Chapter 21

Approximation Algorithms using Linear Programming

21.1 Weighted vertex cover

21.1.0.1 Weighted vertex cover

Weighted Vertex Cover problem \( G = (V, E) \).
Each vertex \( v \in V \): cost \( c_v \).
Compute a vertex cover of minimum cost.

(A) vertex cover: subset of vertices \( V \) so each edge is covered.
(B) NP-Hard
(C) ...unweighted Vertex Cover problem.
(D) ... write as an integer program (IP):
(E) \( \forall v \in V: x_v = 1 \iff \text{v in the vertex cover} \).
(F) \( \forall vu \in E: \text{covered.} \implies x_v \lor x_u \text{ true.} \implies x_v + x_u \geq 1 \).
(G) minimize total cost: \( \min \sum_{v \in V} x_v c_v \).

21.1.1 Weighted vertex cover

21.1.1.1 State as IP \( \implies \) Relax \( \implies \) LP

\[
\begin{align*}
\min & \quad \sum_{v \in V} c_v x_v, \\
\text{such that} & \quad x_v \in \{0, 1\} \quad \forall v \in V \\
& \quad x_v + x_u \geq 1 \quad \forall vu \in E.
\end{align*}
\] (21.1)
(A) ... NP-Hard.
(B) relax the integer program.
(C) allow \( x_v \) get values \( \in \{0, 1\} \).
(D) \( x_v \in \{0, 1\} \) replaced by \( 0 \leq x_v \leq 1 \). The resulting LP is

\[
\min \sum_{v \in V} c_v x_v, \\
\text{s.t.} \quad 0 \leq x_v \quad \forall v \in V, \\
\quad x_v \leq 1 \quad \forall v \in V, \\
\quad x_v + x_u \geq 1 \quad \forall vu \in E.
\]

21.1.1.2 Weighted vertex cover – rounding the LP

(A) Optimal solution to this LP: \( \hat{x}_v \) value of var \( X_v, \forall v \in V \).
(B) optimal value of LP solution is \( \hat{\alpha} = \sum_{v \in V} c_v \hat{x}_v \).
(C) optimal integer solution: \( x^I_v, \forall v \in V \) and \( \alpha^I \).
(D) Any valid solution to IP is valid solution for LP!
(E) \( \hat{\alpha} \leq \alpha^I \).

Integral solution not better than LP.
(F) Got fractional solution (i.e., values of \( \hat{x}_v \)).
(G) Fractional solution is better than the optimal cost.
(H) Q: How to turn fractional solution into a (valid!) integer solution?
(I) Using rounding.

21.1.1.3 How to round?

(A) consider vertex \( v \) and fractional value \( \hat{x}_v \).
(B) If \( \hat{x}_v = 1 \) then include in solution!
(C) If \( \hat{x}_v = 0 \) then do \textbf{not} include in solution.
(D) if \( \hat{x}_v = 0.9 \implies \text{LP considers } v \text{ as being 0.9 useful.} \)
(E) The LP puts its money where its belief is...
(F) ...\( \hat{\alpha} \) value is a function of this “belief” generated by the LP.
(G) \textbf{Big idea}: Trust LP values as guidance to usefulness of vertices.
(H) Pick all vertices \( \geq \) threshold of usefulness according to LP.
(I) \( S = \left\{ v \mid \hat{x}_v \geq 1/2 \right\} \).
(J) Claim: \( S \) a valid vertex cover, and cost is low.
(K) Indeed, edge cover as: \( \forall vu \in E \) have \( \hat{x}_v + \hat{x}_u \geq 1 \).
(L) \( \hat{x}_v, \hat{x}_u \in (0, 1) \implies \hat{x}_v \geq 1/2 \) or \( \hat{x}_u \geq 1/2 \).
\[ \implies v \in S \text{ or } u \in S \text{ (or both).} \]
\[ \implies S \text{ covers all the edges of } G. \]

21.1.1.4 Cost of solution

Cost of \( S \):

\[
c_S = \sum_{v \in S} c_v = \sum_{v \in \hat{S}} c_v \leq \sum_{v \in \hat{S}} 2\hat{x}_v \cdot c_v \leq 2 \sum_{v \in \hat{S}} \hat{x}_v c_v = 2\hat{\alpha} \leq 2\alpha^I,
\]

since \( \hat{x}_v \geq 1/2 \) as \( v \in S \).

