

Approximation Algorithms

Lecture 10:

MINIMUM-DEGREE SPANNING TREE
via Local Search

Part I:

MINIMUM-DEGREE SPANNING TREE

MINIMUM-DEGREE SPANNING TREE

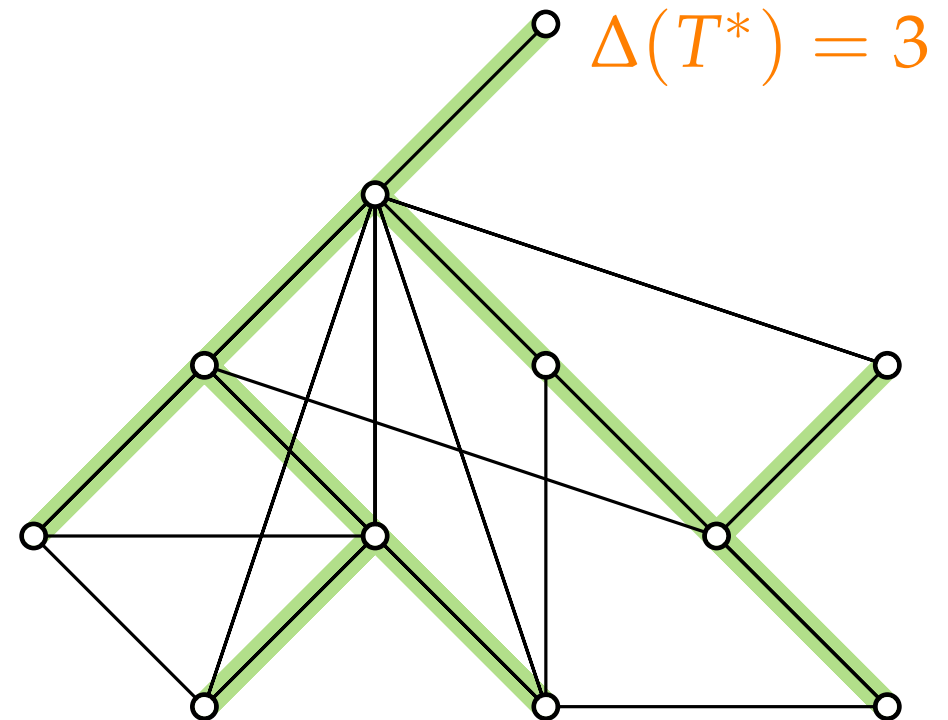
Given: A connected graph G .

Task: Find a **spanning tree** T that has the smallest maximum degree $\Delta(T)$ among all spanning trees of G .

NP-hard. 😞

Why?

Special case of
Hamiltonian Path!



Warm-up

Obs.

A spanning tree T has...

- n vertices and $n - 1$ edges,
- sum of degrees $\sum_{v \in V} \deg_T(v) = 2n - 2$,
- average degree < 2 .

Obs.

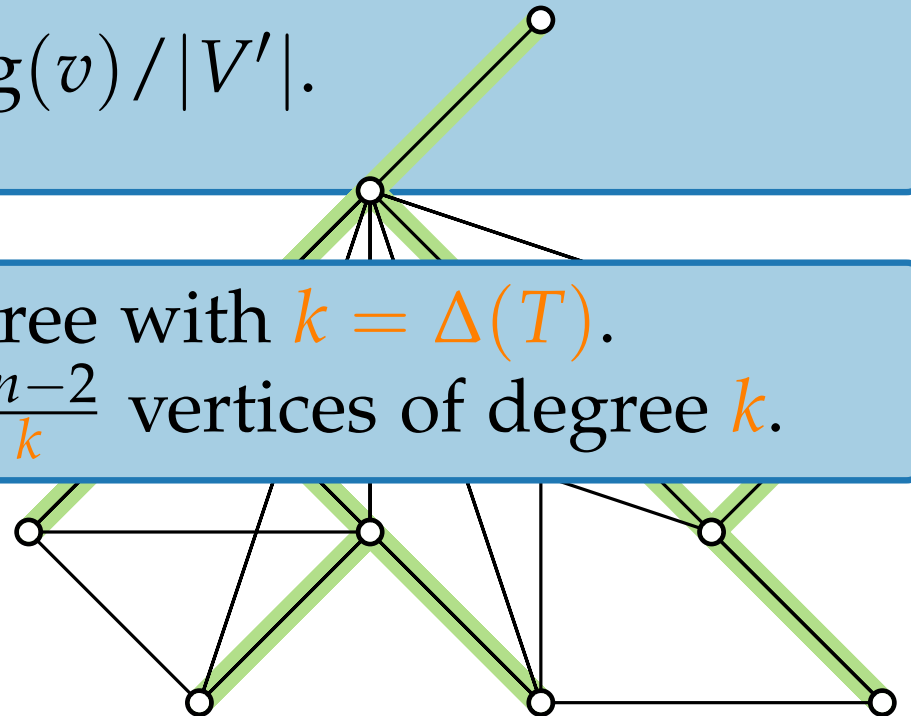
Let $V' \subseteq V(G)$.

