24 April 2024

Approximation Algorithms
Max Cut

Plan

* Max Cut
* Announcements
* Approximation Algorithms
  * → Greedy
  * → Random.

Maximum Cut Problem

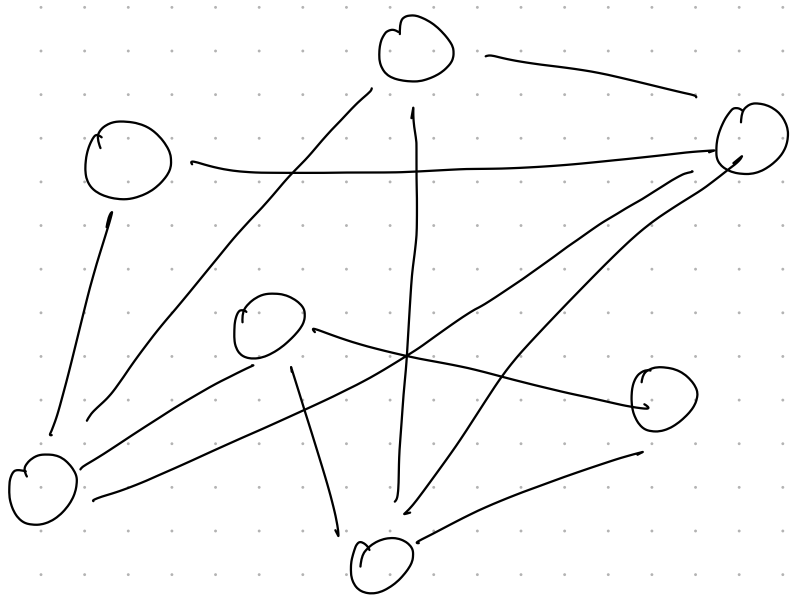
Given: Undirected Graph $G = (V, E)$

Find: Cut $(S, V \setminus S)$ maximizing

$$\delta(S) = \#\text{ edges crossing } (S, V \setminus S)$$

$E(S, V \setminus S)$

$E(S_0, S_1)$

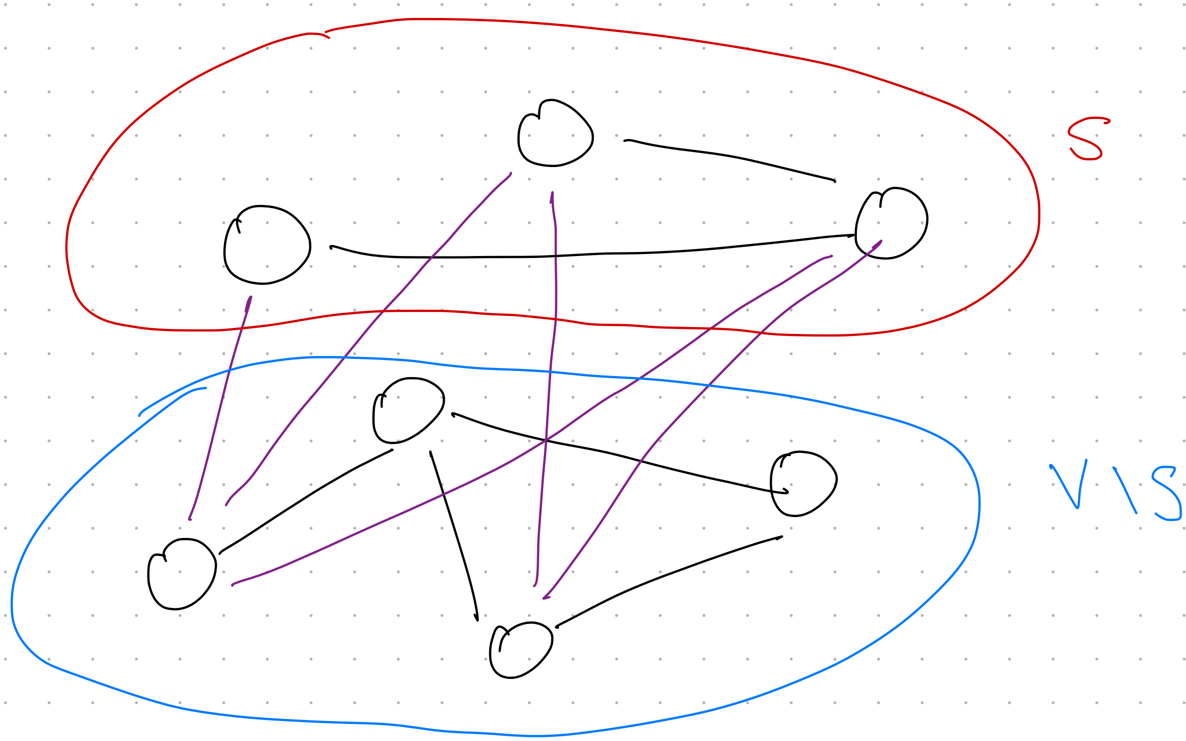


Maximum Cut Problem

Given. Undirected Graph $G = (V, E)$

Find. Cut $(S, V \setminus S)$ maximizing

$\delta(S) =$ # edges crossing $(S, V \setminus S)$



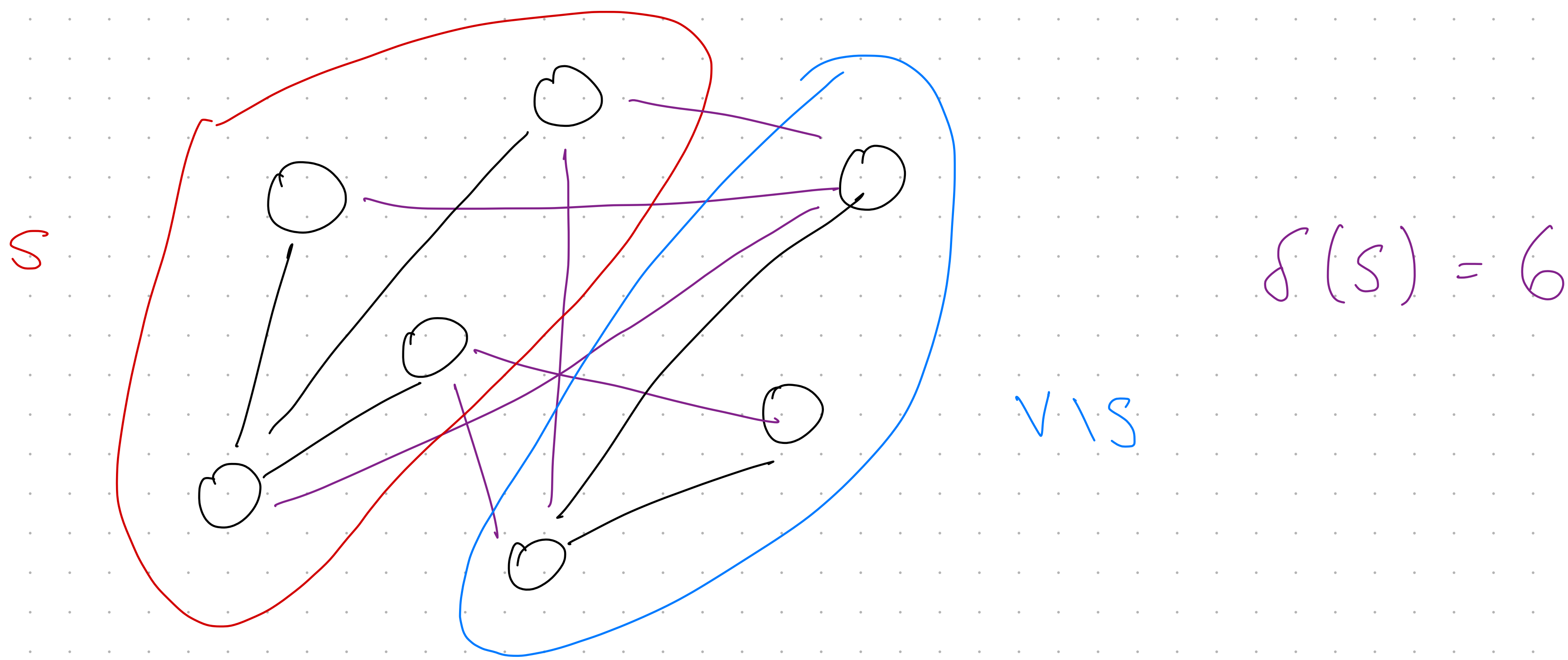
$\delta(S) = 5$

Maximum Cut Problem

