LATTICE BASIS REDUCTION

4. THE LLL ALGORITHM

Alfred Menezes
cryptography101.ca

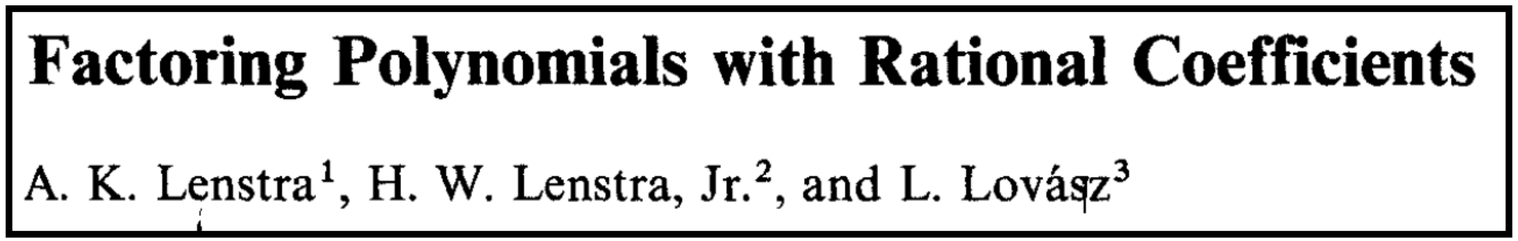
October 2025

Outline

1. LLL-reduced basis
2. Size reduction
3. Swap
4. Full algorithm
5. Termination
6. Running time
7. Example

Introduction

- The LLL algorithm (also known as L^3) is a polynomial-time algorithm that finds a (relatively) short basis for a lattice.
- It was discovered in 1982 by Arjen Lenstra, Hendrik Lenstra, and László Lovász.
- Interestingly, the first major application of LLL was a polynomial-time algorithm for factoring polynomials with rational coefficients.



Factoring Polynomials with Rational Coefficients

A. K. Lenstra[1], H. W. Lenstra, Jr.[2], and L. Lovász[3]

- Since then, LLL has found numerous applications in computational number theory and cryptography.
- Its most prominent application is solving the Shortest Vector Problem (SVP), which is central for assessing the security of lattice-based cryptosystems.

Gram-Schmidt orthogonalization

- Let $B = [b_1, \ldots, b_n]$ be a basis for an (integer) lattice $L \subseteq \mathbb{Z}^n$.
- The corresponding Gram-Schmidt (GS) basis is $B^* = [b_1^*, \ldots, b_n^*]$, where $b_1^* = b_1$ and $b_i^* = \Pi_{i-1}(b_i)$ for $2 \leq i \leq n$, where $\Pi_{i-1}(b_i)$, the projection of b_i onto $\langle b_1^*, \ldots, b_{i-1}^* \rangle^\perp$, is defined as $\Pi_{i-1}(b_i) = b_i - \sum_{j<i} \mu_{ij} b_j^*$.

  The Gram-Schmidt coefficients are $\mu_{ij} = \langle b_i, b_j^* \rangle / \|b_j^*\|^2$.
- B^* is an orthogonal basis for $\mathbb{R}^n$, but in general is not a basis for L.
- We have $\|b_i^*\| \leq \|b_i\|$ and $\langle b_i, b_i^* \rangle = \langle b_i^*, b_i^* \rangle$ for all $1 \leq i \leq n$.
- The volume of L is $\mathrm{vol}(L) = \Pi_{i=1}^n \|b_i^*\| \in \mathbb{Z}_{\geq 1}$.

