{n} z_i \boldsymbol{b}_i : z_i \in \mathbb{Z} \right\}$$

是由 $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n$ 生成的一个格. 称这组向量为格 Λ 的一个基.



由 $\boldsymbol{b}_1$ 和 $\boldsymbol{b}_2$ 生成的一个 2-维格

格的数学名人堂



Joseph-Louis Lagrange
(1736–1813)

Charles Hermite
(1822–1901)

Hermann Minkowski
(1864–1909)

研究过格的 Fields 奖得主

- Gregori Aleksandrovich Margulis (1978)
- Elon Lindenstrauss (2010)
- Stanislav Smirnov (2010)
- Manjul Bhargava (2014)
- Akshay Venkatesh (2018)
- Maryna Viazovska (2022)

提纲

1. 格的理论概要
   - 格的定义
   - 格的不变量
   - 以 q-ary 格为例
   - 格中的计算问题
2. 格基约化算法简介
   - Lagrange 算法
   - LLL 格基约化算法
   - BKZ 算法
3. 应用举例
   - 背包问题的求解
   - 求解 LWE 问题的几何方法

提纲

1. 格的理论概要
   - 格的定义
   - 格的不变量
   - 以 q-ary 格为例
   - 格中的计算问题
2. 格基约化算法简介
3. 应用举例

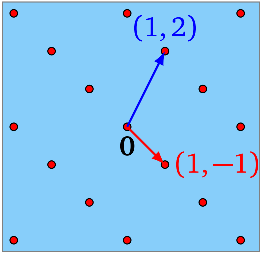
格的定义

设矩阵 $\boldsymbol{B} = (\boldsymbol{b}_1, \cdots, \boldsymbol{b}_d) \in \mathbb{R}^{n\times d}$ 列满秩. 定义由 $\boldsymbol{B}$ 生成的格是

$$\Lambda = \mathcal{L}(\boldsymbol{B}) = \boldsymbol{B}\cdot\mathbb{Z}^d = \{\boldsymbol{B}\boldsymbol{z} \colon \boldsymbol{z} \in \mathbb{Z}^d\} = \left\{ \sum_{i=1}^{d} z_i \cdot \boldsymbol{b}_i : \forall i,\, z_i \in \mathbb{Z} \right\}.$$

矩阵 $\boldsymbol{B}$ 称为格 Λ 的一个基. 整数 d 被称作格的秩, 记作 $\operatorname{rank}(\Lambda)$.
若 $d = n$, 则 $\Lambda = \mathcal{L}(\boldsymbol{B})$ 被称作满秩格.

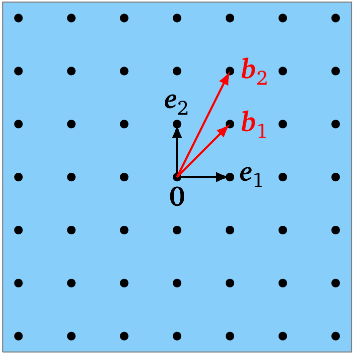
$$\boldsymbol{B} = \begin{pmatrix} 1 & 1 \\ -1 & 2 \end{pmatrix}$$



$\Lambda = \mathcal{L}(\boldsymbol{B})$

格的基

设 Λ 是秩为 d 的格. 当 $d \geq 2$ 时, Λ 可以被不同的基表示.



$$(\boldsymbol{e}_1, \boldsymbol{e}_2) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

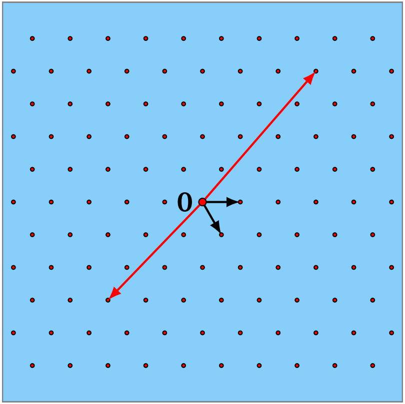
$$(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$$

$$\boldsymbol{e}_1 = 2\boldsymbol{b}_1 - \boldsymbol{b}_2$$

$$\boldsymbol{e}_2 = -\boldsymbol{b}_1 + \boldsymbol{b}_2$$

单位矩阵 $\boldsymbol{I}_2$ 和 $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ 都是格 $\mathbb{Z}^2$ 的基

基的质量

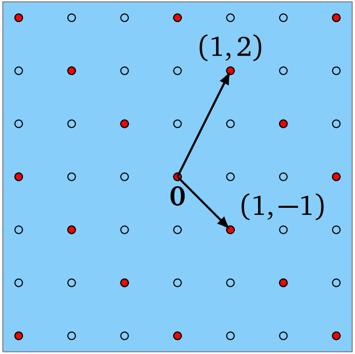


好基的判断标准

- 向量长度较短;
- 向量间接近正交 (垂直).

设 Λ 是一个格. 称 Λ' 为 Λ 的一个子格, 若 Λ' 满足:

- $\Lambda' \subset \Lambda$;
- Λ' 是一个格.



$\mathbb{Z}^2 = \mathscr{L}(\boldsymbol{I}_2)$ 的子格

幺模矩阵: $\mathbb{Z}$ 上的可逆矩阵

称 $\boldsymbol{U} \in \mathbb{Z}^{n\times n}$ 在 $\mathbb{Z}$ 上可逆, 若存在 $\boldsymbol{V} \in \mathbb{Z}^{n\times n}$ 使得 $\boldsymbol{U}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{U} = \boldsymbol{I}_n$.

定理

设 $\boldsymbol{U}$ 为整数方阵. 则 $\boldsymbol{U}$ 在 $\mathbb{Z}$ 上可逆当且仅当 $|\det(\boldsymbol{U})| = 1$.

因此, 亦称 $\mathbb{Z}$ 上的可逆矩阵为幺模矩阵.

定理

设 $\boldsymbol{B} \in \mathbb{R}^{n\times d}$ 和 $\boldsymbol{C} \in \mathbb{R}^{n\times d}$ 为两个格基. 则 $\mathscr{L}(\boldsymbol{B}) = \mathscr{L}(\boldsymbol{C})$ 的充要条件是存在 d 阶幺模矩阵 $\boldsymbol{U}$ 使得 $\boldsymbol{B} = \boldsymbol{C}\boldsymbol{U}$.

证明

$(\Leftarrow)$: 由 $\boldsymbol{U}$ 是幺模矩阵且 $\boldsymbol{B} = \boldsymbol{C}\boldsymbol{U}$ 知

$$\mathscr{L}(\boldsymbol{B}) = \mathscr{L}(\boldsymbol{C}\boldsymbol{U}) \subseteq \mathscr{L}(\boldsymbol{C}) = \mathscr{L}(\boldsymbol{B}\boldsymbol{U}^{-1}) \subseteq \mathscr{L}(\boldsymbol{B}).$$

整数初等变换

矩阵 $\boldsymbol{B} \in \mathbb{R}^{n\times d}$ 的一个整数初等列变换有以下三种:

- swap(i, j): $(\boldsymbol{b}_i, \boldsymbol{b}_j) := (\boldsymbol{b}_j, \boldsymbol{b}_i), i \neq j$.

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

- invert(i): $\boldsymbol{b}_i := (-\boldsymbol{b}_i)$, $i = 1, \cdots, n$.

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

- add(i, c, j): $\boldsymbol{b}_i := \boldsymbol{b}_i + c \cdot \boldsymbol{b}_j, i \neq j$ 且 $c \in \mathbb{Z}$.

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0 \\ c & 1 \end{pmatrix}$$

以上三种整数初等列变换都是幺模变换, 从而不改变原来的格.

