

格的理论、算法和应用

一份入门性的介绍

陈经纬



二〇二〇年十二月四日

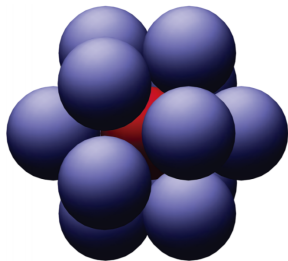


Sir Walter Raleigh (1552–1618)

- Raleigh 爵士: 如何让有限的炮弹仓尽量多地携带加农炮?



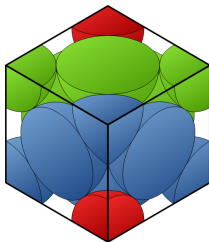
Thomas Harriot (1560–1621)



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)

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



Johannes Kepler (1571–1630)

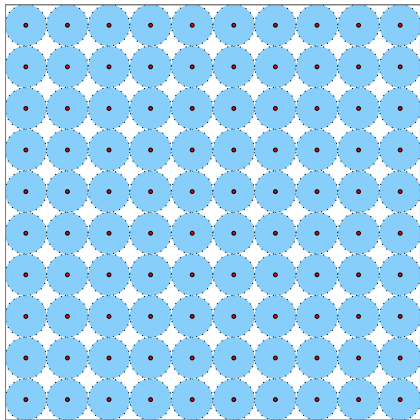
猜想 (J. Kepler. *The Six-Cornered Snow Flake*, 1611)¹

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

¹1998 年, 被 Thomas Hales 用计算机程序证明; 2014 年完成形式化验证.

思考题

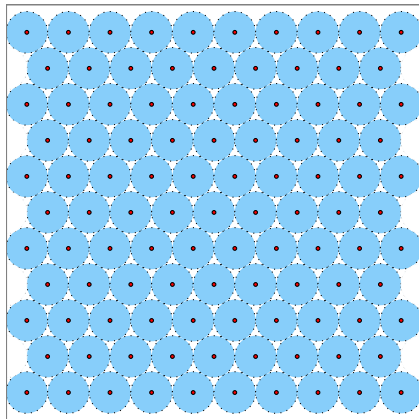
给定一个 10×10 的正方形, 最多可以放进多少个直径为 1 的圆?



可放入 100 个直径为 1 的圆.

思考题

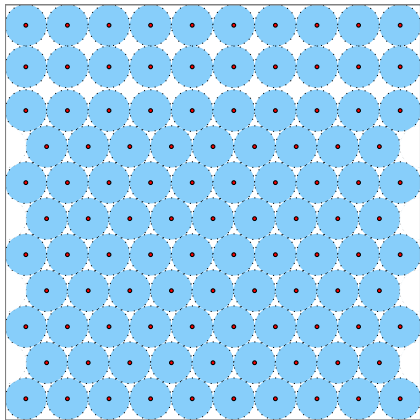
给定一个 10×10 的正方形, 最多可以放进多少个直径为 1 的圆?



可放入 105 个直径为 1 的圆.

思考题

给定一个 10×10 的正方形, 最多可以放进多少个直径为 1 的圆?



可放入 106 个直径为 1 的圆.



Carl Friedrich Gauß (1777–1885)

Gauß 的贡献

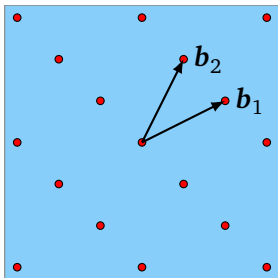
在三维空间中堆同样大小的球. 若它们的球心构成一个格 (或者格的一部分), 那么堆球的密度不会超过 $\pi/\sqrt{18}$.

格 (Lattice)

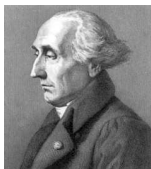
若 $\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n \in \mathbb{R}^n$ 线性无关, 则称

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

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



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



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)

- ① 格的理论概要
 - 格的定义
 - 格的不变量
 - 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
- ③ 应用举例
 - 整数关系问题求解
 - 背包问题的求解

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

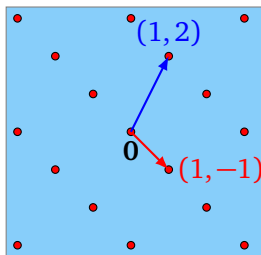
格的定义

设矩阵 $B = (b_1, \dots, b_d) \in \mathbb{R}^{n \times d}$ **列满秩**. 定义由 B 生成的**格**是

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

矩阵 B 称为格 Λ 的一个**基**. 整数 d 被称作格的**秩**, 记作 $\text{rank}(\Lambda)$.
若 $d = n$, 则 $\Lambda = \mathcal{L}(B)$ 被称作**满秩格**.