Given: Undirected Graph $G = (V, E)$

Find: Cut $(S, V \setminus S)$ maximizing

$$\delta(S) = \#\text{ edges crossing } (S, V \setminus S)$$



Theorem. MAX CUT is NP-Hard.

↳ Even NP-Hard to determine the value of the max cut.

So, no efficient algorithm to solve Max Cut.

What about an "almost" max cut?

Announcements.

* Keep working on HW8

* HW9 will be released Friday
  - → Final homework of semester.

Approximate Maximum Cut Problem

Given. Undirected Graph $G = (V, E)$

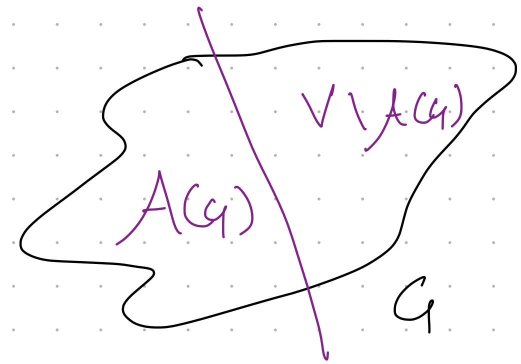
Find. Cut $(S, V \setminus S)$ approximately maximizing

$$\delta(S) = \#\text{ edges crossing } (S, V \setminus S)$$

