Trees, Flows, and Cuts

1 Minimum Spanning Trees

1.1 Minimum Spanning Tree (MST) Problem

Problem. Given a connected, undirected graph $G = (V, E)$ and costs c_e on each edge $e \in E$, find a minimum cost spanning tree of G.

We let $n = |V|$ and $m = |E|$. This problem can be solved efficiently by Kruskal's Algorithm:

Algorithm 1 Kruskal's Algorithm

```
$T = \emptyset$
Sort edges in order of increasing cost
for each edge $e$ in order do
  if $T \bigcup e$ is acyclic then
    $T = T \bigcup e$
  end if
end for
```

1.2 MST LP

By allowing fractional MSTs, we have the MST LP:

$$\begin{aligned} \min \ & \sum_{e\in E} c_e x_e \\ \text{s.t.} \ & \sum_{e\in E} x_e = n-1 \\ & \sum_{e\subseteq X} x_e \le |X| - 1 \ \forall X \subset V \\ & x_e \ge 0 \ \forall e \in E \end{aligned}$$

The interpretation of this LP is that we wish to minimize the total cost of the fractional MST where the first constraint encodes that the tree must have $n-1$ edges, the second constraint encode that G is acyclic, and the third constraint forces the x_e to be positive.

Theorem. *The feasible region of the above LP is the convex hull of spanning trees.*

Proof. By finding a dual solution with cost matching the output of Kruskal's Algorithm. □

Note that this LP has exponential (in n) constraints. Thus we need an efficient separation oracle (an algorithm which, given an input vector, either certifies that it is in the feasible region or identifies a violated constraint) to solve it efficiently. In this case, our separation oracle must find a nonempty $X \subset V$ with $\sum_{e\subseteq X} x_e > |X| - 1$, if such an X exists. We can do this by minimizing $|X| - \sum_{e\subseteq X} x_e$ and checking if it is less than one, which is efficient using submodular minimization.

1.3 Application of MST LP

It seems unintuitive to solve for a fractional MST, but it is useful when designing robust MSTs via a randomized selection algorithm, perhaps to foil hackers. Fix some probability p, the maximum probability with which you want to select any edge in the graph (perhaps $p = 0.1$). Then, we have the robust MST LP:

$$\begin{aligned}
\min \quad & \sum_{e \in E} c_e x_e \\
\text{s.t.} \quad & \sum_{e \in E} x_e = n - 1 \\
& \sum_{e \subseteq X} x_e \leq |X| - 1 \ \forall X \subset V \\
& x_e \leq p \ \forall e \in E \\
& x_e \geq 0 \ \forall e \in E
\end{aligned}$$

The interpretation of this LP is that the fractional MST can be interpreted as a probability distribution over integral MSTs – we know that a solution to this LP must exist! Evidently, $\boldsymbol{x}$ is in the original MST LP polytope. By Caratheodory's Theorem, $\boldsymbol{x}$ is a convex combination of $m + 1$ vertices of the MST polytope. Then by the integrality of those vertices, $\boldsymbol{x}$ is the expectation of a probability distribution over spanning trees.

2 Maximum Flows

2.1 Maximum Flow LP

Problem. Given a directed graph $G = (V, E)$ with capacities u_e on each edge e and source/sink nodes $s, t \in V$, find the maximum flow from $s \to t$ respecting the capacities.

By allowing fractional flows, we obtain the LP:

$$\begin{aligned}
\max \quad & \sum_{e \in \delta^+(s)} x_e - \sum_{e \in \delta^-(s)} x_e \\
& \sum_{e \in \delta^+(s)} x_e = \sum_{e \in \delta^-(s)} x_e \ \forall v \in V \setminus \{s, t\} \\
& x_e \leq u_e \ \forall e \in E \\
& x_e \geq 0 \ \forall e \in E
\end{aligned}$$

The interpretation of this LP is that we wish to maximize the amount of flow from s to t while obeying conversation of flow (constraint 1), edge capacities (constraint 2), and positive flow (constraint 3). Taking the dual LP:

$$\begin{aligned}
\min \quad & \sum_{e \in E} u_e z_e \\
\text{s.t.} \quad & y_v - y_u \leq z_e \ \forall e = (u, v) \in E \\
& y_s = 0 \\
& y_t = 1 \\
& z_e \geq 0 \ \forall e \in E
\end{aligned}$$

The interpretation of this LP is that z_e describes the fraction of each edge to cut, while cutting at least one edge on every path from $s \to t$. We minimize the total capacity of this cut, and recognize that the dual LP therefore encodes the minimum cut problem. Every integral $s \to t$ cut is feasible, and strong duality implies the equivalence of maximum flow and minimum cut. Furthermore, the Ford-Fulkerson Algorithm shows that there exists an integral maximum flow when the edge capacities are integer.

