APPROXIMATION ALGOS:

1) LP BASED APPROXIMATION ALGOS

   a) Vertex Cover

   b) 3-WAY CUT

(NEXT) 2) SDP BASED APPROXIMATION ALGOS

MINIMUM VERTEX COVER

INPUT: GRAPH $G = (V, E)$

DEF: A set $S \subseteq V$ is a vertex cover if $\forall$ edge (u,v) "S covers (u,v)"

$\Updownarrow$

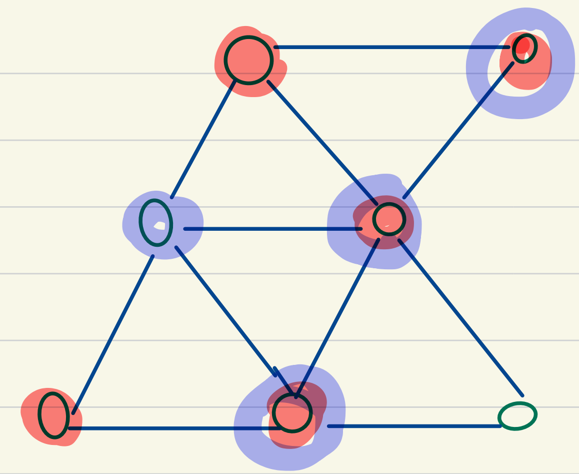
at least one of $u, v \in S$.

SOL:

GOAL: Find a vertex cover of smallest size.

REMARK: NP-hard.