Q: Can we develop an efficient! algorithm that provably achieves an "almost" max cut?

Consider an algorithm $\mathcal{A}$ for solving MaxCut.

$\mathcal{A}(G)$ is some cut $(S, V \setminus S)$

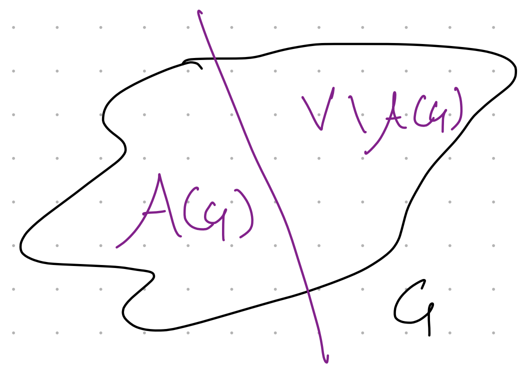


$$\boxed{\delta(\mathcal{A}(G))}$$

How does cut returned by $\mathcal{A}$ compare to optimal?

Consider an algorithm $\mathcal{A}$ for solving MaxCut.

$\mathcal{A}(G)$ is some cut $(S, V \setminus S)$



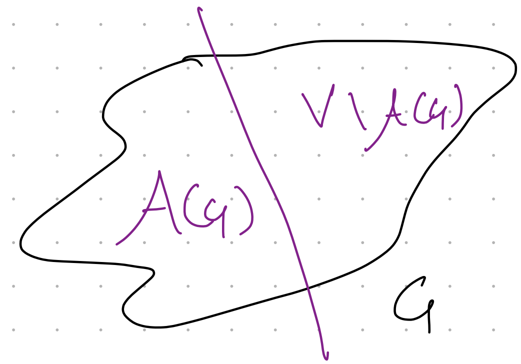
$$\boxed{\delta(\mathcal{A}(G))}$$

How does cut returned by $\mathcal{A}$ compare to optimal?

$$\boxed{\Delta^*(G) = \max_{S \subseteq V} \delta(S)}$$

Consider an algorithm $\mathcal{A}$ for solving MaxCut.

