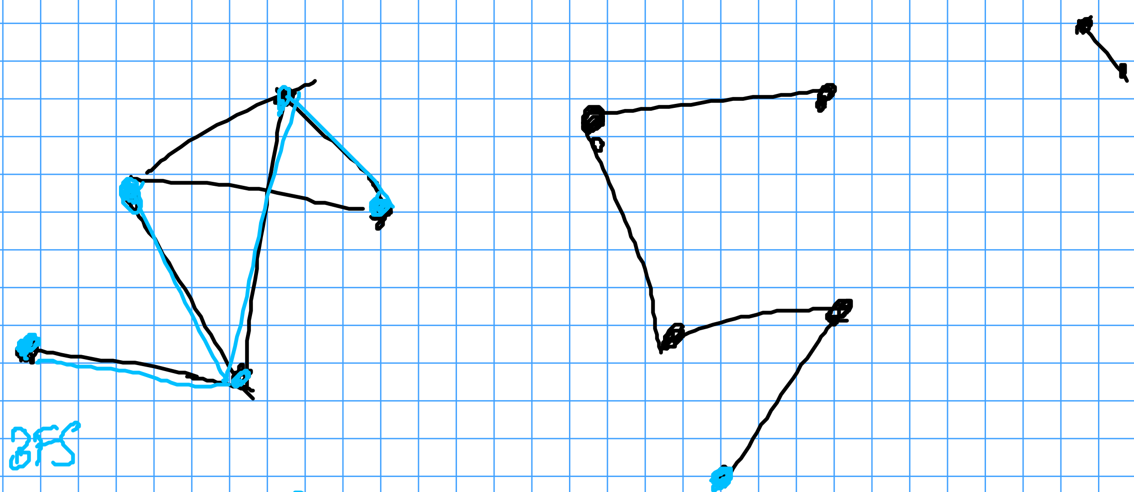
SUBLINEAR ALGORITHMS

Find the number
of connected components in a graph

- undirected graph $G(V,E)$



BFS

$O(|V|+|E|)$

$O(|V|^2)$ in dense graphs

α-additive APX algorithm outputs value ℓ

$$|\ell - OPT| \leq \alpha$$

THEOREM There exists a rand. alg. that for given $G(V,E)$ and ε

outputs an ϵn-additive APX for the number of connected components of G with probability $\frac{2}{3}$.

So it means, if k_{OPT} is the correct number of CC of G w.p. $\frac{2}{3}$ the alg. outputs k' such that

$$|k' - k_{OPT}| \leq \epsilon n$$

Def.: • For a vertex $v \in V$ we let n_v be the number of vertices in the connected component of v $C(v)$, so $n_v = |C(v)|$

Obs: • $\sum_{v \in C_i} \frac{1}{n_v} = \sum_{v \in C_i} \frac{1}{|C_i|} = |C_i| \cdot \frac{1}{|C_i|} = 1$

• $\sum_{v \in V} \frac{1}{n_v} = k_{OPT}$ $\rightsquigarrow$ if n_v is large $\frac{1}{n_v}$ is small, so vertices in small components are more important

IDEA: pick a random sample S of nodes and estimate

$$\sum_{v \in S} \frac{1}{n_v}$$

Def.: Let $\hat{n}_v = \min\{n_v, \frac{2}{\varepsilon}\}$

$$= \begin{cases} n_v & \text{if } v \text{ is in a CC of size } \frac{2}{\varepsilon} \\ \frac{2}{\varepsilon} & \text{otherwise} \end{cases}$$

Lemma: $0 \leq \frac{1}{\hat{n}_v} - \frac{1}{n_v} < \frac{\varepsilon}{2}$

Proof: If v is in a 'small component' then $\frac{1}{\hat{n}_v} - \frac{1}{n_v} = 0$

If v is in a 'large component'

$$0 < \frac{1}{\hat{n}_v} - \frac{1}{n_v} = \frac{\varepsilon}{2} - \frac{1}{n_v} < \frac{\varepsilon}{2}$$

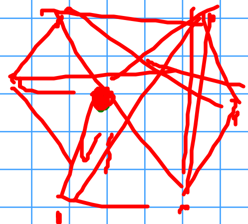
Let $\hat{c} = \sum_{v \in V} \frac{1}{\hat{n}_v}$, then $|\hat{c} - c| < \frac{\varepsilon n}{2}$.

