勉強会形式ゼミ資料②
L. Ducas, L. N. Pulles and M. Stevens Towards a modern LLL implementation[1]

佐藤 新

July 6, 2025

本セミナーで用いられる記号など

- $\boldsymbol{A} \in M_{n,m}(\mathbb{R})$ に対して，$\|\boldsymbol{A}\|_{\max} := \max_{i,j} |a_{i,j}|$
- $\boldsymbol{A} \in M_{n,m}(\mathbb{R})$ に対して，$\boldsymbol{A}_{[r,s]\times[t,u]} := [a_{i,j}]_{r\leq i\leq s, t\leq j\leq u}$

[1] の貢献

- BLASter の提案
  - 高速かつモダンな LLL の実装
  - [2] の分割手法を利用
  - サイズ簡約を Seysen 簡約に置き換え
  - Cholesky 分解を QR 分解に置き換え

[2] の分割手法

Algorithm Reduce[2]

Require: 格子 L の基底行列 $\boldsymbol{B} \in M_n(\mathbb{Z})$，簡約パラメタ $0 < \eta \leq 1$

Ensure: 簡約された基底 $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$

$n_r \leftarrow n$ 以下最大の specific dimension

computes GSO $B \leftarrow [\|\boldsymbol{b}_i\|^\star], \boldsymbol{U} \leftarrow [\mu_{i,j}]$

$\texttt{sizeReduce}(\boldsymbol{B})$

$\overline{\beta} \leftarrow 2 + \lceil \log_2 (\sqrt{n}(\max_{i \leq n} \|\boldsymbol{b}_i\|)^n) \rceil$

for $t = 1$ to $(n/n_r)^2 \lceil \log(\overline{\beta}/\eta) \rceil$ **do**

  for $k = 0$ to $n - n_r$ **do**

    $B' \leftarrow B_{[k+1, n_r+k]}$; $\boldsymbol{U}' \leftarrow \boldsymbol{U}_{[k+1,n_r+k] \times [k+1,n_r+k]}$

    $\boldsymbol{V} \leftarrow \texttt{recReduce}(B', \boldsymbol{U}', \eta, \overline{\beta})$

    $\boldsymbol{V}' \leftarrow \boldsymbol{E}_n$

    $\boldsymbol{B} \leftarrow \boldsymbol{V}'\boldsymbol{B}$

    $\texttt{sizeReduce}(\boldsymbol{B})$

Algorithm L^2 簡約 [?]

Require: パラメタ $\frac{1}{4} < \delta < 1, \frac{1}{2} < \eta < \sqrt{\delta}$，基底 $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$

Ensure: 簡約された基底 $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$

$\overline{\delta} \leftarrow (\delta + 1)/2$; $\overline{\eta} \leftarrow \frac{\eta + 1/2}{2}$

$\overline{r}_{1,1} \leftarrow \diamond(\|\boldsymbol{b}_1\|^2)$; $\kappa \leftarrow 2$

while $\kappa \leq n$ **do**

  $\texttt{sizeReduce}(\overline{\eta}, \kappa)$

  if $\overline{\delta}\overline{r}_{\kappa-1,\kappa-1} < \overline{r}_{\kappa,\kappa} + \overline{\mu}^2_{\kappa,\kappa-1}\overline{r}_{\kappa-1,\kappa-1}$ **then**

    $\kappa \leftarrow \kappa + 1$

  else

Seysen 簡約（1/3）

- $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$: 基底
- $\boldsymbol{B} = (\boldsymbol{b}_1^\top, \ldots, \boldsymbol{b}_n^\top)^\top$: 基底行列
- $\boldsymbol{B} = \boldsymbol{R}\boldsymbol{Q}$ ($\boldsymbol{R}$： 下三角行列, $\boldsymbol{Q}$： 直交行列)
- $\boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{1,1} & \boldsymbol{O}_{\lfloor n/2 \rfloor, n - \lfloor n/2 \rfloor} \\ \boldsymbol{R}_{2,1} & \boldsymbol{R}_{2,2} \end{bmatrix}$

定義 1 (Seysen 簡約)

$\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ が Seysen 簡約されているとは

$$n = 1 \vee \left(\boldsymbol{R}_{1,1},\ \boldsymbol{R}_{2,2} \text{が Seysen 簡約されている} \wedge \left\| \boldsymbol{R}_{2,1} \boldsymbol{R}_{1,1}^{-1} \right\|_{\max} \leq \frac{1}{2} \right)$$

Seysen 簡約（2/3）

Seysen 簡約は大まかには次のようなことを $\boldsymbol{B} = \boldsymbol{R}\boldsymbol{Q}$ なる $\boldsymbol{R}$ に対して再帰的に行う．

1. $\boldsymbol{R}$ が 1 次正方なら何もしない
2. $\boldsymbol{R}$ を $\begin{bmatrix} \boldsymbol{R}_{1,1} & \boldsymbol{O} \\ \boldsymbol{R}_{2,1} & \boldsymbol{R}_{2,2} \end{bmatrix}$ とブロックに分ける
3. $\boldsymbol{R}_{1,1}, \boldsymbol{R}_{2,2}$ をそれぞれ Seysen 簡約

Seysen 簡約（3/3）

Algorithm Seysen 簡約 [1]

