CSC 2421S Approximation Algorithms

Lecture #8: March 4, 2003

Lecturer: Professor Charles Rackoff

Scribe Notes by: Daniel Ivan, Steven Hellebust

Approximating Shortest Vector in a Lattice

1 Introduction

The Shortest Vector Problem (or shortly **SVP**) is a central computational problem in the classical area of geometry of numbers. The approximation algorithm presented in this lecture has many applications in computational number theory and cryptography. Two of these applications in breaking cryptography may be presented in one of the next lectures.

2 Problem Definition

Let $\mathbf{a}_1$, $\mathbf{a}_2$, ..., $\mathbf{a}_n \in \mathbb{Z}^n$ be n linearly independent vectors, and let the lattice generated by the basis $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}$ be:

$$\mathcal{L}(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n) = \{\lambda_1\mathbf{a}_1 + \lambda_2\mathbf{a}_2 + \ldots + \lambda_n\mathbf{a}_n | \lambda_i \in \mathbb{Z}, 1 \leq i \leq n\} \tag{1}$$

Problem: Find a shortest non-zero vector, in Euclidean norm[1], in $\mathcal{L}(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n)$.

3 Complexity Results

Ajtai (1998), in [1], showed that the **SVP** is **NP**-hard for randomized reductions, i.e. if the **SVP** can be done probabilistically in polynomial time, then **NP**=**RP**.

Miccianio (1998), in [2], gave a *lower bound* for the **SVP** by proving that finding a better than $\sqrt{2}$ approximation for **SVP** probabilistically in polynomial time, implies that **NP**=**RP**.

Lenstra, Lenstra and Lovász (1982), in [3], found an *upper bound* for the **SVP** using an algorithm (the "L^3 algorithm") which achieves a $2^{(n-1)/2}$ approximation in polynomial time. In spite of the huge size of this upper bound, it is a beautiful theorem with elegant and unintuitive applications.

[1]Given a vector $\vec{w} = (w(1), w(2), \ldots, w(n)) \in \mathbb{R}^n$, the Euclidean norm of $\vec{w}$ is defined as $||\vec{w}|| = \sqrt{w(1)^2 + w(2)^2 + \ldots + w(n)^2}$. The Euclidean scalar product of two vectors $\vec{w}$, $\vec{v} \in \mathbb{R}^n$ is $\vec{w} \cdot \vec{v} = w(1)v(1) + w(2)v(2) + \ldots + w(n)v(n) = ||\vec{w}|| \cdot ||\vec{v}|| \cdot \cos\theta$, where θ is the angle between $\vec{w}$ and $\vec{v}$.

Schnorr (1987), in [4], improved the upper bound to $2^{O(\frac{n}{\log n}(\log\log n)^2)}$.

Goldreich and Goldwasser (1998), in [5], gave an *un*-**NP** *hardness* result by showing that if it is **NP**-hard for randomized reductions to get a $\sqrt{n}$ approximation, then the Polynomial Time Hierarchy collapses.

4 Motivation for the L^3 algorithm

A 1-dimensional analogue of this problem is to find the shortest vector in the 1-dimensional lattice generated by two integers (although the vectors in this case are not linearly independent). This is simply the greatest common divisor of these integers, which is polynomial time computable by Euclid's algorithm:

---

input a and b $\{$assume w.l.o.g. that $a \geq b > 0\}$
loop
    $c \leftarrow$ remainder of a divided by b
    if $c = 0$ halt and output b
    $a \leftarrow b$
    $b \leftarrow c$
end loop

Figure 1: Euclid's Algorithm

The **SVP** in two dimensions is polynomial time solvable using an algorithm due to Gauss, as shown in Figure 2.

Input b_1 and b_2 $\{$assume that $||b_1|| \leq ||b_2||\}$
let $u_{21} = \frac{b_2 \cdot b_1}{||b_1||^2}$ be a function of b_1, b_2
while $u_{21} > 1/2$
    $b_2 \leftarrow b_2 - m b_1$ $\{$where m is the integer closest to $u_{21}\}$
    if $||b_1|| > ||b_2||$, switch b_1 and b_2
end while
output b_1

Figure 2: Gauss' Algorithm

Gauss' algorithm can be seen as a generalization of Euclid's algorithm. The L^3 algorithm used for the n-dimensional **SVP** is a further generalization. It transforms a basis by subtracting integer multiples of some vectors from others, creating a basis whose shortest element is an approximation for the shortest vector in the lattice.