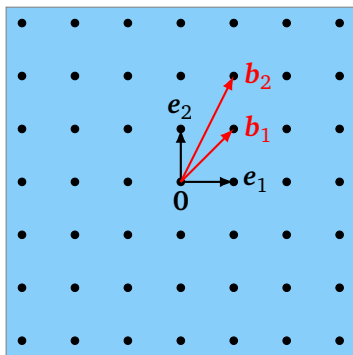
$$B = \begin{pmatrix} 1 & 1 \\ -1 & 2 \end{pmatrix}$$



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

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.



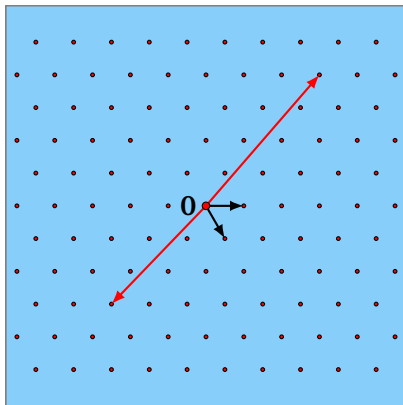
$$(e_1, e_2) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$(b_1, b_2) = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$$

$$e_1 = 2b_1 - b_2$$

$$e_2 = -b_1 + b_2$$

单位矩阵 I_2 和 (b_1, b_2) 都是格 \mathbb{Z}^2 的基



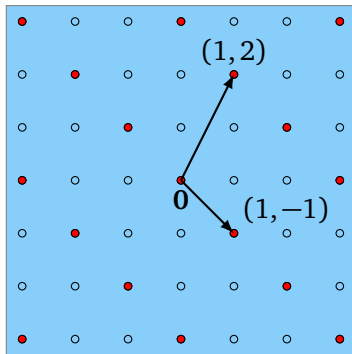
好基的判断标准

- 向量长度较短;
- 向量间接近正交 (垂直).

子格

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.

 $\mathbb{Z}^2 = \mathcal{L}(I_2)$ 的子格

幺模矩阵: \mathbb{Z} 上的可逆矩阵

称 $U \in \mathbb{Z}^{n \times n}$ 在 \mathbb{Z} 上可逆, 若存在 $V \in \mathbb{Z}^{n \times n}$ 使得 $UV = VU = I_n$.

定理

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

因此, 亦称 \mathbb{Z} 上的可逆矩阵为 幺模矩阵.

定理

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

证明

(\Leftarrow): 由 U 是幺模矩阵且 $B = CU$ 知

$$\mathcal{L}(B) = \mathcal{L}(CU) \subseteq \mathcal{L}(C) = \mathcal{L}(BU^{-1}) \subseteq \mathcal{L}(B).$$

整数初等变换

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

- $\text{swap}(i, j): (b_i, b_j) := (b_j, b_i), i \neq j.$

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

- $\text{invert}(i): b_i := (-b_i), i = 1, \dots, n.$

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

- $\text{add}(i, c, j): b_i := b_i + c \cdot b_j, i \neq j \text{ 且 } c \in \mathbb{Z}.$

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

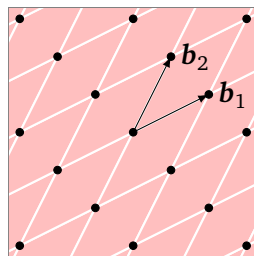
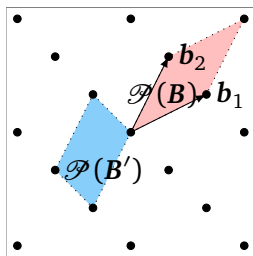
以上三种整数初等列变换都是幺模变换, 从而不改变原来的格.

- 1 格的理论概要
 - 格的定义
 - 格的不变量
 - 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 $B = (b_1, \dots, b_d)$ 为格 Λ 的一个基. 定义格 $\Lambda = \mathcal{L}(B)$ 的**基本平行六面体** 为

$$\mathcal{P}(B) = B[0, 1)^d = \left\{ \sum_{i=1}^d z_i \cdot b_i : \forall i, 0 \leq z_i < 1 \right\}.$$



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 $\mathcal{P}(B)$. 则 $\mathcal{P}(B)$ 的 d -维体积

$$\text{vol}(\mathcal{P}(B)) = \sqrt{\det(B^T B)}.$$

推论

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

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 $\det(\Lambda)$.