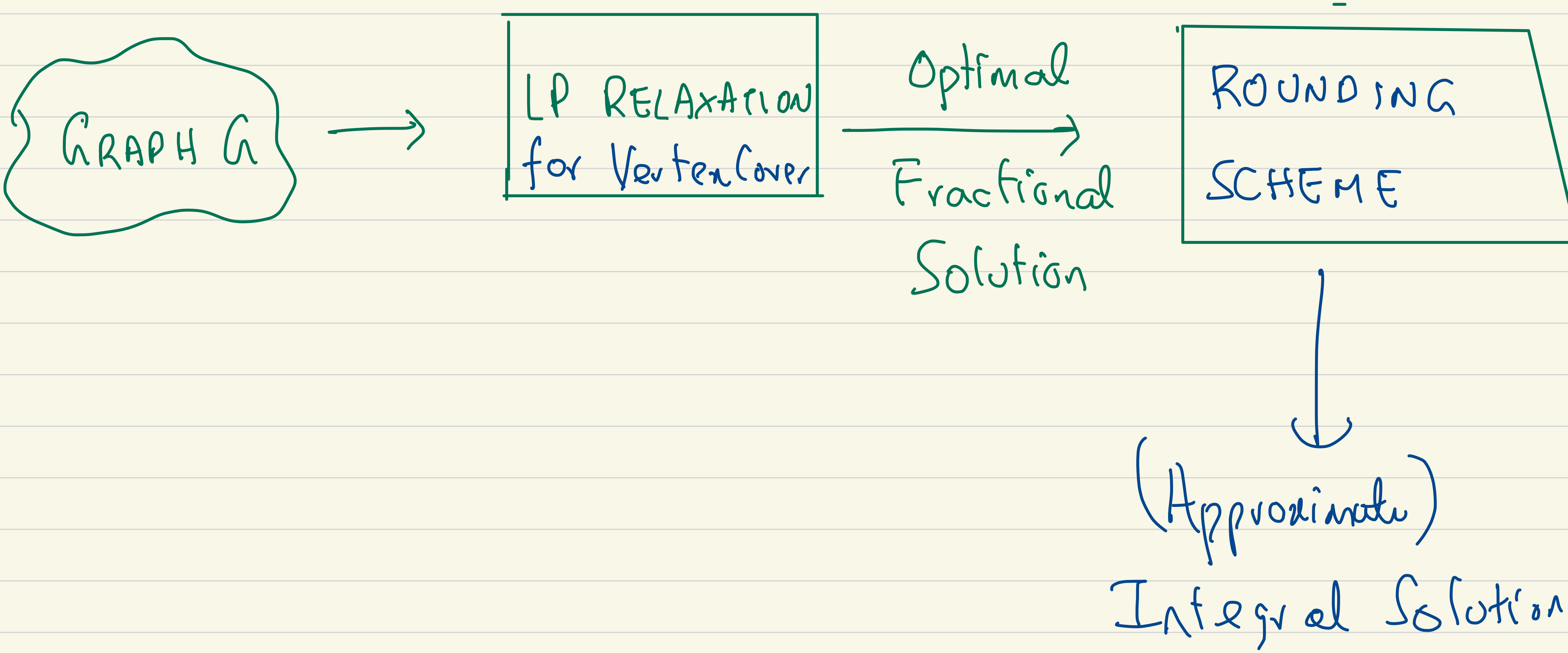
A 2-approximation algorithm



S_1 = covers every edge

S_2

LP- Based Approximation Algorithm



LINEAR PROGRAM:

Variables:

For each vertex i, x_i

$$x_i = \begin{cases} 1 & \text{if } i \in \text{Vertex Cover} \\ 0 & \text{otherwise} \end{cases}$$

MINIMUM VERTEX COVER

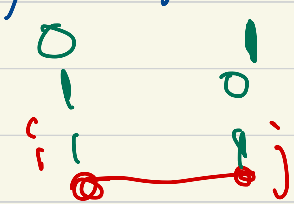
INPUT: Graph $G = (V, E)$

SOL: Smallest Vertex Cover S s.t. $\forall (u,v) \in E$, $u \in S$ or $v \in S$.

Minimize $\sum_{i=1}^{n} x_i$ = Total size of vertex cover

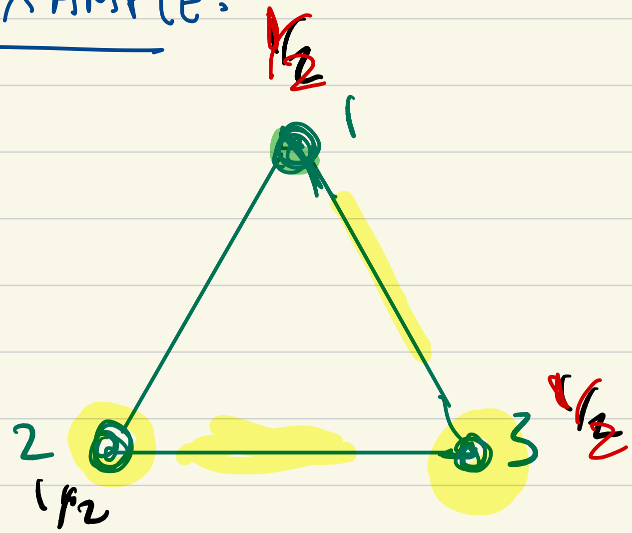
$0 \leq x_i \leq 1$ for each i

$x_i + x_j \geq 1$ for every edge $(i,j) \in E$



Fractional LP-Optima:

EXAMPLE:



Optimal Vertex Cover = 2

$$\text{Min } x_1 + x_2 + x_3$$

$$x_1 + x_2 \geq 1$$

$$x_2 + x_3 \geq 1$$

$$x_3 + x_1 \geq 1$$

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

$$x_1 = x_2 = x_3 = \underline{\underline{0.5}}$$

LP-OPT = 1.5 $\ll$ OPT = 2

Observation: For all graphs G (Minimum Vertex Cover)

$$\text{LP-OPT}(G) \leq \text{OPT}(G).$$

$\text{LP-OPT}(G)$ = Best Solution among ALL (integer/fractional)

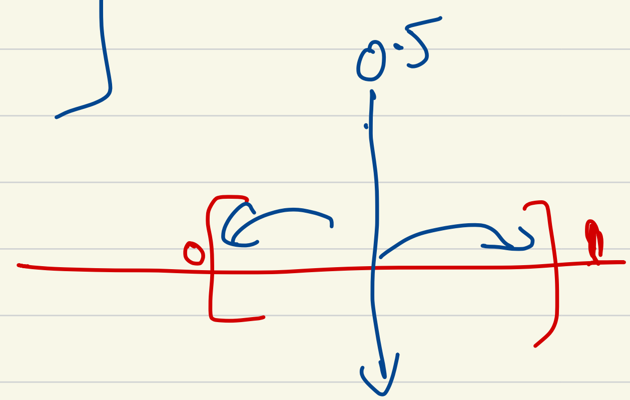
$\text{OPT}(G)$ = Best solution among all integer solutions

