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

1. Choose any vertex $v \in V$ and put it in $S$
2. Delete vertex $v$ and all its incident edges
3. Delete all isolated vertices

*Bad example*
Algorithm 2

While V is non-empty

\{ choose highest-degree vertex v ∈ 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

\[ 2x_1 + x_2 \leq 4 \]
\[ 3x_1 + 4x_2 \leq 12 \]

\[ \text{Max } x_1 + x_2 \]

• 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,

for each $v \in V$, $x_v \in \{0,1\}$

for each edge $(u,v)$, $x_u + x_v \geq 1$

\[ \text{Min} \sum_{v \in V} w_v x_v \]

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

Can we get a linear program?

for each $v \in V$, $0 \leq x_v \leq 1$

for each edge $(u,v)$, $x_u + x_v \geq 1$

\[ \text{Min} \sum_{v \in V} w_v x_v \]

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

LP opt = 2.5

\[ x_1^* + x_2^* = x_4^* = x_5^* = \frac{1}{2} \]

\[ \text{Min} (x_1 + x_2 + x_3 + x_4 + x_5) \]
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**

\[ x^* \in \{0,1\}^V \]

Construct \( S \subseteq V \) such that if \( x^*_v \geq \frac{1}{2} \) 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 \( v \in S \)

\[ \iff \text{either } x^*_u \geq \frac{1}{2} \text{ or } x^*_v \geq \frac{1}{2} \]

\[ \implies x^*_u + x^*_v \geq 1 \]
Let's say $S^*$ is the optimal vertex cover.

$$w(s) \leq \alpha w(s^*) \quad \text{for some } \alpha \geq 1$$

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

**Proof**

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

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

$y$ is a feasible point for LP

Hence, $\sum_u w_u x_u^* \leq \sum_u w_u y_u = w(s^*)$

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

**Proof**

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

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

**Summary**

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

2-approximate solution.

There is no efficient algorithm known for better than 2-approximation for min weight vertex cover.
Special case when $L^p$ optimal correctly gives the size of the minimum vertex cover.

Bipartite Graphs

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

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

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

for any matching $M$,

and for any vertex cover $S$

$|S| \geq |M|$

Claim 1 For any matching $M$,

$L^P - OPT \geq |M|$

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

$\sum_{u \in V} x_u \geq \sum_{u \in S, u \in M} x_u \geq |M|$

Claim 2 Size of min vertex cover $= \frac{\text{Size of maximum matching}}{\text{Homework}}$

Max flow min cut

$\Rightarrow L^P - OPT = \text{min vertex cover}$
LP approximately optimal solution