Scribe Notes for *Algorithmic Number Theory*
Class 23—June 18, 1998

Scribes: Scott A. Guyer, Duxing Cai, and Degong Song

Abstract

The concept of a reduced basis is introduced and some properties are discussed. Also, the Lenstra, Lenstra, Lovasz (L^3) Algorithm is given, which starts with any given basis $\{b_1, b_2, \ldots, b_n\}$ and finds a reduced basis.

1 Reduced Basis and Properties

Vectors $b_1, b_2, \ldots, b_n$ are *quasi-orthogonal* if $|\mu_{ij}| \leq \frac{1}{2}$, $1 \leq j < i \leq n$. Vectors are *quasi-ordered* if $\|b_i^* + \mu_{i,i-1}b_{i-1}^*\|^2 \geq \frac{3}{4}\|b_{i-1}^*\|^2$. A basis is *reduced* if it is quasi-orthogonal and quasi-ordered.

THEOREM 1 *The following properties hold for a reduced basis $b_1, ..., b_n$ of a lattice L:*

1. $\|b_i^*\|^2 \geq \frac{1}{2}\|b_{i-1}^*\|^2$, *for* $2 \leq i \leq n$.
2. $\|b_i^*\|^2 \geq 2^{j-i}\|b_j^*\|^2$, *for* $1 \leq j \leq i \leq n$.
3. $\|b_i\|^2 \leq 2^{i-1}\|b_i^*\|^2$.
4. $\|b_j\|^2 \leq 2^{i-1}\|b_i^*\|^2$, *for* $1 \leq j \leq i \leq n$.
5. *If* $x \in L - \{0\}$, *then* $\|b_i\|^2 \leq 2^{n-1}\|x\|^2$.

Proof:

1. Consider

$$\|b_i^* + \mu_{i,i-1}b_{i-1}^*\|^2 \quad = \quad \|b_i^*\|^2 + |\mu_{i,i-1}|^2\|b_{i-1}^*\|^2 \geq \frac{3}{4}\|b_{i-1}^*\|^2.$$

   Since $|\mu_{i,j}| \leq \frac{1}{2}$ implies $|\mu_{i,i-1}|^2 \leq \frac{1}{4}$, we get

$$\|b_i^*\|^2 + \frac{1}{4}\|b_{i-1}^*\|^2 \geq \frac{3}{4}\|b_{i-1}^*\|^2.$$

   Hence $\|b_i^*\|^2 \geq \frac{1}{2}\|b_{i-1}^*\|^2$.

2. This follows from property 1 of this theorem.

3. Recall that

$$b_i = b_i^* + \sum_{j=1}^{i-1} \mu_{ij} b_j^*.$$

Squaring yields the following:

$$\begin{aligned} \|b_i\|^2 &= \|b_i^*\|^2 + \sum_{j=1}^{i-1} \mu_{ij}^2 \|b_j^*\|^2 \\ &\leq \|b_i^*\|^2 + \frac{1}{4} \sum_{j=1}^{i-1} 2^{i-j} \|b_i^*\|^2 = \|b_i^*\|^2 \left(1 + \frac{1}{4} \sum_{j=1}^{i-1} 2^{i-j}\right) \\ &\leq \|b_i^*\|^2 \left(1 + \frac{2^i}{4}\right) \leq 2^{i-1} \|b_i^*\|^2. \end{aligned}$$

4. By properties 2 and 3, we know that $\|b_j\|^2 \leq 2^{j-1}\|b_j^*\|^2$ and $\|b_j^*\|^2 \leq 2^{i-j}\|b_i^*\|^2$. Combining these and letting $j = 1$, we get $\|b_1\|^2 \leq 2^{i-1}\|b_i^*\|^2$.

5. Write $x = \sum_{i=1}^{n} r_i b_i$, where $r_i \in \mathbb{Z}$. Let k be the maximum integer such that $r_k \neq 0$ and hence $x = \sum_{i=1}^{k} r_i b_i$. It is easy to see that there exists an $r_i^* \in Q$ such that $x = \sum_{i=1}^{k} r_i^* b_i^*$. From

$$\begin{aligned} x &= \sum_{i=1}^{k} r_i b_i \\ &= \sum_{i=1}^{k} r_i \left(b_i^* + \sum_{j=1}^{i-1} \mu_{ij} b_j^*\right) \end{aligned}$$

and

$$r_i^* = \frac{\langle x, b_i^* \rangle}{\langle b_i^*, b_i^* \rangle},$$

we see that

$$r_i^* = r_i + \sum_{j=i+1}^{k} \mu_{ji},$$

for $0 \leq i \leq k-1$, and that

$$\begin{aligned} r_k^* &= \frac{\langle x, b_k^* \rangle}{\langle b_k^*, b_k^* \rangle} \\ &= r_k. \end{aligned}$$

Since $r_k \in \mathbb{Z}$ and $r_k \neq 0$, we have $|r_k| \geq 1$ and hence

$$\|x\|^2 = \sum_{i=1}^{k} |r_i^*|^2 \|b_i^*\|^2 \geq |r_k^*|^2 \|b_k^*\|^2 = |r_k|^2 \|b_k^*\|^2 \geq \|b_k^*\|^2.$$

By property 4, we get

$$\|b_1\|^2 \leq 2^{k-1}\|b_k^*\|^2 \leq 2^{k-1}\|x\|^2 \leq 2^{n-1}\|x\|^2. \qquad \square$$

As a consequence of property 5, every reduced basis has the following property:

$$\|b_1\|^2 \leq 2^{n-1}\sqrt{m(L)}.$$

2 Lenstra, Lenstra, Lovasz (L^3) Algorithm

The goal of this algorithm is to start with a basis $\{b_1, b_2, \ldots, b_n\}$ and find a reduced basis.

