Lecture 10

*Lecturer: Madhu Sudan* *Scribe: Pritish Kamath*

# 1 Introduction

In last class we saw Hensel Lifting and how to factorize bivariate polynomials over finite fields. In this lecture we will see how to factor univariate polynomials over $\mathbb{Q}$. Apart from the technique of Hensel lifting, another important routine in this algorithm will be the Lenstra-Lenstra-Lovasz (LLL) algorithm for obtaining an approximation for the Shortest Vector problem in lattices. A preliminary version of this algorithm was given by Gauss, which albeit works only for 2 dimensions.

# 2 Factorizing in $\mathbb{Q}[x]$

Suppose we want to factorize $f \in \mathbb{Z}[X]$ which has degree $n$ and $|\text{coeffs}(f)| \leq 2^{O(n)}$ (where we define $|\text{coeffs}(f)|$ to be the sum of absolute values of the coefficients of $f$). We can assume without loss of generality that $f$ is square-free[1]. Suppose $f = A.B$ where $A$ is irreducible. But first, we need to know that the factors of $f$ have small coefficients, otherwise we will not be able to even represent them efficiently. To this end, we have the following lemma:

**Lemma 1.** *All factors $f_i$ of $f$ have $|\text{coeffs}(f_i)| \leq 2^{\text{poly}(n)}$, where $\deg(f) \leq n$ and $|\text{coeffs}(f)| \leq 2^{O(n)}$*

**Proof** The main idea is that all complex roots of $f$ have magnitude $\leq 2^{\text{poly}(n)}$. This is because the leading term of $f$ will dominate all the other terms if $|x| > 2^{\Omega(n)}$, and thus $f$ cannot have roots outside a certain radius around 0. Thus, writing $g$ as $\prod_\alpha (x - \alpha)$ we get that $|\text{coeffs}(g)| \leq 2^{\text{poly}(n)}$. □

We take an approach similar to what we did for bivariate factorization.

(a) We find a "nice" prime $p$, and polynomials $g$ and $h$ such that $f = g.h (\text{mod } p)$ where $g$ is irreducible, monic, rel. prime to $h$ with $\deg_x(g), \deg_x(h) \geq 1$.

(b) We lift $g$ and $h$ to get $f = g_t h_t (\text{mod } p^t)$ where $g_t = g(\text{mod } p)$ and $h_t = h(\text{mod } p)$.

(c) Find $\widetilde{A}$ s.t. $1 \leq \deg(\widetilde{A}) < \deg(f)$ of minimum degree s.t. $\exists \widetilde{h}$ s.t. $\widetilde{A} = g_t.\widetilde{h}(\text{mod } p^t)$ and $|\text{coeffs}(\widetilde{A})| < M = 2^{\text{poly}(n)}$

[1] otherwise $\gcd(f, f')$ would have been a non-trivial factor of $f$ already

(d) $\gcd(\widetilde{A}, f)$ gives a non-trivial factor of $f$.

Steps (a) and (b) are very natural, following bivariate factorization over finite fields. We now justify step (d). We know that $A$ is such that $|\text{coeffs}(A)| \leq M_1 = M$ (from Claim 1) and $\widetilde{A}$ is such that $|\text{coeffs}(\widetilde{A})| \leq M$ and $\deg(A), \deg(\widetilde{A}) < n$. From Hensel lifting we know that there exist $h_1$ and $h_2$ such that $\alpha A = g_t h_1 (\text{mod } p^t)$ and $\beta \widetilde{A} = g_t h_2 (\text{mod } p^t)$. And hence $\alpha A + \beta \widetilde{A} = g_t(\alpha h_1 + \beta h_2)(\text{mod } p^t)$.

Suppose for contradiction that $A \nmid \widetilde{A}$. Then $R = \text{Res}(A, \widetilde{A}) \in \mathbb{Z}$ (see Lecture 8. Also follows easily from Bezout's theorem). We have that $R < n! M^{2n} \ll p^t$ (we choose $p$ and $t$ large enough for this to hold). But then $R = g_t \widetilde{h} (\text{mod } p^t)$ for some $\widetilde{h}$. This is a contradiction because $g_t \widetilde{h}$ is a polynomial with non-zero degree and leading coefficient less than $p^t$, but $R \in \mathbb{Z}$. Hence $A | \widetilde{A}$.

Thus, once we find the $\widetilde{A}$ as in Step (iii), we can get $A = \gcd(\widetilde{A}, f)$ which will be a non-trivial factor of $f$.