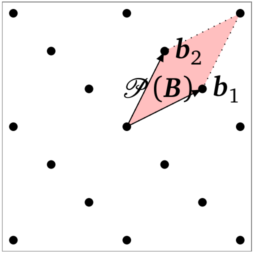
提纲

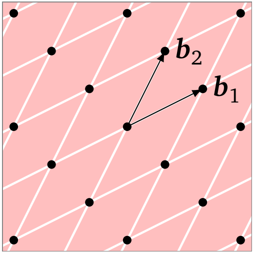
1. 格的理论概要
   - 格的定义
   - 格的不变量
   - 以 q-ary 格为例
   - 格中的计算问题
2. 格基约化算法简介
3. 应用举例

基本平行六面体

设 $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ 为格 Λ 的一个基. 定义格 $\Lambda = \mathscr{L}(\boldsymbol{B})$ 的基本平行六面体 为

$$\mathscr{P}(\boldsymbol{B}) = \boldsymbol{B}[0,1)^d = \left\{ \sum_{i=1}^{d} z_i \cdot \boldsymbol{b}_i : \forall i, 0 \le z_i < 1 \right\}.$$

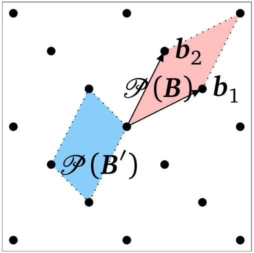


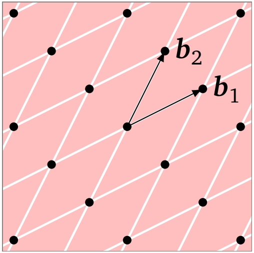


基本平行六面体

设 $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ 为格 Λ 的一个基. 定义格 $\Lambda = \mathscr{L}(\boldsymbol{B})$ 的基本平行六面体 为

$$\mathscr{P}(\boldsymbol{B}) = \boldsymbol{B}[0, 1)^d = \left\{ \sum_{i=1}^{d} z_i \cdot \boldsymbol{b}_i : \forall i, 0 \le z_i < 1 \right\}.$$





不同的基定义不同的基本平行六面体

格的行列式

命题

设 $B \in \mathbb{R}^{m\times d}$ 的列是格 Λ 的一个基, 其基本平行六面体为 $\mathscr{P}(B)$. 则 $\mathscr{P}(B)$ 的 d-维体积

$$\mathrm{vol}(\mathscr{P}(B)) = \sqrt{\det(B^{\mathrm{T}}B)}.$$

推论

设 B 和 C 是格 Λ 的任意两个基. 证明: $\mathrm{vol}(\mathscr{P}(B)) = \mathrm{vol}(\mathscr{P}(C))$.

定义

定义格 Λ 的行列式为其任意基本平行六面体的体积, 记为 $\det(\Lambda)$.

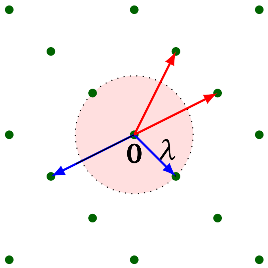
性质

格的行列式是幺模变换下的不变量.

最小距离

对任意的格 $\Lambda \subseteq \mathbb{R}^n$，定义其最小距离 (又称 Minkowski 极小值)为任意两个格点间距离的最小值:

$$\begin{aligned}\lambda(\Lambda) &= \min\{\|\boldsymbol{x}-\boldsymbol{y}\| : \boldsymbol{x},\boldsymbol{y}\in\Lambda, \boldsymbol{x}\neq\boldsymbol{y}\} \\ &= \min\{\|\boldsymbol{b}\| : \boldsymbol{b}\in\Lambda\setminus\boldsymbol{0}\}.\end{aligned}$$

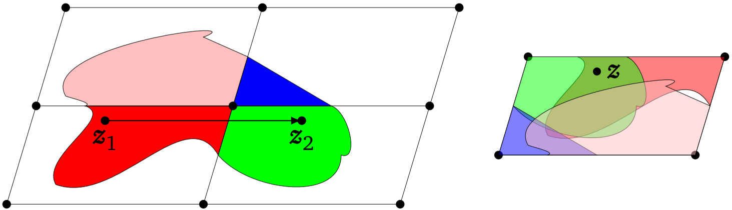


性质

格的最小距离是幺模变换下的不变量.

Blichfeldt 引理

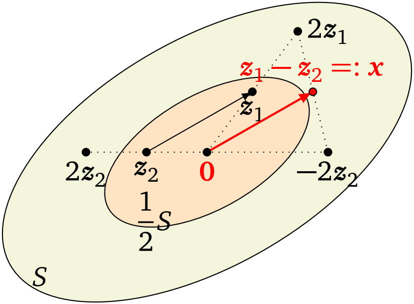
设 Λ 是一个格, 并设集合 $S \subseteq \mathrm{Span}(\Lambda)$ 有体积. 若 $\mathrm{vol}(S) > \det(\Lambda)$, 则存在 $\boldsymbol{z}_1 \neq \boldsymbol{z}_2 \in S$ 使得 $\boldsymbol{z}_1 - \boldsymbol{z}_2 \in \Lambda$.



Blichfeldt 引理

Minkoski 凸胞定理

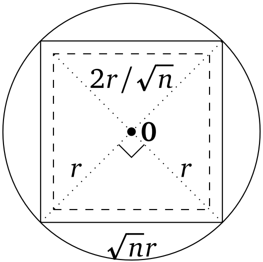
设 $\Lambda \subseteq \mathbb{R}^n$ 为一个满秩格. 若 $S \in \mathbb{R}^n$ 是一个中心对称的凸胞且 $\mathrm{vol}(S) > 2^n \det(\Lambda)$, 则存在 $\boldsymbol{x} \in \Lambda \setminus \{\boldsymbol{0}\}$ 使得 $\boldsymbol{x} \in S$.



Minkowski 凸胞定理

Minkowski 第一定理

对任意的满秩格 $\Lambda \in \mathbb{R}^n$, $\lambda(\Lambda) \leq \sqrt{n}\,\det(\Lambda)^{1/n}$.



圆的内接正方形: $r = \lambda(\Lambda)$

$$\left(\frac{2\lambda(\Lambda)}{\sqrt{n}}\right)^n \leq \mathrm{vol}(\mathscr{B}(\mathbf{0}, \lambda(\Lambda))) \leq 2^n \det(\Lambda).$$

提纲

1. 格的理论概要
   - 格的定义
   - 格的不变量
   - 以 q-ary 格为例
   - 格中的计算问题
2. 格基约化算法简介
3. 应用举例

q-ary 格：定义

设 $m \geq n \geq 1$，$q \geq 2$ 为素数，$\boldsymbol{A} \in \mathbb{Z}_q^{m\times n}$. 则

$$\Lambda_q(\boldsymbol{A}) := \boldsymbol{A} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m \subseteq \mathbb{Z}^m$$

是一个格，被称作由 $\boldsymbol{A}$ 生成的 q-ary 格.

q-ary 格的基

设 $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{pmatrix}$，其中 $\boldsymbol{A}_1 \in \mathbb{Z}_q^{n\times n}$ 在 $\mathbb{Z}_q$ 上可逆，$\boldsymbol{A}_2 \in \mathbb{Z}^{(m-n)\times n}$. 则

$$\begin{aligned}
\Lambda_q(\boldsymbol{A}) &= \begin{pmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{pmatrix} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m = \begin{pmatrix} \boldsymbol{I}_n \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} \end{pmatrix} \cdot \boldsymbol{A}_1\mathbb{Z}_q^n + q \cdot \mathbb{Z}^m \\
&= \begin{pmatrix} \boldsymbol{I}_n \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} \end{pmatrix} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m = \begin{pmatrix} \boldsymbol{I}_n & q\boldsymbol{I}_n & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^{m+n} \\
&= \begin{pmatrix} \boldsymbol{I}_n & & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & -q\boldsymbol{A}_2\boldsymbol{A}_1^{-1} & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^{m+n} = \begin{pmatrix} \boldsymbol{I}_n & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^m.
\end{aligned}$$

q-ary 格：不变量

- 设 $m \geq n \geq 1, q \geq 2$ 为素数, $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{pmatrix} \in \mathbb{Z}_q^{m\times n}$.
- $\boldsymbol{A}_1 \in \mathbb{Z}_q^{n\times n}$ 在 $\mathbb{Z}_q$ 上可逆.