$\mathcal{A}(G)$ is some cut $(S, V \setminus S)$



$$\boxed{\delta(\mathcal{A}(G))}$$

How does cut returned by $\mathcal{A}$ compare to optimal?

$$\boxed{\Delta^*(G) = \max_{S \subseteq V} \delta(S)}$$

Approximation Ratio of Algorithm $\mathcal{A}$

$$\boxed{r_{\mathcal{A}} = \min_{G} \frac{\delta(\mathcal{A}(G))}{\Delta^*(G)}}$$

r-Approximate Maximum Cut Problem

Given. Undirected Graph $G = (V, E)$

Find. Cut $(S, V \setminus S)$

$$\text{s.t.} \quad \frac{\delta(S)}{\Delta^*(G)} \geq r$$

r-Approximate Maximum Cut Problem

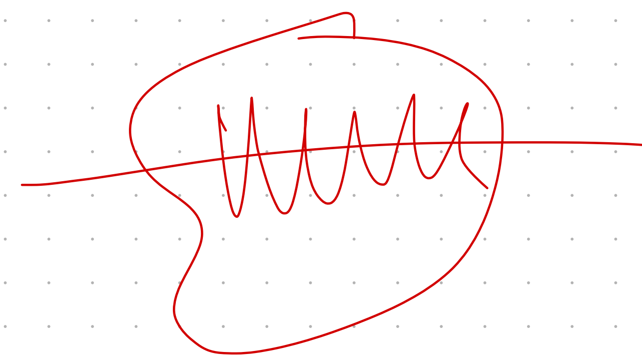
Given: Undirected Graph $G = (V, E)$

Find: Cut $(S, V \setminus S)$

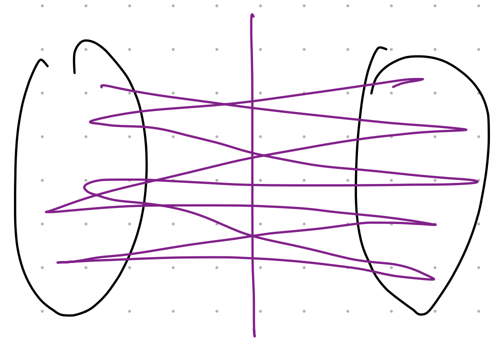
$$\text{s.t.} \quad \frac{\delta(S)}{\Delta^*(G)} \geq r$$

Useful Observation

Max Cut at most # edges



$$\Delta^*(G) \leq m$$



$\delta(S) = m$

S $\qquad$ $V \setminus S$

Realized by Bipartite Graphs.

