Randomized Rounding

- Brief Introduction to Linear Programming and Its Usage in Combinatorial Optimization
- Randomized Rounding for Cut Problems
- Randomized Rounding for Satisfiability Problems
- Randomized Rounding for Covering Problems
- Randomized Rounding and Semi-definite Programming

Approximate Sampling and Counting

- ...
(Randomized) Rounding

- A (minimization) combinatorial problem $\Pi \Leftrightarrow$ an ILP
- Let $\bar{y}$ be an optimal solution to the ILP
- Relax ILP to get an LP; let $y^*$ be an optimal solution to the LP
- Then,

$$\text{OPT}(\Pi) = \text{cost}(\bar{y}) \geq \text{cost}(y^*)$$

(If $\Pi$ is maximization, reverse the inequality!)

- Carefully “round” $y^*$ (rational) to get a feasible solution $y^A$ (integral) to the ILP, such that $y^A$ is not too bad, say $\text{cost}(y^A) \leq \alpha \cdot \text{cost}(y^*)$
- Conclude that $\text{cost}(y^A) \leq \alpha \cdot \text{OPT}(\Pi)$
- Thus, we get an $\alpha$-approximation algorithm for $\Pi$
- If $\alpha = 1$, then we have solved $\Pi$ exactly!
An Integer Linear Program for Minimum Cut

Definition (Min-Cut Problem)

Given a (undirect/directed) graph $G = (V, E)$, edge capacities $c : E \to \mathbb{N}$, a source $s \in V$, sink $t \in V$, find a subset $C$ of edges such that removing $C$ disconnects $t$ from $s$ (i.e. there's no path from $s$ to $t$), such that $C$ has minimum total capacity.

Let $\mathcal{P}$ be the set of all $s, t$-paths

$$
\begin{align*}
\min & \quad \sum_{e \in E} c_e y_e \\
\text{subject to} & \quad \sum_{e \in P} y_e \geq 1, \quad \forall P \in \mathcal{P}, \\
& \quad y_e \in \{0, 1\}, \quad \forall e \in E.
\end{align*}
$$

Let $\bar{y}$ be an optimal solution to this ILP.
To Relax and Integer LP is to relax the integral constraints
The relaxation of the ILP is a linear program:

\[
\begin{align*}
\text{min} & \quad \sum_{e \in E} c_e y_e \\
\text{subject to} & \quad \sum_{e \in P} y_e \geq 1, \quad \forall P \in \mathcal{P}, \\
& \quad y_e \geq 0, \quad \forall e \in E.
\end{align*}
\]
The Randomized Rounding Step

- Let $y^*$ be an optimal solution to the LP
- Think of $y_e^*$ as the “length” of $e$. Let $d(s, u)$ be the distance from $s$ to $u$ in terms of $y^*$-length. Then, $d(s, t) \geq 1$.
- For each $r \in [0, 1]$, let $B(r) := \{u \mid d(s, u) \leq r\}$ and
  
  $C(r) = [B(r), \overline{B(r)}]$  
  
- Choose $r \in [0, 1)$ uniformly at random (a continuous distribution now!). Output the cut $C = C(r)$
(Expected) Quality of the Solution

- Expected quality of the solution

\[ E[\text{cap}(C')] = \sum_{e=(u,v) \in E} c_e \text{Prob}[e \in C'] \]
\[ \leq \sum_{e=(u,v) \in E} c_e \frac{d(s,v) - d(s,u)}{1 - 0} \leq \sum_{e \in E} c_e y_e^* = \text{cost}(y^*). \]

- And so,

\[ E[\text{cap}(C')] \leq \text{cost}(y^*) \leq \text{cost}(\bar{y}) = \text{min-cut capacity of } G \]

- Anything “weird”?

- Conclude that, just output \( C(r) \) for any \( r \in [0, 1) \) and we have a minimum cut!
Computers cannot choose \( r \in [0, 1) \) uniformly at random! (They can’t deal with continuous things.)

Fortunately, there are only finitely many \( B(r) \), even though \( r \in [0, 1) \).

There are \( 0 < r_1 < r_2 < \cdots < r_k < 1 \) such that

- For \( r \in [0, r_1) \) we get cut \( C_0 \) (with prob \( r_1 \))
- For \( r \in [r_1, r_2) \) we get cut \( C_1 \) (with prob \( r_2 - r_1 \))
- ...

Let \( \mathcal{C} \) be this set of \( k \) cuts, then,

\[
\text{cost}(y^*) \geq \mathbb{E}[	ext{cap}(\mathcal{C})] = \sum_{i=0}^{k-1} \text{cap}(C_i) \text{ Prob}[C_i]
\]
The Dual Linear Program (DLP)

Here’s the dual linear program of LP (5)

\[
\begin{align*}
\text{max} & \quad \sum_{P \in \mathcal{P}} f_P \\
\text{subject to} & \quad \sum_{P : e \in P} f_P \leq c_e, \quad \forall e \in E, \\
& \quad f_P \geq 0, \quad \forall P \in \mathcal{P}.
\end{align*}
\]  

(3)

- This is precisely the **maximum flow** problem!
- Let \( f^* \) be a maximum flow, then by “strong duality”

\[
\text{cost}(f^*) = \text{cost}(y^*)
\]
Lemma (Maxflow-Mincut, Weak Duality)

For every cut $C$ of $G$, $\text{cap}(C) \geq \text{maxflow}$.

Theorem (Maxflow-Mincut, Strong Duality)

There exists a cut $C$ such that $\text{cap}(C) = \text{maxflow}$. 
Definition (Multiway-Cut Problem)

Given a graph $G = (V, E)$, edge capacities $c : E \rightarrow \mathbb{N}$, and $k$ terminals $t_1, t_2, \ldots, t_k \in V$, find a subset $C$ of edges such that removing $C$ disconnect all terminals from each other such that $C$ has minimum total capacity.

Let $\mathcal{P}$ be the set of all $t_i, t_j$-paths, $i \neq j$, $i, j \in [k]$

\[
\begin{align*}
\min & \quad \sum_{e \in E} c_ey_e \\
\text{subject to} & \quad \sum_{e \in P} y_e \geq 1, \quad \forall P \in \mathcal{P}, \\
& \quad y_e \in \{0, 1\}, \quad \forall e \in E.
\end{align*}
\] (4)

Let $\tilde{y}$ be an optimal solution to this ILP.
Relaxation

The relaxation of the ILP is a linear program:

\[
\begin{align*}
\min & \quad \sum_{e \in E} c_e y_e \\
\text{subject to} & \quad \sum_{e \in P} y_e \geq 1, \quad \forall P \in \mathcal{P}, \\
& \quad y_e \geq 0, \quad \forall e \in E.
\end{align*}
\]
The Randomized Rounding Step

- Let $y^*$ be an optimal solution to the LP
- Think of $y^*_e$ as the “length” of $e$. Let $d(t_i, u)$ be the distance from $t_i$ to $u$ in terms of $y^*$-length. Then, $d(t_i, t_j) \geq 1$ for every pair $i, j \in [k], i \neq j$.
- For each $r \in [0, 1]$, let $B_i(r) := \{u \mid d(t_i, u) \leq r\}$ and
  \[
  C_i(r) = [B_i(r), \overline{B_i(r)}]
  \]
- Choose $r \in [0, 1/2)$ uniformly at random

Output the cut $C = C_1(r) \cup \cdots \cup C_k(r)$

The rest is a homework problem! We get a 2-approximation algorithm for multiway cut

©Hung Q. Ngo (SUNY at Buffalo) CSE 694 – A Fun Course