LLL algorithm: Main ideas

- As with Gauss reduction, it's relatively simple to find a lattice basis $B = [b_1, \ldots, b_n]$ for which $|\mu_{ij}| \leq \frac{1}{2}$ for all $1 \leq j < i \leq n$.
  - This ensures that the basis vectors are reasonably orthogonal to each other, but does not guarantee that they are short.
- Now, for a randomly selected lattice basis B, one expects that the GS vectors $b_1^*, \ldots, b_n^*$ rapidly decrease in length (since $b_i^* = \Pi_{i-1}(b_i)$).
- The LLL algorithm repeatedly applies a "swap" operation, whose purpose is to reduce the rate at which the GS vectors decrease in length.
  - Since $\mathrm{vol}(L) = \prod_{i=1}^{n} \|b_i^*\|$ is a lattice invariant, this reduction results in a more uniform distribution in the lengths of the GS vectors b_i^*.
  - In particular, b_1^* is expected to relatively short.
  - Since $b_1 = b_1^*$, the lattice basis vector b_1 is also relatively short; more precisely $\|b_1\| \leq 2^{(n-1)/2}\lambda_1(L)$.

LLL-reduced basis

Definition. A lattice basis $[b_1, \ldots, b_n]$ is size-reduced if $|\mu_{ij}| \leq \frac{1}{2}$ for all $1 \leq j < i \leq n$.

A size-reduced basis is LLL-reduced if the Lovász condition is satisfied:

$$\|b_k^*\|^2 \geq (\tfrac{3}{4} - \mu_{k,k-1}^2)\, \|b_{k-1}^*\|^2 \text{ for all } 2 \leq k \leq n.$$

Notes:

1. Since $0 \leq \mu_{k,k-1}^2 \leq \frac{1}{4}$, the Lovász condition implies that $\|b_k^*\|^2 \geq \frac{1}{2}\|b_{k-1}^*\|^2$.
   Thus, the GS basis vectors do not get small too quickly.

2. An alternate formulation of the Lovász condition is: $\|b_k^* + \mu_{k,k-1} b_{k-1}^*\|^2 \geq \frac{3}{4}\|b_{k-1}^*\|^2$.

3. Exercise: Show that another formulation of the Lovász condition is $\|\Pi_{k-2}(b_k)\|^2 \geq \frac{3}{4}\|\Pi_{k-2}(b_{k-1})\|^2$.

4. The constant $\frac{3}{4}$ in the Lovász condition can be replaced by any δ, $\frac{1}{4} < \delta < 1$.

Shortness of the first vector in an LLL-reduced basis

- **Theorem**. Let $[b_1, \ldots, b_n]$ be a lattice basis. Then $\lambda_1(L) \geq \min_i \|b_i^*\|$.
- **Theorem**. Let $[b_1, \ldots, b_n]$ be an LLL-reduced basis for a lattice L.
  Then $\|b_1\| \leq 2^{(n-1)/2}\lambda_1(L)$.
- Proof. For each $2 \leq i \leq n$, we have

$$\|b_i^*\|^2 \geq (\tfrac{3}{4} - \mu_{i,i-1}^2)\|b_{i-1}^*\|^2 \geq (\tfrac{3}{4} - \tfrac{1}{4})\|b_{i-1}^*\|^2 = \tfrac{1}{2}\|b_{i-1}^*\|^2,$$

  whence $\|b_{i-1}^*\|^2 \leq 2\|b_i^*\|^2$.

  Hence, $\|b_1\|^2 = \|b_1^*\|^2 \leq 2^{i-1}\|b_i^*\|^2$, so $\|b_1\|^2 \leq 2^{n-1}\min_i \|b_i^*\|^2 \leq 2^{n-1}\lambda_1(L)^2$.

  We conclude that $\|b_1\| \leq 2^{(n-1)/2}\lambda_1(L)$. $\square$

- **Exercise**. Prove that $\|b_i\| \leq 2^{(n-1)/2}\lambda_i(L)$ for $1 \leq i \leq n$.

LLL lattice basis reduction

- The LLL lattice basis reduction algorithm has two components:
    1. Size reduction
    2. Swap
- Note. I'm presenting an unoptimized version of the algorithm. Some of the optimizations and variants of LLL, including BKZ, will be mentioned in V6.

Size reduction

Input: Lattice basis $B = [b_1, \ldots, b_n]$.

Output: Size-reduced basis $\tilde{B} = [\tilde{b}_1, \ldots, \tilde{b}_n]$.