r-Approximate Maximum Cut Problem

Given. Undirected Graph $G = (V, E)$

Find. Cut $(S, V \setminus S)$

$$\text{s.t.} \quad r \leq \frac{\delta(S)}{\Delta^*(G)}$$

Useful Observation

Max Cut at most # edges

$$\Delta^*(G) \leq m$$

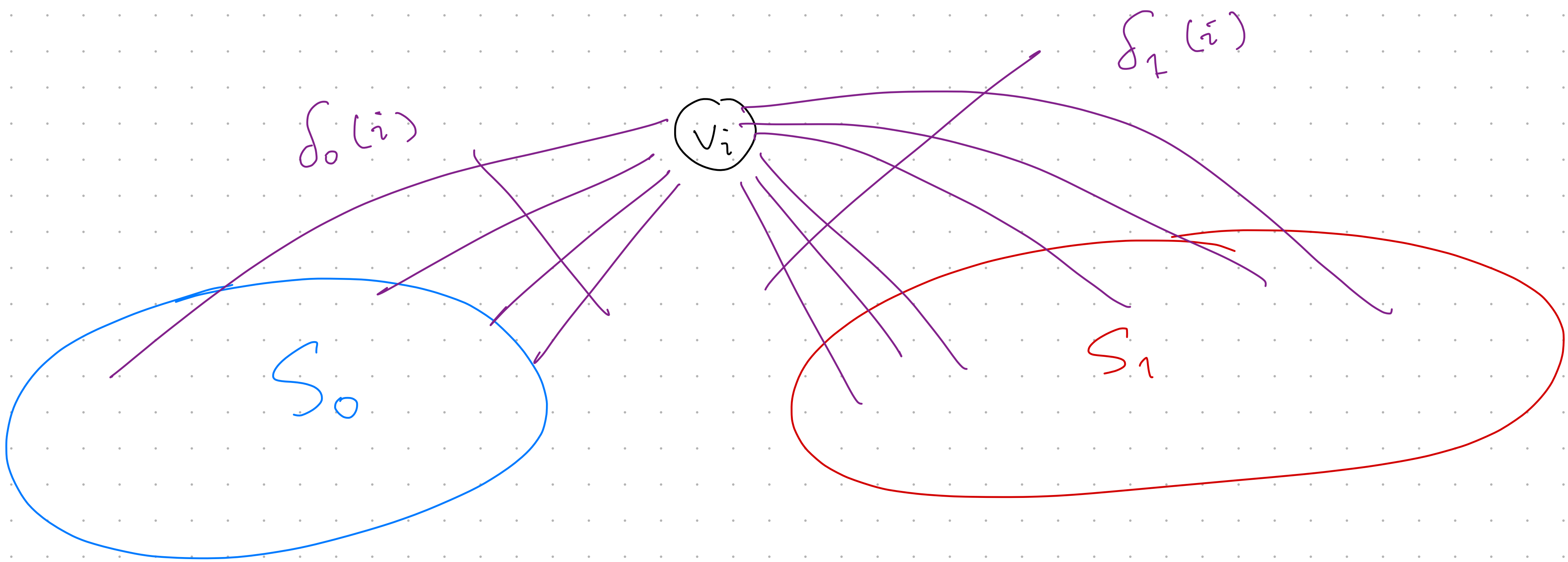
$$\Rightarrow \frac{\delta(S)}{m} \leq \frac{\delta(S)}{\Delta^*(G)}$$

To lower bound r we lower bound $\delta(S)/m$.

Greedy Max Cut.

Initialize $S_0 = \emptyset$, $S_1 = \emptyset$.

For each vertex v_i



$\delta_b(i)$ = # edges cut if v_i placed in S_{1-b}

Greedy Max Cut.

Initialize $S_0 = \emptyset$, $S_1 = \emptyset$.

For $i = 1 \ldots n$.

$$\delta_0(i) \leftarrow |\{(v_i, u_0) : u_0 \in S_0\}|$$

$$\delta_1(i) \leftarrow |\{(v_i, u_1) : u_1 \in S_1\}|$$

if $\delta_0(i) > \delta_1(i)$. // Vertex v_i has more edges to S_0 than to S_1

Add v_i to S_1

else. // Add v_i to S_0 greedily.

Add v_i to S_0

Return (S_0, S_1)

Theorem. Greedy Max Cut solves the $\frac{1}{2}$-approximate Max Cut problem.

Theorem. Greedy Max Cut solves the $\frac{1}{2}$-approximate Max Cut problem.

For every graph G, Greedy Max Cut guarantees that $\delta(S)$ is at least 50% of $\Delta^*(G)$

Theorem. Greedy Max Cut solves the $\frac{1}{2}$-approximate Max Cut problem.

For every graph G, Greedy Max Cut guarantees that $\delta(S)$ is at least 50% of $\Delta^*(G)$

Analysis

of cut edges "stays ahead" of non-cut edges

↳ Greedy Max Cut guarantees 50% of $|E|$.

Greedy MaxCut

For $i = 1 \ldots n$

if v_i has more edges into S_0, add to S_1

else, add to S_0

For $i = 1 \ldots n$, let $V_i = \{v_1 \ldots v_i\}$

Let E_i be the set of edges

with both endpoints in V_i

Greedy MaxCut

For $i = 1 \ldots n$

if v_i has more edges into S_0, add to S_1

else, add to S_0

For $i = 1 \ldots n$, let $V_i = \{v_1 \ldots v_i\}$

Let E_i be the set of edges with both endpoints in V_i

Claim. For all $i = 1 \ldots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

Claim. For all $i = 1 \ldots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

Proof. By induction on i.