5 The L^3 algorithm

Let $\mathcal{L}$ be a lattice. In general we will identify a basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ for $\mathcal{L}$ with the n×n matrix $\mathbf{B}$ whose rows are $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$. We say that a basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ is orthogonal if $\mathbf{b}_i \cdot \mathbf{b}_j = 0$ for all $i \neq j$.

A matrix $\mathbf{M}$ is called *unimodular* if it is a square matrix with integer entries and has determinant equal to ± 1.

Claim Given a basis $\mathbf{B}$ and a unimodular matrix $\mathbf{M}$, (the rows of) $\mathbf{M}\cdot\mathbf{B}$ will also be a basis for the same lattice, and every basis for that lattice can be obtained in this way.

Notice that for unimodular $\mathbf{M}$, the determinant of $\mathbf{M}\cdot\mathbf{B}$ is equal to the determinant of $\mathbf{B}$ in absolute value. Hence the determinant of the basis is invariant for any lattice to within a change of sign. For an n-dimensional lattice $\mathcal{L}$, the absolute value of the determinant of the basis is equal to the n-dimensional volume of the parallelohedron defined by the basis' vectors.

Figure 3 shows the lattice generated by vectors $\mathbf{b}_1$ and $\mathbf{b}_2$. The node x is an element in this lattice which has coordinates 3 and 2 with respect to this basis.

In Figure 4 the same lattice is represented with a different basis. The node x has coordinates 1 and 1 with respect to this new basis. The parallelograms in each figure have equal areas. A

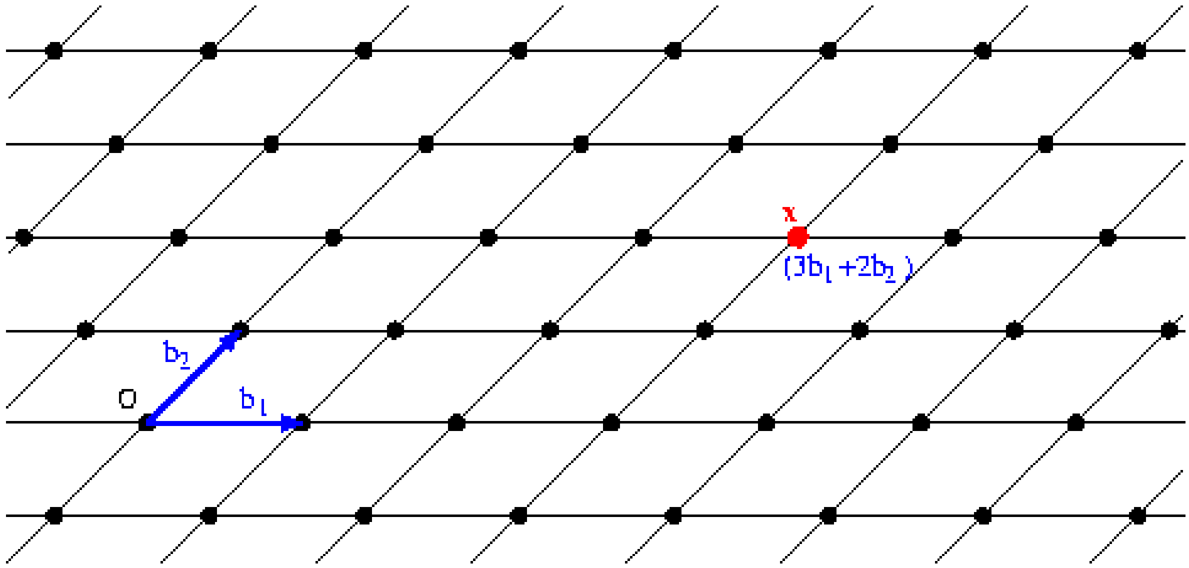

Figure 3: A lattice in 2 dimensions

lattice can be generated by more than one basis. It would be nice if we could obtain an orthogonal basis since this would contain the shortest vector of the lattice. However, in general lattices do not admit an orthogonal basis.