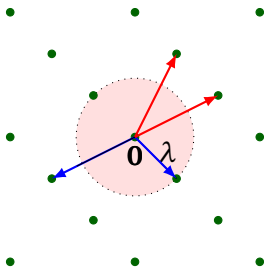
性质

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:

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

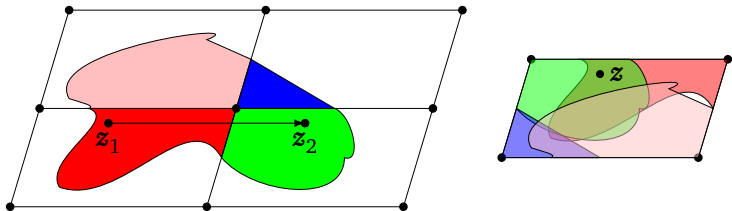


性质

格的最小距离是幺模变换下的不变量.

Blichfeldt 引理

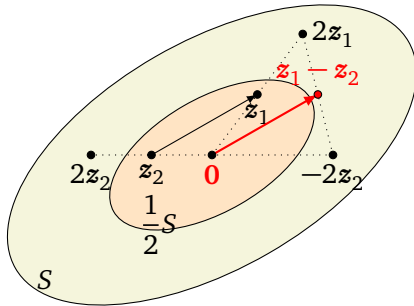
设 Λ 是一个格, 并设集合 $S \subseteq \text{Span}(\Lambda)$ 有体积. 若 $\text{vol}(S) > \det(\Lambda)$, 则存在 $z_1 \neq z_2 \in S$ 使得 $z_1 - z_2 \in \Lambda$.



Blichfeldt 引理

Minkowski 凸胞定理

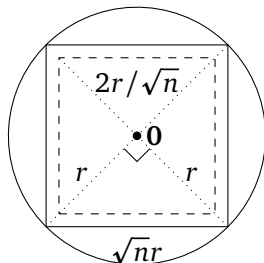
设 $\Lambda \subseteq \mathbb{R}^n$ 为一个满秩格. 若 $S \in \mathbb{R}^n$ 是一个中心对称的凸胞且 $\text{vol}(S) > 2^n \det(\Lambda)$, 则存在 $x \in \Lambda \setminus \{0\}$ 使得 $x \in S$.



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: $r = \lambda(\Lambda)$

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

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

最短向量问题 (SVP)

- 搜索版 SVP: 给一个格基 $B \in \mathbb{Z}^{n \times n}$, 计算 $v \in \mathcal{L}(B)$ 使得

$$\|v\| = \lambda(\mathcal{L}(B)).$$

- 判定版 SVP: 给定一个格基 B 和 $\mu \in \mathbb{Q}$, 判断下式是否成立

$$\lambda(\mathcal{L}(B)) \leq \mu.$$

- SVP_γ : 给定 B , 求 $v \in \Lambda = \mathcal{L}(B)$ 使得

$$\|v\| \leq \gamma \cdot \lambda(\mathcal{L}(B)).$$

- GapSVP_γ : 给定 B 和 $\mu \in \mathbb{Q}$, 判断属于下面哪种情况

$$\lambda(\mathcal{L}(B)) \leq \mu \quad \text{或} \quad \lambda(\mathcal{L}(B)) \geq \gamma \cdot \mu.$$

复杂性结果

- 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)

- 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
- 3 应用举例

给定格的一个基, 通过一系列么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

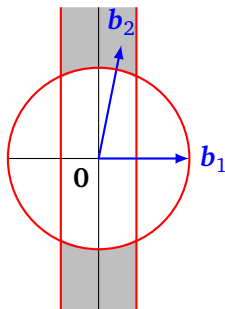
$$\Lambda = \{sa + tb : s, t \in \mathbb{Z}\}.$$

- $\Lambda \subseteq \mathbb{Z}$ 是由 a 和 b 生成的一个理想 $\Lambda = \langle d \rangle$, $d = \text{gcd}(a, b)$.
- 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 \mathbf{b}_1 和 \mathbf{b}_2 使得

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



Lagrange 约化基

Lagrange 算法 (1773)

输入: 二维格 Λ 的一个基 (b_1, b_2) .