\( \alpha^I \) is cost of the optimal solution \[ \implies \]

\textbf{Theorem 21.1.1.} The \textbf{Weighted Vertex Cover} problem can be 2-approximated by solving a single LP. Assuming computing the LP takes polynomial time, the resulting approximation algorithm takes polynomial time.
21.1.2 The lessons we can take away

21.1.2.1 Or not - boring, boring, boring.

(A) Weighted vertex cover is simple, but resulting approximation algorithm is non-trivial.
(B) Not aware of any other 2-approximation algorithm does not use \( LP \). (For the weighted case!)
(C) Solving a relaxation of an optimization problem into a \( LP \) provides us with insight.
(D) But... have to be creative in the rounding.

21.2 Revisiting Set Cover

21.2.0.1 Revisiting Set Cover

(A) Purpose: See new technique for an approximation algorithm.
(B) Not better than greedy algorithm already seen \( O(\log n) \) approximation.

**Set Cover**

**Instance:** \((S, \mathcal{F})\)
- \(S\): set of \( n \) elements
- \(\mathcal{F}\): family of subsets of \( S\), s.t. \( \bigcup_{X \in \mathcal{F}} X = S\).

**Question:** The set \( X \subseteq \mathcal{F} \) such that \( X \) contains as few sets as possible, and \( X \) covers \( S \).

21.2.0.2 Set Cover – IP & LP

\[
\begin{align*}
\text{min} \quad & \alpha = \sum_{U \in \mathcal{F}} x_U, \\
\text{s.t.} \quad & x_U \in \{0, 1\} \quad \forall U \in \mathcal{F}, \\
& \sum_{U \in \mathcal{F}, s \in U} x_U \geq 1 \quad \forall s \in S.
\end{align*}
\]

Next, we relax this IP into the following LP.

\[
\begin{align*}
\text{min} \quad & \alpha = \sum_{U \in \mathcal{F}} x_U, \\
& 0 \leq x_U \leq 1 \quad \forall U \in \mathcal{F}, \\
& \sum_{U \in \mathcal{F}, s \in U} x_U \geq 1 \quad \forall s \in S.
\end{align*}
\]

21.2.0.3 Set Cover – IP & LP

(A) LP solution: \( \forall U \in \mathcal{F}, \hat{x}_U \), and \( \hat{\alpha} \).
(B) Opt IP solution: \( \forall U \in \mathcal{F}, \hat{x}^I_U \), and \( \hat{\alpha}^I \).
(C) Use LP solution to guide in rounding process.
(D) If \( \hat{x}_U \) is close to 1 then pick \( U \) to cover.
(E) If \( \hat{x}_U \) close to 0 do not.
(F) Idea: Pick \( U \in \mathcal{F} \): randomly choose \( U \) with probability \( \hat{x}_U \).
(G) Resulting family of sets $\mathcal{G}$.
(H) $Z_S$: indicator variable. 1 if $S \in \mathcal{G}$.
(I) Cost of $\mathcal{G}$ is $\sum_{S \in \mathcal{F}} Z_S$, and the expected cost is $E[\text{cost of } \mathcal{G}] = E[\sum_{S \in \mathcal{F}} Z_S] = \sum_{S \in \mathcal{F}} E[Z_S] = \sum_{S \in \mathcal{F}} \Pr[S \in \mathcal{G}] = \sum_{S \in \mathcal{F}} \hat{x}_S = \hat{\alpha} \leq \alpha'$. 
(J) In expectation, $\mathcal{G}$ is not too expensive.
(K) Bigus problumos: $\mathcal{G}$ might fail to cover some element $s \in S$.