1. Compute the GS basis $B^* = [b_1^*, \ldots, b_n^*]$.
2. For i from 2 to n do
   (a) For j from $i - 1$ to 1 do
   - Compute $\mu_{ij} = \langle b_i, b_j^* \rangle / \|b_j^*\|^2$.
   - Set $b_i \leftarrow b_i - \lfloor \mu_{ij} \rceil b_j$.
3. Return($\tilde{B} = [b_1, \ldots, b_n]$).

- **Claim 1**. $\tilde{B}$ is a basis for $L(B)$.
- **Claim 2**. The Gram-Schmidt basis of $\tilde{B}$ is identical to that of B.
- **Claim 3**. $\tilde{B}$ is size reduced.

Size reduction doesn't change the GS basis

- **Claim 1**. $\tilde{B}$ is a basis for $L(B)$.
  Proof. Each operation $b_i \leftarrow b_i - \lfloor \mu_{ij} \rceil b_j$ preserves the basis. □

- **Claim 2**. The Gram-Schmidt basis of $\tilde{B}$ is identical to that of B.

  Proof. We need to prove that $\tilde{b}_i^* = b_i^*$ for all $1 \leq i \leq n$.

  Let $V_i = \text{Span}(b_1, \ldots, b_i)$ and $\tilde{V}_i = \text{Span}(\tilde{b}_1, \ldots, \tilde{b}_i)$ for $1 \leq i \leq n$.

  We have $\tilde{V}_i = V_i$ for all $1 \leq i \leq n$.

  Thus $\tilde{b}_i^* = \Pi_{(\tilde{V}_{i-1})^\perp}(\tilde{b}_i) = \Pi_{(V_{i-1})^\perp}(\tilde{b}_i) = \Pi_{(V_{i-1})^\perp}(b_i) = b_i^*$,

  since $\tilde{b}_i = b_i + c_i$ where c_i is an integer linear combination of $b_1, \ldots, b_{i-1}$

  whence $c_i \in V_{i-1}$. □

Size reduction works (1)

- **Claim 3**. $\tilde{B} = [\tilde{b}_1, \ldots, \tilde{b}_n]$ is size-reduced.
- Proof. We need to prove that $|\tilde{\mu}_{ij}| \leq \frac{1}{2}$ for all $1 \leq j < i \leq n$.
  We'll show this for $i = 2$ and $i = 3$.
  The general statement can be proven by induction (exercise).
- Note that $\tilde{b}_1 = b_1$.
- $(i = 2, j = 1)$ The operation is: $\tilde{b}_2 \leftarrow b_2 - \lfloor \mu_{21} \rceil b_1$, where $\mu_{21} = \langle b_2, b_1^* \rangle / \|b_1^*\|^2$.

  The new GS coefficient is

  $$\tilde{\mu}_{21} = \langle \tilde{b}_2, b_1^* \rangle / \|b_1^*\|^2 = \langle b_2, b_1^* \rangle / \|b_1^*\|^2 - \lfloor \mu_{21} \rceil \langle b_1, b_1^* \rangle / \|b_1^*\|^2 = \mu_{21} - \lfloor \mu_{21} \rceil,$$

  since $\langle b_1, b_1^* \rangle = \langle b_1^*, b_1^* \rangle = \|b_1^*\|^2$. Thus, $|\tilde{\mu}_{21}| \leq \frac{1}{2}$ as required.
- $(i = 3, j = 2)$ The operation is: $b_3' \leftarrow b_3 - \lfloor \mu_{32} \rceil \tilde{b}_2$, where $\mu_{32} = \langle b_3, b_2^* \rangle / \|b_2^*\|^2$.
- $(i = 3, j = 1)$ The operation is: $\tilde{b}_3 \leftarrow b_3' - \lfloor \mu_{31}' \rceil b_1$, where $\mu_{31}' = \langle b_3', b_1^* \rangle / \|b_1^*\|^2$.