The first step is to compute the Gram-Schmidt orthogonalization $\{b_1^*, b_2^*, \ldots, b_n^*\}$ and the μ_{ij}'s. We obtain the matrix equation

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \mu_{21} & 1 & 0 & \cdots & \vdots \\ \mu_{31} & \mu_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \mu_{n1} & \cdots & \cdots & \cdots & 1 \end{bmatrix} \begin{bmatrix} b_1^* \\ b_2^* \\ b_3^* \\ \vdots \\ b_n^* \end{bmatrix},$$

or simply

$$B = MB^*.$$

Let $[x]$ be the unique integer in $(x - \frac{1}{2}, x + \frac{1}{2}]$. For any fixed i, j, we can make $|\mu_{ij}| \leq \frac{1}{2}$ as follows:

STRAIGHTEN-ELEMENT(i, j)
1 $m \leftarrow [\mu_{ij}]$
2 $b_i \leftarrow b_i - mb_j$
3 **for** $k \leftarrow 1$ **to** j
4 **do** $\mu_{ik} \leftarrow \mu_{ik} - m\mu_{jk}$

STRAIGHTEN-ELEMENT takes $O(n)$ arithmetic operations. For any fixed i, we can make all $|\mu_{ij}| \leq \frac{1}{2}$ in row i as follows:

STRAIGHTEN-ROW(i)
1 **for** $k \leftarrow i - 1$ **down to** 1
2 **do** STRAIGHTEN-ELEMENT(i,k)

STRAIGHTEN-ROW takes $O(n^2)$ time. Now, we need to work on getting quasi-ordering. Suppose B is not quasi-ordered and k is the smallest integer such that

$$\frac{3}{4}\|b_{k-1}^*\|^2 > \|b_k^* + \mu_{k,k-1}b_{k-1}^*\|^2.$$

Then we swap rows $k-1$ and k in B.

The old $(k-1)$th orthogonal basis element is

$$b_{k-1}^* = b_{k-1} - \sum_{j=1}^{k-2} \mu_{k-1,j} b_j^*.$$

The new $(k-1)$th orthogonal basis element is

$$b_k - \sum_{j=1}^{k-2} \mu_{k,j} b_j^* = b_k^* + \mu_{k,k-1} b_{k-1}^*.$$

Hence the norm square of the $(k-1)$th row has decreased by a factor strictly less than $\frac{3}{4}$. These are all the ideas used in the Lenstra-Lenstra-Lovasz algorithm. Pseudocode for the algorithm follows.

$L^3(b_1, b_2, ..., b_n)$

```
 1  for i ← 1 to n
 2      do GRAM-SCHMIDT(b_i)
 3         b_i^* ← b_i
 4         for j ← 1 to i − 1
 5             do μ_ij ← ⟨b_i, b_j^*⟩/B_i
 6                b_i^* ← b_i^* − μ_ij b_j^*
 7         B_i ← ⟨b_i^*, b_i^*⟩        ▷ B_i = ‖b_i^*‖^2
 8  k ← 2
 9  while k ≤ n
10      do STRAIGHTEN-ELEMENT(k, k − 1)
11         if B_k ≥ (3/4 − μ_{k,k−1}^2) B_{k−1}
12           then STRAIGHTEN-ROW(k)
13                k ← k + 1
14           else ▷ Swap row k − 1 and k
15                μ ← μ_{k,k−1}
16                B ← B_k + μ^2 B_{k−1}
17                μ_{k,k−1} ← μ B_{k−1}/B
18                B_k ← B_{k−1} B_k/B
19                B_{k−1} ← B
20                (b_{k−1}, b_k) ← (b_k, b_{k−1})
21                for j ← 1 to k − 2
22                    do ▷(μ_{k−1,j}, μ_{k,j}) ← (μ_{k,j}, μ_{k−1,j})
23         for i ← k + 1 to n
24             do ( μ_{i,k−1} ) ← [ 1  μ_{k,k−1} ] [ 0   1 ] ( μ_{i,k−1} )
                  ( μ_{i,k}   )   [ 0  1       ] [ 1  −μ ] ( μ_{i,k}   )
25         if k > 2
26           then k ← k − 1
27  return (b_1, b_2, ..., b_n)
```

The following begins a proof that algorithm L^3 terminates. The proof is concluded tomorrow.

Proof: Define

$$d_i = \left|\det(\langle b_j, b_k\rangle)_{1\le j,k\le i}\right| .$$

This is equal to

$$\begin{aligned} &= \left|\det(\langle b_j^*, b_k^*\rangle)_{1\le j,k\le i}\right| \\ &= \prod_{j=1}^{i} \|b_j^*\|^2 \end{aligned}$$

for $0 \le i \le n$. It is easy to see each d_i is a positive integer, $d_0 = 1$, and $d_n = (d(L))^2$. □