ROUNDING ALGORITHM:

Let x^* be the optimal LP-solution

$$x_i^* \in [0,1] \quad \forall i$$

$$S = \{\, i \mid x_i^* \geq 0.5 \,\}$$



Lemma 1: S is a valid vertex cover.

PROOF: For every edge (i, j)

$$x_i^* + x_j^* \geq 1$$

$\Rightarrow$ Either $x_i^* \geq 0.5$ or $x_j^* \geq 0.5$ or BOTH

$\Rightarrow$ Either $i \in S$ or $j \in S$ or BOTH,

$\Rightarrow$ Every edge (i,j) is covered by S.

$$S = \{ i \mid x_i^* \geq 1/2 \}$$

Claim: $|S| \leq 2 \cdot \text{LP-OPT}$

PROOF:

$$|S| \leq 2 \cdot \sum_i x_i^*$$

Consider any vertex $i \in S$

LHS: to $|S|$ ← it contributes 1.

RHS: $2x_i^* \geq 1$ because $x_i^* \geq 1/2$ for all $i \in S$,

$$|S| = \sum_{i=1}^{n} \mathbb{1}[i \in S]$$

For each i,

$$\mathbb{1}[i \in S] \leq 2 x_i^*$$

because $i \in S \iff x_i^* \geq 1/2$

$$\Rightarrow \leq 2 \cdot \sum_{i=1}^{n} x_i^* = 2 \cdot \text{LP-OPT}$$

ALG OUTPUT $\leq 2 \cdot$ (Cost of LP)

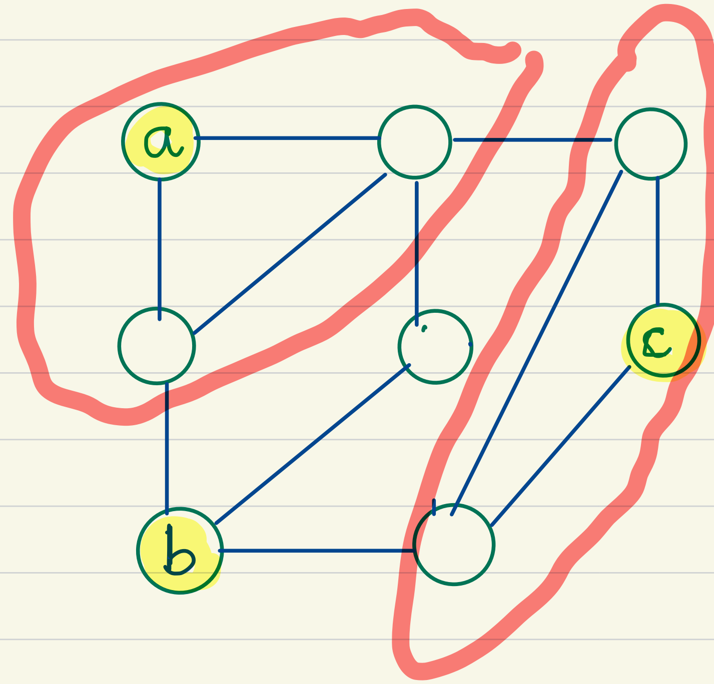
$\leq 2 \cdot$ (Cost of Optimal Sol)

MINIMUM 3-WAY CUT

INPUT: 1) Graph $G = (V, E)$

2) 3 vertices $a, b, c \in V$.

