LP-based approximation algorithms

- Rounding LP-solutions
- Proving approx. guarantees via LP-duality (dual solutions)

* Find "suitable" LP-formulation.
  Solve LP-relax. in poly. time. 
  "Round" fractional solution to "good" feas. one.

Simple rounding

Example: Weighted Vertex Cover:

Given an undirected graph $G = (V,E)$ with nonnegative node weights $\omega : V \rightarrow \mathbb{Q}^+$, find a minimum weight vertex cover, i.e., a set $V' \subseteq V$ such that any edge $e \in E$ has at least one endpoint incident at $V'$.

(LP) \quad \text{with } \quad x_v = \begin{cases} 1 & \text{if } v \text{ is in } V' \\ 0 & \text{otherwise} \end{cases} \quad \forall v \in V

\begin{align*}
\min & \quad \sum_{v \in V} \omega_v x_v \\
\text{s.t.} & \quad x_u + x_v \geq 1 \quad \forall (u,v) \in E \\
& \quad x_v \in \{0,1\} \quad \forall v \in V
\end{align*} \quad (1)

\begin{align*}
\text{(LP)} \quad \min & \quad \sum_{v \in V} \omega_v x_v \\
\text{s.t.} & \quad (1) \\
& \quad x_v \geq 0 \quad \forall v \in V
\end{align*} \quad (2)

Simple rounding alg.

- Let $x^{LP}$ be opt. sol. of (LP).
- Round $x^{LP}$ to $x$:
  \[ x_v = \begin{cases} 1 & \text{if } x^{LP} \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \]
Thus, the simple rounding algo. is a \( 2 \)-approximation.

**Pf.** (a) \( x \) is feasible for (LP), i.e., it is a vertex cover.

Why? Consider some edge \( e = (u, v) \in E \).
\[
x_u^* + x_v^* \geq 1 = \Rightarrow x_u^* \geq 0.5 \quad \text{as} \quad x_u^* \geq 0.5
\]
\[
\Rightarrow x_u = 1 \quad \text{or} \quad x_v = 1 \quad = \text{edge} \ (u, v) \text{ is covered.}
\]

(b) algorithms cost: \( \text{Alg} \leq 2 \cdot x^* \)

Why? \( x_v \leq 2 \cdot x_v^* \quad \forall \ v \in V \)
\[
\Rightarrow \text{Alg} = \sum_{v \in V} w_v x_v \leq 2 \sum_{v \in V} w_v x_v^* = 2 \cdot x^*.
\]

\( \equiv \) Since \( x^* \leq \text{OPT} \), algo. is a \( 2 \)-approx.

and close look at the (LP) gives

Thus, any basic solution to (LP) is \underline{half-integral}, i.e., \( x^* \in \{0, \frac{1}{2}, 1\} \), \( \forall \ v \in V \).

**Pf.** Recall, a feasible solution is basic if it cannot be expressed as a convex comb. of two other feas. solutions.

Consider a feas. solution \( x \). Let
\[
V_+ := \{v \in V \mid \frac{1}{2} \leq x_v < 1\}
\]
\[
V_- := \{v \in V \mid 0 < x_v \leq \frac{1}{2}\}.
\]

Gains new solutions \( y \) and \( z \) for \( \varepsilon > 0 \):
\[
y_v := \begin{cases} x_v + \varepsilon & \text{if} & x_v \in V_+, \\ x_v - \varepsilon & \text{if} & x_v \in V_-, \\ x_v & \text{otherwise}. \end{cases}
\]
\[
z_v := \begin{cases} x_v - \varepsilon & \text{if} & x_v \in V_+, \\ x_v + \varepsilon & \text{if} & x_v \in V_-, \\ x_v & \text{otherwise}. \end{cases}
\]

We can choose \( \varepsilon \) s.t. \( y \) and \( z \) are feasible solutions to (LP).
(\( \varepsilon \leq \min_{V \in V} x_v \) for \( x_v \neq 0 \)) Clearly, \( y, z \neq x \).

But \( \frac{1}{2} y + \frac{1}{2} z = x \). \( \equiv \) \( x \) not a basic feas. solution!

[[Turnhause, Trotter '73]] All extreme points of VC polytope are \underline{half-integral}.

Thus, the algo. that solves (LP) and rounds all \( x_v^* \in \{\frac{1}{2}, 1\} \) to \( x_v = 1 \) is a \( 2 \)-approx. for. min. vertex cover.
Randomized rounding

Idea: Interpret fractional LP values \( x_j \) as probability with which \( x_j \) is set to 1.

