# Equivalence of Separation and Optimization

## 1 Equivalence for Polytopes

We will show that the optimization of linear functions over a polytope reduces to implementing a separation oracle for the polytope, and vice versa; that is, the two tasks are polynomial time equivalent. Given a polytope $P$, we can define the linear optimization problem:

- Input: linear objective $\boldsymbol{c} \in \mathbb{R}^n$
- Output: $\arg\max_{\boldsymbol{x} \in P} \boldsymbol{c}^\intercal \boldsymbol{x}$

We also have the linear separation problem:

- Input: $\boldsymbol{y} \in \mathbb{R}^n$
- Output: Decide that $\boldsymbol{y} \in P$ or else find $\boldsymbol{h} \in \mathbb{R}^n$ such that $\boldsymbol{h}^\intercal \boldsymbol{x} < \boldsymbol{h}^\intercal \boldsymbol{y}$ for all $\boldsymbol{x} \in P$.

For example, in the MST LP, optimization corresponds to finding the MST, while separation corresponds to finding a dense subgraph which violates the LP constraints.

**Theorem 1.** *Consider a family $\mathcal{P}$ of polytopes of the form $P = \mathbf{A}\boldsymbol{x} \preceq \boldsymbol{b}$ described implicitly using $\langle P \rangle$ bits and satisfying $\langle a_{ij} \rangle, \langle b_i \rangle \leq poly(\langle P \rangle, n)$. Then, the separation problem is solvable in $poly(\langle P \rangle, n, \langle \boldsymbol{y} \rangle)$ if and only if the optimization problem is solvable in $poly(\langle P \rangle, n, \langle c \rangle)$. We say that such a $\mathcal{P}$ is solvable.*

*Proof.* We already saw that separation implies optimization by the Ellipsoid Algorithm. For the other direction, we will have to utilize polars. Recall that $S^\circ = \{\boldsymbol{y} : \boldsymbol{y}^\intercal \boldsymbol{x} \leq 1 \; \forall \boldsymbol{x} \in S\}$ for some convex convex $S \subseteq \mathbb{R}^n$ containing the origin. Recall also that $S^\circ$ is bounded and contains the origin and $S^{\circ\circ} = S$. Given a polytope represented as $\mathbf{A}\boldsymbol{x} \preceq \mathbf{1}$, its polar is the convex hull of the rows of $\mathbf{A}$.

**Lemma 2.** *Separation over $S$ reduces in constant time to optimization over $S^\circ$, and vice versa because $S^{\circ\circ} = S$.*

*Proof.* Given a vector $\boldsymbol{x}$, we must check $\boldsymbol{x} \in S$ and if it is not, output a separating hyperplane. We know $\boldsymbol{x} \in S \iff \boldsymbol{y}^\intercal \boldsymbol{x} \leq \mathbf{1} \; \forall \boldsymbol{y} \in S^\circ$ by definition. This is equivalent to $\boldsymbol{x} \in S \iff \sup_{\boldsymbol{y} \in S^\circ} \boldsymbol{y}^\intercal \boldsymbol{x} \leq 1$, but this is exactly saying that $\boldsymbol{x}$ is the objective of the optimization problem. Thus if we find $\boldsymbol{y} \in S^\circ$ such that $\boldsymbol{y}^\intercal \boldsymbol{x} > 1$, then $\boldsymbol{y}$ is a separating hyperplane because $\boldsymbol{y}^\intercal \boldsymbol{z} \leq 1 < \boldsymbol{y}^\intercal \boldsymbol{x} \; \forall \boldsymbol{z} \in S$. This is optimization over $S^\circ$, so we can solve it using the Ellipsoid Algorithm! □

Technical Note 1: We need to "center" the polytope about the origin, but we can do that by running the Ellipsoid Algorithm to find a strictly feasible point in $P$, then change our coordinate system to center that point.

Technical Note 2: To apply the Ellipsoid Algorithm to $P^\circ$, we need polynomial bit complexity of the facts of $P^\circ$, but this follows from the polynomial bit complexity of the vertices of $P$ (since facets of the polar correspond to vertices of the polytope). □

## 2 Equivalence beyond Polytopes

The equivalence of separation and optimization approximately extends to arbitrary convex sets as long as you can circumscribe the convex set. This allows you to solve weak separation and weak optimization problems, and allows an approximation $\epsilon$ in both feasibility and optimality. These problems are both solvable in polynomial time if and only if the other is, with terms for circumscription accuracy and $\log(1/\epsilon)$.

This implies that there must exist operations which preserve solvability. Given that you can efficiently optimize over convex sets $P$ and $Q$, it turns out that you can:

- Optimize over $P \cap Q$: You can separate over $P$ and $Q$ individually, therefore you can separate over $P \cap Q$ which by the equivalence of separation and optimization implies that you can optimize over $P \cap Q$. This has applications in colorful spanning tree, cardinality constrained matching, etc.
- Optimize over $P \cup Q$: Simply optimize over each set individually, then take the better result of the two. This is equivalent to optimizing over $convHull(P \cup Q)$, because taking the convex hull will not add any vertices. So by the equivalence of separation and optimization, there exists a polynomial time separation oracle for $convHull(P \cup Q)$. Note that it must be a constant or polynomially sized union, not an exponentially sized union like TSP.

Note that the sets $P$ and $Q$ are not explicit, just described by a separation oracle, so we don't even need to know what the sets actually are to optimize over their intersection or union! A final implication is that by the equivalence of separation and optimization, there exists a polynomial time algorithm for the Constructive Caratheodory problem: Given a point $\boldsymbol{x} \in P$, where $P \in \mathbb{R}^n$ is a solvable polytope, write $\boldsymbol{x}$ as a convex combination of $n + 1$ vertices of $P$ (proof on next homework).