GOAL: Separate a, b, c by cutting fewest number of edges

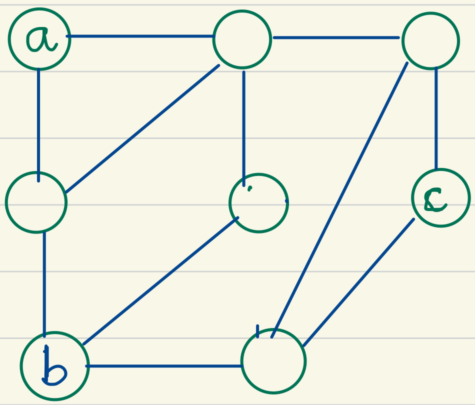


Cut = 4

MINIMUM 2-WAY CUT $\in P$

MINIMUM 3-WAY CUT is NP-hard.

LP-Based Approximation Algo



LP Relaxation
for 3-way cut

Optimal
Fractional
Solution

Rounding Algo

Approximate
3-way
cut

LP RELAXATION for 3-WAY CUT

Decision Variables: $\forall$ vertex v $\to$ Component 1, Component 2, Component 3

Vertex

$$v \longrightarrow (v_1, v_2, v_3)$$

$$\boxed{\begin{aligned} v \to 1 &\iff (v_1, v_2, v_3) = (1, 0, 0) \\ v \to 2 &\iff (v_1, v_2, v_3) = (0, 1, 0) \\ v \to 3 &\iff (v_1, v_2, v_3) = (0, 0, 1) \end{aligned}}$$

LP variables: $\forall$ vertex v, v_1, v_2, v_3

$v_i = 1$ if vertex $v \in i$

0 otherwise

Constraints: $0 \leq v_1, v_2, v_3 \leq 1$ $\forall$ vertex v.

$v_1 + v_2 + v_3 = 1$ $\forall$ vertex v.

For vertex a $(a_1, a_2, a_3) = (1, 0, 0)$

vertex b $(b_1, b_2, b_3) = (0, 1, 0)$

c $(c_1, c_2, c_3) = (0, 0, 1)$

Objective Function:

$$\#\text{ of edges cut} = \sum_{(u,v)\in E} \mathbb{1}[(u,v)\text{ is cut}]$$

Minimize: $\frac{1}{2} \sum_{(u,v)\in E} \left(|u_1 - v_1| + |u_2 - v_2| + |u_3 - v_3| \right)$

NOT linear because of $|\;|$!!

[Can be made into LP]

