Apr 11, 2023

Vertex Cover: A set S of vertices such that for every edge (u,v) in the graph, u is in S or v is in S.

Finding minimum size vertex cover is NP-hard.

Can we design an approximation algorithm for this problem?

2-approximation: Given a graph, output a vertex cover whose size is at most twice of the minimum vertex cover.

Algorithm 1

While V is non-empty

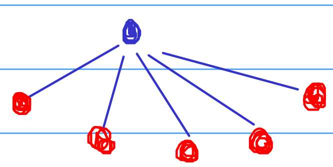
{

- choose any vertex $v \in V$ and put it in S
- Delete vertex v and all its incident edges

Delete all isolated vertices

}

Bad example



Algorithm 2

While V is non-empty

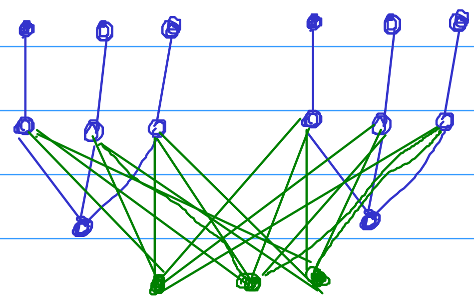
{ choose highest degree vertex $v \in V$ and
put it in S

· Delete vertex v and all its incident edges

Delete all isolated vertices

}

Bad example



Algorithm 3

While V is non-empty

{ choose an edge (u, v) and
put both u and v in S

Delete vertices u and v
and all their incident edges
Delete all isolated vertices

}

Gives a 2- approximation algorithm.

Proof for 2-approximation: consider all the edges chosen during the algorithm. Note that they are disjoint, that is no two of them share a common vertex (matching).

If k edges are chosen during the algorithm then that means there are k disjoint edges, and hence we need at least k vertices in any vertex cover. While the algorithm will output a vertex cover of size 2k.

Hence, this is a 2-approximation algorithm.

Weighted vertex cover

Given weights on vertices,

find a vertex cover of minimum weight.

Any variants of the greedy algorithms fail to give a good approximation for the minimum weight vertex cover problem.

Linear Programming

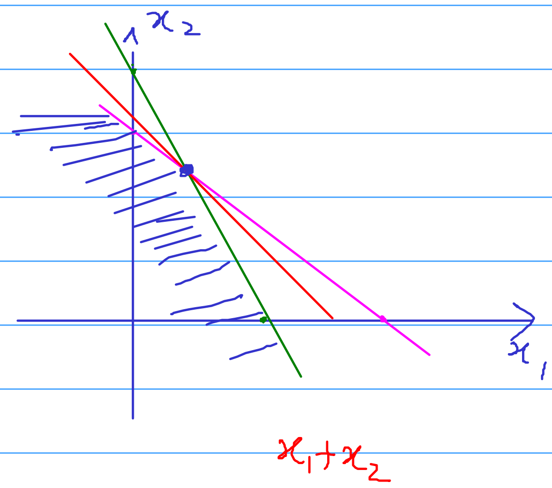
→ linear objective function
→ linear constraints

subject to

$2x_1 + x_2 \le 4$

$3x_1 + 4x_2 \le 12$

Max $x_1 + x_2$



- Optimal solutions are at the boundary.
- There is always a corner point which is optimal.

Integer Linear Programming (ILP)

- Linear program where some/all variables are restricted to be integers

Algorithms for linear programming

- Simplex method (efficient in practice)
- Ellipsoid method (poly time)
- Interior point methods (poly time + efficient in practice)

- Integer linear programming is NP-hard.

Use of linear programming to solve discrete problems.

For example minimum weight vertex cover

For a graph $G(V,E)$ with weights $\{w_v\}$ on vertices.

$$\left.\begin{array}{ll} \text{for each } v \in V & x_v \in \{0,1\} \\ \text{for each edge } (u,v): & x_u + x_v \geq 1 \\ & \text{Min } \sum_{v \in V} w_v x_v \end{array}\right\} \text{ ILP}$$

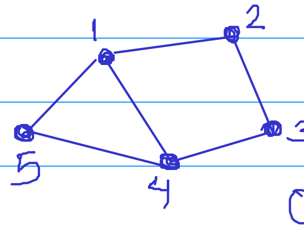
The ILP optimal value will be exactly equal to the weight of the min weight vertex cover.