Base Case $i=1$. $|E_1| = 0$.

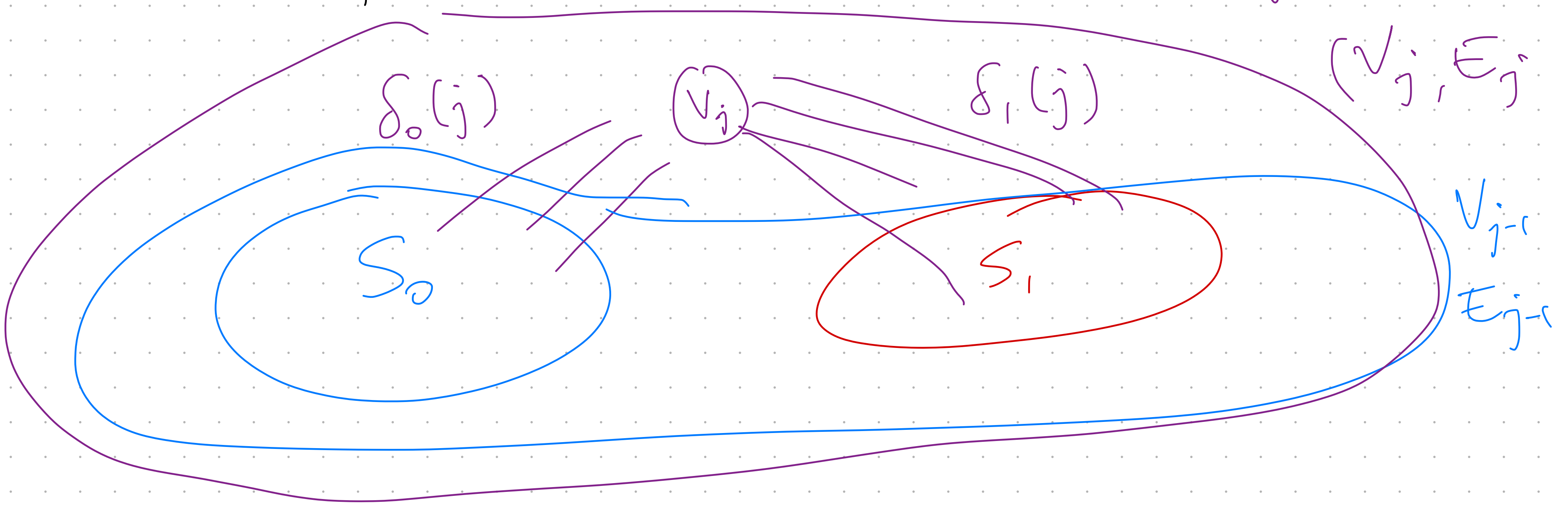
<u>Claim</u>. For all $i = 1 \ldots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

<u>Inductive Step</u>. Before iteration j

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_{j-1}|$$

How many edges does v_j add to E_j ?



Claim: For all $i = 1 \dots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

Inductive Step: Before iteration j

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_{j-1}|$$

How many edges does v_j add to E_j?

$$\delta_0(j) + \delta_1(j)$$

Claim: For all $i = 1 \ldots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

Inductive Step: Before iteration j

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_{j-1}|$$

How many edges does v_j add to E_j?

$$\delta_0(j) + \delta_1(j)$$

How many edges does v_j add to $E(S_0, S_1)$?

$$\max\{\delta_0(j), \delta_1(j)\}$$

Claim: For all $i = 1 \ldots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

Inductive Step: Before iteration j

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_{j-1}|$$

After iteration j: $+ \max\{\delta_0(j), \delta_1(j)\}$ $\quad \frac{1}{2}\left(\delta_0(j) + \delta_1(j)\right)$

Claim. For all $i = 1 \dots n$, after the i^{th} iteration of Greedy MaxCut

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_i|$$

Inductive Step. Before iteration j

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_{j-1}|$$

After iteration j: $+ \max\{\delta_0(j), \delta_1(j)\}$ $\quad \frac{1}{2}(\delta_0(j) + \delta_1(j))$