Then $\Delta(G) \geq \sum_{v \in V'} \deg(v) / |V'|$.

Obs.

Let T be a spanning tree with $k = \Delta(T)$.

Then T has at most $\frac{2n-2}{k}$ vertices of degree k .



Approximation Algorithms

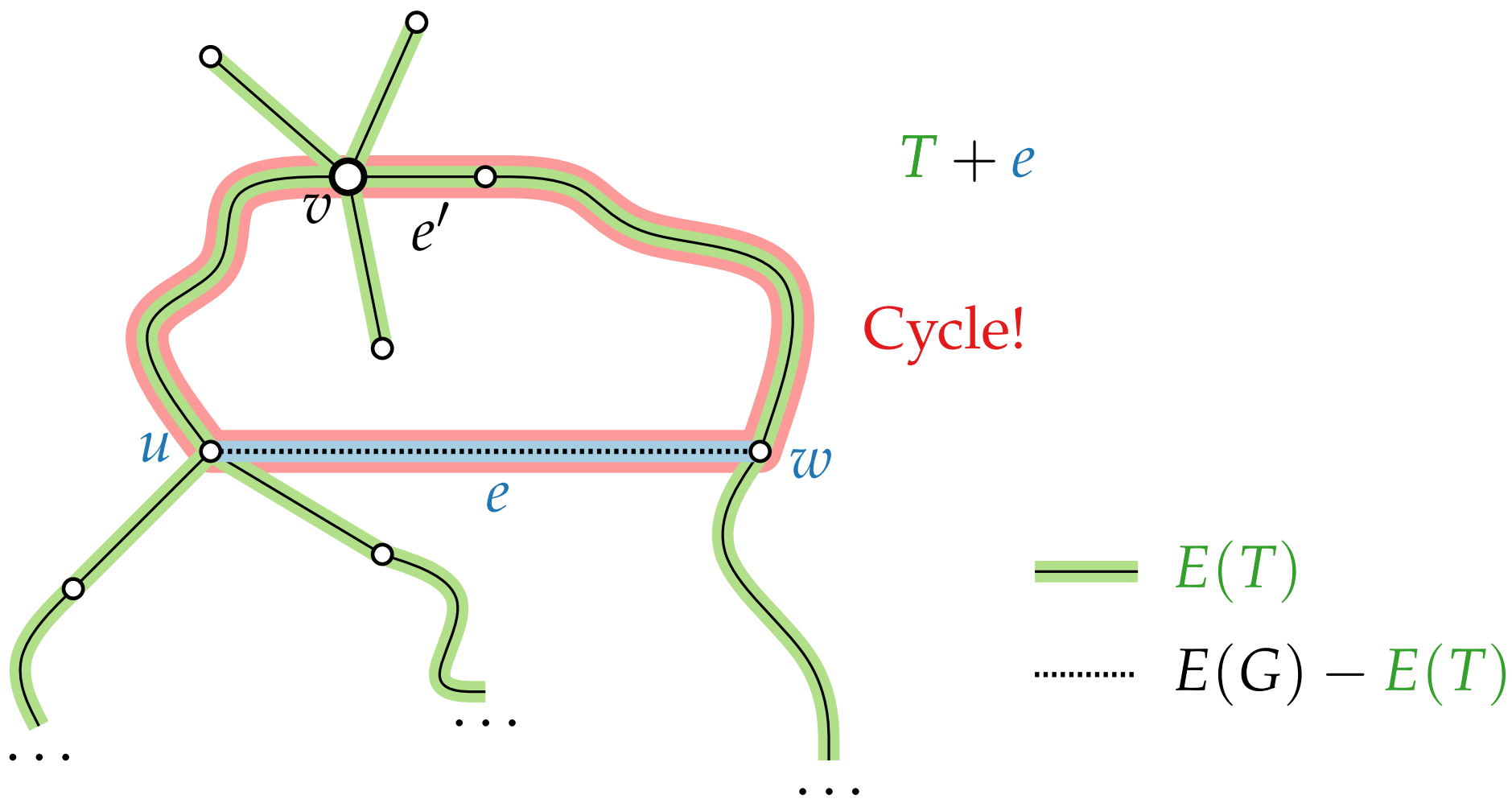
Lecture 10:

MINIMUM-DEGREE SPANNING TREE
via Local Search

Part II:

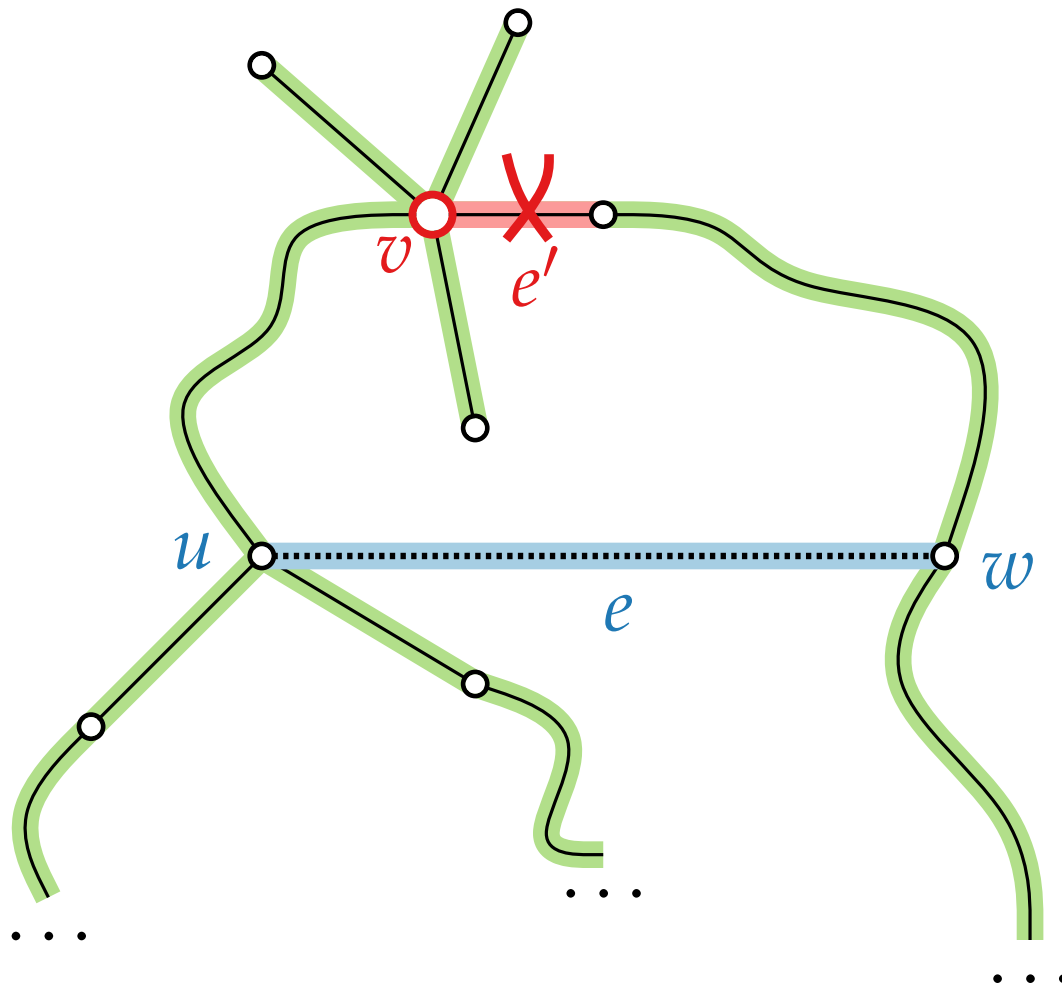
Edge Flips and Local Search

Edge Flips



Edge Flips

Def. An **improving flip** in T for a vertex v and an edge $uw \in E(G) \setminus E(T)$ is a flip with $\deg_T(v) > \max\{\deg_T(u), \deg_T(w)\} + 1$.