Example: MAX SAT

- \( m \) clauses \( C_1, \ldots, C_m \) and \( n \) boolean variables \( v_j, j = 1, \ldots, n \)
- \( \Phi = \left( \frac{x_{11} \lor x_{12}}{C_1} \right) \land \left( \frac{x_{21} \lor x_{22} \lor x_{23}}{C_2} \right) \land \cdots \land \left( \frac{x_{m1} \lor x_{m2} \lor \cdots}{C_m} \right) \)

- \( x_j \in \{v, \overline{v}\} \) literals, \( v \in \{v_1, v_2, \ldots, v_n\} \) boolean variables
- each clause \( C_i \) has non-negative weight \( w_i \).

Task: Find a truth assignment to the boolean variables such that the total weight of satisfied clauses is maximized.

Notation:

- Let \( k(i) \) denote the number of literals in clause \( C_i \).
  
  Note: - problem with \( k(i) \leq k \), \( i = 1, \ldots, m \), also called Max-KSAT.
  
  - Max-2SAT is NP-hard, while 2SAT is in P.

- Let \( W \) be a random variable that denotes the total weight of satisfied clauses in a randomized truth assignment.

- Let \( W_i \) be the weight contributed by \( C_i \).

1.) A trivial algorithm (no LPs) \[ \text{Johnson '74} \]

Alg: set every boolean var. \( v_j \) to TRUE independent with prob. \( \frac{1}{2} \).

Then: Alg yields a 2-approx. for MAX-SAT.

Lemma: \[ \mathbb{E}[W_i] = \left(1 - \frac{1}{2^{k(i)}}\right) w_i \]

Pf: \( C_i \) satisfied if not all literals are FALSE.

Probability for this event:

\[ \text{Prob}[C_i \text{ satisfied}] = 1 - \frac{1}{2^{k(i)}}. \]
Proof of Theorem:

For \( k \geq 1 \):
\[
1 - \frac{1}{2^{k+1}} \geq 1 - \frac{1}{2^k}
\]

From Lemma, then follows:

\[
ECW_J = \sum_{i=1}^{m} ECW_i \geq \left(1 - \frac{1}{2^k}\right) \sum_{i=1}^{m} w_i \geq \left(1 - \frac{1}{2^k}\right) \text{OPT} \geq \frac{1}{2} \text{OPT}.
\]

2. Randomized rounding of LP solutions

0/1 vars. \( y_j \) for bool. vars. \( y_j \), \( j = 1, \ldots, n \)

0/1 vars. \( z_i \) for clause \( C_i \), \( i = 1, \ldots, m \)

For any clause \( C_i \), let \( f^+(i) \) \( f^-(i) \) denote the set of indices of non-negated (negated) variables in \( C_i \).

(1LP)
\[
\text{max} \quad \sum_{i=1}^{m} w_i z_i
\]
\[
\text{s.t.} \quad \sum_{j \in f^+(i)} y_j + \sum_{j \in f^-(i)} (1 - y_j) \geq z_i, \quad i = 1, \ldots, m \quad (1)
\]
\[
z_i \in \{0, 1\}, \quad i = 1, \ldots, m \quad (2)
\]
\[
y_j \in \{0, 1\}, \quad j = 1, \ldots, n \quad (3)
\]

(LP)
\[
\text{max} \quad \sum_{i=1}^{m} w_i z_i
\]
\[
\text{s.t.} \quad (1)
\]
\[
0 \leq z_i \leq 1, \quad i = 1, \ldots, m
\]
\[
0 \leq y_j \leq 1, \quad j = 1, \ldots, n
\]

Alg.

* Solve LP relaxation \( (y_{\text{lp}}, z_{\text{lp}}) \)
* Set \( y_j \) to TRUE \((=1)\) with probab. \( y_{j, \text{lp}} \) indep. \( j = 1, \ldots, n \)

Thus. Alg. yields an approx. factor of \( \frac{e}{e-1} \approx 1.57 \) for MAX-SAT.

