Lecture 4: Linear Programming: Extreme Points

Notes by Ola Svensson[1]

In this lecture we do the following:

- We recall linear programming.

- We define extreme points and show that any optimal solution can be taken to be an extreme point (if feasible region is bounded).

- We then show how to exploit extreme point structure for max-weight bipartite matching and bipartite vertex cover.

These notes are based on [1].

1 Recall LPs, Extreme Points, and Bipartite Matching

Recall that a linear program is the problem of optimizing a linear objective subject to linear constraints (inequalities and equalities):

Definition 1 *A linear program (LP) is the problem of finding values for n variables $x_1, x_2, \ldots, x_n \in \mathbb{R}$ that minimize (or equivalently, maximize) a given linear objective function, subject to m linear constraints*

$$
\begin{aligned}
\textbf{minimize:} \quad & \sum_{i=1}^{n} c_i x_i \\
\textbf{Subject to:} \quad & \sum_{i} e_{i,j} x_i = b_j && \textit{for } j = 1, \ldots, m_1 \\
& \sum_{i} d_{i,k} x_i \geq g_k && \textit{for } k = 1, \ldots, m_2 \\
& \sum_{i} f_{i,p} x_i \leq l_p && \textit{for } p = 1, \ldots, m_3
\end{aligned}
$$

where $m_1 + m_2 + m_3 = m$.

An example of a linear program is as follows:

$$
\begin{aligned}
\text{Maximize} \quad & x + y \\
\text{Subject to} \quad & x + y \leq 2 \\
& y \leq 1 \\
& x, y \geq 0
\end{aligned}
$$

The following figure shows the feasible area.

---

[1]**Disclaimer:** These notes were written as notes for the lecturer. They have not been peer-reviewed and may contain inconsistent notation, typos, and omit citations of relevant works.

Feasible Solution
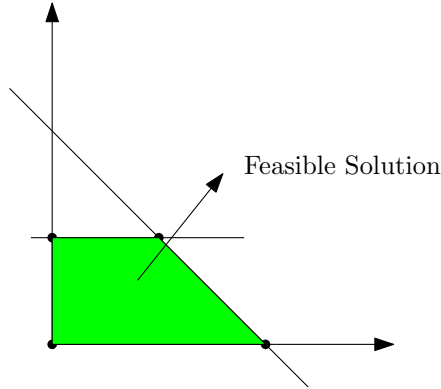
When solving discrete optimization problems (such as matchings, vertex cover, spanning tree, etc.) it will be important that optimal solutions have certain structure. That is where we use extreme points and their structure (which is also important for the simplex algorithm).

1.1 Extreme Points

Let us first define an extreme point:

Definition 2 *A feasible solution is an extreme point if it cannot be written as a convex combination of other feasible solutions.*

Just to recall:

Definition 3 *A convex combination of points $x_1, x_2, ..., x_n$ is a point of the form $\sum_{i=1}^{n} \lambda_i x_i$ where the real numbers λ_i satisfy $\sum_{i=1}^{n} \lambda_i = 1$ and $\lambda_i \in [0,1]$ for all i.*

Now, we state a theorem about extreme points. Extreme points are important because sometimes they have useful structural properties, which we can exploit to design algorithms. We see two such examples in this lecture.

Theorem 4 *If the feasible region is bounded, there always exists an optimum which is an extreme point.*

Proof   As the feasible region is bounded, there is an optimal solution $x^\star$ and any feasible point can be written as a convex combination of the extreme points. In particular, we have $x^\star = \sum_i \lambda_i x^{(i)}$ where $x^{(i)}$'s are feasible extreme points and $\lambda's$ are non-negative real numbers satisfying $\sum_i \lambda_i = 1$. Now let c be the vector defining the objective that we wish to maximize (the proof is the same if we minimize). Then we have

$$c^T x^\star = c^T (\sum_i \lambda_i x^{(i)}) = \sum_i \lambda_i c^T x^{(i)} \,,$$

which implies that there is an extreme point $x^{(i)}$ such that $c^T x^{(i)} \geq c^T x^\star$, i.e., it is an optimal solution.
∎

If the feasible region is not bounded, then we might not have any extreme points. For example, the following LP does not contain any extreme points.

$$\begin{aligned} \text{Maximize} \quad & y \\ \text{Subject to} \quad & y \leq 1 \\ & y \geq 0 \end{aligned}$$