Size reduction works (2)

- Claim 3 proof (cont'd). We have

$$\begin{aligned}\tilde{\mu}_{31} &= \langle \tilde{b}_3, b_1^* \rangle / \|b_1^*\|^2 \\ &= \langle b_3', b_1^* \rangle / \|b_1^*\|^2 - \lfloor \mu_{31}' \rceil \langle b_1, b_1^* \rangle / \|b_1^*\|^2 \quad \text{(note: } \langle b_1, b_1^* \rangle = \langle b_1^*, b_1^* \rangle = \|b_1^*\|^2\text{)} \\ &= \mu_{31}' - \lfloor \mu_{31}' \rceil, \text{ so } |\tilde{\mu}_{31}| \leq \tfrac{1}{2} \text{ as required .}\end{aligned}$$

- Also,

$$\begin{aligned}\tilde{\mu}_{32} &= \langle \tilde{b}_3, b_2^* \rangle / \|b_2^*\|^2 \\ &= \langle b_3', b_2^* \rangle / \|b_2^*\|^2 - \lfloor \mu_{31}' \rceil \langle b_1, b_2^* \rangle / \|b_2^*\|^2 \quad \text{(note: } \langle b_1, b_2^* \rangle = 0\text{)} \\ &= \langle b_3, b_2^* \rangle / \|b_2^*\|^2 - \lfloor \mu_{32} \rceil \langle \tilde{b}_2, b_2^* \rangle / \|b_2^*\|^2 \quad \text{(note: } \langle \tilde{b}_2, b_2^* \rangle = \langle b_2^*, b_2^* \rangle = \|b_2^*\|^2\text{)} \\ &= \mu_{32} - \lfloor \mu_{32} \rceil \text{ so } |\tilde{\mu}_{32}| \leq \tfrac{1}{2} \text{ as required .} \quad \square\end{aligned}$$

Swap

Input: Size-reduced lattice basis $B = [b_1, \ldots, b_n]$.

Output: A basis $\tilde{B} = [\tilde{b}_1, \ldots, \tilde{b}_n]$.

1. If there exists an index $j \in [2, n]$ such that $\|b_j^*\|^2 < (\frac{3}{4} - \mu_{j,j-1}^2)\,\|b_{j-1}^*\|^2$ then
   - Swap b_{j-1} and b_j.
2. Return($\tilde{B} = [b_1, \ldots, b_n]$).

- Goal. Make progress towards satisfying the Lovász condition.
- Note: The output basis $\tilde{B}$ is not necessarily size-reduced.

The LLL algorithm

Input: Lattice basis $B = [b_1, \ldots, b_n]$.

Output: LLL-reduced basis $[b_1, \ldots, b_n]$.

1. Size-reduce $[b_1, \ldots, b_n]$.
2. If there exists an index $j \in [2, n]$ such that $\|b_j^*\|^2 < (\frac{3}{4} - \mu_{j,j-1}^2)\,\|b_{j-1}^*\|^2$ then
   - Swap b_{j-1} and b_j.
   - Go to step 1.
3. Return($[b_1, \ldots, b_n]$).