$$T + e - e'$$

is a new **spanning tree**

— $E(T)$

..... $E(G) - E(T)$

Local Search

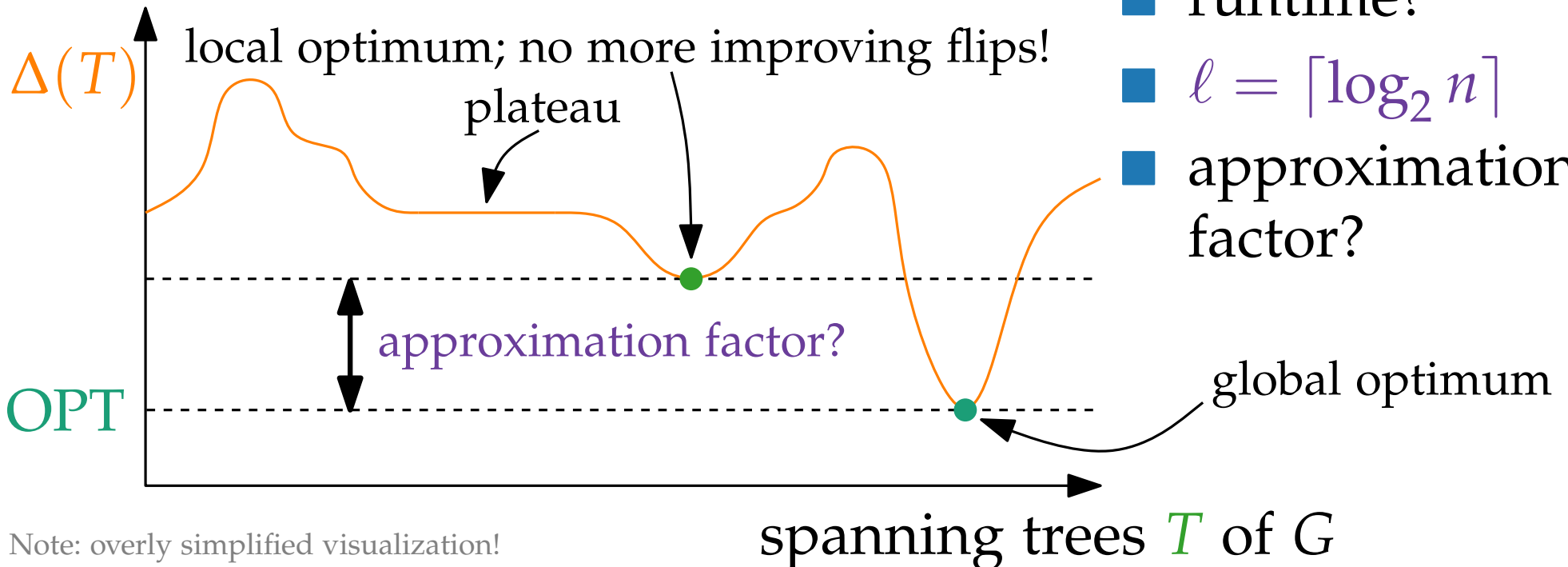
```
MinDegSpanningTreeLocalSearch(graph  $G$ )  
   $T \leftarrow$  any spanning tree of  $G$   
  while  $\exists$  improving flip in  $T$  for a vertex  $v$   
    with  $\deg_T(v) \geq \Delta(T) - \ell$  do  
    | do the improving flip  
  return  $T$ 
```

■ Termination?

■ runtime?

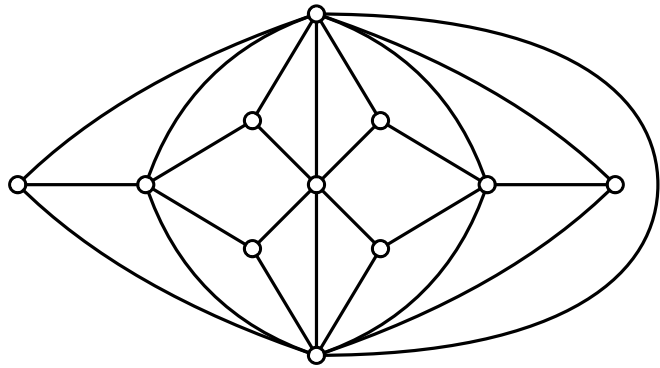
■ $\ell = \lceil \log_2 n \rceil$

■ approximation factor?



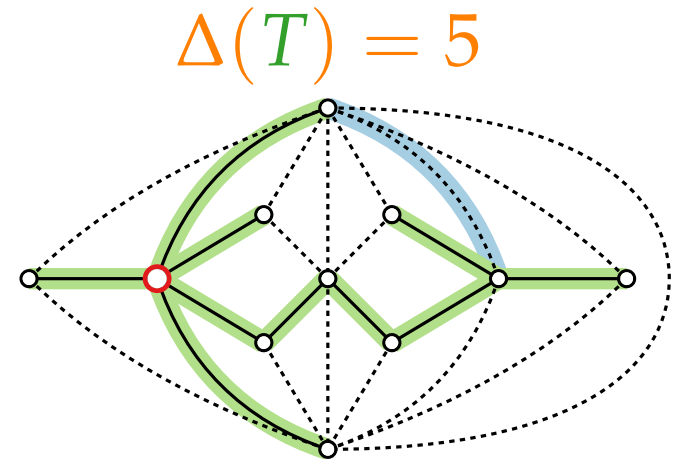
Note: overly simplified visualization!