# 3 Shortest Vector Problem: realizing Step (c)

**Problem 1.** *Given $f, g, M, p, t$, find $\widetilde{A}$ as in Step (c) of approach.*

We want to find $\widetilde{A}$ such that $\widetilde{A} = g.\widetilde{h} (\text{mod } p^t)$. We think of polynomials in $\mathbb{Z}^{<k}[x]$ as vectors in $\mathbb{Z}^k$. This way, the above condition can be written as,

$$\widetilde{A} = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ \vdots \\ \vdots \\ c_{k-1} \end{bmatrix} = \begin{bmatrix} g_0 & & 0 & p^t & 0 & \cdots & \cdots & \cdots & 0 \\ g_1 & \ddots & & 0 & p^t & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & g_0 & 0 & & \ddots & & & \vdots \\ g_\ell & & g_1 & \vdots & & & \ddots & & \vdots \\ & \ddots & & \vdots & & & & \ddots & 0 \\ & & g_\ell & 0 & 0 & \cdots & \cdots & 0 & p^t \end{bmatrix} \begin{bmatrix} \\ \widetilde{h} \\ \\ - \\ \\ e \\ \\ \end{bmatrix}$$

where $\widetilde{A} = \sum_{i=0}^{k-1} c_i x^i$. The set of attainable $(c_0, \cdots, c_{k-1})$ is a subset of $\mathbb{Z}^k$ which is closed under addition.

**Definition 2** (Lattice). *A subset $L \subseteq \mathbb{R}^k$ which is* discrete *and* additive *is a lattice, where,*
discrete: $\forall x \in L$, $\exists \delta > 0$ *such that* $B_\delta(x) \cap L = \{x\}$
additive: $\forall x, y \in L$, $x - y \in L$

**Problem 2** (Shortest vector problem). *Given a basis $v_1, \cdots, v_k \in \mathbb{Z}^k$. Find $\alpha_1, \cdots, \alpha_k$ that minimizes $\|\sum_{i=1}^{k} \alpha_i v_i\|_2$*

## 3.1 Known results about SVP

Ajtai showed that SVP is NP-hard under randomized reductions [1]. But we only need to approximate SVP here. That is, find $\alpha_1, \cdots, \alpha_k$ such that $\|\sum_{i=1}^{k} \alpha_i v_i\|_2 \leq \gamma(k).\beta$ where the minimum of $\|\sum_{i=1}^{k} \alpha_i v_i\|_2$ is $\beta$.

In that sense, Ajtai only showed that $\gamma = 1$ is NP-hard. Daniele Micciancio (grad student at MIT then) was given Ajtai's paper to read and do something about it. He showed achieving $\gamma = \sqrt{2}$ is also NP-hard under randomized reductions [2]. Further work in this area has shown that $\gamma(k) = 2^{\log^{1-\delta}(k)}$ is also 'hard'. Modern Cryptography relies on the hardness of $\gamma(k) = k^{10}$ or so.

However for our purposes, it suffices to have a $\gamma$-approximation, where $\gamma = 2^k$. The Lenstra-Lenstra-Lovasz algorithm gives such an approximation in polynomial time [3].

# 4 Gauss's algorithm for 2-dim

In this lecture, we will only study Gauss' algorithm which works in the two dimensional case, although the LLL algorithm can be thought of as a generalization of Gauss' algorithm.

**Problem 3.** *Given vectors $v_1, v_2 \in \mathbb{Z}^2$, find $\alpha_1, \alpha_2$ minimizing* $\|\alpha_1 v_1 + \alpha_2 v_2\|_2$

The algorithm is similar in flavor to Euclid's GCD algorithm. We start with two vectors $s$ and $b$, where $s$ is smaller than $b$. We repeatedly take $(s, b)$ to $(s, b' = b - s)$. The algorithm is as follows,

**Repeat:**

- Set $i = \text{argmin}_j(\|b - js\|_2)$
- Set $b = b - is$
- If vertical part of $b$ has length $\leq s/2$ then swap $(s, b)$.
  Else stop and output $min(\|b\|_2, \|s\|_2)$.

# References


[1] Miklos Ajtai. The Shortest Vector Problem in L2 is NP-hard for Randomized Reductions *STOC*, 1998.

[2] Daniele Micciancio. The Shortest Vector in a Lattice is Hard to Approximate to within Some Constant. *SIAM Journal of Computing*, 2001

[3] Lenstra, A. K.; Lenstra, H. W., Jr.; Lovsz, L. Factoring Polynomials with Rational Coefcients *Mathematische Annalen*, 1982