21.2.0.4 Set Cover – Rounding continued

(A) Sol: Repeat rounding $m = 10 \lceil \lg n \rceil = \mathcal{O}(\log n)$ times.
(B) $n = |S|$.
(C) $\mathcal{G}_i$: random cover computed in $i$th iteration.
(D) $\mathcal{H} = \bigcup_i \mathcal{G}_i$. Return $\mathcal{H}$ as the required cover.

21.2.0.5 The set $\mathcal{H}$ covers $S$

(A) For an element $s \in S$, we have that

\[
\sum_{U \in \mathcal{F}, s \in U} \hat{x}_U \geq 1, \quad (21.2)
\]

(B) probability $s$ not covered by $\mathcal{G}_i$ (ith iteration set).
\[
\Pr[s \text{ not covered by } \mathcal{G}_i] = \Pr[\text{ no } U \in \mathcal{F}, \text{ s.t. } s \in U \text{ picked into } \mathcal{G}_i] = \prod_{U \in \mathcal{F}, s \in U} \Pr[U \text{ was not picked into } \mathcal{G}_i] = \prod_{U \in \mathcal{F}, s \in U} (1 - \hat{x}_U) \leq \prod_{U \in \mathcal{F}, s \in U} \exp(-\hat{x}_U) = \exp\left(-\sum_{U \in \mathcal{F}, s \in U} \hat{x}_U\right) \leq \exp(-1) \leq \frac{1}{2}, \leq \frac{1}{2}
\]

(C) probability $s$ is not covered in all $m$ iterations \(\leq \left(\frac{1}{2}\right)^m < \frac{1}{n^m}\),

(D) ...since $m = \mathcal{O}(\log n)$.

(E) probability one of $n$ elements of $S$ is not covered by $\mathcal{H}$ is \(\leq n(1/n^{10}) = 1/n^9\).

21.2.0.6 Cost of solution

(A) Have: $E[\text{cost of } \mathcal{G}_i] \leq \alpha'$. 

(B) $\implies$ Each iteration expected cost of cover $\leq$ cost of optimal solution (i.e., $\alpha'$).

(C) Expected cost of the solution is

\[
c_{\mathcal{H}} \leq \sum_i c_{B_i} \leq ma' = \mathcal{O}(\alpha' \log n).
\]

21.2.0.7 The result

Theorem 21.2.1. By solving an LP one can get an $\mathcal{O}(\log n)$-approximation to set cover by a randomized algorithm. The algorithm succeeds with high probability.
21.2.0.8 Same algorithms works for...

**Corollary 21.2.2.** By solving an LP one can get an $O(\log n)$-approximation to set cover by a randomized algorithm. The algorithm also works for the weighted case.

\[
\min \alpha = \sum_{U \in \mathcal{F}} w_U x_U
\]

\[
0 \leq x_U \leq 1 \quad \forall U \in \mathcal{F},
\]

\[
\sum_{U \in \mathcal{F}, s \in U} x_U \geq 1 \quad \forall s \in S.
\]

Rounding algorithm as before...

21.2.1 Cost of solution (weighted case)...

21.2.1.1 Same same, not the same.

(A) Fractional LP solution. Target: $\hat{\alpha} \quad \forall U \in \mathcal{F}: \hat{x}_U \in [0, 1]$.

(B) Integral opt solution. Target: $\alpha^I \quad \forall U \in \mathcal{F}: x^I_U \in \{0, 1\}$.

(C) $\alpha^I = \sum_{U \in \mathcal{F}} w_U x^I_U$.

(D) Rounding. $\forall U \in \mathcal{F}: \Pr[X_U = 1] = \hat{x}_U$.

\[
\mathbb{E}[\text{cost } G_i] = \sum_{U \in \mathcal{F}} \mathbb{E}[w_U X_U] = \sum_{U \in \mathcal{F}} w_U \hat{x}_U = \hat{\alpha} \leq \alpha^I.
\]

(E) Have: $\mathbb{E}[\text{cost of } G_i] \leq \alpha^I$.

(F) $\implies$ Each iteration expected cost of cover $\leq$ cost of optimal solution (i.e., $\alpha^I$).