Require: 下三角行列 $\boldsymbol{R} \in M_n(\mathbb{R})$ s.t. $\boldsymbol{B} = \boldsymbol{RQ}$
Ensure: $\boldsymbol{UB}$ が簡約基底行列となるような unimodular 行列 $\boldsymbol{U} \in M_n(\mathbb{R})$

if $\boldsymbol{R} \in M_1(\mathbb{R})$ **then**
 return $[1]$
$\begin{bmatrix} \boldsymbol{R}_{1,1} & \boldsymbol{O} \\ \boldsymbol{R}_{2,1} & \boldsymbol{R}_{2,2} \end{bmatrix} \leftarrow \boldsymbol{R}$ /* $\boldsymbol{R}_{1,1} \in M_{\lfloor n \rfloor}(\mathbb{R})$, $\boldsymbol{R}_{2,1} \in M_{n-\lfloor n \rfloor, \lfloor n \rfloor}(\mathbb{R})$, $\boldsymbol{R}_{2,2} \in M_{n-\lfloor n \rfloor, n-\lfloor n \rfloor}(\mathbb{R})$ */
$\boldsymbol{U}_{1,1} \leftarrow \texttt{seysenReduce}(\boldsymbol{R}_{1,1})$; $\boldsymbol{U}_{2,2} \leftarrow \texttt{seysenReduce}(\boldsymbol{R}_{2,2})$
$\boldsymbol{R}_{2,1} \leftarrow \boldsymbol{U}_{2,2}\boldsymbol{R}_{2,1}$
$\boldsymbol{U}_{2,1} \leftarrow \lfloor -\boldsymbol{R}_{2,1}\boldsymbol{R}_{1,1}^{-1} \rceil$
$\boldsymbol{R}_{2,1} \leftarrow \boldsymbol{U}_{2,1}\boldsymbol{R}_{1,1} + \boldsymbol{R}_{2,1}$
return $\begin{bmatrix} \boldsymbol{U}_{1,1} & \boldsymbol{O} \\ \boldsymbol{U}_{2,1}\boldsymbol{U}_{1,1} & \boldsymbol{U}_{2,2} \end{bmatrix}$

BLASter LLL

weakly-LLL 簡約

定義 2 (weakly-LLL 簡約基底)

$\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ が weakly-LLL 簡約されているとは

$$\left\|\boldsymbol{b}_k^\star\right\|^2 \geq (\delta - \mu_{k,k-1}^2)\left\|\boldsymbol{b}_{k-1}^\star\right\|^2$$

なるときをいう．

- Lovász 条件のみを満たす

Algorithm BLASter LLL アルゴリズム [1]

Require: 下三角行列 $\boldsymbol{R} \in M_n(\mathbb{R})$ s.t. $\boldsymbol{B} = \boldsymbol{R}\boldsymbol{Q}$

Ensure: $\boldsymbol{U}\boldsymbol{B}$ が簡約基底行列となるような unimodular 行列 $\boldsymbol{U} \in M_n(\mathbb{R})$

$i_0 \leftarrow 0$; $\boldsymbol{U} \leftarrow \boldsymbol{E}_n$

do

  $\boldsymbol{R} \leftarrow \boldsymbol{B} = \boldsymbol{R}\boldsymbol{Q}$ なる下三角行列 $\boldsymbol{R}$ ($\boldsymbol{Q}$: 直交行列)

  $i_0 \leftarrow \ell/2 - i_0$

  $\mathcal{I} \leftarrow \{(i_0 + k\ell + 1, \min\{n, i_0 + k\ell + \ell\}) \mid 0 \le k < (n - i_0)/\ell\}$

  for $(i, j) \in \mathcal{I}$ **do** $\boldsymbol{V}_i \leftarrow \mathrm{LLL}(\boldsymbol{R}_{[i,j]\times[i,j]}, \delta)$ **endfor** /* $\boldsymbol{V}_i$ は LLL 基底を与えるユニモジュラ行列 */

  for $(i, j) \in \mathcal{I}$ **do**

    $\boldsymbol{B}_{[i,j]\times[1,n]} \leftarrow \boldsymbol{V}_i \boldsymbol{B}_{[i,j]\times[1,n]}$; $\boldsymbol{U}_{[i,j]\times[1,n]} \leftarrow \boldsymbol{V}_i \boldsymbol{U}_{[i,j]\times[1,n]}$

  $\boldsymbol{R} \leftarrow \boldsymbol{B} = \boldsymbol{R}\boldsymbol{Q}$ なる下三角行列 $\boldsymbol{R}$ ($\boldsymbol{Q}$: 直交行列)

  $\boldsymbol{W} \leftarrow \texttt{seysenReduce}(\boldsymbol{R})$; $\boldsymbol{B} \leftarrow \boldsymbol{W}\boldsymbol{B}$; $\boldsymbol{U} \leftarrow \boldsymbol{W}\boldsymbol{U}$

  $f \leftarrow \texttt{true}$ /* $\boldsymbol{B}$ が weakly-LLL 簡約されているか */

  for $i = 1$to$n - 1$ **do**

    if $\delta r_{i,i}^2 > r_{i,i+1}^2 + r_{i+1,i+1}^2$ **then** $f \leftarrow \texttt{false}$; **break endif**

while not f

参考文献 I

[1] Léo Ducas, Ludo N. Pulles, and Marc Stevens. Towards a modern LLL implementation. Cryptology ePrint Archive, Paper 2025/774, 2025.

[2] Arnold Neumaier and Damien Stehle. Faster LLL-type reduction of lattice bases. Cryptology ePrint Archive, Paper 2016/852, 2016.