输出: 格 Λ 的一个 Lagrange 约化基 (b_1, b_2) .

```
1: repeat  
2:    $(b_1, b_2) := (b_2, b_1)$   
3:    $k := \left\lceil \frac{\langle b_2, b_1 \rangle}{\langle b_1, b_1 \rangle} \right\rceil$  /*  $\lceil a \rceil := \lfloor a + 0.5 \rfloor$  */  
4:    $b_2 := b_2 - kb_1$   
5: until  $\|b_1\| \leq \|b_2\|$ 
```

定理

Lagrange 格基约化算法是正确的; 所需的循环次数不超过

$$O\left(\log \frac{\|b_1\|}{\sqrt{\det \Lambda}}\right).$$

- 输入 $(b_1, b_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.

- 输入 $(b_1, b_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.
- $(b_1, b_2) := \begin{pmatrix} 13 & 12 \\ 4 & 2 \end{pmatrix}$.

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

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

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

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

- 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
- ③ 应用举例

Gram-Schmidt 正交化

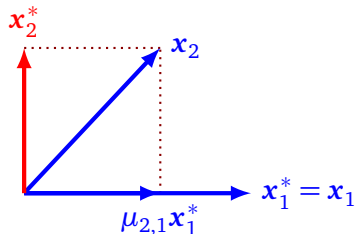
称 $\mathbf{x}_1^*, \mathbf{x}_2^*, \dots, \mathbf{x}_n^*$ 是 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ 的 Gram-Schmidt 正交化, 若

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

$$\mathbf{x}_i^* = \mathbf{x}_i - \sum_{j=1}^{i-1} \mu_{i,j} \mathbf{x}_j^*, \quad (2 \leq i \leq n)$$

$$\mu_{i,j} = \frac{\langle \mathbf{x}_i, \mathbf{x}_j^* \rangle}{\langle \mathbf{x}_j^*, \mathbf{x}_j^* \rangle}, \quad (1 \leq j < i \leq n)$$

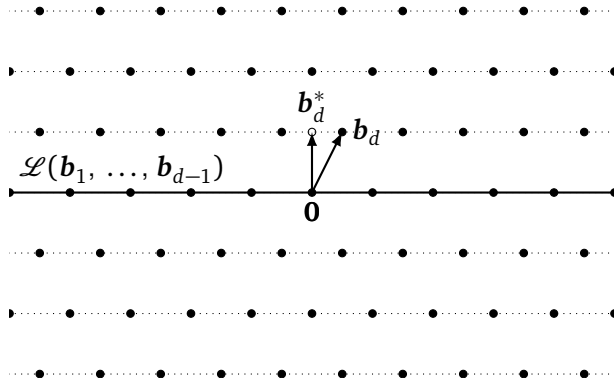
其中 $\mu_{i,j}$ 被称为 Gram-Schmidt 正交化系数.



Gram-Schmidt 正交化和最小距离

定理

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

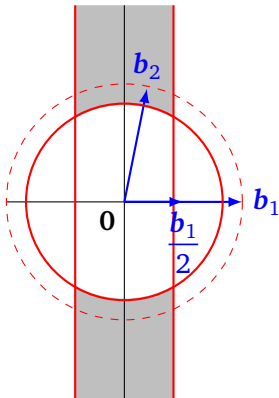


$\mathcal{L}(B)$ 分层: 每层与 b_d^* 正交, 层与层之间的距离为 $\|b_d^*\|$.

Lenstra-Lenstra-Lovász (LLL) 约化基

称一组基 $\mathbf{b}_1, \dots, \mathbf{b}_d \in \mathbb{R}^n$ 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) $1 \leq i \leq n-1, \|\mathbf{b}_i^*\|^2 \leq 2\|\mathbf{b}_{i+1}^*\|^2.$



LLL 约化基的性质

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

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

证明

由 LLL 约化基的定义知

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

于是对任意的 $i \leq n$

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

所以

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

LLL 算法 (1982)

输入: 格 $\Lambda \subseteq \mathbb{Z}^n$ 的一组基 $(\mathbf{b}_i)_{i \leq n}$, $\delta \in (\frac{1}{4}, 1)$.

输出: 格 Λ 的一组 δ -LLL 约化基.