$$\Lambda_q(\boldsymbol{A}) := \boldsymbol{A} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m = \begin{pmatrix} \boldsymbol{I}_n & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^m.$$

- $\operatorname{rank}(\Lambda_q(\boldsymbol{A})) = m$

q-ary 格：不变量

- 设 $m \geq n \geq 1, q \geq 2$ 为素数, $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{pmatrix} \in \mathbb{Z}_q^{m\times n}$.
- $\boldsymbol{A}_1 \in \mathbb{Z}_q^{n\times n}$ 在 $\mathbb{Z}_q$ 上可逆.

$$\Lambda_q(\boldsymbol{A}) := \boldsymbol{A} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m = \begin{pmatrix} \boldsymbol{I}_n & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^m.$$

- $\mathrm{rank}(\Lambda_q(\boldsymbol{A})) = m$
- $\det(\Lambda_q(\boldsymbol{A})) = q^{m-n}$

q-ary 格：不变量

- 设 $m \geq n \geq 1, q \geq 2$ 为素数, $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{pmatrix} \in \mathbb{Z}_q^{m\times n}$.
- $\boldsymbol{A}_1 \in \mathbb{Z}_q^{n\times n}$ 在 $\mathbb{Z}_q$ 上可逆.

$$\Lambda_q(\boldsymbol{A}) := \boldsymbol{A} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m = \begin{pmatrix} \boldsymbol{I}_n & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^m.$$

- $\operatorname{rank}(\Lambda_q(\boldsymbol{A})) = m$
- $\det(\Lambda_q(\boldsymbol{A})) = q^{m-n}$
- 由 Minkowski 第一定理知：$\lambda(\Lambda_q(\boldsymbol{A})) \leq \min\left\{\sqrt{m} \cdot q^{\frac{m-n}{m}}, q\right\}$

q-ary 格：不变量

- 设 $m \geq n \geq 1, q \geq 2$ 为素数, $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{pmatrix} \in \mathbb{Z}_q^{m\times n}$.
- $\boldsymbol{A}_1 \in \mathbb{Z}_q^{n\times n}$ 在 $\mathbb{Z}_q$ 上可逆.

$$\Lambda_q(\boldsymbol{A}) := \boldsymbol{A} \cdot \mathbb{Z}_q^n + q \cdot \mathbb{Z}^m = \begin{pmatrix} \boldsymbol{I}_n & \\ \boldsymbol{A}_2\boldsymbol{A}_1^{-1} & q\boldsymbol{I}_{m-n} \end{pmatrix} \cdot \mathbb{Z}^m.$$

- $\operatorname{rank}(\Lambda_q(\boldsymbol{A})) = m$
- $\det(\Lambda_q(\boldsymbol{A})) = q^{m-n}$
- 由 Minkowski 第一定理知：$\lambda(\Lambda_q(\boldsymbol{A})) \leq \min\left\{\sqrt{m} \cdot q^{\frac{m-n}{m}}, q\right\}$
- 若 $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{m\times n}$，则上面的界在相差常数倍的意义下是紧的.

- q-ary 格: $\Lambda_q(\boldsymbol{A}) = \boldsymbol{A}\cdot\mathbb{Z}_q^n + q\cdot\mathbb{Z}^m$
- 由 Minkowski 第一定理知：$\lambda_1(\Lambda_q(\boldsymbol{A})) \le \min\left\{\sqrt{m}\cdot q^{\frac{m-n}{m}}, q\right\}$
- $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{m\times n} \Rightarrow \Pr\left[\lambda(\Lambda_q(\boldsymbol{A})) \le \frac{1}{2}\sqrt{m}q^{\frac{m-n}{m}}\right] \lesssim 2^{-m}$.

$$
\begin{aligned}
&\Pr[\lambda(\Lambda_q(\boldsymbol{A})) \le B] \\
=&\Pr[\exists \boldsymbol{s}\in\mathbb{Z}_q^n\setminus\boldsymbol{0}, \exists \boldsymbol{y}\in\mathbb{Z}^m \text{ 使得 } \boldsymbol{y} = \boldsymbol{A}\boldsymbol{s} \mod q, \text{ 并且 } \|\boldsymbol{y}\|\le B] \\
\le& \sum_{\substack{\boldsymbol{s}\in\mathbb{Z}_q^n\setminus\boldsymbol{0}\\ \boldsymbol{y}\in\mathbb{Z}^m,\|\boldsymbol{y}\|\le B}} \Pr[\boldsymbol{y} = \boldsymbol{A}\boldsymbol{s} \mod q] \le \sum_{\substack{\boldsymbol{s}\in\mathbb{Z}_q^n\setminus\boldsymbol{0}\\ \boldsymbol{y}\in\mathbb{Z}^m,\|\boldsymbol{y}\|\le B}} \prod_{i\le m} \Pr[\langle\boldsymbol{a}_i,\boldsymbol{s}\rangle = y_i \mod q] \\
\le& \sum_{\substack{\boldsymbol{s}\in\mathbb{Z}_q^n\setminus\boldsymbol{0}\\ \boldsymbol{y}\in\mathbb{Z}^m,\|\boldsymbol{y}\|\le B}} \prod_{i\le m}\frac{1}{q} \lesssim q^n\cdot B^m\cdot\frac{1}{\sqrt{m}^m}\cdot q^{-m}.
\end{aligned}
$$

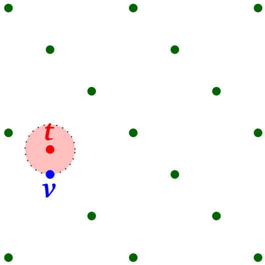
提纲

搜索版 CVP

给定 $\boldsymbol{t} \in \mathbb{Q}^n$ 和格的一个基 $\boldsymbol{B} \in \mathbb{Z}^{n\times n}$, 求满足

$$\|\boldsymbol{v} - \boldsymbol{t}\| = \min_{\boldsymbol{x}\in\mathbb{Z}^n} \|\boldsymbol{B}\boldsymbol{x} - \boldsymbol{t}\| =: \mathrm{dist}(\boldsymbol{t}, \Lambda)$$

的 $\boldsymbol{v} \in \Lambda = \mathscr{L}(\boldsymbol{B})$.



CVP 问题的变种及其困难性

- 判定版: 给定 $(\boldsymbol{B}, \boldsymbol{t}, d)$，其中 $d \in \mathbb{Q}$, 判定是否存在满足

$$\|\boldsymbol{v} - \boldsymbol{t}\| \leq d$$

  的 $\boldsymbol{v} \in \mathscr{L}(\boldsymbol{B})$.

- CVP_γ: 给定 $\boldsymbol{t}$ 和格的基 $\boldsymbol{B}$, 求满足

$$\|\boldsymbol{v} - \boldsymbol{t}\| \leq \gamma \cdot \mathrm{dist}(\boldsymbol{t}, \Lambda), \quad \gamma \geq 1$$

  的 $\boldsymbol{v} \in \mathscr{L}(\boldsymbol{B})$.