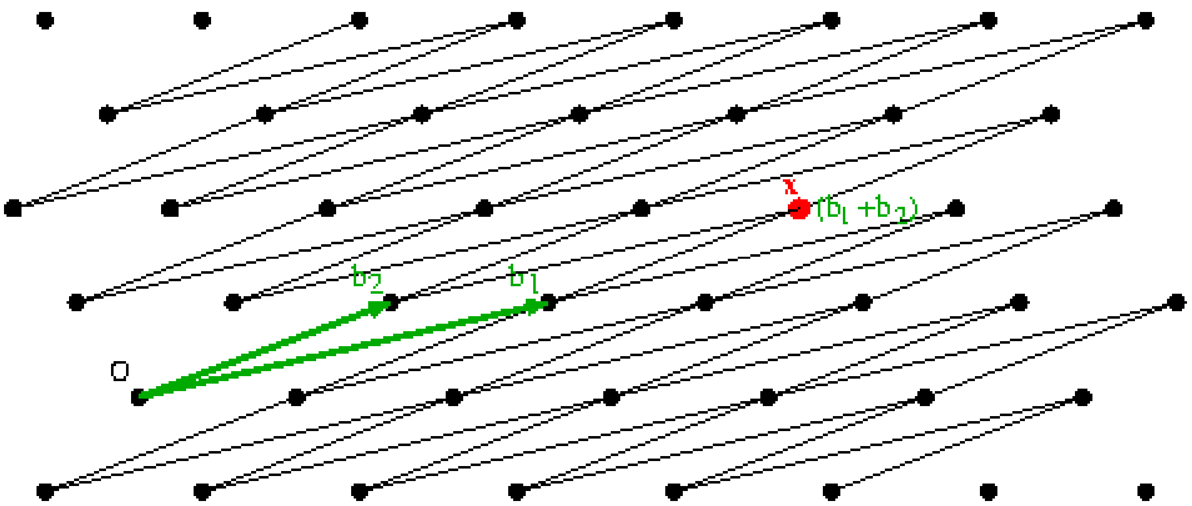

Figure 4: The same lattice with a different basis

When the basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ is orthogonal $|\det(\mathbf{B})| = ||\mathbf{b}_1|| \cdot ||\mathbf{b}_2|| \cdot \ldots \cdot ||\mathbf{b}_n||$. In general if the basis is not orthogonal $|\det(\mathbf{B})| \leq ||\mathbf{b}_1|| \cdot ||\mathbf{b}_2|| \cdot \ldots \cdot ||\mathbf{b}_n||$, and we can define the orthogonality defect of $\mathbf{B}$ as $\frac{||\mathbf{b}_1|| \cdot ||\mathbf{b}_2|| \cdot \ldots \cdot ||\mathbf{b}_n||}{|det(\mathbf{B})|}$.

Therefore we can state our goal in finding a "good" approximation for the **SVP**: given a basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ find an alternative basis that is reasonably orthogonal and choose the smallest vector of the new basis as an approximation.

The Gram-Schmidt Orthogonalization (GSO) method takes a basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ and creates rational vectors $\mathbf{b}_1^*, \mathbf{b}_2^*, \ldots, \mathbf{b}_n^*$ which are an orthogonal basis (using real coefficents) for n-dimensional space. Intuitively, $\mathbf{b}_i^*$ is the component of $\mathbf{b}_i$ orthogonal to the space spanned by $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{i-1}$. Thus $\mathbf{b}_1^*, \mathbf{b}_2^*, \ldots, \mathbf{b}_i^*$ spans (using real coefficients) the same space as $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_i$. The vectors are obtained as follows:

$$\mathbf{b}_1^* = \mathbf{b}_1 \tag{2}$$

and for all $2 \leq i \leq n$:

$$\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \frac{\mathbf{b}_i \cdot \mathbf{b}_j^*}{||\mathbf{b}_j^*||^2} \cdot \mathbf{b}_j^* \tag{3}$$