$u \to (u_1, u_2, u_3) \to \begin{matrix}(1,0,0)\\(0,1,0)\\(0,0,1)\end{matrix}$

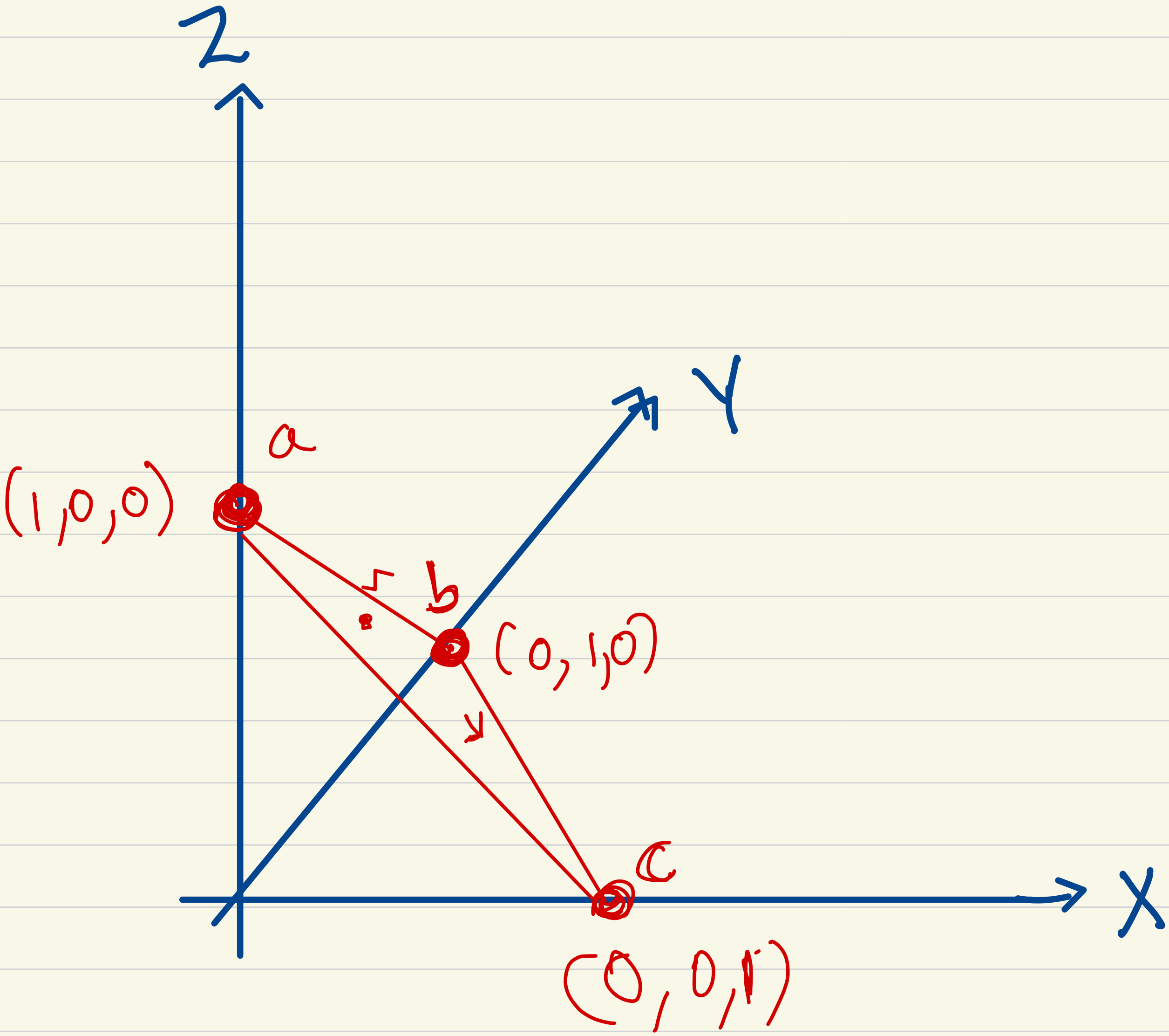
$v \to (v_1, v_2, v_3)$

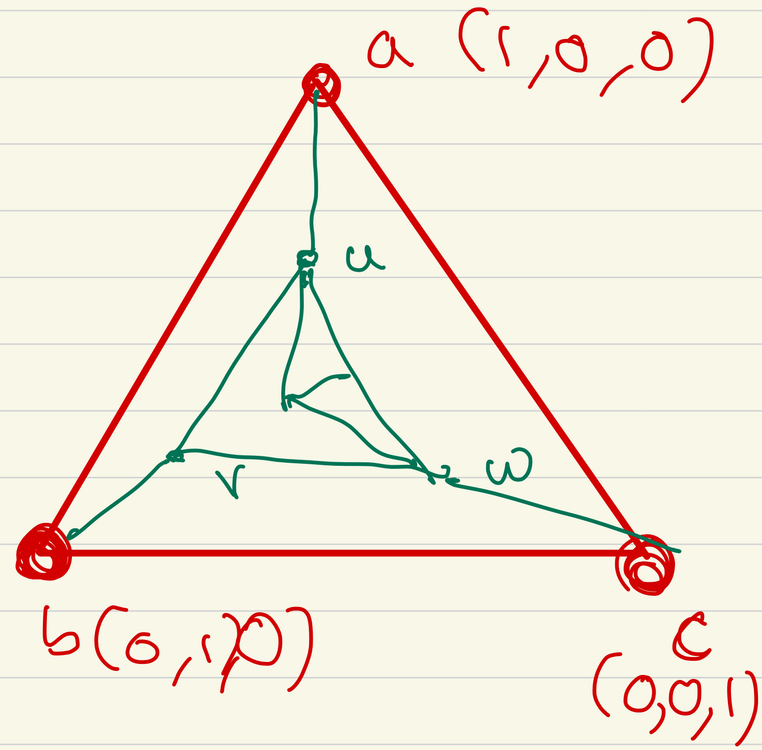
$$\mathbb{1}[(u,v) \text{ is cut}] = \frac{1}{2}\left(\overset{1}{|u_1} - \overset{0}{v_1}| + |u_2 - v_2| + |u_3 - v_3|\right)$$

