Topics: Integer Programming in Bounded Dimension
Book: Chapter 9.1

Summary

In this lecture, we discussed Lenstra's polynomial time algorithm for integer linear programming in bounded dimension. Using binary search, it suffices to solve the integer feasibility problem: given a rational system $Ax \le b$, find an integral feasible solution or (correctly) conclude is has no integral solution.

The rough idea is the following. Either it is easy to find an integral point in $K = \{x \in \mathbb{R}^n : Ax \le b\}$, or we can find a direction in which K is very flat. In the latter case, $K \cap \mathbb{Z}^n$ is contained in a bounded number of affine hyperplanes and the ILP can be reduced to a bounded number of ILP's with only $n-1$ variables.

Lattice width

Let $K \subseteq \mathbb{R}^n$ be a convex body. Given $d \in \mathbb{R}^n \setminus \{0\}$, the *width of K along direction d* is given by

$$w_d(K) := \max\{\langle d, x\rangle : x \in K\} - \min\{\langle d, x\rangle : x \in K\}.$$

This is equal to $\|d\|$ times the Euclidean width of K in the direction d. The *lattice width* is defined by

$$w(K) := \min\{w_d(K) : d \in \mathbb{Z}^n \setminus \{0\}\}.$$

We leave it as an exercise to show that the minimum is indeed attained. Note that we can restrict to d for which $\gcd(d_1, d_2, \ldots, d_n) = 1$. Denoting $H_c := \{x \in \mathbb{R}^n : \langle d, x\rangle = c\}$ for $c \in \mathbb{Z}$, the integral points in K are contained in at most $\lfloor w_d(K)\rfloor + 1$ hyperplanes H_c. The integral points in H_0 form a sublattice of rank $n-1$.

Lemma 1. *Let $d \in \mathbb{Z}^n$ with $\gcd(d_1, \ldots, d_n) = 1$. Then we can find in polynomial time a basis $u^1, \ldots, u^n$ of $\mathbb{Z}^n$ such that $u^1, \ldots, u^{n-1}$ form a basis of the lattice $H_0 = \{x \in \mathbb{Z}^n : \langle d, x\rangle = 0\}$ and $\langle d, u^n\rangle = 1$.*

Conversely, given a basis $u^1, \ldots, u^n$ of $\mathbb{Z}^n$, we can find in polynomial time a $d \in \mathbb{Z}^n$ such that $\langle d, u^k\rangle = 0$ for $k = 1, \ldots, n-1$ and $\langle d, u^n\rangle = 1$.

The first part is Corollary 1.9. The second part follows by solving $Ud = e^n$ where U is the unimodular matrix whose rows are $u^1, \ldots, u^n$ and e^n is the n-th standard basis vector.

Flatness theorem for ellipsoids

An ellipsoid is a set $E(C, a) := \{x \in \mathbb{R}^n : \|C(x-a)\| \le 1\}$ where C is a nonsingular matrix[1] and $a \in \mathbb{R}^n$. In other words: $E(C, a) = a + C^{-1}(B(0,1))$ is an affine image of the unit ball $B(0,1) := \{y \in \mathbb{R}^n : \|y\| \le 1\}$.

Theorem 1 (Thm 9.8)**.** *Let $E = E(C, a)$ be an ellipsoid that does not contain an integral point. Then $w(E) \le n2^{n(n-1)/4}$.*

Proof. Let Λ be the lattice with basis C and let $a' = C(a)$. Then $C(E) = a' + B(0,1)$ contains no points in Λ. Consider a reduced basis B of Λ. After reordering the basis, we may assume that $\|b_n\| \ge \|b_i\|$ for $i = 1, \ldots, n$.