For $1 \leq j < i \leq n$, we define

$$u_{ii} = 1 \tag{4}$$

and

$$u_{ij} = \frac{\mathbf{b}_i \cdot \mathbf{b}_j^*}{||\mathbf{b}_j^*||^2} \tag{5}$$

Then for $1 \le i \le n$, we have

$$\mathbf{b}_i = \sum_{j=1}^{i} u_{ij}\mathbf{b}_j^* \tag{6}$$

Note that $||\mathbf{b}_i^*|| \le ||\mathbf{b}_i||$, and that subtracting multiples of rows from other rows preserves the determinant. Hence

$$|det(\mathbf{B})| = |det(\mathbf{B}^*)| = ||\mathbf{b}_1^*|| \cdot ||\mathbf{b}_2^*|| \cdot \ldots \cdot ||\mathbf{b}_n^*|| \le ||\mathbf{b}_1|| \cdot ||\mathbf{b}_2|| \cdot \ldots \cdot ||\mathbf{b}_n|| \tag{7}$$

Theorem 1 *Let OPT be the length of a shortest vector in $\mathcal{L}(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n)$. Let $\mathbf{b}_1^*, \mathbf{b}_2^*, \ldots, \mathbf{b}_n^*$ be the Gram-Schmidt Orthogonalization. Then $OPT \ge \min\{||\mathbf{b}_1^*||, ||\mathbf{b}_2^*||, \ldots, ||\mathbf{b}_n^*||\}$.*

Proof: Let v be a shortest vector in $\mathcal{L}$, and let k be the index of the largest basis vector used in generating v:

$$v = \sum_{i=1}^{k} \lambda_i \mathbf{b}_i, \quad \lambda_i \in \mathbb{Z}, 1 \le i \le k, \lambda_k \ne 0, \text{ so } |\lambda_k| \ge 1 \tag{8}$$

We can express v in the orthogonal basis $\mathbf{b}_1^*, \mathbf{b}_2^*, \ldots, \mathbf{b}_k^*$ by using the equation (6). In this basis the coefficient of $\mathbf{b}_k^*$ is just λ_k because $u_{kk} = 1$ by (4). So,

$$||v|| \ge ||\lambda_k \mathbf{b}_k^*|| \ge ||\mathbf{b}_k^*|| \ge \min\{||\mathbf{b}_1^*||, ||\mathbf{b}_2^*||, \ldots, ||\mathbf{b}_n^*||\} \tag{9}$$

$\square$

For $j \le i$ we denote by $\mathbf{b}_i(j)$ the component of $\mathbf{b}_i$ orthogonal to $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{j-1}$. So we have

$$\mathbf{b}_i(i) = \mathbf{b}_i^* \tag{10}$$

$$\mathbf{b}_{i+1}(i) = u_{i+1,i} \cdot \mathbf{b}_i^* + \mathbf{b}_{i+1}^* \tag{11}$$

or more generally:

$$\mathbf{b}_i(j) = u_{ij}\mathbf{b}_j^* + u_{i,j+1}\mathbf{b}_{j+1}^* + \ldots + \mathbf{b}_i^* \tag{12}$$

Definition We say that $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ is *Gauss reduced* if

1. $|u_{i+1,i}| \le \frac{1}{2}$ for all i
2. $||\mathbf{b}_i(i)|| \le \frac{2}{\sqrt{3}}||\mathbf{b}_{i+1}(i)||$

In Figure 5 we represent a simplified version of the L^3 algorithm as in the text.

