11 Linear Programming

11.1 Definition and Importance

The final topic in this course is Linear Programming. We say that a problem is an instance of linear programming when it can be effectively expressed in the linear programming framework. A linear programming problem is an *optimization* problem where the optimization function is a linear function. As we have seen before an optimization problem is the following:

Definition 11.1 (Optimization Problem)**.** An optimization problem is a function $f : \Sigma \to \mathbb{R}$ along with a subset $F \subseteq \Sigma$. The goal of the problem is to find the $x \in F$ such that for all $y \in F$, $f(x) \leq f(y)$. We often call Γ the *feasible region* of the problem.

In the case of linear programming, we take $\Sigma = \mathbb{R}^n$, require f to be linear i.e. $f(x) = c^T x$ for some $c \in \mathbb{R}^n$, and the feasible region F is a convex polytope sitting in n-dimensional space. Another way of thinking of a convex polytope is that it is the intersection of finitely many half-planes or as the set of points that satisfy a finite number of affine inequalites.

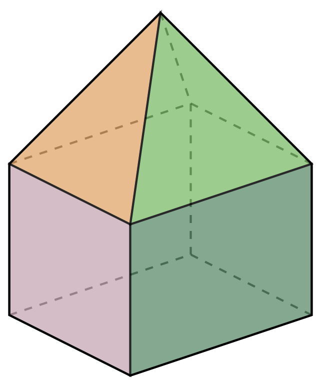

Figure 11.1: An example of a convex polytope. We can consider each face of the polytope as an affine inequality and then the polytope is all the points that satisfy each inequality. Notice that an affine inequality defines a half-plane and therefore is also the intersection of the half-planes.

Definition 11.2 (Convex Polytope)**.** The following are equivalent:

1. For $u_1, \ldots, u_m \in \mathbb{R}^n$ and $b_1, \ldots, b_m \in RR$, the set of $x \in \mathbb{R}^n$ s.t. $u_i^T x \leq b_i$ is a convex polytope.

2. Given a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $b \in \mathbb{R}^m$, the set of $x \in \mathbb{R}^n$ s.t. $Ax \leq b$ is a convex polytope.

3. Given a set of points $y_1, \ldots, y_k \in \mathbb{R}^n$, the convex hull $\mathbf{conv}(y_1, \ldots, y_k)$ is a convex polytope.A convex hull $\mathbf{conv}(y_1, \ldots, y_k)$ is the intersection of all convex sets containing $y_1, \ldots, y_k$.

I leave it as an exercise to show that the three definitions are equivalent.

Remark 11.3. *If F is a convex region and $x, y \in F$ then for every $\lambda \in [0,1]$, $\lambda x + (1-\lambda)y \in F$. Equivalently, the line segment $\overline{xy} \subseteq F$.*

The standard form of a Linear Program is the following. We will see later how to convert all linear programs into the standard form with only a $O(1)$ blowup in complexity.

Standard form of a (Primal) Linear Program.

$$(\mathcal{P}) = \begin{cases} \max & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \geq 0 \end{cases} \tag{11.1}$$

The reason that we care about linear programs is that most of the problems we are interested in have an equivalent formulation as a linear program. Therefore, if we build and optimize techniques for solving such problems, they will directly produce better algorithms for all the problems. Let's first see how max flow can be expressed as a linear program.

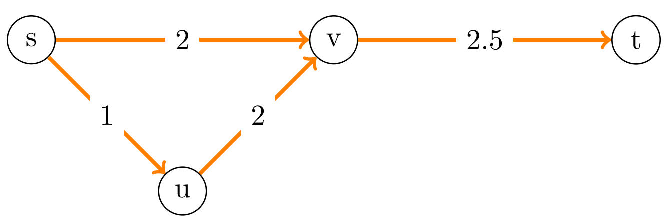

Figure 11.2: An example capacitated network

If we wanted to calculate the max flow in the capacitated network in Figure 11.2 then we could write that as the following linear program

$$
\begin{aligned}
\max \quad & f_{su} + f_{sv} \\
\text{s.t.} \quad & 0 \le f_{sv} \le 2 \\
& 0 \le f_{su} \le 1 \\
& 0 \le f_{uv} \le 2 \\
& 0 \le f_{vt} \le 2.5 \\
& f_{su} - f_{uv} = 0 \\
& f_{sv} + f_{uv} - f_{vt} = 0
\end{aligned}
\tag{11.2}
$$