- BDD_γ (Bounded distance decoding): 对 $\gamma > 0$, 给定

$$\mathrm{dist}(\boldsymbol{t}, \Lambda) \leq \gamma \cdot \lambda_1(\Lambda),$$

  求距离 $\boldsymbol{t}$ 最近的 $\boldsymbol{v} \in \Lambda$.

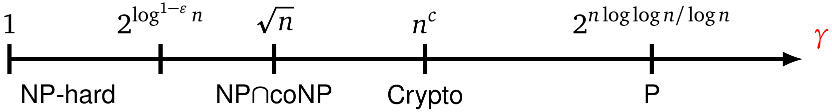
- 搜索版 CVP $\leq$ 判定版 CVP.
- 判定版 CVP 是 NP-完全的.

最短向量问题 (SVP) 及其复杂性

- 搜索版 SVP: 给一个格基 $\boldsymbol{B} \in \mathbb{Z}^{n\times n}$, 计算 $\boldsymbol{v} \in \mathscr{L}(\boldsymbol{B})$ 使得
$$\|\boldsymbol{v}\| = \lambda(\mathscr{L}(\boldsymbol{B})).$$
- 判定版 SVP: 给定一个格基 $\boldsymbol{B}$ 和 $\mu \in \mathbb{Q}$, 判断下式是否成立
$$\lambda(\mathscr{L}(\boldsymbol{B})) \leq \mu.$$
- SVP_γ: 给定 $\boldsymbol{B}$, 求 $\boldsymbol{v} \in \Lambda = \mathscr{L}(\boldsymbol{B})$ 使得
$$\|\boldsymbol{v}\| \leq \gamma \cdot \lambda(\mathscr{L}(\boldsymbol{B})).$$
- GapSVP_γ: 给定 $\boldsymbol{B}$ 和 $\mu \in \mathbb{Q}$, 判断属于下面哪种情况
$$\lambda(\mathscr{L}(\boldsymbol{B})) \leq \mu \quad \text{或} \quad \lambda(\mathscr{L}(\boldsymbol{B})) \geq \gamma \cdot \mu.$$

- 搜索版 SVP $\leq$ 判定版 SVP.
- 对常数 γ, GapSVP_γ 在随机归约下是 NP-难问题. (Khot '04)

GapSVP$_\gamma$ 的困难性



Taken from (Regev, Crypto '06)

提纲

1 格的理论概要

2 格基约化算法简介
- Lagrange 算法
- LLL 格基约化算法
- BKZ 算法

3 应用举例

格基约化算法

给定格的一个基, 通过一系列幺模变换逐步改善基的质量, 得到该格一个质量更好的基是常常采用的一种计算策略. 称这种策略为格基约化 (lattice basis reduction).

一维情形: Euclid 算法 (辗转相除法)

将 $\mathbb{R}$ 看成是一个一维欧氏空间, 则整数 a 和 b 是 $\mathbb{R}$ 中的两个“向量”, 它们生成的格是

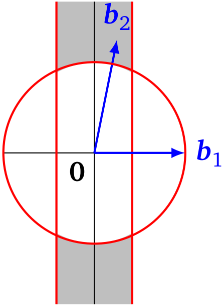
$$\Lambda = \{sa + tb : s, t \in \mathbb{Z}\}.$$

- $\Lambda \subseteq \mathbb{Z}$ 是由 a 和 b 生成的一个理想 $\Lambda = \langle d \rangle$, $d = \gcd(a, b)$.
- d 是 Λ 中最小的正整数 且 Λ 中的每一个元素都是 d 的倍数.
- “向量” d 形成了 Λ 的一个基, 并且 $\lambda(\Lambda) = \|d\| = d$.

定理

对任意的二维格 Λ, 存在 Λ 的一组基 $\boldsymbol{b}_1$ 和 $\boldsymbol{b}_2$ 使得

- $\|\boldsymbol{b}_1\| = \lambda_1(\Lambda)$.
- $|\langle \boldsymbol{b}_2, \boldsymbol{b}_1\rangle| \leq \frac{1}{2}\|\boldsymbol{b}_1\|^2$.



Lagrange 约化基

Lagrange 算法 (1773)

输入: 二维格 Λ 的一个基 $(\boldsymbol{b}_1, \boldsymbol{b}_2)$.
输出: 格 Λ 的一个 Lagrange 约化基 $(\boldsymbol{b}_1, \boldsymbol{b}_2)$.

1: **repeat**
2: $\quad (\boldsymbol{b}_1, \boldsymbol{b}_2) := (\boldsymbol{b}_2, \boldsymbol{b}_1)$
3: $\quad k := \left\lceil \dfrac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor$ $/_* \lceil a \rfloor := \lfloor a + 0.5 \rfloor {}^*/$
4: $\quad \boldsymbol{b}_2 := \boldsymbol{b}_2 - k\boldsymbol{b}_1$
5: **until** $\|\boldsymbol{b}_1\| \le \|\boldsymbol{b}_2\|$

定理

Lagrange 格基约化算法是正确的; 所需的循环次数不超过

$$O\left(\log \frac{\|\boldsymbol{b}_1\|}{\sqrt{\det \Lambda}}\right).$$

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 13 & 12 \\ 4 & 2 \end{pmatrix}$.

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 13 & 12 \\ 4 & 2 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{164}{185} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$.

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 13 & 12 \\ 4 & 2 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{164}{185} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$.
- $\|\boldsymbol{b}_1\| > \|\boldsymbol{b}_2\|$, 故 $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} -1 & 13 \\ -2 & 4 \end{pmatrix}$.

例

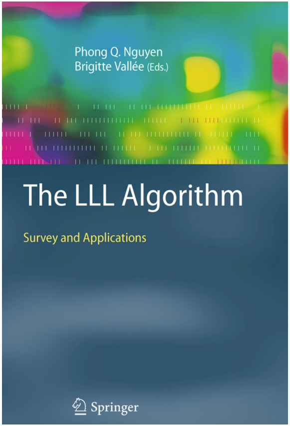
- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 13 & 12 \\ 4 & 2 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{164}{185} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$.
- $\|\boldsymbol{b}_1\| > \|\boldsymbol{b}_2\|$, 故 $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} -1 & 13 \\ -2 & 4 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil -\frac{21}{5} \right\rfloor = -4$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 + 4 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 9 \\ -4 \end{pmatrix}$.

例

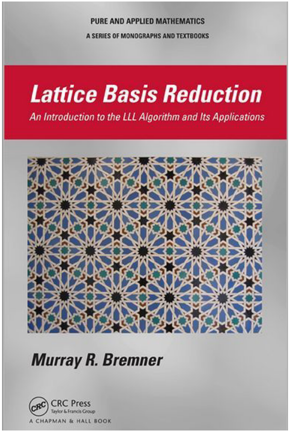
- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 13 & 12 \\ 4 & 2 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{164}{185} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$.
- $\|\boldsymbol{b}_1\| > \|\boldsymbol{b}_2\|$, 故 $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} -1 & 13 \\ -2 & 4 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil -\frac{21}{5} \right\rfloor = -4$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 + 4 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 9 \\ -4 \end{pmatrix}$.
- $\|\boldsymbol{b}_1\| \leq \|\boldsymbol{b}_2\|$, 故输出 $\begin{pmatrix} -1 & 9 \\ -2 & -4 \end{pmatrix}$.