(G) Expected cost of the solution is

\[
c_{\alpha I} \leq \sum_{i=1}^{O(\log n)} c_{B_i} \leq m \alpha^I = O(\alpha^I \log n).
\]

21.3 Minimizing congestion

21.3.0.1 Minimizing congestion by example

21.3.0.2 Minimizing congestion

(A) $G$: graph. $n$ vertices.

(B) $\pi_i, \sigma_i$ paths with the same endpoints $v_i, u_i \in V(G)$, for $i = 1, \ldots, t$.  

\[\pi_1, \pi_2, \pi_3, \sigma_1, \sigma_2, \sigma_3\]
(C) Rule I: Send one unit of flow from \( v_i \) to \( u_i \).

(D) Rule II: Choose whether to use \( \pi_i \) or \( \sigma_i \).

(E) Target: No edge in \( G \) is being used too much.

**Definition 21.3.1.** Given a set \( X \) of paths in a graph \( G \), the **congestion** of \( X \) is the maximum number of paths in \( X \) that use the same edge.

### 21.3.0.3 Minimizing congestion

(A) \( \mathbf{IP} \implies \mathbf{LP}: \)

\[
\begin{align*}
\min & \quad w \\
\text{s.t.} & \quad x_i \geq 0 & i = 1, \ldots, t, \\
& \quad x_i \leq 1 & i = 1, \ldots, t, \\
& \quad \sum_{e \in \pi_i} x_i + \sum_{e \in \sigma_i} (1 - x_i) \leq w & \forall e \in E.
\end{align*}
\]

(B) \( \hat{x}_i \): value of \( x_i \) in the optimal \( \mathbf{LP} \) solution.

(C) \( \hat{w} \): value of \( w \) in \( \mathbf{LP} \) solution.

(D) Optimal congestion must be bigger than \( \hat{w} \).

(E) \( X_i \): random variable one with probability \( \hat{x}_i \), and zero otherwise.

(F) If \( X_i = 1 \) then use \( \pi \) to route from \( v_i \) to \( u_i \).

(G) Otherwise use \( \sigma_i \).

### 21.3.0.4 Minimizing congestion

(A) Congestion of \( e \) is

\[
Y_e = \sum_{e \in \pi_i} X_i + \sum_{e \in \sigma_i} (1 - X_i).
\]

(B) And in expectation

\[
\alpha_e = \mathbb{E}[Y_e] = \mathbb{E} \left[ \sum_{e \in \pi_i} X_i + \sum_{e \in \sigma_i} (1 - X_i) \right]
\]

\[
= \sum_{e \in \pi_i} \mathbb{E}[X_i] + \sum_{e \in \sigma_i} \mathbb{E}[1 - X_i]
\]

\[
= \sum_{e \in \pi_i} \hat{x}_i + \sum_{e \in \sigma_i} (1 - \hat{x}_i) \leq \hat{w}.
\]

(C) \( \hat{w} \): Fractional congestion (from \( \mathbf{LP} \) solution).

### 21.3.0.5 Minimizing congestion - continued

(A) \( Y_e = \sum_{e \in \pi_i} X_i + \sum_{e \in \sigma_i} (1 - X_i) \).

(B) \( Y_e \) is just a sum of independent 0/1 random variables!

(C) Chernoff inequality tells us sum can not be too far from expectation!

### 21.3.0.6 Minimizing congestion - continued

(A) By Chernoff inequality:

\[
\Pr[Y_e \geq (1 + \delta)\alpha_e] \leq \exp \left( -\frac{\alpha_e \delta^2}{4} \right) \leq \exp \left( -\frac{\hat{w} \delta^2}{4} \right).
\]
Let $\delta = \sqrt{\frac{400}{\widehat{w}} \ln t}$. We have that

$$\Pr\left[Y_e \geq (1 + \delta)\alpha_e\right] \leq \exp\left(-\frac{\delta^2 \widehat{w}}{4}\right) \leq \frac{1}{t^{100}},$$

If $t \geq n^{1/50} \implies \forall$ edges in graph congestion $\leq (1 + \delta)\widehat{w}$.