Feasible Solution
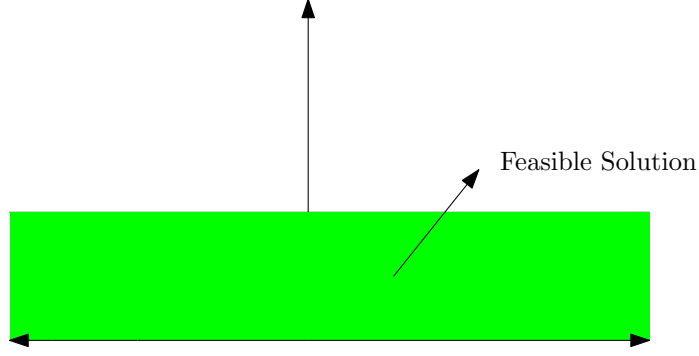
2 Maximum Weight Bipartite Perfect Matching

In this section we are going to concentrate on maximum weight/profit bipartite perfect matching. Given a bipartite graph $G = (V, E)$ with V partitioned into A and B and edge-weights $w \to \mathbb{R}$, this is the problem where we wish to find a perfect matching M that maximizes $w(M) = \sum_{e \in M} w(e)$. Recall that a matching is *perfect* if every vertex is incident to exactly one edge in the matching, i.e., every vertex is matched. The LP for the maximum weight bipartite perfect matching problem can be formulated as

$$\text{Maximize} \quad \sum_{e \in E} x_e w_e$$
$$\text{Subject to} \quad \sum_{e=(a,b) \in E} x_e = 1 \quad \forall a \in A$$
$$\sum_{e=(a,b) \in E} x_e = 1 \quad \forall b \in B$$
$$x_e \geq 0 \quad \forall e \in E$$

In the linear program, we have a variable x_e for each edge e with the intended meaning that it should take value 1 if that edge is picked by the matching. The constraints say that each vertex should have exactly one incident edge in the matching.

Note that in a feasible solution the variables x_e may take values strictly between 0 and 1, which does not really correspond to a matching. However, we have the following key structural result:

Claim 5 *For bipartite graphs, any extreme point solution to the LP is integral.*

Proof Let x^* be an extreme point for the graph $G = (V_1, V_2, E)$ and let $E_f = \{e \in E : 0 < x_e^* < 1\}$. Suppose towards contradiction that $E_f \neq \emptyset$. Note that E_f must then contain a cycle: indeed any vertex incident to an edge in E_f is incident to at least two edges in E_f. All these edges are fractional and we
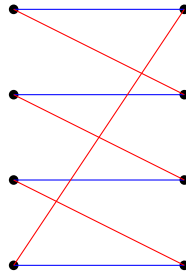
want to define y and z so that they are feasible solutions and $x^* = \frac{1}{2}(y + z)$ which will contradict the

fact that x^* is an extreme point. Let $e_1, e_2, ..., e_{2k}$ be the edges of the cycle. Let y, z be

$$y_e = \begin{cases} x_e^* + \epsilon & \text{if } e \in \{e_1, e_3, e_5, ..., e_{2k-1}\} \\ x_e^* - \epsilon & \text{if } e \in \{e_2, e_4, e_6, ..., e_{2k}\} \\ x_e^* & \text{otherwise} \end{cases}$$

$$z_e = \begin{cases} x_e^* - \epsilon & \text{if } e \in \{e_1, e_3, e_5, ..., e_{2k-1}\} \\ x_e^* + \epsilon & \text{if } e \in \{e_2, e_4, e_6, ..., e_{2k}\} \\ x_e^* & \text{otherwise} \end{cases}$$

Notice that the degree constraints are still satisfied by y and z as we are alternating between increasing and decreasing the edge values in a cycle of even length. Hence, to ensure feasibility, we need to choose such a small ϵ so as to guarantee that all y_e and z_e are in $[0, 1]$. For example $\epsilon = \min\{x_e^*, (1 - x_e^*) : e \in E_f\}$ gives that both y and z are feasible. Now one can easily see that $x^* = \frac{1}{2}(y + z)$ which contradicts the assumption that x^* is an extreme point. ∎

Because of the above claim, the polytope

$$P = \Big\{ x : \sum_{b:(a,b)\in E} x_{(a,b)} = 1 \qquad a \in A,$$

$$\sum_{a:(a,b)\in E} x_{(a,b)} = 1 \qquad b \in B,$$

$$x_{(a,b)} \geq 0 \qquad (a, b) \in E \Big\}$$

is called the *bipartite perfect matching polytope*. Also, it says that we can solve the maximum weight bipartite matching problem by simply solving the above linear program.