提纲

LLL 算法



P. Nguyen, B. Vallée, Springer, 2010



M. R. Bremner, CRC Press, 2012

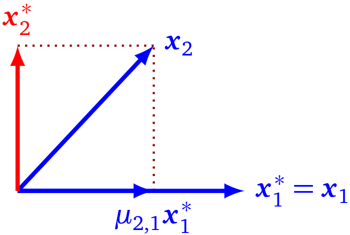
称 $\boldsymbol{x}_1^*, \boldsymbol{x}_2^*, \ldots, \boldsymbol{x}_n^*$ 是 $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ 的 Gram-Schmidt 正交化, 若

$$\boldsymbol{x}_1^* = \boldsymbol{x}_1,$$

$$\boldsymbol{x}_i^* = \boldsymbol{x}_i - \sum_{j=1}^{i-1} \mu_{i,j} \boldsymbol{x}_j^*, \qquad (2 \le i \le n)$$

$$\mu_{i,j} = \frac{\langle \boldsymbol{x}_i, \boldsymbol{x}_j^* \rangle}{\langle \boldsymbol{x}_j^*, \boldsymbol{x}_j^* \rangle}, \qquad (1 \le j < i \le n)$$

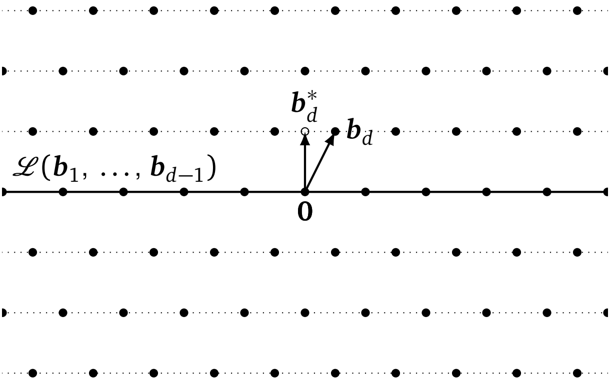
其中 $\mu_{i,j}$ 被称为 Gram-Schmidt 正交化系数.



Gram-Schmidt 正交化和最小距离

定理

对格基 $\boldsymbol{B}$ 和它的 Gram-Schmidt 正交化 $\boldsymbol{B}^*$, $\lambda(\mathscr{L}(\boldsymbol{B})) \geq \min_i \|\boldsymbol{b}_i^*\|$.

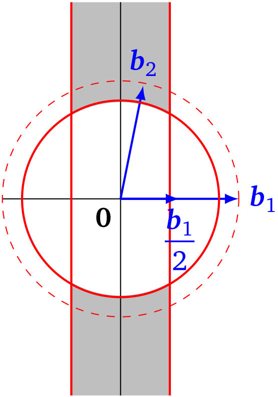


$\mathscr{L}(\boldsymbol{B})$ 分层: 每层与 $\boldsymbol{b}_d^*$ 正交, 层与层之间的距离为 $\|\boldsymbol{b}_d^*\|$.

Lenstra-Lenstra-Lovász (LLL) 约化基

称一组基 $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{R}^n$ 是LLL 约化的, 若如下条件都成立:

- (规模约减) $\forall 1 \leq j < i \leq n, |\mu_{i,j}| \leq \frac{1}{2}$,
- (Siegel 条件) $1 \leq i \leq n-1, \|\boldsymbol{b}_i^*\|^2 \leq 2\|\boldsymbol{b}_{i+1}^*\|^2$.



设 $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^n$ 是格 Λ 的一个 LLL 约化基. 则

$$\|\boldsymbol{b}_1\| \leq 2^{\frac{n-1}{2}} \lambda(\Lambda).$$

证明

由 LLL 约化基的定义知

$$\|\boldsymbol{b}_n^*\|^2 \geq \frac{1}{2}\|\boldsymbol{b}_{n-1}^*\|^2 \geq \cdots \geq \frac{1}{2^{n-1}}\|\boldsymbol{b}_1^*\|^2 = \frac{1}{2^{n-1}}\|\boldsymbol{b}_1\|^2.$$

于是对任意的 $i \leq n$

$$\|\boldsymbol{b}_1\| \leq 2^{\frac{i-1}{2}}\|\boldsymbol{b}_i^*\| \leq 2^{\frac{n-1}{2}}\|\boldsymbol{b}_i^*\|,$$

所以

$$\|\boldsymbol{b}_1\| \leq 2^{\frac{n-1}{2}} \min_i \|\boldsymbol{b}_i^*\| \leq 2^{\frac{n-1}{2}} \lambda(\Lambda).$$

LLL 算法 (1982)

输入: 格 $\Lambda \subseteq \mathbb{Z}^n$ 的一组基 $(\boldsymbol{b}_i)_{i\le n}$.
输出: 格 Λ 的一组 LLL 约化基.

```
计算 (b_i)_{i≤n} 的 GSO (b_i^*)_{i≤n} 和 GSO 系数 (μ_{i,j}).
for i = 2, 3, ⋯, n do
    for j = i−1, i−2, ⋯, 1 do
        b_i := b_i − ⌈μ_{i,j}⌋ b_j, 更新 GSO
    end for
    if ‖b_i^*‖^2 ≤ 2‖b_{i+1}^*‖^2 then
        i := i + 1
    else
        交换 b_i 和 b_{i+1}, 更新 GSO, 令 i := max{i−1, 2}
    end if
end for
return (b_i)_{i≤n}
```

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{41}{37} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{41}{37} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.
- $\|\boldsymbol{b}_1^*\|^2 > 2\|\boldsymbol{b}_2^*\|^2$, 故 $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 1 & 12 \\ 2 & 2 \end{pmatrix}$.

例

- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{41}{37} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.
- $\|\boldsymbol{b}_1^*\|^2 > 2\|\boldsymbol{b}_2^*\|^2$, 故 $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 1 & 12 \\ 2 & 2 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{16}{5} \right\rfloor = 3$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 3 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 9 \\ -4 \end{pmatrix}$.

例

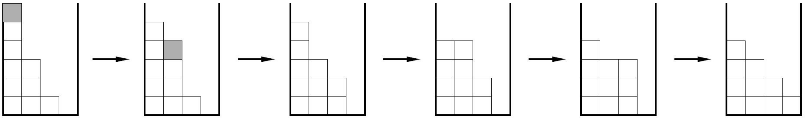
- 输入 $(\boldsymbol{b}_1, \boldsymbol{b}_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{41}{37} \right\rfloor = 1$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 1 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.
- $\|\boldsymbol{b}_1^*\|^2 > 2\|\boldsymbol{b}_2^*\|^2$, 故 $(\boldsymbol{b}_1, \boldsymbol{b}_2) := \begin{pmatrix} 1 & 12 \\ 2 & 2 \end{pmatrix}$.
- $\left\lceil \frac{\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle}{\langle \boldsymbol{b}_1, \boldsymbol{b}_1 \rangle} \right\rfloor = \left\lceil \frac{16}{5} \right\rfloor = 3$. 于是 $\boldsymbol{b}_2 := \boldsymbol{b}_2 - 3 \cdot \boldsymbol{b}_1 = \begin{pmatrix} 9 \\ -4 \end{pmatrix}$.
- $\|\boldsymbol{b}_1^*\|^2 \leq 2\|\boldsymbol{b}_2^*\|^2$, 故输出 $\begin{pmatrix} 1 & 9 \\ 2 & -4 \end{pmatrix}$.

LLL 算法的正确性和终止性

定理 (Lenstra-Lenstra-Lovász, 1982)

对 $B \in \mathbb{Z}^{n\times n}$, LLL 算法在对位长不超过 $O(n \log \|B\|)$ 的整数进行不超过 $O(n^4 \log \|B\|)$ 次算术操作后输出 $\mathscr{L}(B)$ 的一组 LLL 约化基.