Example



Goldner-Harary graph (minus two edges)

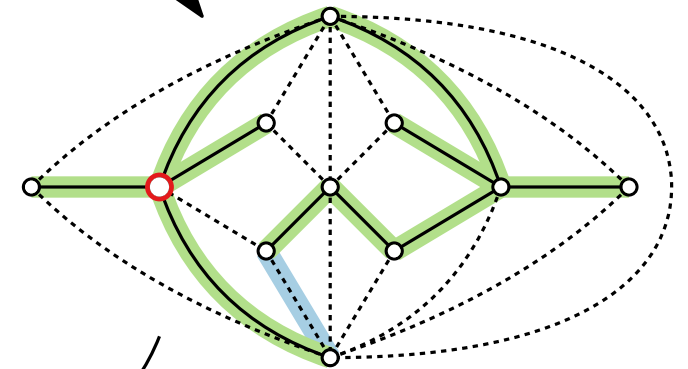
choose any
spanning tree
 T



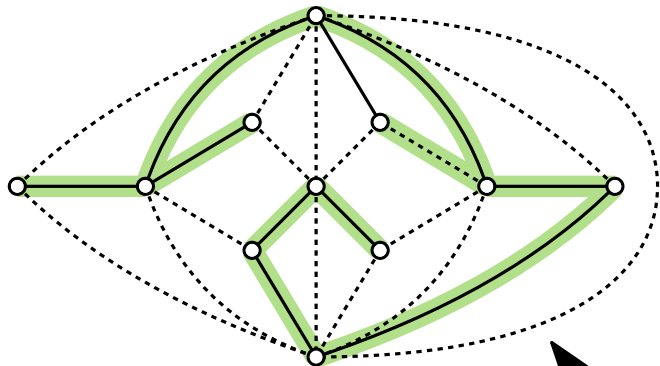
$$\Delta(T) = 5$$

improving flip

$$\Delta(T') = 4$$

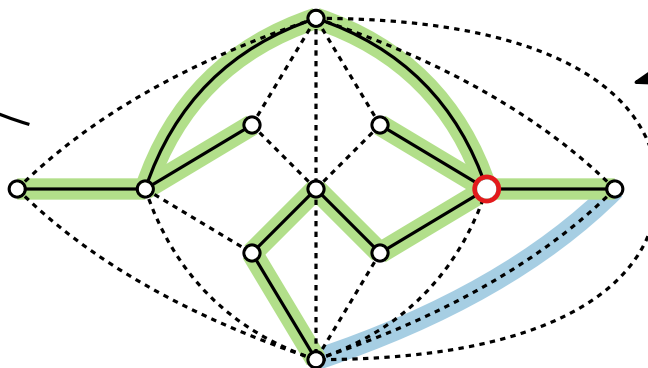


$$\Delta(T''') = 3 \text{ but } \Delta(T^*) = 2$$



improving flip

$$\Delta(T'') = 4$$



improving flip

Approximation Algorithms

Lecture 10:

MINIMUM-DEGREE SPANNING TREE

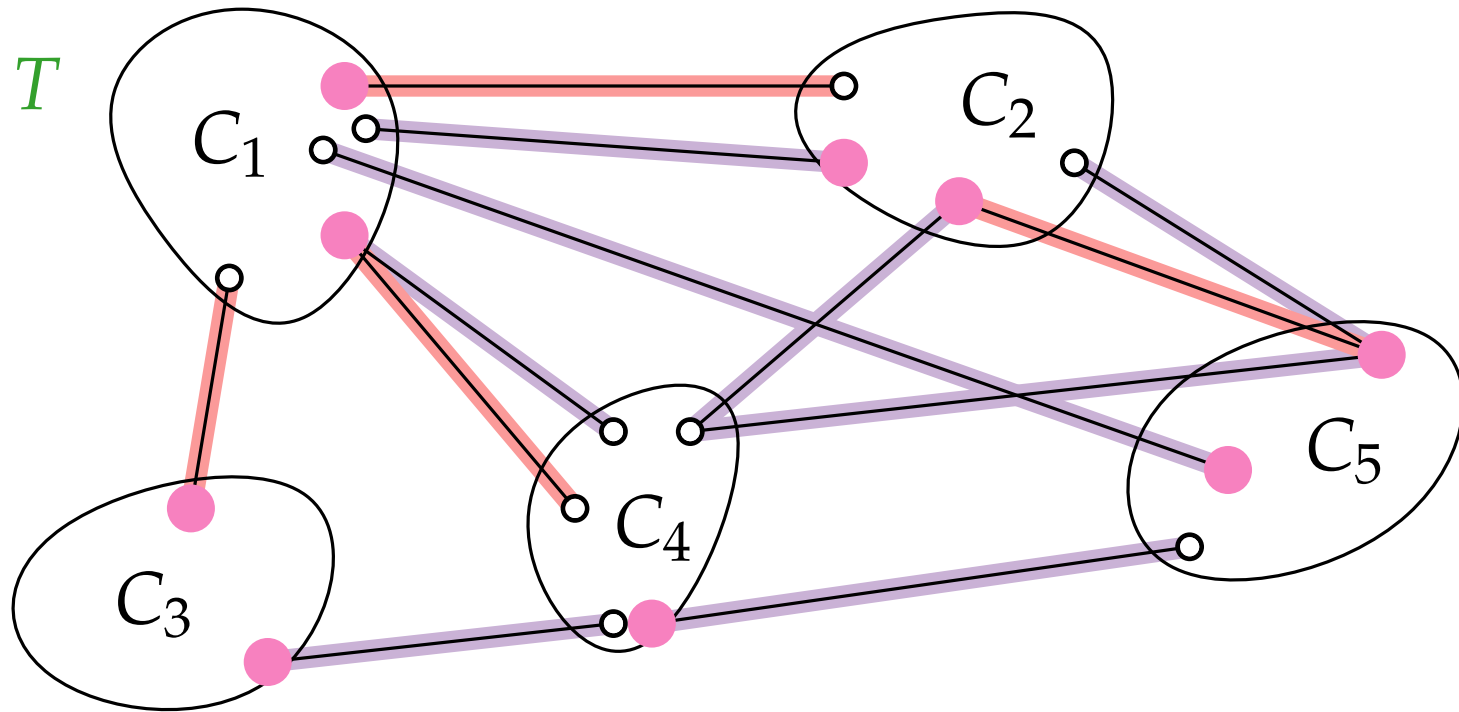
via Local Search

Part III:

Lower Bound

Decomposition \Rightarrow Lower Bound for **OPT**

- Removing k edges decomposes T into $k + 1$ components
- $E' = \{\text{edges in } G \text{ between different components } C_i \neq C_j\}$.
- $S := \text{vertex cover of } E'$.



- $|E(T^*) \cap E'| \geq k$ for opt. spanning tree T^*
- $\sum_{v \in S} \deg_{T^*}(v) \geq k$

Lemma 1.

$\Rightarrow \text{OPT} \geq k / |S|$

Approximation Algorithms

Lecture 10:

MINIMUM-DEGREE SPANNING TREE

via Local Search

Part IV:

More Lemmas

More Lemmas

$$\begin{aligned} &\Rightarrow S_1 \supseteq S_2 \supseteq \dots \\ &\Rightarrow S_1 = V(G) \\ &\Rightarrow E_1 = E(T) \end{aligned}$$

Let S_i be the set of vertices v in T with $\deg_T(v) \geq i$.

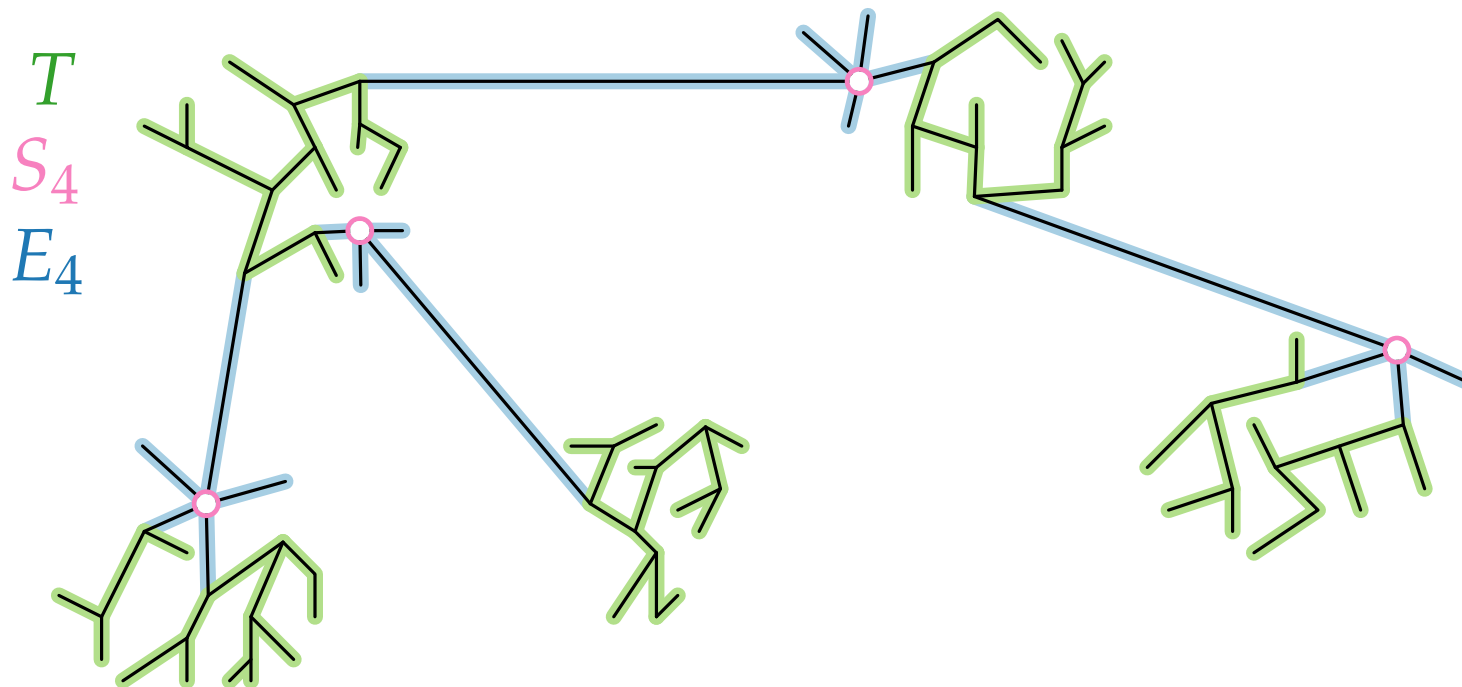
Let E_i be the set of edges in T incident to S_i .