Write a' with respect to the basis B. That is, $a' = \lambda_1 b^1 + \cdots + \lambda_n b^n$. The point $a'' := \lfloor \lambda_1 \rceil b^1 + \cdots + \lfloor \lambda_n \rceil b^n$ is an element of Λ. Hence, $a'' \notin a' + B(0,1)$ and we have

$$1 < \|a'' - a'\| \le \tfrac{1}{2}(\|b^1\| + \cdots + \|b^n\|) \le \tfrac{n}{2}\|b^n\|. \tag{1}$$

Now compute the Gram-Schmidt basis G for the basis B. Recall that $\|b^1\| \cdots \|b^n\| \le 2^{n(n-1)/4} \det B = \|g^1\| \cdots \|g^n\|$. The inequality follows from Theorem 9.2. Since $\|g_i\| \le \|b_i\|$ holds for every i, we get

$$\|g^n\| \ge 2^{-n(n-1)/4}\|b^n\| > \tfrac{2}{n2^{n(n-1)/4}}. \tag{2}$$

[1] The matrix C can be taken symmetric since $C^\mathsf{T} C = D^\mathsf{T} D$ for some symmetric D.

Let $g := \|g^n\|^{-2} \cdot g^n$ be a scaled version of g^n such that

$$\begin{aligned} \langle g, b^n \rangle &= 1, \\ \langle g, b^i \rangle &= 0 \qquad (i = 1, \ldots, n-1). \end{aligned}$$

Note that $\|g\| = \|g^n\|^{-1} \leq \frac{n2^{n(n-1)/4}}{2}$. Since the unit ball has diameter 2, it follows that for all $y \in a' + B(0,1)$ we have

$$\langle g, y \rangle \in [\alpha, \beta],$$

where α, β are such that $\beta - \alpha = 2\|g\| \leq n2^{n(n-1)/4}$. Let $d = C^{\mathsf{T}} g$. Note that $\langle d, x \rangle = \langle g, Cx \rangle$ for every $x \in \mathbb{R}^n$. Since g has integral inner product with all $y \in \Lambda = C(\mathbb{Z}^n)$, it follows that d has integral inner product with all elements in $\mathbb{Z}^n$. Hence d is itself integral and $w(E) \leq w_d(E) \leq n2^{n(n-1)/4}$. □

Khinchine's Flatness theorem

Given a full-dimensional convex body $K \subseteq \mathbb{R}^n$, the unique ellipsoid $E = E(C, a)$ of minimum volume containing K is called the Löwner-John ellipsoid. It has the nice property that $E(nC, a)$ is contained in K. When K is a polytope P, the Löwner-John ellipsoid can be computed (up to arbitrary precision) in polynomial time using convex optimisation. Hence, we can in polynomial time find an ellipsoid $E = E(C, a)$ with C and a rational and $E((n+1)C, a) \subseteq P \subseteq E(C, a)$. Using the flattness theorem for ellipsoids, we obtain the following corollary.

Theorem 2 (Thm 9.7)**.** *Let $K \subseteq \mathbb{R}^n$ be a full-dimensional convex body. If K does not contain an integral point, then $w(K) \leq n^2 2^{n(n-1)/4}$.*

Integer programming algorithm

We want to solve the integer feasibility problem for $P = \{x : Ax \leq b\}$. By Lemma 4.35 we may assume that P is bounded. First, we can determine if P is full-dimensional by solving $\min\{a^i x \leq b_i\}$ for $i = 1, \ldots, m$. Suppose that P is contained in a hyperplane $d \cdot x = \beta$, where d is integral. We may assume that $\gcd(d_1, \ldots, d_n) = 1$. If β is not integral, the P has no integral points. If β is integral, then by Lemma 1 we can reduce the integer program by one in $n-1$ variables.

Now suppose that P is full-dimensional. By the previous section, we can either find an integral point in an ellipsoid contained in P, or we can solve the integer program by solving at most $n(n+1)2^{(n-1)n/4}$ integer programs in $n-1$ variables (by Lemma 1).

Using binary search, the integer optimisation problem can be solved in polynomial time as well by reduction to polynomially many integer feasibility problems.