Size reduction

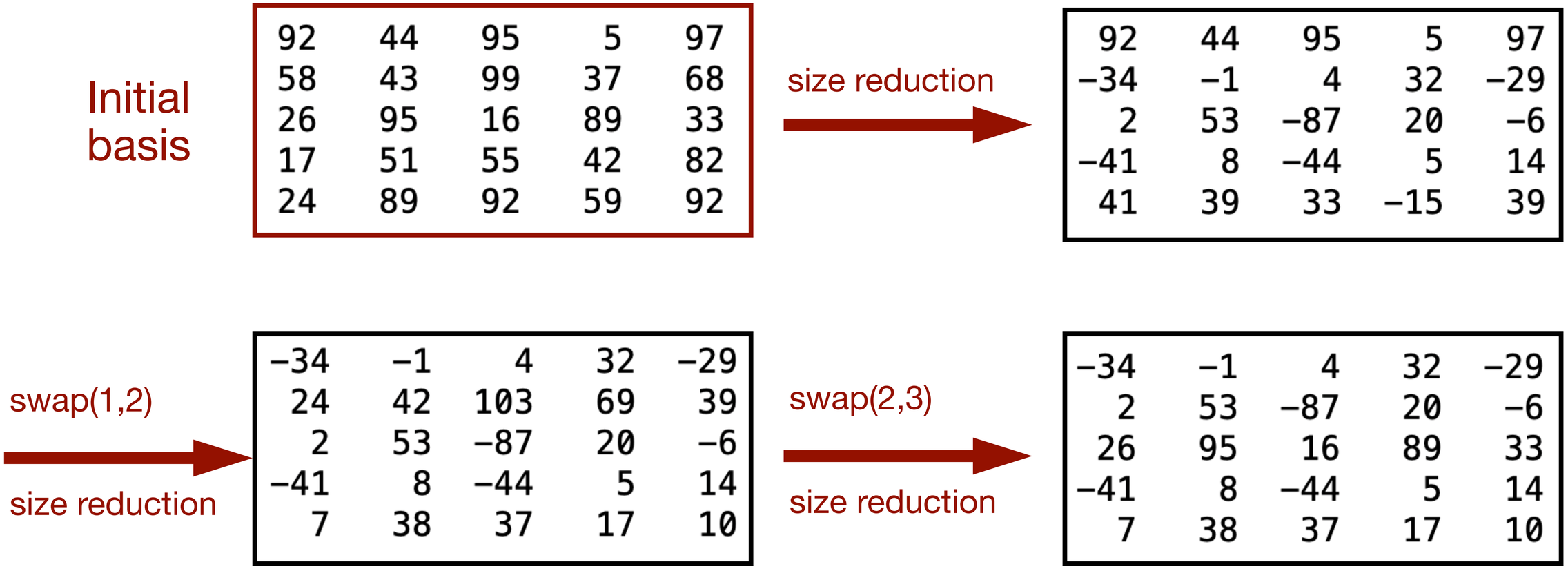
Input: Lattice basis $B = [b_1, \ldots, b_n]$.

Output: Size-reduced basis $\tilde{B} = [\tilde{b}_1, \ldots, \tilde{b}_n]$.

1. Compute the GS basis $B^* = [b_1^*, \ldots, b_n^*]$.
2. For i from 2 to n do
   (a) For j from $i - 1$ to 1 do
   - Compute $\mu_{ij} = \langle b_i, b_j^* \rangle / \|b_j^*\|^2$.
   - Set $b_i \leftarrow b_i - \lfloor \mu_{ij} \rceil b_j$.
3. Return($\tilde{B} = [b_1, \ldots, b_n]$).

Example: LLL algorithm (1)

- $n = 5$, each entry randomly selected from [0,99], basis vectors are *rows*.



Example: LLL algorithm (2)

swap(3,4)

size reduction

−34	−1	4	32	−29
2	53	−87	20	−6
−41	8	−44	5	14
17	51	55	42	82
−34	46	−7	22	24

swap(2,3)

size reduction

−34	−1	4	32	−29
−41	8	−44	5	14
43	45	−43	15	−20
17	51	55	42	82
7	38	37	17	10

swap(4,5)

size reduction

−34	−1	4	32	−29
−41	8	−44	5	14
43	45	−43	15	−20
7	38	37	17	10
10	−34	29	35	19

swap(3,4)

size reduction

−34	−1	4	32	−29
−41	8	−44	5	14
7	38	37	17	10
43	45	−43	15	−20
10	−34	29	35	19

swap(4,5)

size reduction

−34	−1	4	32	−29
−41	8	−44	5	14
7	38	37	17	10
10	−34	29	35	19
53	11	−14	50	−1

LLL-reduced basis

The lengths of the vectors in the original basis are 169.9, 144.9, 137.7, 120.0, and 170.0, respectively.

The lengths of the vectors in the LLL-reduced basis are 55.1, 62.4, 57.0, 60.7, and 75.0, respectively

Termination: potential function

- Consider a lattice L with basis $B = [b_1, \ldots, b_n]$.
- Let the associated Gram-Schmidt basis be $B^* = [b_1^*, \ldots, b_n^*]$.
- For $1 \le \ell \le n$, define $B_\ell = [b_1, \ldots, b_\ell]$, and $L_\ell = L(B_\ell)$.
  - Note that L_ℓ is a rank-ℓ sublattice of L.
  - The volume of L_ℓ is $\text{vol}(L_\ell) = \sqrt{\det(B_\ell^T B_\ell)}$.
- **Fact**. $\text{vol}(L_\ell) = \prod_{i=1}^{\ell} \|b_i^*\|$.