- 1: 计算 $(\mathbf{b}_i)_{i \leq n}$ 的 GSO $(\mathbf{b}_i^*)_{i \leq n}$ 和 GSO 系数 $(\mu_{i,j})$.
- 2: **for** $i = 2, 3, \dots, n$ **do**
- 3: **for** $j = i - 1, i - 2, \dots, 1$ **do**
- 4: $\mathbf{b}_i := \mathbf{b}_i - \lceil \mu_{i,j} \rceil \mathbf{b}_j$, 更新 GSO
- 5: **end for**
- 6: **if** $\|\mathbf{b}_i^*\|^2 \leq 2\|\mathbf{b}_{i+1}^*\|^2$ **then**
- 7: $i := i + 1$
- 8: **else**
- 9: 交换 \mathbf{b}_i 和 \mathbf{b}_{i+1} , 更新 GSO, 令 $i := \max\{i - 1, 2\}$
- 10: **end if**
- 11: **end for**
- 12: **return** $(\mathbf{b}_i)_{i \leq n}$

- 输入 $(b_1, b_2) = \begin{pmatrix} 12 & 13 \\ 2 & 4 \end{pmatrix}$.

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

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

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

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

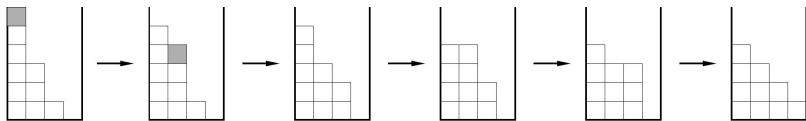
LLL 算法的正确性和终止性

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 $\mathcal{L}(B)$ 的一组 LLL 约化基.



左起: L. Lovász, H. Lenstra, A. Lenstra



沙堆模型

设 \mathbf{B} 是一个秩为 n 的格的基. 经典的分析工具

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

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

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

- ① 格的理论概要
- ② 格基约化算法简介
- ③ 应用举例
 - 整数关系问题求解
 - 背包问题的求解

- 给定 $\mathbf{x} \in \mathbb{R}^n$, 称 $\mathbf{m} \in \mathbb{Z}^n$ 是 \mathbf{x} 的一个整数关系, 若

$$\langle \mathbf{x}, \mathbf{m} \rangle = \sum_{i=1}^n m_i x_i = 0.$$

- 集合 $\Lambda_{\mathbf{x}} = \{\mathbf{m} \in \mathbb{Z}^n : \langle \mathbf{x}, \mathbf{m} \rangle = 0\}$ 是一个格.
- 整数关系问题: 给定 $\mathbf{x} \in \mathbb{R}^n$, 判定 $\Lambda_{\mathbf{x}} = \{\mathbf{0}\}$? 若否, 如何找到一个整数关系或 $\Lambda_{\mathbf{x}}$ 的一个基?
 - (Babai, *et al.* 1988): 在精确的算术模型下是不可判定的.

整数关系问题: 实际应用的考量

输入的是近似的浮点数据

$$(\tilde{x}_1, \dots, \tilde{x}_n).$$

如 HJLS/PSLQ² 和基于 LLL 的整数关系算法, 在计算机实现时

- 要么返回一个 可能的答案 和对应的“可信度”,
- 要么就精度用尽.

²[Meichsner, 2001]: HJLS \approx PSLQ

$$\bullet \mathbf{x} = \begin{bmatrix} t \\ 1 \\ \ln 2 \\ (\ln 2)^2 \\ \pi^2 \end{bmatrix}, \quad t = \int_0^1 \int_0^1 \left(\frac{x-1}{x+1} \right)^2 \left(\frac{y-1}{y+1} \right)^2 \left(\frac{xy-1}{xy+1} \right)^2 dx dy$$

$$\bullet \mathbf{x} = \begin{bmatrix} t \\ 1 \\ \ln 2 \\ (\ln 2)^2 \\ \pi^2 \end{bmatrix}, \quad t = \int_0^1 \int_0^1 \left(\frac{x-1}{x+1} \right)^2 \left(\frac{y-1}{y+1} \right)^2 \left(\frac{xy-1}{xy+1} \right)^2 dx dy$$

$$\bullet (1, -5, 4, -16, 1) \cdot \begin{bmatrix} 0.04505509936 \\ 1.0 \\ 0.6931471806 \\ 0.4804530139 \\ 9.869604401 \end{bmatrix} = \varepsilon \ (\approx -8.2938 \times 10^{-14})$$