$u = (1, 0, 0)$

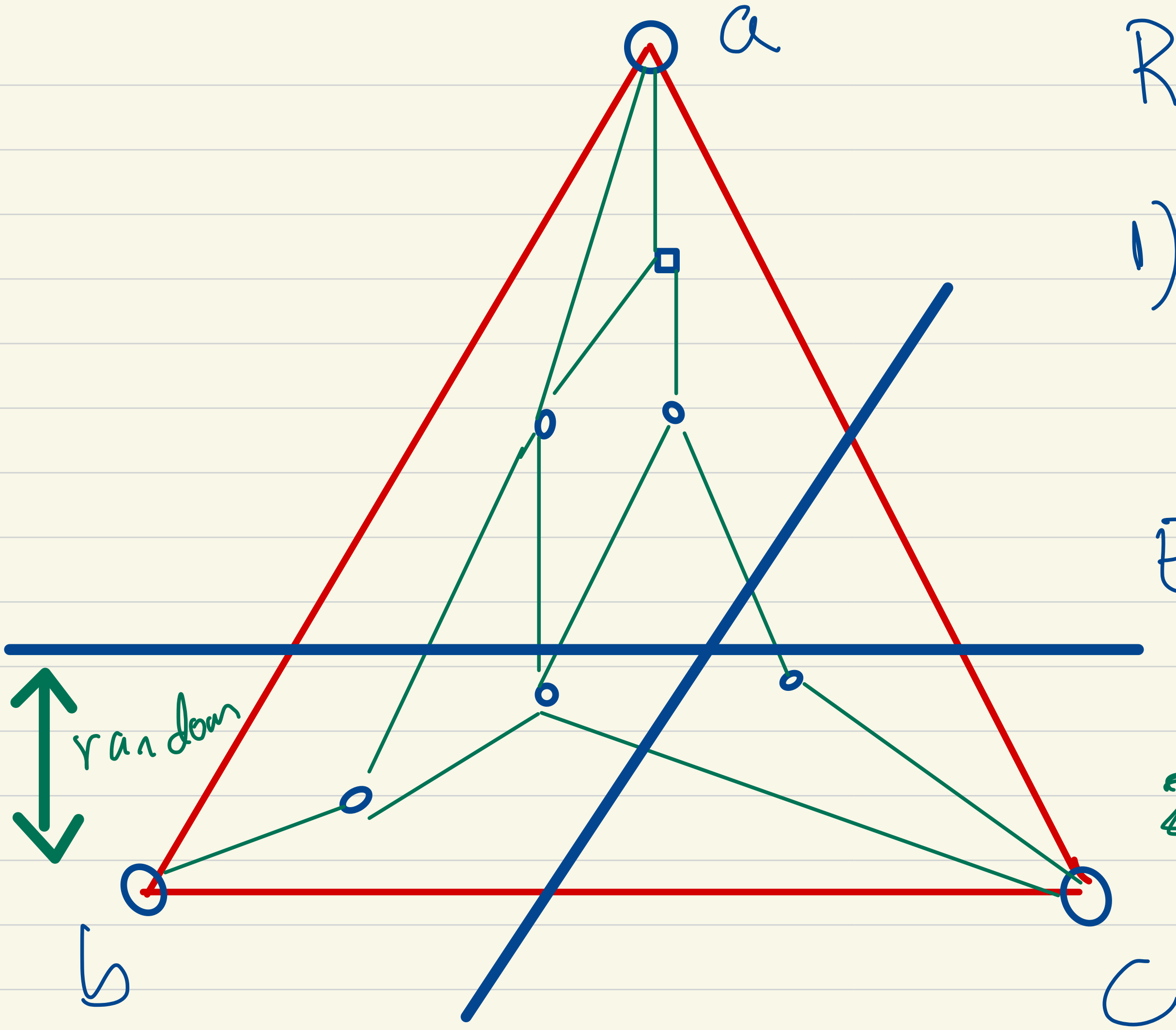
$v = (0, 1, 0)$

$u - v = (1, -1, 0) \Rightarrow 2$





$$\forall u \quad \begin{cases} u_1 + u_2 + u_3 = 1 \\ 0 \le u_1, u_2, u_3 \le 1 \end{cases}$$