Lemma 2. There is some $i \geq \Delta(T) - \ell + 1$ with $|S_{i-1}| \leq 2|S_i|$.

Proof. $|S_{\Delta(T) - \ell}| > 2^\ell |S_{\Delta(T)}| = 2^{\lceil \log_2 n \rceil} |S_{\Delta(T)}| \geq n \cdot |S_{\Delta(T)}|$ ⚡

Otherwise

TODO: What if $\ell > \Delta(T)$?



More Lemmas

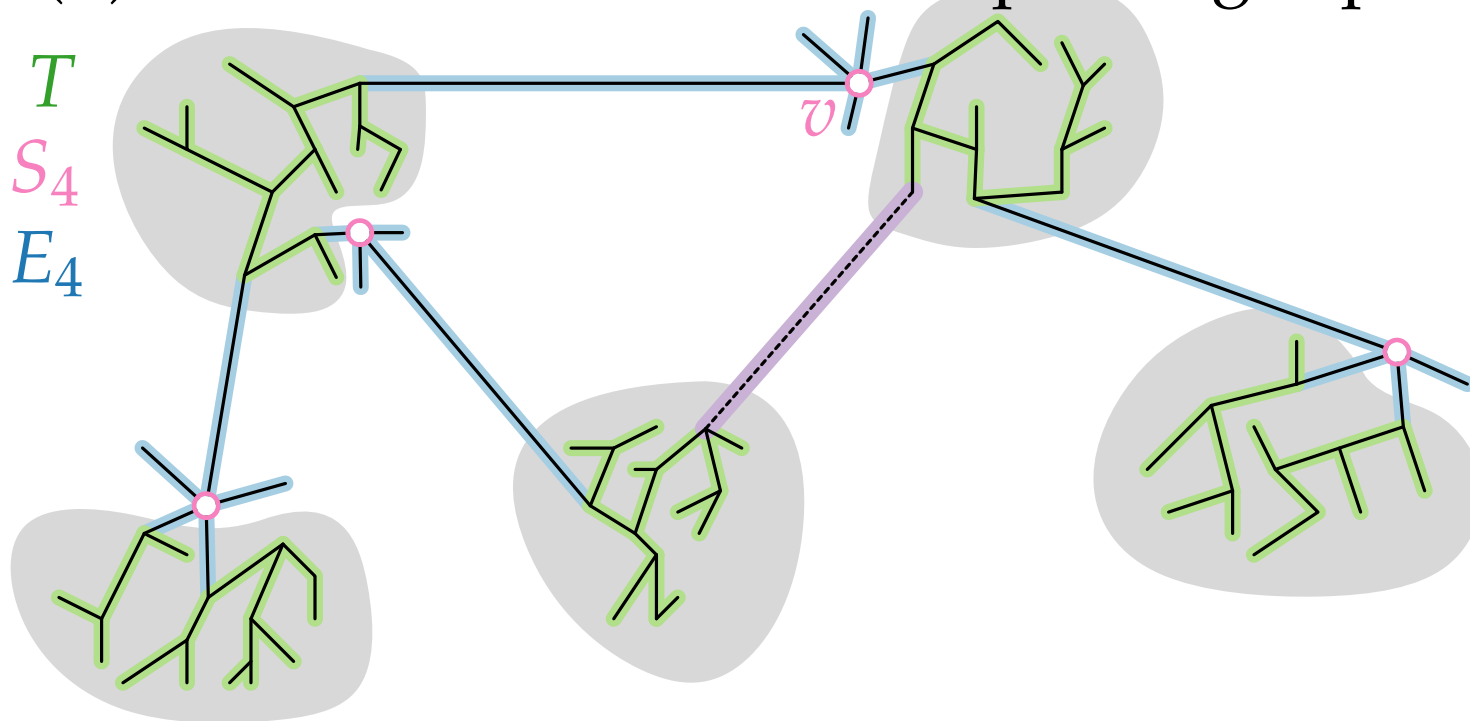
Lemma 3. For $i \geq \Delta(T) - \ell + 1$,