- **Definition**. For each $1 \le \ell \le n$, let $d_\ell = \prod_{i=1}^{\ell} \|b_i^*\|^2$.

  The potential function is
  $D = D(b_1, \ldots, b_n) = \prod_{\ell=1}^{n} d_\ell = \prod_{i=1}^{n} \|b_i^*\|^{2(n+1-i)}$.
- **Notes**:
  1. More weight is given to the first few GS basis vectors.
  2. Since we are only considering integer lattices, $\text{vol}(L_\ell)^2 = \det(B_\ell^T B_\ell) \in \mathbb{Z}_{\ge 1}$ and so $D \in \mathbb{Z}_{\ge 1}$.

Termination (1)

- **Theorem.** The LLL algorithm terminates after $O(n^2 \log X)$ swaps, where $X = \max_i \|b_i\|$.
- Proof. The only operation in the LLL algorithm that changes any L_ℓ is Swap.
- Consider Swap(b_{j-1}, b_j), which only changes L_{j-1}.
- After this swap, the new lattice basis is $\tilde{B} = [\tilde{b}_1, \ldots, \tilde{b}_n]$, where $\tilde{b}_{j-1} = b_j$, $\tilde{b}_j = b_{j-1}$, and $\tilde{b}_i = b_i$ for all $i \neq j-1, j$.
- The new GS basis is $\tilde{B}^* = [\tilde{b}_1^*, \ldots, \tilde{b}_n^*]$, where $\tilde{b}_i^* = b_i^*$ for all $i \neq j-1, j$.
  - Now, $\|\tilde{b}_{j-1}^*\|^2 = \|\Pi_{j-2}(\tilde{b}_{j-1})\|^2 = \|\Pi_{j-2}(b_j)\|^2 = \|b_j^* + \mu_{j,j-1} b_{j-1}^*\|^2 < \frac{3}{4}\|b_{j-1}^*\|^2$.
  - Thus, $\|\tilde{b}_{j-1}^*\|^2 < \frac{3}{4}\|b_{j-1}^*\|^2$, say $\|\tilde{b}_{j-1}^*\|^2 = \epsilon \|b_{j-1}^*\|^2$ for some $\epsilon < \frac{3}{4}$.
  - Hence, $\tilde{d}_{j-1} = \epsilon d_{j-1}$, whereas $\tilde{d}_i = d_i$ for all $i \neq j-1$.

Termination (2)

- It follows that $\tilde{D} = \prod_{\ell=1}^{n} \tilde{d}_\ell = \epsilon \prod_{\ell=1}^{n} d_\ell = \epsilon D$.
- Thus, the potential D decreases by a multiplicative factor of at least $\frac{4}{3}$ after each swap operation.
- Since D is a positive integer, we can conclude that the LLL algorithm must terminate.
- Now, the original potential is
$$D = \prod_{i=1}^{n} \|b_i^*\|^{2(n+1-i)} \leq \prod_{i=1}^{n} \|b_i\|^{2(n+1-i)} \leq \left(\max_i \|b_i\|\right)^{2(1+2+\cdots+n)} = X^{n^2+n},$$
where $X = \max_i \|b_i\|$.
- Thus, the maximum number of swaps before the LLL algorithm terminates is $\log_{4/3} X^{n^2+n} = O(n^2 \log X)$. □

Running time

- The maximum number of swaps before the LLL algorithm terminates is $O(n^2 \log X)$, where $X = \max_i \|b_i\|$.
- **Fact**. The LLL algorithm performs arithmetic operations on integers of bitlength at most $O(n \log X)$.
- **Theorem**. The running time of the LLL algorithm is $O(n^6 \log^3 X)$ bit operations.
    - Note: LLL is a polynomial-time algorithm.

Example: LLL algorithm

- $n = 15$, each entry randomly selected from $[-2, 2]$, basis vectors are *rows*.

