Convexity

Prof. Friedrich Eisenbrand
Christoph Hunkenschröder

Assignment Sheet 7 - Solutions
November 17, 2016

Exercise 1
A polytope $P \subseteq \mathbb{R}^n$ is called *integral* if $P = \text{conv}(P \cap \mathbb{Z}^n)$ holds. Show that a polytope is integral if and only if all of its vertices are integer points.

Solution:
As a polytope is known to be the convex hull of its vertices, it certainly is integral if all of its vertices are integral.

For the other direction, assume P is an integral polytope and let $v \in P$ be a vertex. As a vertex cannot be written as the convex combination of other points in P (by the characterization of vertices on a previous exercise sheet) and it is contained in $\text{conv}(P \cap \mathbb{Z}^n)$ it has to be integral itself.

Exercise 2
Let $\Lambda \subseteq \mathbb{R}^n$ be a lattice, $\mathcal{V}$ its voronoi cell. For $p \in \Lambda$, denote the voronoi cell translated by p with $\mathcal{V}(p)$. Let $p \in \Lambda$ such that $\mathcal{V}(p) \cap (2\mathcal{V}) \neq \emptyset$.

1. Show that the line segment $[0, p]$ is contained in $2\mathcal{V} \cup \mathcal{V}(p)$.
2. Using the previous fact, argue that $\mathcal{V}(p) \subseteq (4\mathcal{V})$.
3. Show that at most 4^n of the voronoi cells $\mathcal{V}(p)$, $p \in \Lambda$, have a non-empty intersection with $(2\mathcal{V})^\circ$.

Solution:

1. Assume $\mathcal{V}(p)$ is a translate of $\mathcal{V}$ intersecting $2\mathcal{V}$. Let $z \in [0, p]$ the unique point on the boundary of $2\mathcal{V}$ and $z' \in [0, p]$ be the unique point on the boundary of $\mathcal{V}(p)$. As $[0, p]$ is a line segment and the considered sets are convex, z and z' are well-defined. If $z \in \mathcal{V}(p)$, we are done already, so assume not. But then we can find a supporting hyperplane $a^T x = \beta$ of $2\mathcal{V}$ in z, and by central symmetry, the hyperplane $-a^T x = -\beta/2 - a^T p$ is a supporting hyperplane for $\mathcal{V}(p)$ in z'. But as the supporting hyperplanes are parallel, they either coincide or prove that $\mathcal{V}(p)$ and $2\mathcal{V}$ are strictly separable, which was a contradiction to our assumption of a non-empty intersection.

   Remark: For polytopes, the supporting hyperplane we chose is given by the facet-defining inequality. However, for general convex bodies you have to show that you are always able to find such a hyperplane. Alternatively, you can work with a separating hyperplane of $2\mathcal{V}$ and $\frac{z+z'}{2}$ and use the same argumentation.

2. First, by shifting $\mathcal{V}(p)$ further in direction p and using the previous part, we may assume that they touch only in the boundary and $z' = z$ (however, p might not be a lattice point any more, but is still the center). With the notation as before, we prolong $[0, p]$ to $[0, p']$ where p' is the point where we leave $\mathcal{V}(p)$ (the point opposite of z'. By central symmetry, $\|p - z'\| = \|p' - p\| = 1/2\,\|z\|$, Hence, we see that the whole line segment is contained in $4\mathcal{V}$. Using $aK + bK = (a + b)K$ for convex bodies (a generalization of a previous exercise), we see that $\mathcal{V}(p)$ is contained in $4\mathcal{V}$.

3. By the aforementioned, all voronoi cells intersecting $(2\mathcal{V})^\circ$ are contained in $4\mathcal{V}$. Moreover, their interiors are disjoint. Hence, writing N for the number of voronoi cells intersecting $2\mathcal{V}$, we find

$$N \cdot \operatorname{vol} \mathcal{V} \leq \operatorname{vol}(4\mathcal{V}) = 4^n \operatorname{vol} \mathcal{V},$$

implying the desired bound.

Exercise 3
Let $\Lambda \subseteq \mathbb{R}^n$ be a lattice, and $\mathcal{V}$ its voronoi cell. For $p \in \Lambda$, denote the voronoi cell translated by p with $\mathcal{V}(p)$. Assume that Λ is of such form that the following holds.

$(\star)$ Any point $p \in \Lambda$ for which $(2\mathcal{V})^\circ \cap \mathcal{V}(p)^\circ \neq \emptyset$ lies on the boundary of $2\mathcal{V}$.

Show that we have

$$|\{p \in \Lambda : \ (2\mathcal{V})^\circ \cap \mathcal{V}(p)^\circ \neq \emptyset\}| \leq 3^n.$$

Show that the property $(\star)$ is not true in general.

Solution:
Here, the same argumentation as in the exercise before works just fine, the additional property $(\star)$ allows us to replace the bound of 4 by 3.

Let $\mathcal{V}(p)$ be a voronoi cell intersecting $(2\mathcal{V})^\circ$. By property $(\star)$, we know that p is on the boundary of $2\mathcal{V}$. Hence, any $x \in \mathcal{V}(p)$ can be written as $x = p + y$ with $y \in \mathcal{V}$ and $p/2 \in \mathcal{V}$. Thus $x = 3(1/3y + 2/3\frac{p}{2}) \in 3\mathcal{V}$ by convexity. As the $\mathcal{V}(p)$ are voronoi cells, their intersections has measure 0, and we find

$$|2\mathcal{V} \cap \Lambda| \cdot \operatorname{vol} \mathcal{V} \leq \operatorname{vol}(3\mathcal{V}),$$

implying the desired bound.