Can we get a linear program?

$$\left.\begin{array}{ll} \text{for each } v \in V & 0 \leq x_v \leq 1 \\ \text{for each edge } (u,v): & x_u + x_v \geq 1 \\ & \text{Min } \sum_{v \in V} w_v x_v \end{array}\right\} \text{ LP}$$

Now, the LP optimal value may be different from the minimum weight of a vertex cover.

Example



All vertices have weight 1.

$$0 \leq x_1, x_2, x_3, x_4, x_5 \leq 1$$

min weight vertex cover = 3

$x_1 + x_5 \geq 1$ $\quad x_1 + x_4 \geq 1$

$x_1 + x_2 \geq 1$ $\quad x_2 + x_3 \geq 1$

$x_4 + x_5 \geq 1$ $\quad x_3 + x_4 \geq 1$

LP opt = 2.5

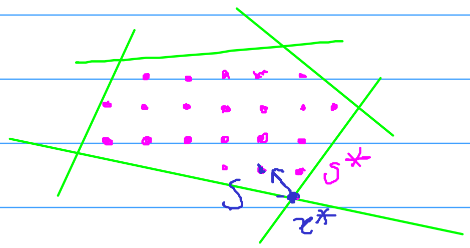
$x_1^* = x_2^* = x_3^* = x_4^* = x_5^* = 1/2$

$\min\ (x_1 + x_2 + x_3 + x_4 + x_5)$

Nonetheless, the above LP can help us design an approximation algorithm for minimum weight vertex cover.

Algorithm outline

1. Find an optimal solution x^* for the above LP
2. Apply rounding on x^* to obtain an integral solution



Rounding Scheme

$x^* \in \mathbb{R}^V$

Construct $S \subseteq V$ s.t. if $x^*_v \geq 1/2$ then put $v \in S$
otherwise don't put v in S

We need to show:

1. S is a vertex cover
2. S is an approximately optimal vertex cover

Proof 1: Show that for any edge (u,v) either $u \in S$ or $v \in S$

$\iff$ either $x^*_u \geq 1/2$ or $x^*_v \geq 1/2$

$\Leftarrow$ $x^*_u + x^*_v \geq 1$

Let's say S^* is the optimal vertex cover.

$$w(S) \leq \alpha\, w(S^*) \qquad \text{for some } \alpha \geq 1$$

Obs: $\sum_u w_u x_u^* \leq w(S^*)$

proof

$$y_u = \begin{cases} 1 & u \in S^* \\ 0 & u \notin S^* \end{cases}$$

$$w(S^*) = \sum_u w_u y_u$$

y is a feasible point for LP

Hence, $\sum w_u x_u^* \leq \sum w_u y_u = w(S^*)$

Claim: $w(S) \leq 2 \sum_u w_u x_u^*$

Proof

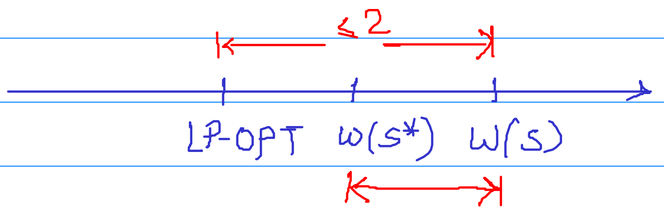
$$w(S) = \sum_{u \in S} w_u \cdot 1 \leq 2 \sum_{u \in S} w_u x_u^* \leq 2 \sum_{u \in V} w_u x_u^*$$

$1 \leq 2 \cdot x_u^*$ whenever $u \in S$

Summary

$$w(S) \leq 2 \cdot \text{LP-OPT} \leq 2\, w(S^*)$$

2-approximate solution.



There is no efficient algorithm known for better than 2-approximation for min weight vertex cover.

Special Case when LP optimal correctly gives the size of the minimum vertex cover.