Unfortunately, the degree-constraints are not sufficient for general graphs which can be seen by the following example:
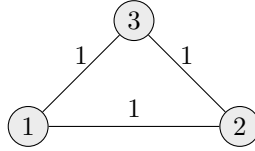

Here, the solution that sets $x_e = 1/2$ for every edge e gives the optimal value $3/2$, but any integral matching has only value 1 (and the graph does not even have a perfect matching).

3 Vertex Cover

Here, we consider minimum weight vertex cover on bipartite graphs. The definition of the vertex cover problem is:

Definition 6 *Given a graph $G = (V, E)$ with node-weights $w : V \to \mathbb{R}$ find a vertex cover C (i.e., $e \cap C \neq \emptyset$ for all $e \in E$) that minimizes $w(C) = \sum_{v \in C} w(v)$.*

Example 1 *The minimum weight vertex cover (depicted in black) of the following graph is 3 (obtained by taking the vertices of weight 1 and 2).*
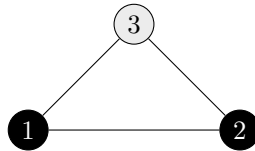
Similarly to the case of maximum weight bipartite perfect matching, we formulate a linear program for the vertex cover problem. Here, we have a variable x_v for each vertex v with the intended meaning that it takes value 1 if that vertex is in the vertex cover. The constraints say that each edge needs to be covered. The LP can be formulated as follows:

$$\text{Minimize} \quad \sum_{v \in V} x_v w(v)$$
$$\text{Subject to} \quad x_u + x_v \geq 1 \quad \forall \{u, v\} \in E$$
$$0 \leq x_v \leq 1 \quad \forall v \in V$$

We have the following structural result for bipartite graphs:

Claim 7 *For bipartite graphs, any extreme point to the vertex cover LP is integral.*

Proof Consider an extreme point x^* and let $V_f = \{v : 0 < x_v^* < 1\}$ be those vertices with fractional values in x^*. Suppose toward contradiction that $V_f \neq \emptyset$. Further, let A, B be the bipartition of V and let $A_f = V_f \cap A$ and $B_f \cap V_f$ be the fractional vertices in A and B, respectively. As for bipartite matchings, we reach a contradiction by defining feasible solutions y and z so that $x^* = \frac{1}{2}(y + z)$ which contradicts that x^* is an extreme point. Let $\epsilon = \min\{x_v, (1 - x_v) : v \in A_f \cup B_f\}$ and define y and z by

$$y_v = \begin{cases} x_v^* + \epsilon & \text{if } v \in A_f \\ x_v^* - \epsilon & \text{if } v \in B_f \\ x_v^* & \text{otherwise} \end{cases}$$

$$z_v = \begin{cases} x_v^* - \epsilon & \text{if } v \in A_f \\ x_v^* + \epsilon & \text{if } e \in B_f \\ x_v^* & \text{otherwise} \end{cases}$$

Clearly, we have that $x^* = \frac{1}{2}(y + z)$. It remains to verify that y and z are feasible solutions. Let us verify y (the argument that z is feasible is the same). By the selection of ϵ, we have that $0 \leq y_v \leq 1$ for all $v \in V$. We now need to verify that the constraint for every edge is satisfied. Consider an edge $\{a, b\} \in E$ in the bipartite graph. We need to verify that $y_a + y_b \geq 1$. If $x_a = 1$ (or $x_b = 1$), then we have that $y_a = 1$ (or $y_b = 1$) and so the constraint holds. Otherwise $0 < x_a, x_b < 1$ and so $a \in A_f$ and $b \in B_f$. This in turn implies that y is feasible since

$$y_a + y_b = (x_a + \epsilon) + (x_b - \epsilon) = x_a + x_b \geq 1\,,$$

where the last inequality holds because x^* is a feasible solution. ∎

The above structural result says that we can solve minimum weighted vertex cover on bipartite graphs by simply solving the above linear program to find an optimal extreme point. It does not work for general graphs which can be seen by considering the triangle (as for matchings). In contrast to matchings, the vertex cover problem on general graphs is an NP-hard problem and we do not expect to have efficient algorithms for it.

References

[1] Ashkan Norouzi-Fard, Christos Kalaitzis: *Scribes of Lecture 9 in Topics in TCS 2014.* *http://theory.epfl.ch/courses/topicstcs/Lecture9.pdf*