(i) $|E_i| \geq (i - 1)|S_i| + 1$,

(ii) Each edge $e \in E(G) \setminus E_i$ connecting distinct components of $T \setminus E_i$ is incident to a node of S_{i-1} .

Proof. (i) $|E_i| \geq \underbrace{i|S_i|}_{\text{vertex-deg}} - \underbrace{(|S_i| - 1)}_{\text{counted twice?}} = (i - 1)|S_i| + 1$

(ii) Otherwise, there is an improving flip for $v \in S_i$.



Approximation Algorithms

Lecture 10:

MINIMUM-DEGREE SPANNING TREE
via Local Search

Part V:

Approximation Factor

Approximation Factor

[Fürer & Raghavachari:
SODA'92, JA'94]

Theorem. Let T be a locally optimal spanning tree.
Then $\Delta(T) \leq 2 \cdot \text{OPT} + \ell$, where $\ell = \lceil \log_2 n \rceil$.

Proof. Let S_i be the vertices v in T with $\deg_T(v) \geq i$.
Let E_i be the edges in T incident to S_i .

Lemma 1. $\text{OPT} \geq k/|S|$ if $k = |\text{removed edges}|$, S vertex cover.

Lemma 2. There is an $i \geq \Delta(T) - \ell + 1$ with $|S_{i-1}| \leq 2|S_i|$.

Lemma 3. For $i \geq \Delta(T) - \ell + 1$,

(i) $|E_i| \geq (i-1)|S_i| + 1$,

(ii) Each edge $e \in E(G) \setminus E_i$ connecting distinct components of $T \setminus E_i$ is incident to a node of S_{i-1} .

Remove E_i for this $i!$ $\Rightarrow S_{i-1}$ covers edges between comp.

$$\text{OPT} \underset{\text{Lemma 1}}{\geq} \frac{k}{|S|} = \frac{|E_i|}{|S_{i-1}|} \underset{\text{Lemma 3}}{\geq} \frac{(i-1)|S_i|+1}{|S_{i-1}|} \underset{\text{Lemma 2}}{\geq} \frac{(i-1)|S_i|+1}{2|S_i|} > \frac{(i-1)}{2} \geq \frac{\Delta(T)-\ell}{2} \quad \square$$

Approximation Algorithms

Lecture 10:

MINIMUM-DEGREE SPANNING TREE
via Local Search

Part VI:

Termination, Running Time & Extensions

Termination and Running Time

Theorem. The algorithm finds a locally optimal spanning tree after $O(n^4)$ iterations.

Proof. Via potential function $\Phi(T)$ measuring the value of a solution where (hopefully): $\Phi(T) = \sum_{v \in V(G)} 3^{\deg_T(v)}$

- Each iteration decreases the potential of a solution.

Lemma. After each flip $T \rightarrow T'$, $\Phi(T') \leq (1 - \frac{2}{27n^3})\Phi(T)$.

- The function is bounded both from above and below.

Lemma. For each spanning tree T , $\Phi(T) \in [3n, n3^n]$.

- Executing $f(n)$ iterations would exceed the lower bound.

Let $f(n) = \frac{27}{2}n^4 \cdot \ln 3$. How does $\Phi(T)$ change?

$\Phi(T)$ decreases by: $(1 - \frac{2}{27n^3})^{f(n)} \leq (e^{-\frac{2}{27n^3}})^{f(n)} = e^{-n \ln 3} = 3^{-n}$

Goal: After $f(n)$ iterations: $\Phi(T) = n < 3n$ □

Extensions

[Fürer & Raghavachari:
SODA'92, JA'94]

Corollary. For any constant $b > 1$ and $\ell = \lceil \log_b n \rceil$, the local search algorithm runs in polynomial time and produces a spanning tree T with $\Delta(T) \leq b \cdot \text{OPT} + \lceil \log_b n \rceil$.

Proof. Similar to previous pages. **Homework** \square

Theorem. There is a local search algorithm that runs in $O(EV^\alpha(E, V) \log V)$ time and produces a spanning tree T with $\Delta(T) \leq \text{OPT} + 1$.