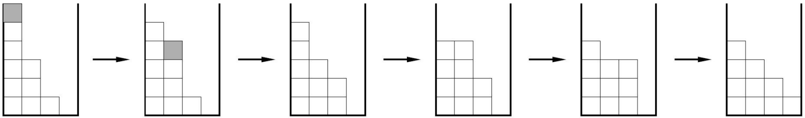
左起: L. Lovász, H. Lenstra, A. Lenstra



沙堆模型

设 $\boldsymbol{B}$ 是一个秩为 n 的格的基. 经典的分析工具

$$\Pi(\boldsymbol{B}) = \sum_{i=1}^{n-1} (n-i) \log \|\boldsymbol{b}_i^*\|.$$



沙堆模型

设 $\boldsymbol{B}$ 是一个秩为 n 的格的基. 经典的分析工具

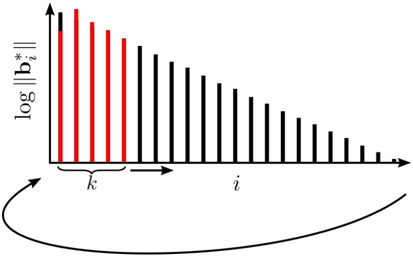
$$\Pi(\boldsymbol{B}) = \sum_{i=1}^{n-1} (n-i) \log \|\boldsymbol{b}_i^*\|.$$

针对一类特殊格的一个新工具 (C.-Stehlé-Villard, 2018):

$$\Pi_k(\boldsymbol{B}) = \sum_{j=1}^{k-1} (k-j) \log \|\boldsymbol{b}_{\ell_j}^*\| - \sum_{i=1}^{n-k} i \log \|\boldsymbol{b}_{s_i}^*\| + \sum_{i=1}^{n-k} s_i.$$

提纲

1. 格的理论概要
2. 格基约化算法简介
   - Lagrange 算法
   - LLL 格基约化算法
   - BKZ 算法
3. 应用举例



BKZ 算法示意图

- 开销：由 k 维的 SVP$_1$ 求解主导.
- 质量：第一个向量满足

$$\|\boldsymbol{b}_1\| = \kappa^n \det(\mathscr{L}(\boldsymbol{B}))^{\frac{1}{n}},$$

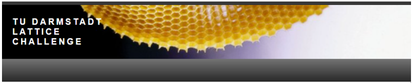
其中 κ 与分块大小 k 相关.

- 开销：$\mathrm{poly}(n)\cdot 2^{c\cdot k\log k}$，其中

$$c = 1/8 = 0.125 \quad \text{(理论)} \quad \text{或} \quad c = 1/(2\mathrm{e}) \approx 0.184 \quad \text{(实际)}.$$

- 质量：$\|\boldsymbol{b}_1\| = \kappa^n \det(\mathscr{L}(\boldsymbol{B}))^{\frac{1}{n}}$，其中

$$\kappa = k^{\frac{1}{2k}+o(1)} \quad \text{(理论)} \quad \text{或} \quad \kappa = \left(\frac{k}{2\pi\mathrm{e}}(\pi k)^{\frac{1}{k}}\right)^{\frac{1}{2(k-1)}} \quad \text{(实际)}.$$



HALL OF FAME

Position	Dimension	Euclidean norm	Contestant	Submission	Date
1	1000	155.21	Yao Sun	Details	2022-11-1
	1000	158.30	Yao Sun	Details	2022-10-24
2	975	144.55	Yao Sun	Details	2022-10-7
	975	146.25	Yao Sun	Details	2022-10-2
	975	151.42	Yao Sun	Details	2022-09-3
3	950	132.17	Yao Sun	Details	2022-10-25
	950	135.98	Yao Sun	Details	2022-10-2
	950	139.84	Yao Sun	Details	2022-08-26
	950	146.81	Yao Sun	Details	2022-08-10
4	925	138.70	Yao Sun	Details	2022-08-7
5	900	115.98	Yao Sun	Details	2022-08-7
	900	121.09	Yao Sun	Details	2022-08-4
6	875	104.96	Yao Sun	Details	2022-08-7
	875	109.66	Yao Sun	Details	2022-08-4
	875	133.67	Yao Sun	Details	2022-07-18
7	850	102.40	Yao Sun	Details	2022-08-1
	850	117.99	Yao Sun	Details	2022-07-18
	850	130.63	Shengxuan Jin	Details	2021-11-16
8	825	95.05	Yao Sun	Details	2022-08-2
	825	95.16	Yao Sun	Details	2022-08-1
	825	103.99	Yao Sun	Details	2022-07-18
	825	111.22	Minghao Sun Longjiang Qu Chao Li	Details	2022-06-8
	825	114.79	Shengxuan Jin	Details	2021-11-16
	825	117.64	Yoshinori Aono Phong Nguyen	Details	2017-01-4
	825	120.37	Yuanmi Chen Phong Nguyen	Details	2013-03-22
	825	122.38	Yoshinori Aono Ken Naganuma	Details	2012-10-1
9	800	103.47	Minghao Sun Wei Zhang Longjiang Qu Chao Li	Details	2022-05-3
	800	103.95	Yoshinori Aono Phong Nguyen	Details	2017-01-4
	800	106.60	Yuanmi Chen Phong Nguyen	Details	2013-03-26
	800	111.33	Yuanmi Chen Phong Nguyen	Details	2013-03-24

https://latticechallenge.org/

提纲

1. 格的理论概要
2. 格基约化算法简介
3. 应用举例
   - 背包问题的求解
   - 求解 LWE 问题的几何方法

背包问题

给定正整数 $a_1, \ldots, a_n$ (重量) 和 s, 找到 $e_1, \ldots, e_n \in \{0, 1\}$ 使得

$$\sum_{i=1}^{n} e_i a_i = s.$$

- 背包问题是 NP-完全问题.

背包问题

给定正整数 $a_1, \ldots, a_n$ (重量) 和 s, 找到 $e_1, \ldots, e_n \in \{0, 1\}$ 使得

$$\sum_{i=1}^{n} e_i a_i = s.$$

- 背包问题是 NP-完全问题.
- 密码学应用: Merkle-Hellman 加密系统...

给定正整数 $a_1, \ldots, a_n$ (重量) 和 s, 找到 $e_1, \ldots, e_n \in \{0, 1\}$ 使得

$$\sum_{i=1}^{n} e_i a_i = s.$$

- 背包问题是 NP-完全问题.
- 密码学应用: Merkle-Hellman 加密系统...
- 几乎所有的基于背包问题的加密系统都已被攻破...

一个容易求解的背包问题实例

例

对 $i = 1, 2, \ldots, n$, 设 $a_i = 2^{i-1}$. 则

$$\text{背包问题有解} \iff 0 \le s \le 2^n - 1;$$

并且, 解向量 $(e_1, \ldots, e_n)$ 刚好对应于 s 的二进制表示:

$$s = \sum_{i=1}^{n} e_i a_i = \sum_{i=1}^{n} e_i 2^{i-1}.$$

超增长序列

称一个正整数序列 $a_1, \ldots, a_n$ 是超增长 (superincreasing) 的, 若

$$a_i > \sum_{j=1}^{i-1} a_j, \qquad i = 2, 3, \ldots, n.$$

对于超增长序列, 背包问题是容易求解的, 因为

$$e_n = 1 \Leftrightarrow s \geq a_n,$$

且对于 $i = n-1, n-2, \ldots, 1$,

$$e_i = 1 \Leftrightarrow s - \sum_{j=i+1}^{n} e_j a_j \geq a_i.$$

“均匀”假设

$a_1, \ldots, a_n$ 是从 $\{1, 2, \cdots, A\}$ 中独立随机选取得到的, 其中 $A \in \mathbb{Z}_+$.