[Seppänen, Williamson '93]
Lemma \[ EC[W_i] \geq \left( 1 - (1 - \frac{1}{k(i)})^k \right) w_i \cdot z_i \]

**Pf.**
- wlog. all literals in \( C_i \) appear non-negated. (otherwise replace \( v_j \) by \( v_j \) in all clauses; no effect on \( w_i \) or \( z_i \).)
- Rename variables s.t. \( C_i = (v_{a} \lor v_{b} \lor \ldots \lor v_{w(i)}) \)
  
\( C_i \) satisfied if not all \( v_{a}, v_{b}, \ldots, v_{w(i)} \) are set to \( \text{FALSE} \).

\[ \Pr[ C_i \text{ satisfied}] = 1 - \left( 1 - \frac{k(i)}{\sum_{j=1}^{k(i)} (1 - y_{j})} \right)^k \]
\[ \geq 1 - \left( 1 - \frac{\sum_{j=1}^{k(i)} (1 - y_{j})}{k(i)} \right)^k \]
\[ \geq 1 - \left( 1 - \frac{z_i}{k(i)} \right)^k \]

Now \( g(z) = 1 - (1 - \frac{z}{k})^k \) is a concave function in \( z \) with \( g(0) = 0 \) and \( g(1) = 1 - (1 - \frac{1}{k})^k \).

And \( f(z) = \left( 1 - (1 - \frac{1}{k})^k \right)^z \) is a binomial function in \( z \) with \( f(0) = g(0) \) and \( f(1) = g(1) \).

\[ g(z) \geq f(z) \text{ on interval } [0,1] \text{ since } g(z) \geq f(z) \text{ at } \text{endpoints of interval} \]

\[ \Pr[ C_i \text{ satisfied}] \geq \left( 1 - (1 - \frac{1}{k(i)})^k \right) \cdot z_i \cdot \omega_i \]

**Pf. (of Thm).**

\[ EC[W] = \sum_{i=1}^{n} EC[W_i] \geq \sum_{i=1}^{n} \left( 1 - (1 - \frac{1}{k(i)})^k \right) z_i \cdot \omega_i \]

\[ \geq \left( 1 - (1 - \frac{1}{k})^k \right) \sum_{i=1}^{n} z_i \cdot \omega_i = \left( 1 - (1 - \frac{1}{k})^k \right) \cdot \text{OPT} \]

\[ \geq \left( 1 - (1 - \frac{1}{k})^k \right) \cdot \frac{e-1}{e} \cdot \text{OPT} \]
3.) A combined algorithm.

Alg. Flip a fair coin to decide which of the two previous algorithms to run.

Remark: - Actually we set variable \( y_i \) to TRUE with probab. \( \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} \).
- but they are not set independently any more.

Then. The combined alg. is a \( \frac{4}{3} \) -approximation for MAX-SAT.

Pf. Prob. \([C_i \text{ satisfied}] \geq \frac{1}{2} \left( 1 - \frac{1}{2^{k(i)}} \right) + \frac{1}{2} \left( 1 - \left( 1 - \frac{1}{k(i)} \right)^{\ell(i)} \right) \zeta_{i}^{\text{up}} \)

\[ \geq \frac{1}{2} \left( 1 - \frac{1}{2^{k(i)}} \right) \zeta_{i}^{\text{up}} \quad \text{since } \zeta_{i}^{\text{up}} \leq 1 \]

\[ \geq f(k(i)) \cdot \zeta_{i}^{\text{up}} \quad \text{with } f(k(i)) := \frac{1}{2} \left( 1 - \frac{1}{2^{k(i)}} \right) + \frac{1}{2} \left( 1 - \left( 1 - \frac{1}{k(i)} \right) \right) \]

analyze function \( f(k(i)) \):

- \( f(1) = f(2) = \frac{3}{4} \)
- \( f(k(i)) \geq \frac{1}{2} \cdot \frac{3}{8} + \frac{1}{2} \cdot (1 - \frac{1}{2}) \geq \frac{3}{4} \)

\( \implies \quad E[W] = \sum_{i=1}^{\infty} E[W_i] = \sum_{i=1}^{\infty} f(k(i)) \cdot \zeta_{i}^{\text{up}} \cdot \omega_i \)

\[ \geq \frac{3}{4} \cdot \zeta^{\text{up}} \geq \frac{3}{4} \cdot \text{OPT.} \]

\[ \forall \lambda \in \mathbb{Z}^+ \quad \left( 1 - \frac{1}{\lambda} \right)^k \leq \frac{1}{e} \]

\[ 1 + x \leq e^x, \quad \forall x \in \mathbb{R} \]

\[ x = -\frac{1}{\lambda k} \leq -\frac{1}{k} \]

\[ x = 1 - \frac{1}{\lambda k} \leq e^{-\frac{1}{k}} \]

\[ \left( 1 - \frac{1}{\lambda} \right)^k \leq \frac{1}{e} \]