Bipartite Graphs.

$$\min \sum_{u \in V} x_u$$

$$\text{s.t.} \quad 0 \leq x_u \leq 1 \quad \text{for } u \in V$$

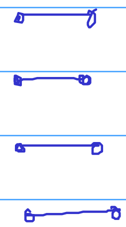
$$x_u + x_v \geq 1 \quad \text{for edge } (u,v) \in E$$

for any matching M,
and for any vertex cover S

$$|S| \geq |M|$$

Claim 1 For any matching M,

$$\text{LP-OPT} \geq |M|$$

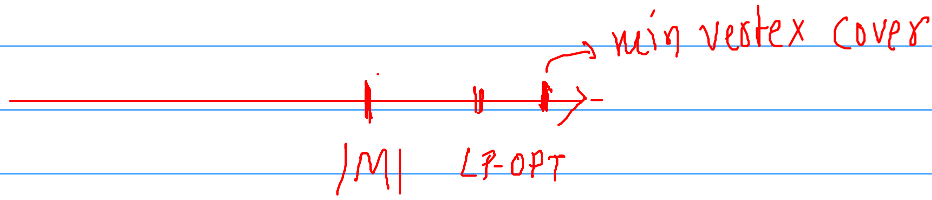


for any edge $(u,v) \in M$ $\quad x_u + x_v \geq 1$

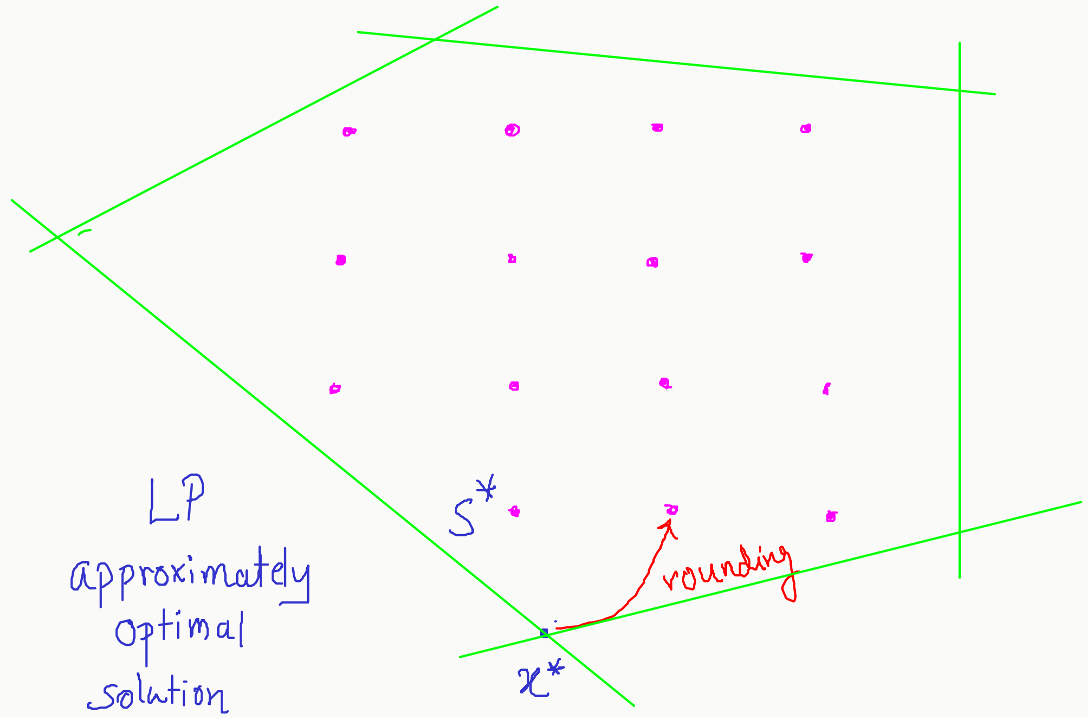
$$\sum_{u \in V} x_u \geq \sum_{u \text{ is in } M} x_u \geq |M|$$

Claim 2 size of min vertex cover = size of maximum matching

Homework using Max flow min cut



$$\Rightarrow \text{LP-OPT} = \text{min vertex cover}$$


LP
approximately
optimal
solution
S^*
x^*
rounding