$B =$

															Length
2	−1	2	−1	0	1	1	2	0	−2	2	−1	0	−1	−1	5.19
1	0	0	−2	−1	2	1	2	−1	0	−1	1	−2	2	0	5.09
−2	2	1	0	2	0	2	1	0	−1	0	−2	−2	0	2	5.56
2	−1	−1	−1	0	−1	1	−1	−2	2	−2	1	1	2	0	5.29
−1	−2	−1	1	2	0	1	−2	−2	−2	−1	−1	0	1	2	5.56
−2	1	−2	−1	−2	2	−2	−2	1	0	2	1	1	1	−1	5.91
−1	−1	0	2	2	1	0	0	2	0	0	2	0	1	0	4.47
−1	2	−1	0	0	1	0	−2	1	−1	0	−1	−2	−2	−2	5.09
2	−1	1	2	1	−2	0	−2	0	−1	1	−2	2	−1	0	5.47
−1	1	−2	−2	−1	−2	2	0	−1	2	2	0	−2	2	1	6.08
1	2	0	0	−2	2	0	−2	−1	−2	1	2	−2	−2	−2	6.24
0	−2	−2	0	0	−2	1	2	1	−2	−1	2	2	−2	−1	6.00
−2	2	1	1	2	1	−1	−2	1	−2	2	−2	−2	2	−2	6.70
1	2	1	2	−2	2	−1	2	2	2	−1	0	−2	1	1	6.16
−1	−1	−1	1	−2	2	−1	2	−1	2	2	−2	2	−1	−1	6.00

Example: randomized basis

- A "random basis" for the same lattice was generated by repeating the following operation 100 times:
  - Select $i, j \in_R [1, 15]$ with $i \neq j$ and select $t \in_R [-25, 25]$.
  - Set $b_i \leftarrow b_i - tb_j$.
- Here are the first two vectors in the resulting (large) lattice basis:

```
-115876334270    95334076546    44582453025    57364088199    86794272932    48376818744
 -62515108200   -91698909830    60554412628   -99007760874   102291706978   -91931888448
 -92286074031    67369918728   -85178843944

-413892649229   340531262846   159244076714   204897045599   310035529595   172790730091
-223283060344  -327547371938   216281025848  -353634883086   365359051013  -328376636917
-329645272684   240667714177  -304246036603
```

Example: LLL-reduced basis

- LLL-reduced basis

$B =$

															Length
−1	1	0	2	0	0	−1	−1	1	−1	0	−1	−1	0	1	3.61
2	0	1	0	−2	1	0	0	−2	−1	1	3	0	0	0	5.00
0	0	−1	−2	−1	−2	1	1	0	0	−1	−1	−1	−1	−2	4.47
2	−1	0	1	−2	−1	1	0	−2	0	−1	1	0	−1	0	4.36
0	0	1	−1	−1	−1	−1	2	0	−1	1	1	0	2	1	4.12
−1	−1	1	−1	1	0	0	0	1	0	−1	−1	−1	−2	−2	4.12
0	0	−1	1	−1	−1	1	2	0	0	1	−1	1	−1	1	3.74
−1	−1	−1	0	1	−1	1	1	2	0	−1	1	−1	0	−2	4.24
1	2	1	0	−1	2	−1	−1	1	−1	1	0	2	0	1	4.58
1	1	1	1	−1	1	−1	2	−2	0	1	−1	−1	1	0	4.36
3	−1	1	0	0	0	1	0	−1	−1	0	−1	0	1	0	4.00
0	0	−1	−1	1	1	0	0	1	−1	0	0	−3	−1	0	4.00
−2	−1	1	−1	−1	0	1	1	−1	0	0	0	0	0	1	3.46
0	−1	2	−1	−2	−2	1	0	0	1	0	−1	−1	−1	1	4.47
0	−2	1	0	−1	1	−2	−1	1	−1	1	−1	1	2	0	4.58