设 $\boldsymbol{e} = (e_1, \ldots, e_n) \in \{0, 1\}^n$ 是背包问题的解. 令 $t = \sum_{i=1}^{n} a_i$. 事实上, 可以假设

$$s \geq \frac{t}{2}.$$

否则, 可以考虑求解如下问题

$$\sum_{i=1}^{n} g_i a_i = t - s, \quad g_i = 1 - e_i \in \{0, 1\}, \quad i = 1, 2, \ldots, n.$$

设 N 为一个充分大的正整数. 考虑由如下矩阵的列生成的格 Λ:

$$B = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ -Na_1 & -Na_2 & \cdots & -Na_n & Ns \end{pmatrix} \in \mathbb{Z}^{(n+1)\times(n+1)}.$$

- $\hat{\boldsymbol{e}} = (e_1, \ldots, e_n, 0) \in \Lambda$, 且 $\|\hat{\boldsymbol{e}}\| \le \sqrt{n}$, 从而 $\hat{\boldsymbol{e}}$ 是 Λ 中的短向量.
- LLL 算法可以找到一个向量 $\hat{\boldsymbol{x}} \in \Lambda$ 使得

$$\|\hat{\boldsymbol{x}}\| \le 2^{n/2}\lambda(\Lambda) \le 2^{n/2}\sqrt{n} =: M.$$

- 因此, 可以对 B 调用 LLL 算法, 然后检验算法是否输出 $\pm\hat{\boldsymbol{e}}$.

定理 (Lagarias-Odlyzko, 1983)

设 $\hat{\boldsymbol{x}}$ 是 LLL 算法输出的基中的最短向量, $a_1, \ldots, a_n$ 的分布服从“均匀”假设, 其中 $A \geq 2^{(1/2+\varepsilon)n^2}$, $\varepsilon > 0$. 则

$$\Pr[\hat{\boldsymbol{x}} \neq \pm\hat{\boldsymbol{e}}] \leq \frac{(4M+1)(2M+1)^n}{A} = O(2^{-\varepsilon n^2/2}).$$

$$
\boldsymbol{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 1/2 \\ 0 & 1 & \cdots & 0 & 1/2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1/2 \\ Na_1 & Na_2 & \cdots & Na_n & Ns \end{pmatrix}
$$

定理 (Coster-Joux-LaMacchia-Odlyzko-Schnorr-Stern, 1992)

对 $A = 2^{cn}$ $(c > c_0 = 1.0628\cdots)$ 的随机背包问题用 LLL 算法求解:

$$
\lim_{n\to\infty} \Pr[\boldsymbol{x} \neq \pm\hat{\boldsymbol{e}}] = 0.
$$

一个改进

$$
\boldsymbol{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 1/2 \\ 0 & 1 & \cdots & 0 & 1/2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1/2 \\ Na_1 & Na_2 & \cdots & Na_n & Ns \end{pmatrix}
$$

定理 (Coster-Joux-LaMacchia-Odlyzko-Schnorr-Stern, 1992)

对 $A = 2^{cn}$ $(c > c_0 = 1.0628\cdots)$ 的随机背包问题用 LLL 算法求解:

$$
\lim_{n\to\infty} \Pr[\boldsymbol{x} \neq \pm\hat{\boldsymbol{e}}] = 0.
$$

- 这是目前关于 c_0 最小的一个结果.

一个改进

$$B = \begin{pmatrix} 1 & 0 & \cdots & 0 & 1/2 \\ 0 & 1 & \cdots & 0 & 1/2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1/2 \\ Na_1 & Na_2 & \cdots & Na_n & Ns \end{pmatrix}$$

定理 (Coster-Joux-LaMacchia-Odlyzko-Schnorr-Stern, 1992)

对 $A = 2^{cn}$ $(c > c_0 = 1.0628\cdots)$ 的随机背包问题用 LLL 算法求解:

$$\lim_{n\to\infty} \Pr[\boldsymbol{x} \neq \pm\hat{\boldsymbol{e}}] = 0.$$

- 这是目前关于 c_0 最小的一个结果.
- worst-case vs average-case?

随机自归约

定义 (random self-reducible)

称一个计算问题是随机自归约的, 若存在多项式时间算法, 该算法可以将问题的任何给定实例转换为均匀分布的随机实例, 从而可以在多项式时间内从新实例的解中获取原始实例的解.

- 含义: worst-case $\leq$ average-case.

随机自归约

定义 (random self-reducible)

称一个计算问题是随机自归约的, 若存在多项式时间算法, 该算法可以将问题的任何给定实例转换为均匀分布的随机实例, 从而可以在多项式时间内从新实例的解中获取原始实例的解.

- 含义: worst-case $\leq$ average-case.
- 这一性质对密码学应用非常重要.

随机自归约

定义 (random self-reducible)

称一个计算问题是随机自归约的, 若存在多项式时间算法, 该算法可以将问题的任何给定实例转换为均匀分布的随机实例, 从而可以在多项式时间内从新实例的解中获取原始实例的解.

- 含义: worst-case $\leq$ average-case.
- 这一性质对密码学应用非常重要.
- 背包问题不是随机自归约的.

随机自归约

定义 (random self-reducible)

称一个计算问题是随机自归约的, 若存在多项式时间算法, 该算法可以将问题的任何给定实例转换为均匀分布的随机实例, 从而可以在多项式时间内从新实例的解中获取原始实例的解.

- 含义: worst-case $\leq$ average-case.
- 这一性质对密码学应用非常重要.
- 背包问题不是随机自归约的.
- 离散对数、RSA 求逆、LWE 问题都是随机自归约的.

提纲

1. 格的理论概要
2. 格基约化算法简介
3. 应用举例
   - 背包问题的求解
   - 求解 LWE 问题的几何方法

LWE 问题 (Learning with errors)

搜索版 LWE 问题

给定

$$\left\{(\boldsymbol{a}_i, b_i = \langle \boldsymbol{a}_i, \boldsymbol{s}\rangle + e_i \mod q) : \boldsymbol{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \boldsymbol{a}_i \leftarrow \mathcal{U}(\mathbb{Z}_q^n), e_i \leftarrow \chi\right\}_{i=1}^m,$$

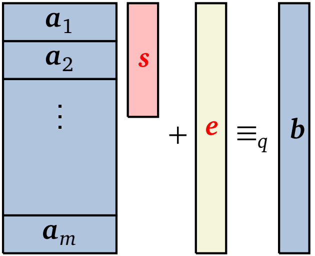
计算 $\boldsymbol{s} \in \mathbb{Z}_q^n$, 其中 χ 是 $\mathbb{Z}$ 上的离散高斯分布.

$$\begin{cases} a_{11}s_1 + a_{12}s_2 + \cdots + a_{1n}s_n + e_1 = b_1 \mod q \\ a_{21}s_1 + a_{22}s_2 + \cdots + a_{2n}s_n + e_2 = b_2 \mod q \\ \vdots \\ a_{m1}s_1 + a_{m2}s_2 + \cdots + a_{mn}s_n + e_m = b_m \mod q \end{cases}$$

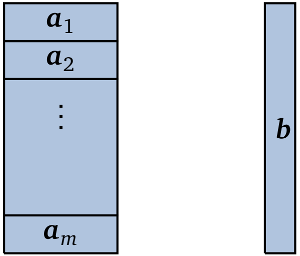
- 维数：n
- 模数：素数 $q = \mathrm{poly}(n)$ 满足 $q \approx n^2$
- 误差分布：χ，标准差为 $\sigma = \alpha q$ 的离散高斯分布
- 噪声规模：$\alpha = 1/\mathrm{poly}(n)$ ($\Longrightarrow e_i \ll q$)，例如 $\alpha q \gtrsim 2\sqrt{n}$.
  - 这样的参数设置允许 worst-case $\leq$ average-case 归约
  - 存在复杂度为 $2^{\widetilde{O}(\alpha^2 q^2)}$ 的算法求解 LWE
- 样本数：$m \approx n \log q$
  - $O(n \log q)$ 个样本 $\Rightarrow$ 任意多“好而新”的样本

(Gentry-Peikert-Vaikuntanathan ’08)

判定版 LWE 问题



World 1: LWE 分布



World 2: 均匀分布

判定版 LWE 问题

给定样本 $(\boldsymbol{a}_i, b_i)_{i\leq m}$，判定这些样本来自于哪一个 World.