We can make some simplifications by equating $f_{su} = f_{uv}$ and $f_{vt} = f_{uv} + f_{sv}$. Then we can rewrite the linear program in the standard form as

$$
\begin{aligned}
\max \quad & \begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} f_{su} \\ f_{sv} \end{pmatrix} \\
\text{s.t.} \quad & \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} f_{su} \\ f_{sv} \end{pmatrix} \le \begin{pmatrix} 2 \\ 1 \\ 2 \\ 2.5 \end{pmatrix} \\
& f_{su}, f_{sv} \ge 0
\end{aligned}
\tag{11.3}
$$

This is just one instance of writing a problem we have seen as a linear programming problem. We will see more and we will also see an algorithm for how to solve linear programs.

11.2 Optimums in the Feasible Region

Definition 11.4 (Feasibility)**.** For a linear program in the standard form of (11.1), we say the *feasible region* is the set $F = \{x : Ax \le b\}$. If $F \neq \emptyset$, we say F is *feasbile*. We say that the linear program is unbounded if F is unbounded.

A first question we need to ask ourselves when solving a linear program is whether a linear program is feasible. This itself is non-trivial. Assuming, however, that F is feasible, how would we go about finding an optimum? We exploit the two properties that the objective function $c^T x$ is linear and that F is a convex polytope.

Lemma 11.5 (Local optimums are global optimums). *Given $F \neq \emptyset$, if a point $x \in F$ is a local optimum then it is a global optimum.*

Proof. Assume for contradiction, x isn't a global optimum. Then $\exists y \in F$ s.t. $c^T x < c^T y$. But the line $\overline{xy} \in F$. Then for any $\lambda \in (0, 1]$,

$$c^T(\lambda x + (1 - \lambda)y) = \lambda c^T x + (1 - \lambda)c^T y > c^T x \tag{11.4}$$

Meaning, x is not a local optimum as moving towards y increases the objective function. □

Definition 11.6 (Vertex of a Polytope). Recall that we say z is on the line segement $\overline{xy}$ if there $\exists \lambda \in [0, 1]$ s.t. $z = \lambda x + (1 - \lambda)y$. A point $z \in F$ is a vertex of the polytope F if it is on no proper line segment contained in F.[1]

Remark 11.7. *If $v_1, \ldots, v_k$ are the vertices of a polytope F, then $F = \boldsymbol{conv}(v_1, \ldots, v_k)$. i.e. F is the convex hull of the vertices.*[2]

Theorem 11.8. *Let* **OPT** *be the optimal value of a standard form linear program and assume* **OPT** *is finite. Then* **OPT** *is achieved at a vertex.*

Proof. Let $v_1, \ldots, v_k$ be the vertices of F. Then every point $x \in F$ can be expressed as $\sum_{i=1}^{k} \lambda_i v_i$ with each $\lambda_i \geq 0$ and $\sum \lambda_i = 1$. By linearity of the objective function, if $\mathbf{OPT} = c^T x$ for $x \in \mathbb{R}^n$, then necessarily $c^T v_i \geq \mathbf{OPT}$ for some v_i, so the optimal value is achieved at the vertex. □

This statement is rather strong in that it means that we only need to check the vertices of a polytope in order to find the optimum. In particular, it also demonstrates that the proper way of thinking about a polytope is not by the number of linear equations that define it, but rather by the number of vertices. This is actually a very fruitful section of algorithm theory. There has been considerable work done to consider how to reduce the number of vertices. However, this is not always optimal. Consider the n-dimensional sube. It has 2^n vertices. In general if the polytope is defined by m half planes there are up $\binom{m+n}{n}$ vertices. We will soon see the simplex algorithm which will reduce the number of vertices we need to consider.

[1] A vertex is just the n-dimensional analog of what you think of as the vertex of a polytope in low dimension.

[2] This can be shown by induction on the dimension. The proof is a little tedious, so it has been ommitted.