Example: Swaps

```
Swap:  1   2   potential = 2550.75  Difference = -2.447960
Swap:  1   2   potential = 2547.60  Difference = -3.147762
Swap:  1   2   potential = 2536.84  Difference = -10.758942
Swap:  2   3   potential = 2533.09  Difference = -3.751924
Swap:  2   3   potential = 2529.88  Difference = -3.202972
Swap:  2   3   potential = 2520.17  Difference = -9.709751
Swap:  1   2   potential = 2517.88  Difference = -2.294698
Swap:  1   2   potential = 2514.19  Difference = -3.688553
Swap:  2   3   potential = 2512.07  Difference = -2.125418
Swap:  2   3   potential = 2505.52  Difference = -6.547939
.....................................................
Swap:  6   7   potential = 421.81  Difference  = -0.587346
Swap: 11  12   potential = 420.86  Difference  = -0.950476
Swap: 10  11   potential = 420.28  Difference  = -0.579370
Swap:  9  10   potential = 419.70  Difference  = -0.575864
Swap: 12  13   potential = 418.84  Difference  = -0.867230
Swap: 13  14   potential = 417.89  Difference  = -0.941581
Swap: 14  15   potential = 417.24  Difference  = -0.655858
```

- Total number of swaps: 1157.
- "Potential" is $\log_2(D)$.
- "Difference" is $\log_2$ of New potential / Old potential.
- Note: $\log_2(3/4) = -0.415037$.

Example: Lengths of original and LLL-reduced basis vectors

	Lengths of original basis vectors	Lengths of original Gram-Schmidt basis vectors	Lengths of LLL	Lengths of GS
1	320436945197.4	320436945197.359360255541512	3.61	3.61
2	1144568732312.6	46732111.664529318780066	5.00	4.71
3	195578567373.0	12819.739043033123886	4.47	3.81
4	81148130911.1	397137.610643187052484	4.36	3.15
5	180908523242.6	14000.36711745820053 1	4.12	3.57
6	5340229.8	1676.733789170875920	4.12	2.88
7	1722666532.5	0.080881385402629	3.74	3.02
8	17775727743.2	0.276952900412341	4.24	3.35
9	104620743288.4	0.000045519492794	4.58	2.74
10	25839996203.4	2.172605704403767	4.36	3.09
11	2197024540111.8	0.574351076442320	4.00	2.97
12	4900011449631.2	0.019296310688108	4.00	2.44
13	656730.3	0.000000000176541	3.46	2.28
14	21423670206853.1	0.002453963889001	4.47	2.52
15	1622962205744.5	0.000000001243810	4.58	3.15

Example: Gram-Schmidt coefficients

- Product of the lengths of the original and LLL-reduced Gram-Schmidt basis vectors is 23677332.
- Matrix of Gram-Schmidt coefficients μ_{ij} of the LLL-reduced basis (where $\mu_{ii} = 1$):

```
 1.00
-0.46  1.00
-0.46 -0.35  1.00
-0.38  0.48  0.37  1.00
-0.15  0.27  0.01 -0.48  1.00
-0.08 -0.43  0.36 -0.18 -0.43  1.00
 0.00 -0.09  0.16  0.41  0.35 -0.36  1.00
-0.15 -0.40  0.34 -0.13 -0.05  0.33 -0.09  1.00
 0.31  0.40 -0.42 -0.45 -0.15 -0.28  0.26 -0.49  1.00
 0.08  0.38  0.03 -0.08  0.35 -0.09  0.24 -0.45 -0.46  1.00
-0.31  0.23 -0.00  0.40  0.15  0.05 -0.19 -0.29  0.17  0.38  1.00
 0.23 -0.07  0.33 -0.49 -0.35  0.24 -0.06  0.23 -0.42 -0.06  0.36  1.00
-0.23 -0.02  0.03 -0.02  0.36  0.29  0.24 -0.19 -0.41 -0.31 -0.24 -0.09  1.00
-0.15 -0.04  0.47  0.42  0.38  0.46  0.09 -0.48 -0.05 -0.37  0.00 -0.01  0.38  1.00
 0.23  0.11 -0.40 -0.29  0.36  0.33 -0.25 -0.26  0.41  0.01  0.32 -0.38  0.05 -0.11  1.00
```