判定版 vs 搜索版

判定版 LWE 问题$\Longleftrightarrow$搜索版 LWE 问题.

LWE 问题的特性

- 判定版 LWE 问题是随机自归约的 (worst case $\leq$ average case)

LWE 问题的特性

- 判定版 LWE 问题是随机自归约的 (worst case $\leq$ average case)
  - 输入：$(\boldsymbol{A}, \boldsymbol{b})$

- 判定版 LWE 问题是随机自归约的 (worst case $\leq$ average case)
  - 输入：$(\boldsymbol{A}, \boldsymbol{b})$
  - 构造 $\boldsymbol{t} \leftarrow \mathbb{Z}_q^n$，计算 $(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{t} + \boldsymbol{b})$

LWE 问题的特性

- 判定版 LWE 问题是随机自归约的 (worst case $\leq$ average case)
  - 输入：$(\boldsymbol{A}, \boldsymbol{b})$
  - 构造 $\boldsymbol{t} \leftarrow \mathbb{Z}_q^n$，计算 $(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{t} + \boldsymbol{b})$
- 任意数量的样本

LWE 问题的特性

- 判定版 LWE 问题是随机自归约的 (worst case $\leq$ average case)
  - 输入：$(\boldsymbol{A}, \boldsymbol{b})$
  - 构造 $\boldsymbol{t} \leftarrow \mathbb{Z}_q^n$，计算 $(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{t} + \boldsymbol{b})$
- 任意数量的样本
- 搜索版本和判定版本等价

LWE 问题的特性

- 判定版 LWE 问题是随机自归约的 (worst case $\leq$ average case)
  - 输入：$(\boldsymbol{A}, \boldsymbol{b})$
  - 构造 $\boldsymbol{t} \leftarrow \mathbb{Z}_q^n$，计算 $(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{t} + \boldsymbol{b})$
- 任意数量的样本
- 搜索版本和判定版本等价
- 对多种不同的误差分布可以保持困难性

	Reduction	q	Worst-case problem
Regev (2005)	Quantum	Poly.	GapSVP, SIVP
Peikert (2009)	Classical	Exp.	GapSVP
Peikert (2009)	Classical	Poly.	Non-standard
Brakerski *et al.* (2013)	Classical	Poly.	GapSVP, dim = $\sqrt{n}$

LWE 假设

在一定参数条件下，$(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{s}+\boldsymbol{e})$ 与均匀分布 $(\boldsymbol{U}, \boldsymbol{u})$ 不可区分.

通过 q-ary 格求解 LWE (1/2)

- 给定 $(\boldsymbol{A} \in \mathbb{Z}_q^{m\times n}, \boldsymbol{b} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{e} \mod q)$，其中 $e_i \leftarrow D_{\mathbb{Z},\alpha q}$，求 $\boldsymbol{e}$.
- 可以看作 q-ary 格上的一个 CVP/BDD 实例：

$$\begin{aligned}\Lambda_q(\boldsymbol{A}) &= \boldsymbol{A}\cdot\mathbb{Z}_q^n + q\cdot\mathbb{Z}^m \\ &= \left\{\boldsymbol{x}\in\mathbb{Z}^m : \exists \boldsymbol{y}\in\mathbb{Z}^n \text{ s.t. } \boldsymbol{x} = \boldsymbol{A}\boldsymbol{y} \mod q\right\}.\end{aligned}$$

  - 以 $\boldsymbol{b}$ 为目标向量对格 $\Lambda_q(\boldsymbol{A})$ 求解 CVP/BDD.
  - 回顾：$\mathrm{rank}(\Lambda_q(\boldsymbol{A})) = m, \quad \det(\Lambda_q(\boldsymbol{A})) = q^{m-n}$.
  - 转化成秩为 $m+1$ 的一个 SVP 实例.

Kannan 嵌入

设 $\boldsymbol{B}$ 是 $\Lambda_q(\boldsymbol{A})$ 的一个基. 构造

$$\boldsymbol{B}' = \begin{pmatrix} \boldsymbol{B} & \boldsymbol{b} \\ \boldsymbol{0} & 1 \end{pmatrix} \Rightarrow \begin{pmatrix} \boldsymbol{B} & \boldsymbol{b} \\ \boldsymbol{0} & 1 \end{pmatrix} \cdot \begin{pmatrix} * \\ 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{e} \\ 1 \end{pmatrix},$$

$*$ 是格点 $-(\boldsymbol{A}\boldsymbol{s} + q\boldsymbol{c})$ 的在基下 $\boldsymbol{B}$ 下的系数向量 ($\boldsymbol{b} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{e} + q\boldsymbol{c}$).

- 在一定条件下，对 $m \approx -\dfrac{2n\log q}{\log\alpha}$，基于 BKZ 的 LWE 求解算法开销为

$$\left(\frac{n\log q}{\log^2\alpha}\right)^{O\left(\frac{n\log q}{\log^2\alpha}\right)}.$$

- 代数攻击：Arora-Ge 算法
- 组合攻击：Blum-Kalai-Wasserman 算法
- 几何攻击：格 (约化) 算法

Algorithms for LWE	(Some) broken parameter settings
Arora-Ge	$\Omega(n^B)$ samples + time[2]
Blum-Kalai-Wasserman	$> q^{n/\log(q/B)}$ samples + time
Geometric attack	$\mathrm{poly}(n)$ samples + $2^{O(n)}$ time

[2] B bounds the width of error.

总结

包含的内容

- 从 Kepler 猜想到 Kepler 定理
- 格的简介：q-ary 格、行列式、最小距离、Minkowski 定理
- 格约化算法简介：Lagrange、LLL、BKZ
- 在密码分析中的应用：背包问题、LWE 问题

总结

包含的内容

- 从 Kepler 猜想到 Kepler 定理
- 格的简介：q-ary 格、行列式、最小距离、Minkowski 定理
- 格约化算法简介：Lagrange、LLL、BKZ
- 在密码分析中的应用：背包问题、LWE 问题

未包含的内容

- Hermite 常数，对偶格，格上的概率分布，代数格...
- 求解 SVP 的算法 (筛法，穷举+剪枝...)
- 与格有关的其他困难问题及算法 (SIS...)
- 用于攻击 RSA 系统 (Coppersmith 方法)
- 基于格的密码学构造 (PQC，FHE...)
- 格的其他应用

Hilbert 第 18 问题第 3 小问 (Kepler 定理的推广)

David Hilbert (1862–1943)

- 在 n 维欧氏空间中, 如何堆放无穷多个同样的物体, 比如球和正四面体, 使得堆积的密度最大?——尚待解决.

主要参考文献

- 宗传明. 堆球的故事. 2014.
- J. Lagrange. Nouveaux Mémoires de l'Académie de Berlin, 1773.
- A. Lenstra, H. Lenstra, L. Lovász. Math. Ann., 261:515–534, 1982.
- C.-P. Schnorr. Comb. Probab. Comput., 3:507-522, 1994
- J. Lagarias, A. Odlyzko. In FOCS '83, p. 1–10, 1983.
- M. Coster, et al. Comput. Complex, 2: 111-128, 1992.
- O. Regev. J. ACM, 56(6):34:1-40, 2009.

- 本课件中人物肖像来自 wikipedia.org.
- 本课件内容可从如下网址下载:

`https://chen-jingwei.github.io/download/intro2lattice22.pdf`

THANKS