2.2 Generalizations

Encoding a combinatorial problem as an LP allows us to easily extend it to more general situations. In the case of maximum flow, it is easy to write upper and lower bounds on the flow for each edge, find the minimum cost flow of at least a certain amount, encode commodity sharing, and so on. In particular, by extending this problem from an LP to a convex program, we can encode the minimum congestion flow problem. Here, we are given congestion functions $c_e(\cdot)$ on each edge e, and we want to find the minimum average congestion flow from $s \to t$:

$$\begin{aligned} \min \ & \sum_{e \in E} x_e c_e(x_e) \\ & \sum_{e \in \delta^+(s)} x_e - \sum_{e \in \delta^-(s)} x_e \ \forall v \in V \setminus \{s,t\} \\ & \sum_{e \in \delta^+(s)} x_e - \sum_{e \in \delta^-(s)} x_e = r \\ & x_e \geq 0 \ \forall e \in E \end{aligned}$$

Note that this is a convex program when the $c_e(\cdot)$ are polynomials with non-negative coefficients.

3 Maximum Cuts

3.1 Maximum Cut Problem

Problem. Given an undirected graph $G = (V, E)$, find a 2-partition of G maximizing the number of edges which are cut.

We have the combinatorial program:

$$\begin{aligned} \max \ & \sum_{(i,j) \in E} \frac{1 - x_i x_j}{2} \\ \text{s.t. } & x_i \in \{-1, 1\} \ \forall i \in V \end{aligned}$$

Recall that we were able to relax this problem as a semi-definite program because we are solely interested in the pairwise relationships between each x_i. We have:

$$\begin{aligned} \max \ & \sum_{(i,j) \in E} \frac{1 - \mathbf{Y}_{ij}}{2} \\ \text{s.t. } & Y_{ii} = 1 \ \forall i \in V \\ & Y \in S_+^n \end{aligned}$$

3.2 Approximation Algorithm

The Goemans-Williamson Algorithm provides a method to calculate an approximate maximum cut given the solution to the semi-definite program relaxation. First, we solve the program to obtain an optimal semi-definite $\mathbf{Y}$ in $\mathbb{R}^{n \times n}$. Then, we decompose $\mathbf{Y} = \mathbf{V}\mathbf{V}^\intercal$. We select a random hyperplane $\boldsymbol{r}$ on the unit sphere and place nodes i with $\boldsymbol{v_i r} \geq 0$ on one side of the cut, and other nodes on the opposite side.

Lemma. *The random hyperplane cuts each edge (i, j) with probability at least $0.878\frac{1-\mathbf{Y}_{ij}}{2}$.*

Proof. Note that for all angles $\theta \in [0, \pi]$, $\theta/\pi \geq 0.878\frac{1-\cos(\theta)}{2}$. Consider an edge (i, j). It will only be cut if $\operatorname{sgn}(\boldsymbol{v_i r}) \neq \operatorname{sgn}(\boldsymbol{v_j r})$. So, we can zoom in on the 2D plane containing these two vectors, and notice that since $\boldsymbol{r}$ is a random hyperplane, the component of $\boldsymbol{r}$ in this plane must also be random. Let θ_{ij} be the angle between $\boldsymbol{v_i}$ and $\boldsymbol{v_j}$. Then $\mathbf{Y}_{ij} = \boldsymbol{v_i v_j} = \cos(\theta_{ij})$. Thus $\boldsymbol{r}$ cuts these two vectors with probability

$$\frac{2\theta_{ij}}{2\pi} = \frac{\theta_{ij}}{\pi} \geq 0.878\frac{1-\cos(\theta)}{2} = 0.878\frac{1-\mathbf{Y}_{ij}}{2}$$

□

Theorem. *As a result of the above lemma, the Goemans-Williamson Algorithm outputs a random cut of expected size at least* $0.878 \cdot OPT$. *This is the best possible approximation unless the Unique Games Conjecture is false.*