$\delta$: Number of pairs, $n$: Number of vertices in $G$.

### 21.3.0.7 Minimizing congestion - continued

(A) Got: For $\delta = \sqrt{\frac{400}{\widehat{w}} \ln t}$. We have

$$\Pr\left[Y_e \geq (1 + \delta)\alpha_e\right] \leq \exp\left(-\frac{\delta^2 \widehat{w}}{4}\right) \leq \frac{1}{t^{100}},$$

(B) Play with the numbers. If $t = n$, and $\widehat{w} \geq \sqrt{n}$. Then, the solution has congestion larger than the optimal solution by a factor of

$$1 + \delta = 1 + \sqrt{\frac{20}{\widehat{w}} \ln t} \leq 1 + \sqrt{\frac{20 \ln n}{n^{1/4}}},$$

which is of course extremely close to 1, if $n$ is sufficiently large.

### 21.3.0.8 Minimizing congestion: result

**Theorem 21.3.2.** (A) $G$: Graph $n$ vertices.

(B) $(s_1, t_1), \ldots, (s_t, t_t)$: pairs of vertices

(C) $\pi_i, \sigma_i$: two different paths connecting $s_i$ to $t_i$

(D) $\widehat{w}$: Fractional congestion at least $n^{1/2}$.

(E) opt: Congestion of optimal solution.

(F) $\implies$ In polynomial time (LP solving time) choose paths

(A) congestion $\forall$ edges: $\leq (1 + \delta)\text{opt}$

(B) $\delta = \sqrt{\frac{20}{\widehat{w}} \ln t}$.

### 21.3.0.9 When the congestion is low

(A) Assume $\widehat{w}$ is a constant.

(B) Can get a better bound by using the Chernoff inequality in its more general form.

(C) set $\delta = c \ln t / \ln \ln t$, where $c$ is a constant. For $\mu = \alpha_e$, we have that

$$\Pr\left[Y_e \geq (1 + \delta)\mu\right] \leq \left(\frac{e^\delta}{(1 + \delta)^{1+\delta}}\right)^{\mu}$$

$$\leq \exp\left(\mu(\delta - (1 + \delta)\ln(1 + \delta))\right)$$

$$= \exp\left(-\mu c' \ln t\right) \leq \frac{1}{t^{O(1)}},$$

where $c'$ is a constant that depends on $c$ and grows if $c$ grows.
21.3.0.10  When the congestion is low

(A) Just proved that...
(B) if the optimal congestion is $O(1)$, then...
(C) algorithm outputs a solution with congestion $O(\log t / \log \log t)$, and this holds with high probability.

21.4  Reminder about Chernoff inequality

21.4.1  The Chernoff Bound — General Case

21.4.1.1  Chernoff inequality

Problem 21.4.1. Let $X_1, \ldots, X_n$ be $n$ independent Bernoulli trials, where

$$\Pr[X_i = 1] = p_i, \quad \Pr[X_i = 0] = 1 - p_i,$$

$$Y = \sum_i X_i, \quad \text{and} \quad \mu = \mathbb{E}[Y].$$

We are interested in bounding the probability that $Y \geq (1 + \delta)\mu$.

21.4.1.2  Chernoff inequality

Theorem 21.4.2 (Chernoff inequality). For any $\delta > 0$,

$$\Pr\left[Y > (1 + \delta)\mu\right] < \left(\frac{e^{\delta}}{(1 + \delta)^{1+\delta}}\right)^\mu.$$

Or in a more simplified form, for any $\delta \leq 2e - 1$,

$$\Pr\left[Y > (1 + \delta)\mu\right] < \exp(-\mu\delta^2/4),$$

and

$$\Pr\left[Y > (1 + \delta)\mu\right] < 2^{-\mu(1+\delta)},$$

for $\delta \geq 2e - 1$.

21.4.1.3  More Chernoff...

Theorem 21.4.3. Under the same assumptions as the theorem above, we have

$$\Pr\left[Y < (1 - \delta)\mu\right] \leq \exp\left(-\mu \frac{\delta^2}{2}\right).$$