Because $0 \leq \sum_{v \in V} \left(\frac{1}{\hat{n}_v} - \frac{1}{n_v} \right) < n \frac{\varepsilon}{2}$

ALGORITHM

1. Pick $s = O\left(\frac{1}{\varepsilon^2}\right)$ vertices in a sample $S = \{v_1, \ldots, v_s\}$ $O\left(\frac{1}{\varepsilon^2}\right)$

2. For each $v_i \in S$: run 'local' BFS, i.e.



Stop after at most $\frac{2}{\varepsilon}$ vertices have been visited $\quad O\left(\left(\frac{2}{\varepsilon}\right)^2 \cdot \frac{1}{\varepsilon^2}\right)$

$O\left(\frac{1}{\varepsilon^2}\right)$

3. For each $v_i \in S$: set $\hat{n}_v$ to the number of vertices visited in the BFS run

4. Output $c' = \left(\frac{1}{s}\sum_{v \in S} \frac{1}{\hat{n}_v}\right) \cdot n$

how much does a randomly chosen vertex contribute to the total sum

Lemma: With prob. $\frac{2}{3}$, the alg. outputs c' such that $|c' - c_{opt}| < \varepsilon n$, and it runs in $O\left(\frac{1}{\varepsilon^4}\right)$.

Lemma: $P\left(|c' - \hat{c}| > \frac{\varepsilon n}{2}\right) < \frac{1}{3}$

Prove: Let $X_i = \frac{1}{\hat{n}_{v_i}}$ be a random variable. Then

$$E(X_i) = \frac{1}{n} \underbrace{\sum_{v \in V} \frac{1}{\hat{n}_v}}_{\hat{c}} = \frac{\hat{c}}{n}$$

for every i.

Let $X = \sum_{i=1}^{s} X_i$. Then

$$E(X) = E\left(\sum_{i=1}^{s} X_i\right) = \sum_{i=1}^{s} E(X_i)$$

$$= \sum_{i=1}^{s} \frac{\hat{c}}{n} = \frac{s}{n}\hat{c}$$

The algorithm outputs $\frac{n}{s} X = c'$,

i.e. $\frac{n}{s} E(X) = \hat{c}$.

Now we apply Chernoff's bound:

$$P\left(|X - E(X)| > \frac{\varepsilon s}{2}\right) < e^{-\frac{\varepsilon^2 s}{8}}$$

$s \in O\left(\frac{1}{\varepsilon^2}\right)$, e.g. use $s = \frac{12}{\varepsilon^2}$

$$\Rightarrow P\left(|X - E(X)| > \frac{\varepsilon \cdot \frac{12}{\varepsilon^2}}{2}\right) < e^{-1.5} < \frac{1}{3}$$

$$P\left(\left|X \cdot \frac{n}{s} - E(X)\frac{n}{s}\right| > \frac{\varepsilon s}{2} \frac{n}{s}\right) < \frac{1}{3}$$

$$P\left(|\underline{c'} - \hat{c}| > \frac{\varepsilon n}{2}\right) < \frac{1}{3}$$

We showed that the alg. outputs an $\frac{\varepsilon n}{2}$ additive APX for $\hat{c}$

with a prob. of $\frac{2}{3}$. And now we show, that this is an εn-additive APX to b_{OPT}.

Proof: We now by the previous lemma that

$$P\left(|b' - \hat{b}| > \frac{\varepsilon n}{2}\right) < \frac{1}{3}$$

and by the other Lemma that $|\hat{b} - b| < \frac{\varepsilon n}{2}$. Recall that we want to show

$$P\left(|b' - b_{OPT}| > \varepsilon n\right) < \frac{1}{3}$$

If $|b' - b_{OPT}| > \varepsilon n$, then

$$\varepsilon n < |b' - b_{OPT}| \leq \underbrace{|b' - \hat{b}|}_{\substack{> \frac{\varepsilon n}{2} \\ \text{with prob. } \frac{1}{3}}} + \underbrace{|\hat{b} - b_{OPT}|}_{< \frac{\varepsilon n}{2}}$$

It follows that

$$P\left(|b' - b_{OPT}| > \varepsilon n\right) < \frac{1}{3}.$$

Runtime: dominated by $O(\frac{\Delta}{\varepsilon^2})$ many local

BFS runs, each run taking at most

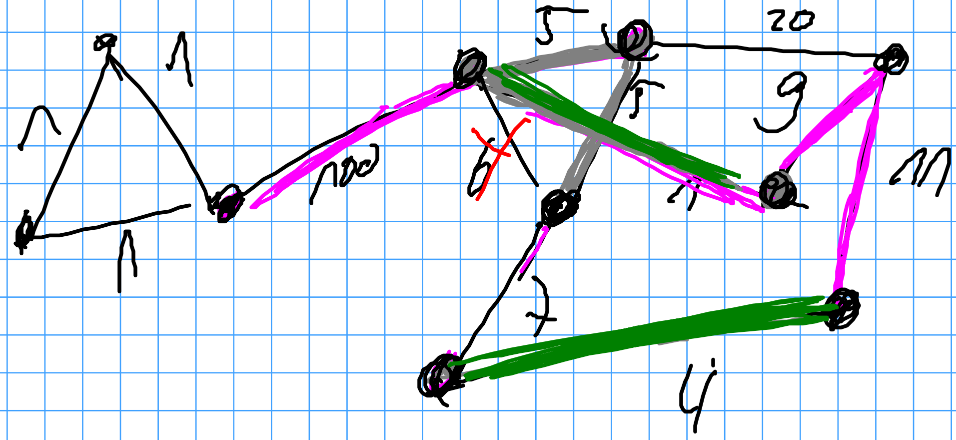
$$O\left(\left(\frac{2}{\frac{\varepsilon}{2}}\right)^2\right) \text{ time} \Rightarrow O\left(\frac{\Delta}{\varepsilon^2} \cdot \frac{4}{\varepsilon^2}\right) = O\left(\frac{\Delta}{\varepsilon^4}\right)$$

EXERCISE: Implement the det. alg. to get the connected comp., implement the rand. alg. & compare results.