ROUNDING SCHEME

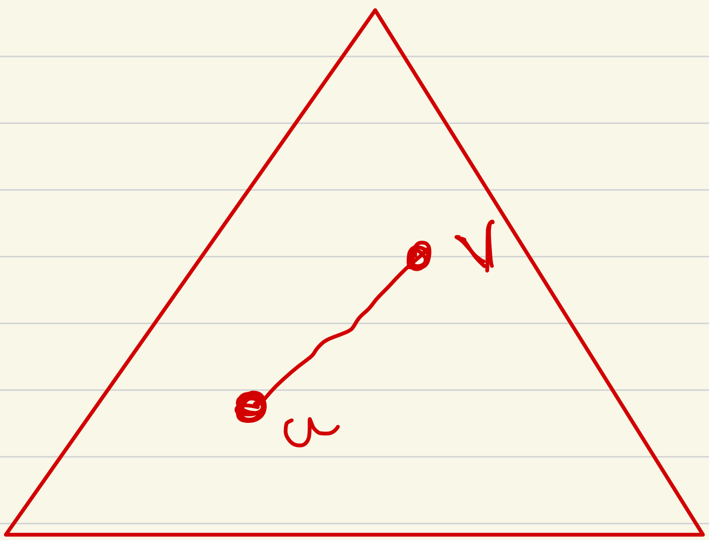
1) Pick 2 out of 3 directions of Δ

Ex: (a, b) AND (b, c)

2) Cut the graph by lines parallel to directions at RANDOM HEIGHT.

Claim: $\Pr[\text{edge } (u,v) \text{ is cut}] = \frac{2}{3} \|\vec{u}-\vec{v}\|_1$

$$= \frac{2}{3}\Big[|u_1-v_1| + |u_2-v_2| + |u_3-v_3|\Big]$$



$$= \frac{4}{3} \cdot \left(\frac{1}{2}\Big[|u_1-v_1| + |u_2-v_2| + |u_3-v_3|\Big]\right)$$

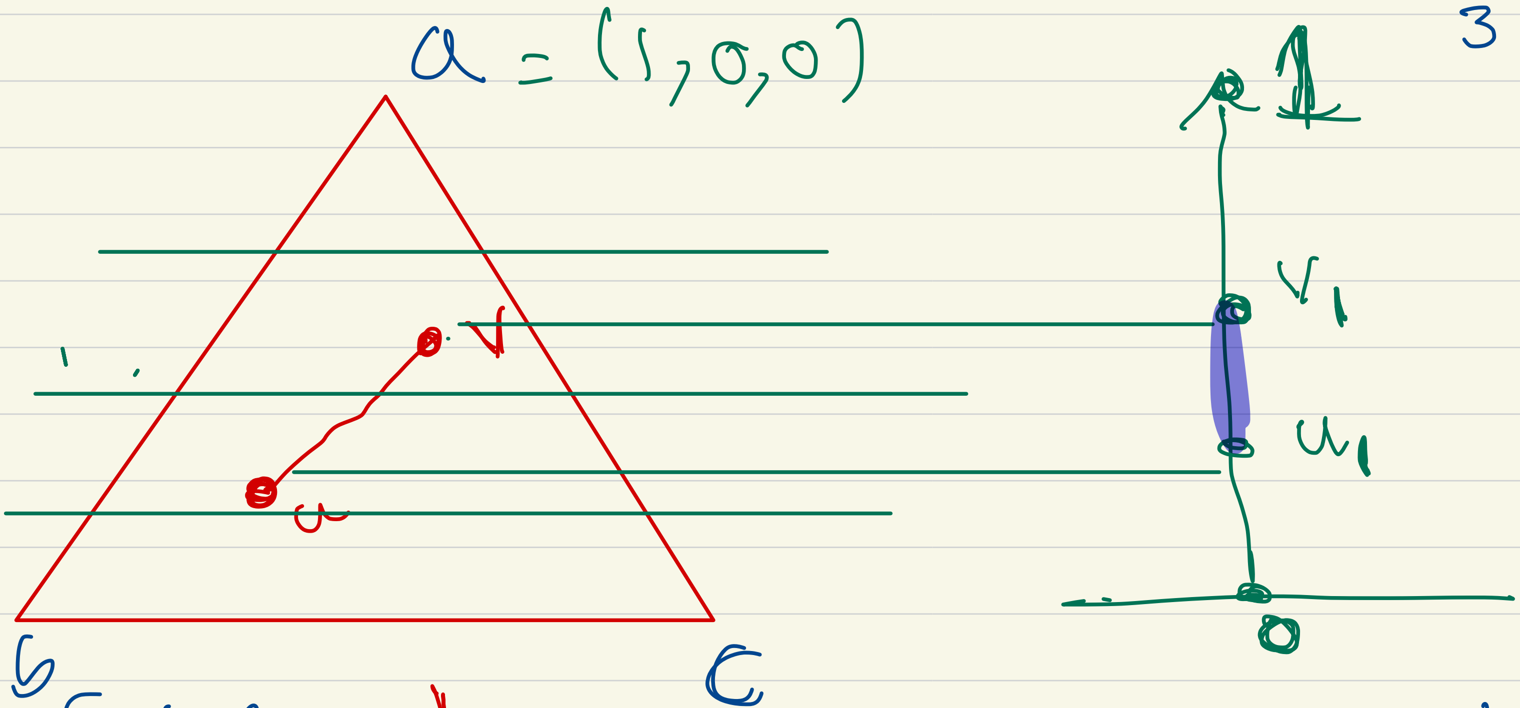
$$\mathbb{E}[\text{\# of edges cut}] = \sum_{(u,v)\in E} \Pr[(u,v) \text{ is cut}]$$