$$\bullet \mathbf{x} = \begin{bmatrix} t \\ 1 \\ \ln 2 \\ (\ln 2)^2 \\ \pi^2 \end{bmatrix}, \quad t = \int_0^1 \int_0^1 \left(\frac{x-1}{x+1} \right)^2 \left(\frac{y-1}{y+1} \right)^2 \left(\frac{xy-1}{xy+1} \right)^2 dx dy$$

$$\bullet (1, -5, 4, -16, 1) \cdot \begin{bmatrix} 0.04505509936 \\ 1.0 \\ 0.6931471806 \\ 0.4804530139 \\ 9.869604401 \end{bmatrix} = \varepsilon \ (\approx -8.2938 \times 10^{-14})$$

$$\bullet 1 \cdot t + (-5) \cdot 1 + 4 \cdot \ln 2 + (-16) \cdot (\ln 2)^2 + 1 \cdot \pi^2 = 0?$$

基于 LLL 的整数关系算法

LLL 算法通过一系列幺模变换最终找到给定格中的“短”向量.

- 考虑下面矩阵的列生成的格

$$\begin{pmatrix} K \cdot x_1 & K \cdot x_2 & \cdots & K \cdot x_n \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \xrightarrow{\text{LLL}} \begin{pmatrix} \varepsilon & * & \cdots & * \\ * & u_2 & \cdots & u_n \\ * & & & \\ \vdots & & & \\ * & & & \end{pmatrix}$$

- 若 K 足够大, 则
 - x 的一个整数关系 \Leftrightarrow 上述格中一个“短”向量.
 - LLL 能够找到.

例: $\alpha = 3^{1/2} - 2^{1/2}$, $\mathbf{x} = (1, \alpha, \dots, \alpha^4)$

$$\begin{pmatrix} 1000000.0 & 317837.2 & 101020.5 & 32108.1 & 10205.1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

⇓ LLL

例: $\alpha = 3^{1/2} - 2^{1/2}$, $\mathbf{x} = (1, \alpha, \dots, \alpha^4)$

$$\begin{pmatrix} 1000000.0 & 317837.2 & 101020.5 & 32108.1 & 10205.1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

↓ LLL

$$\begin{pmatrix} -3.6 \times 10^{-12} & 6.855 & 11.532 & 3.182 & 4.841 \\ 1 & -1 & -1 & 5 & -4 \\ 0 & 1 & 6 & -15 & 13 \\ -10 & 2 & -7 & -1 & -1 \\ 0 & 14 & -4 & -6 & -7 \\ 1 & 3 & -7 & 6 & 19 \end{pmatrix}$$

- ① 格的理论概要
- ② 格基约化算法简介
- ③ 应用举例
 - 整数关系问题求解
 - 背包问题的求解

给定正整数 a_1, \dots, a_n (重量) 和 s , 找到 $e_1, \dots, e_n \in \{0, 1\}$ 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 $i = 1, 2, \dots, n$, 设 $a_i = 2^{i-1}$. 则

背包问题有解 $\Leftrightarrow 0 \leq s \leq 2^n - 1$,

且解向量 (e_1, \dots, e_n) 刚好对应于 s 的二进制表示:

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

一类容易求解的背包问题实例

超增长序列

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

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

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 $i = n-1, n-2, \dots, 1$,

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

假设

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

设 $\mathbf{e} = (e_1, \dots, e_n) \in \{0, 1\}^n$ 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

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

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 :

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

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

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

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

定理 (Lagarias-Odlyzko, 1983)

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

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

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

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

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

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

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

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:

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

- 这是目前关于 c_0 最小的一个结果.

包含的内容

- 从 Kepler 猜想到 Kepler 定理
- 格的基本知识
- 格约化算法简介
- 在零误差计算和密码分析中的应用

包含的内容

- 从 Kepler 猜想到 Kepler 定理
- 格的基本知识
- 格约化算法简介
- 在零误差计算和密码分析中的应用

未包含的内容

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

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.**

-  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.
-  R. Kannan, et al. Math. Comput. 50: 235–250, 1988.
-  J. Lagarias, A. Odlyzko. In FOCS '83, p. 1–10, 1983.
-  M. Coster, et al. Comput. Compl. 2: 111-128, 1992.

- 本课件中人物肖像来自 wikipedia.org.
- 其余非本人制作的图片都有超链接.
- 本课件内容可从如下网址下载:



www.arcnl.org/jchen/download/intro2lattice.pdf

THANKS