Obs.: If we pick $s = O\left(\frac{\Delta}{\varepsilon^2} \log \frac{1}{\delta}\right)$ samples in the above alg. we get $P\left(|\hat{c}' - c_{apr}| > \frac{\varepsilon n}{2}\right) < \delta$.

Estimating the Weight of a Minimum Spanning Tree

weighted $G(V, E)$ $\quad w: E \to \mathbb{R}$



Det. alg.: Prim & Kruskal

$$O(n \log n + m)$$

Rand. alg.: We will use the alg. to estimate the number of CC as a subroutine (picking $s = O(\frac{1}{\varepsilon^2}\log\frac{1}{\delta})$ sample)

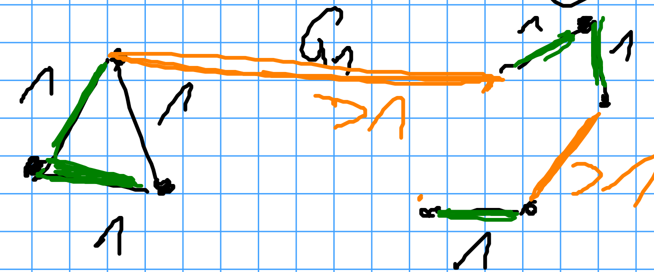
c) Assumption: $w(e) \in \{1, 2, \dots, W\}$ upper weight bound

Goal: estimate $w(MST)$, such that

$$|\hat{w}(MST) - w(MST)| < \varepsilon n.$$

Let G_1 be the subgraph of G which contains only edges of weight 1.

Let T be some MST in G.
We want to estimate how many edges in T have a weight strictly larger than 1.



Obs:

- T cannot have an edge with $w(e) > 1$ between vertices in a CC of G_1 since that would result in cycle.
- Each time we add an edge to G_1 greedily (maintaining the MST structure), the number of disconnected comp. is reduced by one.

Let c_1 be the number of CC in G_1 $\Rightarrow$ # edges in T with weight > 1 $= c_1 - 1$

Define G_i as subgraph of G with only weights $\leq i$ (up to $i = W$).

By the same argument as above, we get the number of edges with weight $> i$ in T is exactly $c_i - 1$

Observe that this is even true for 0, because then we have n disconnected components and we know $|T| = n-1$.

Lemma: $$w(MST) = \sum_{i=0}^{W} (c_i - 1)$$

Proof: Let m_i be the number of edges with weight i in T.

Then
$$w(MST) = \sum_{i=1}^{W} i\, m_i$$

$$= \sum (1+1+1 \ldots +1) m_i$$

$$= \sum_{i=1}^{W} m_i + \sum_{i=2}^{W} m_i + \ldots + \sum_{i=w}^{W} m_i$$

$$= \sum_{i=0}^{W} (c_i - 1)$$

because $\sum_{i=j}^{W} m_j = c_j - 1$

ALGORITHM:

given: G, E and W

1. For $i = 1, \dots, W$ compute $\hat{c}_i$

(approx. of number of CC of G_i)

by running the prev. alg. with

$\varepsilon' = \frac{\varepsilon}{W}$ and $\delta = \frac{1}{3W}$

2. Output: $\hat{w}(MST) = \sum_{i=1}^{W-1} \hat{c}_i - W$

Analysis: If all $\hat{c}_i$ are within

an additive error $\frac{\varepsilon}{W} n$, i.e.

$|\hat{c}_i - c_{i,OPT}| \leq \frac{\varepsilon n}{W}$, then

$$\left| \sum_{i=1}^{W} (\hat{c}_i - c_{i,OPT}) \right| \leq W \cdot \frac{\varepsilon n}{W} = \varepsilon n$$

So with $O\left(\frac{1}{\varepsilon^2} \log \frac{W}{\varepsilon}\right)$ samples,

for each i $P\left(|\hat{c}_i - c_{i,OPT}| > \frac{\varepsilon n}{W}\right) < \frac{1}{3W}$

$$P(\text{one of the estimates is bad}) \leq W \cdot \frac{1}{3W} = \frac{1}{3}$$

$\Rightarrow$ Can approximate $w(MST)$ w.p. $\frac{2}{3}$ in a εn - additive way.

Runtime: $$O\left(s \cdot \left(\frac{1}{\varepsilon}\right)^2 \cdot W\right)$$
$$= O\left(\frac{1}{\varepsilon^4} W \log \frac{W}{\varepsilon}\right).$$