Note. $\max\{\delta_0(j), \delta_1(j)\} \geq \frac{1}{2} \cdot (\delta_0(j) + \delta_1(j))$

$$\Rightarrow \quad |E(S_0, S_1)| \geq \frac{1}{2} \cdot |E_j|$$

Conclusion. After iteration n,

$$|E(S_0, S_1)| \geq \frac{1}{2} \cdot m.$$

$$\Rightarrow \quad r_{greedy} = \frac{|E(S_0, S_1)|}{\Delta^*(G)} \geq \frac{|E(S_0, S_1)|}{m} \geq \frac{\frac{1}{2} m}{m}$$

$$= \frac{1}{2}$$

Random Max Cut

Initialize $S_0 = \emptyset$ $S_1 = \emptyset$

For $i = 1 \ldots n$.

Choose $b \in \{0, 1\}$ uniformly at random

Add v_i to S_b

Return (S_0, S_1)

Note. Algorithm Never looks at the edges!

Random Max Cut

Initialize $S_0 = \phi$ $S_1 = \phi$

For $i = 1 \ldots n$.

Choose $b \in \{0,1\}$ uniformly at random

Add v_i to S_b

Return (S_0, S_1)

Note. Algorithm Never looks at the edges!

Claim. $\mathbb{E}\left[|E(S_0, S_1)| \right] \geq \frac{1}{2} \cdot m$.

$\Rightarrow$ Random Max Cut has Approx Ratio of $\frac{1}{2}$ in expectation.

Proof. Consider an edge $(u,v) \in E$.

Define $X_{uv} = \mathbb{1}\left[(u,v) \text{ cut by } (S_0, S_1)\right]$

What is expected $|E(S_0, S_1)|$?

$$\mathbb{E}\left[\sum_{uv \in E} X_{uv}\right] = \sum_{uv \in E} \underbrace{\mathbb{E}[X_{uv}]}$$

Proof. Consider an edge $(u,v) \in E$.

Define $X_{uv} = \mathbb{1}[(u,v) \text{ cut by } (S_0, S_1)]$

What is expected $|E(S_0, S_1)|$?

$$\mathbb{E}\left[\sum_{uv \in E} X_{uv}\right] = \sum_{uv \in E} \underbrace{\mathbb{E}[X_{uv}]}$$

$$\mathbb{E}[X_{uv}] = \Pr[u \in S_0 \wedge v \in S_1] + \Pr[u \in S_1 \wedge v \in S_0]$$

$$= 2 \cdot \left(\frac{1}{2}\right)^2$$

$$= \frac{1}{2}$$

Proof. Consider an edge $(u,v) \in E$.

Define $X_{uv} = \mathbb{1}[(u,v) \text{ cut by } (S_0, S_1)]$

What is expected $|E(S_0, S_1)|$?

$$\mathbb{E}\left[\sum_{uv \in E} X_{uv}\right] = \sum_{uv \in E} \underbrace{\mathbb{E}[X_{uv}]}$$

$$= \frac{1}{2} \cdot m$$

Can we do better?

(Goemans - $\underbrace{\text{Williamson}}_{\text{Cornell Prof.}}$)

YES!

$r_{GW} \geq \underbrace{0.878}$

$$\min_{\theta} \frac{2}{\pi} \left(\frac{\theta}{1 - \cos\theta} \right)$$

Can we do better?

(Goemans - $\underbrace{\text{Williamson}}_{\text{Cornell Prof.}}$) YES!

$r_{GW} \geq \underbrace{0.878}$

$$\min_{\theta} \frac{2}{\pi}\left(\frac{\theta}{1-\cos\theta}\right)$$

Can we do better than GW?

(Khot, Kindler, Mossel, O'donnell)
(Mossel, O'donnell, Oleszkiewicz)
(Raghavendra)

NO!

Unless $\underbrace{(P \neq NP)^{++}}_{\text{Unique Games Conjecture.}}$ is wrong.