Exercise 4
Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice, $\mathcal{V}$ its voronoi cell, given in the form $\{x : Ax \leq b\}$, and $t \in \mathbb{R}^n$ be a target vector.

1. Using Exercise 2, show that if $t \in 2\mathcal{V}$ and the description $Ax \leq b$ of the voronoi cell is given, then a closest vector can be found in time $2^{O(n)}$ and state the algorithm.

2. State an algorithm with the following specifications. The input is a basis B of Λ and a target vector t, the output is a closest vector to t in Λ. Moreover, the algorithm has access to the description $Ax \leq b$ of the voronoi cell of Λ and has running time $\log(\|t\|+1)2^{O(n)}$, assuming that each basic arithmetic operation can be performed in constant time.

Prove correctness for your algorithms.
[*Hint: What is the difference between the first and the second part? Which running time would you (roughly) get if you used the technique of the first part directly on the second part? Which techniques do you know to reduce the appearing factor? For example, how can you implement* $f(x,n) = x^n$ *only using* $O(\log(n+1))$ *arithmetic operations, for n integral?*]

Solution:

1. We will walk along the line segment $[0,t]$, starting at 0, keeping track of the voronoi cell we are in. When we reach t, this will give us the closest vector.

   CVP-simple (Λ, $\mathcal{V}$, t)

   (a) Set $p = 0$. This variable will store the center of the voronoi cell we are in.

   (b) If $t \in \mathcal{V}(p)$, then return p.

   (c) Determine the point z where $[0,t]$ leaves the voronoi cell $\mathcal{V}(p)$. Let $\mathcal{V}(q)$, $q \in \Lambda$, be the voronoi cell the line segment $[0,t]$ enters at z.

   (d) Update $p \leftarrow q$ and goto (b).

   If we can implement the proceeding above and the algorithm terminates, the output will be correct. Let us see how to implement it. As the voronoi cell $\mathcal{V}$ is given by $Ax \leq b$, the voronoi cell $\mathcal{V}(p)$ for any $p \in \Lambda$ is given by $A(x-p) \leq b$. Hence, $b)$ can be done by checking all inequalities. If we entered a voronoi cell $\mathcal{V}(p)$, this means that there is a λ s.t. $A(\lambda t - p) = \lambda A(t-p) \leq b$. As λ is a scalar, for any inequality $a_i x \leq \beta_i$, we can find a λ_i s.t. $a_i(\lambda_i t - p) = \beta_i$[1]. This equality reads like *At point* $\lambda_i t$*, we will pass the supporting hyperplane defining the facet corresponding to the i-th inequality*. If $\lambda_i \leq \lambda$, this means that we already passed the facet, if $\lambda_i \geq \lambda$, we will pass the facet. Hence, we simply have to choose the smallest positive λ_i, as this is the facet we will pass first, hence where we leave the voronoi cell. But as the inequalities are actually $a_i x \leq \frac{\|a_i\|^2}{2}$ with a lattice vector a_i, the cell we enter is given by $\mathcal{V}(p+a_i)$. As we stay the whole time within $2\mathcal{V}$ by assumption, exercise 3 gives us that we visit at most $3^n = 2^{O(n)}$ voronoi cells, hence we terminate after at most that much iterations.

   Let us consider the running time. Step $a)$ is done in $O(n)$ (as we initialize a vector of dimension n). Step $b)$ is checking $2^{O(n)}$ by the results of the lecture, each checking can be done by $O(n)$ basic arithmetic operations, hence it takes time $2^{O(n)}$ in total. Step $c)$ is mainly solving $2^{O(n)}$ equalities, each with $O(n)$ basic operations. Finally, step $d)$ is again easy, so in total, one iteration has $2^{O(n)}$ basic arithmetic operations. As we have $2^{O(n)}$ iterations in total, we get the desired running time.

[1] Actually, there might be a degenerate case where we cannot find such a λ_i. But this means that we run parallel to the facet, so we will never violate its inequality as long as we start with a feasible point.

2. This will be a recursive algorithm.

   CVP-with-scaling $(\Lambda, \mathcal{V}, t)$

   (a) If $t \in \mathcal{V}$, then return 0.

   (b) Let $p' \leftarrow$ **CVP-with-scaling** $(2\Lambda, 2\mathcal{V}, t)$

   (c) Let $p'' \leftarrow$ **CVP-simple** $(\Lambda, \mathcal{V}, t - p')$

   (d) return $p' + p''$

As the algorithm is obviously correct, we only have to show that $p' + p''$ is indeed a closest vector to t in Λ. Correctness then follows recursively. By induction hypothesis, we have $t \in 2\mathcal{V}(p')$. We also have $(t - p') \in \mathcal{V}(p'')$, implying $t \in p' + \mathcal{V}(p'') = \mathcal{V}(p' + p'')$, hence the output is indeed a closest vector.

For the running time, note that step 1. can be checked in time $2^{O(n)}$ and consider the smallest κ s.t. $t \in 2^{\kappa}\mathcal{V}$. Then we have a recursion depth of $O(\kappa + 1)$, and $\kappa \in O(\log \|t\|)$, hence it is linear in the input size. In each iteration, we have one call to **CVP-simple**, but as $t - p'$ is contained in $2\mathcal{V}$, this subroutine works in $2^{O(n)}$ by part 1., yielding the desired running time.