11.3 Dual Linear Program

Consider the following simple linear program. Imagine there is a salesman who wants to sell either pens or markers. He sells pens for S_1 dollars and markers for S_2 dollars. He has restrictions due to labour, ink, and plastic. These are formalized in the following linear program.

$$\begin{aligned} \max \quad & S_1x_1 + S_2x_2 \\ \text{s.t.} \quad & L_1x_1 + L_2x_2 \leq L \\ & I_1x_1 + I_2x_2 \leq I \\ & P_1x_1 + P_2x_2 \leq P \\ & x_1, x_2 \geq 0 \end{aligned} \tag{11.5}$$

Let's consider the dual of the problem. Imagine now there are market prices for each of the three materials: labour, ink, and plastic. Call these y_L, y_I, y_P. Now the salesman is only going to sell pens if $y_LL_1 + y_II_1 + y_PP_1 \geq S_1$ and analagously for markers. Therefore, it is in the interest of the market prices to minimize the total available labour, ink, and plastic while still allowing the salesman to sell his goods. This is the dual problem and it can be expressed as follows.

$$\begin{aligned} \min \quad & y_LL + y_II + y_PP \\ \text{s.t.} \quad & y_LL_1 + y_II_1 + y_PP_1 \geq S_1 \\ & y_LL_2 + y_II_2 + y_PP_2 \geq S_2 \\ & y_L, y_I, y_P \geq 0 \end{aligned} \tag{11.6}$$

In general, the dual to the standard linear program expressed in (11.1) is

Standard form of a Dual Linear Program.

$$(\mathcal{D}) = \begin{cases} \min & b^T y \\ \text{s.t.} & A^T y \geq c \\ & y \geq 0 \end{cases} \tag{11.7}$$

The dual of a linear program is a powerful thing. An optimal solution to the dual gives us a bound on the solution to the primal.

Theorem 11.9 (Weak LP Duality)**.** *If $x \in \mathbb{R}^n$ is feasible for $(\mathcal{P})$ and $y \in \mathbb{R}^m$ is feasible for $(\mathcal{D})$, then*

$$c^T x \leq y^T A x \leq b^T y \tag{11.8}$$

Thus, if $(\mathcal{P})$ is unbounded, then $(\mathcal{D})$ is infeasible. Conversely if $(\mathcal{D})$ is unbounded, then $(\mathcal{P})$ is infeasible. Moreover, if $c^T x' = b^T y'$ with x' feasible for $(\mathcal{P})$ and y' feasible for $(\mathcal{D})$, then x' is an optimal solution for $(\mathcal{P})$ and y' is an optimal solution for $(\mathcal{D})$.

Proof. Assume $x \in \mathbb{R}^n$ is feasible for $(\mathcal{P})$ and $y \in \mathbb{R}^m$ is feasible for $(\mathcal{D})$. Therefore $Ax \leq b$ and $A^T y \geq c$. Multiplying the first equation by y^T on the left

$$y^T A x \leq y^T b = b^T y \tag{11.9}$$

Taking the transpose of the second equation ($y^T A \geq c^T$) and multiplying by x on the right

$$y^T A x \geq c^T x \tag{11.10}$$

Combining (11.9) and (11.10) generates (11.8). If $(\mathcal{P})$ is unbounded then for any M, $\exists$ a feasible x s.t. $c^T x > M$. By (11.8), any feasible y must satisfy $b^T y > M$ for any M. Clearly this is impossible so $(\mathcal{D})$ is infeasible. The converse follows similarly.

Assume x' wasn't optimal but $c^T x' = b^T y'$. Then $\exists x^\star$ feasible in $(\mathcal{P})$ with $c^T x' < c^T x^\star$. But then $b^T y' < c^T x^\star$, violating (11.8). So x' is optimal. A similar argument shows y' is optimal. $\square$

11.4 Simplex Algorithm

11.5 Approximation Theory

11.6 Two Person Zero-Sum Games

11.7 Randomized Sorting Complexity Lower Bound

11.8 Circuit Evaluation

11.9 Khachiyan's Ellipsoid Algorithm

11.10 Set Cover, Integer Linear Programming