Note that the operations in a) do not change the Gram-Schmidt Orthogonalization, and the change to $\mathbf{b}_{i+1}$ does not change any other $u_{j+1,j}$, for $j \ne i$. To show that step a) makes $|u_{i+1,i}| \le 1/2$ we make the following claim:

input basis $b_1, b_2, \ldots, b_n$
let $u_{j,k}$ be functions of $b_1, b_2, \ldots, b_n$ as in (4) and (5)
  while basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ is not Gauss reduced, **do**
    a) **for all** i, $1 \le i \le n$ {the order doesn't matter here}
      if $|u_{i+1,i}| > \frac{1}{2}$, **then** $\mathbf{b}_{i+1} \leftarrow \mathbf{b}_{i+1} - m\mathbf{b}_i$ {where m is the closest integer to $u_{i+1,i}$}
    b) pick an i (if there is one) such that
      $||\mathbf{b}_i(i)|| > \frac{2}{\sqrt{3}}||\mathbf{b}_{i+1}(i)||$ and interchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$
output b_1

Figure 5: The L^3 Algorithm

Claim If I replace $\mathbf{b}_{i+1}$ by $\mathbf{b}_{i+1} - \mathbf{b}_i$, then $u_{i+1,i}$ decreases by 1 (and in general, if I replace $\mathbf{b}_{i+1}$ by $\mathbf{b}_{i+1} - m\mathbf{b}_i$, then $u_{i+1,i}$ decreases by m).

Proof: $u_{i+1,i} = \frac{\mathbf{b}_{i+1} \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2}$. If $\mathbf{b}_{i+1}$ is decreased by $m\mathbf{b}_i$ to become b'_{i+1}, then

$$u'_{i+1,i} = \frac{(\mathbf{b}_{i+1} - m\mathbf{b}_i) \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} = \frac{\mathbf{b}_{i+1} \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} - m \cdot \frac{\mathbf{b}_i \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} = u_{i+1,i} - m \cdot \frac{\mathbf{b}_i \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} \tag{13}$$

Using (4), (6) and the orthogonality of $\mathbf{b}_1^*, \mathbf{b}_2^*, \ldots, \mathbf{b}_n^*$

$$\frac{\mathbf{b}_i \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} = \sum_{j=1}^{i} \frac{u_{ij}\mathbf{b}_j^* \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} = u_{ii}\frac{\mathbf{b}_i^* \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} + \sum_{j=1}^{i-1} \frac{u_{ij}\mathbf{b}_j^* \cdot \mathbf{b}_i^*}{||\mathbf{b}_i^*||^2} = 1. \tag{14}$$

Thus $u_{i+1,i}$ decreases by m. □

Theorem 2 *The algorithm from Figure 5 achieves an approximation guarantee of* $2^{(n-1)/2}$ *if it halts.*

Proof: Assume that L^3 halts. Therefore the new basis $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ is Gauss-reduced. Then

$$||\mathbf{b}_i^*||^2 = ||\mathbf{b}_i(i)||^2 \le \frac{4}{3}||\mathbf{b}_{i+1}(i)||^2 = \frac{4}{3}(u_{i+1,i}^2||\mathbf{b}_i^*||^2 + ||\mathbf{b}_{i+1}^*||^2) \le \frac{||\mathbf{b}_i^*||^2}{3} + \frac{4}{3}||\mathbf{b}_{i+1}^*||^2 \tag{15}$$

Hence

$$||\mathbf{b}_i^*||^2 \le 2||\mathbf{b}_{i+1}^*||^2 \tag{16}$$

and so

$$||\mathbf{b}_1|| = ||\mathbf{b}_1^*|| \le 2^{(n-1)/2} \cdot \min_{1 \le i \le n}\{||\mathbf{b}_i^*||\} \le 2^{(n-1)/2} \cdot OPT. \tag{17}$$

□

References

[1] M. Ajtai. The Shortest Vector In L_2 is **NP**-hard for randomized reductions. In *Proc. 30th ACM Symposium on the Theory of Computing*, pages 10-19, 1998

[2] D. Micciancio. The Shortest Vector in a Lattice is Hard to Approximate to Within Some Constant. *In Proc. 39th IEEE Annual Symposium on Foundations of Computer Science*, 92-98, 1998

[3] A. K. Lenstra, H. W. Lenstra, Jr., L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Ann.*, 261:513-534, 1982

[4] C. P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical Computer Science*, 73:201-224, 1987

[5] O. Goldreich and S. Goldwasser. On the limits of non-approximability of lattice problems. *In Proc. of 30th STOC. ACM*, 1998

Acknowledgement

Figures 3 and 4 were taken from http://www.cs.ucsd.edu/~daniele/lattice/lattice.html