$$= 4/3 \cdot \text{LP-OPT}.$$

$$\leq 4/3 \ \text{OPT}$$

Claim: $\Pr[\text{edge } (u,v) \text{ is cut}] = \frac{2}{3} \|\vec{u} - \vec{v}\|_1$

$$= \frac{2}{3}\left[|u_1 - v_1| + |u_2 - v_2| + |u_3 - v_3| \right]$$



Sub Claim 1: For a random cut parallel to edge (b, c)

$$\Pr[(u,v) \text{ is cut}] = |u_1 - v_1|$$

$$\Pr\left[\begin{array}{c}(u, v) \text{ is cut} \\ \text{by a single} \\ \text{line}\end{array}\right] = \Pr\left(\begin{array}{c}\text{cutting line} \\ \text{lands between } u \,\&\, v\end{array}\right)$$

$$= |u_1 - v_1|$$

$$\Pr\left(\begin{array}{c}[(u, v) \text{ is cut}] \\ \text{in rounding} \\ \text{scheme}\end{array}\right) = \frac{2}{3}\Big(|u_1 - v_1| + |u_2 - v_2| + |u_3 - v_3|\Big)$$

$$= 4/3 \left[\underbrace{\frac{1}{2}\Big(|u_1 - v_1| + |u_2 - v_2| + |u_3 - v_3|\Big)}\right]$$

$$E\left[(u,v) \text{ is cut}\right] = \frac{1}{3} \underbrace{E\left[(u,v) \text{ is cut} \;\middle|\; \begin{matrix}\text{direction}\\ 1,2\end{matrix}\right]}$$

$$+ \frac{1}{3} E\left[(u,v) \text{ is cut} \;\middle|\; \begin{matrix}\text{direction}\\ 2,3\end{matrix}\right]$$

$$+ \frac{1}{3} E\left[(u,v) \text{ is cut} \;\middle|\; \begin{matrix}\text{direction}\\ 1,3\end{matrix}\right]$$

$$= \frac{1}{3}\left(|u_1 - v_1| + |u_2 - v_2|\right) + \frac{1}{3}\left(|u_2 - v_2| + |u_3 - v_3|\right)$$

$$+ \frac{1}{3}\left(|u_3 - v_3| + |u_1 - v_1|\right)$$