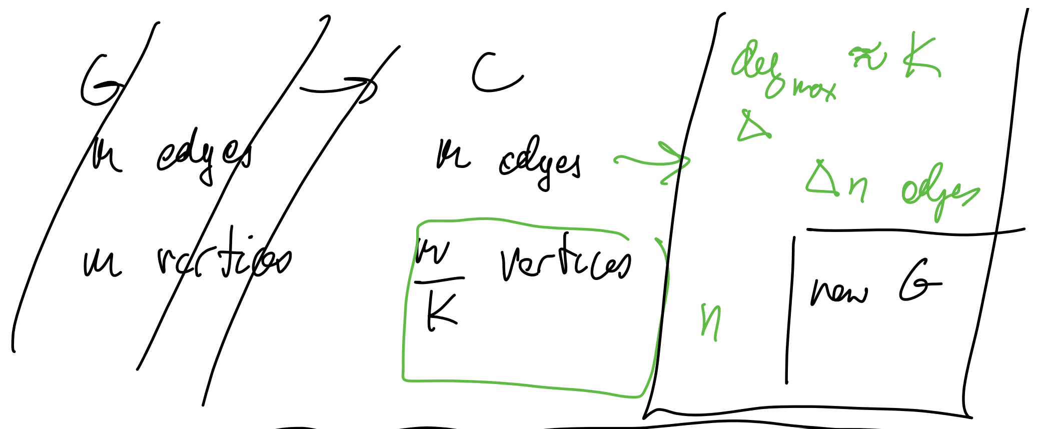


DEFN Decremental graph 1) G is undergoing vertex splits & edge deletions

2) G is connected

Spanner should survive through ~ n updates

Decremental Connected Spanner THM $G = (V, E)$

n vertices

$= \Delta n$ edges

Δ : max deg

Sequence of graphs $G = G_0, \dots, G_t$

time $0, \dots, t$

we can maintain a spanner H_i of G_i

with H_0 has $|E_0| \leq \tilde{O}(n)$

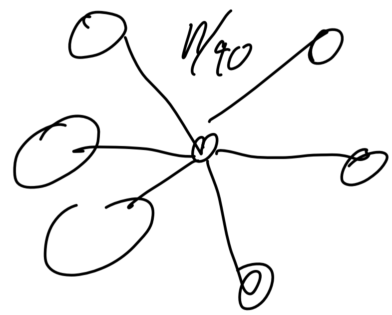
\# of changes from H_i to H_{i+1} is amortized $n^{o(1)}$ $\ll \Delta$

$n \cdot n^{o(1)} \ll n\Delta$

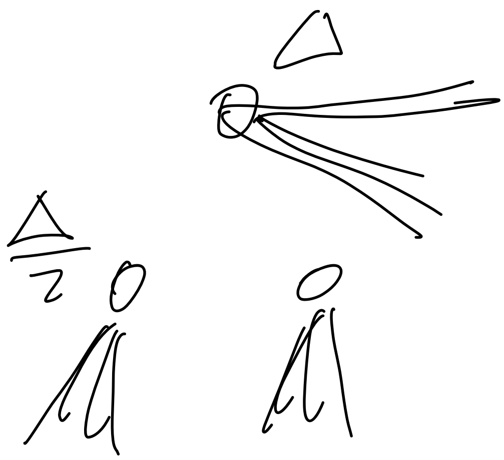
This works against adaptive adversaries.

The spanner can survive through $\Omega(n)$ updates.

Running time will total $n^{1+o(1)} \cdot \Delta$



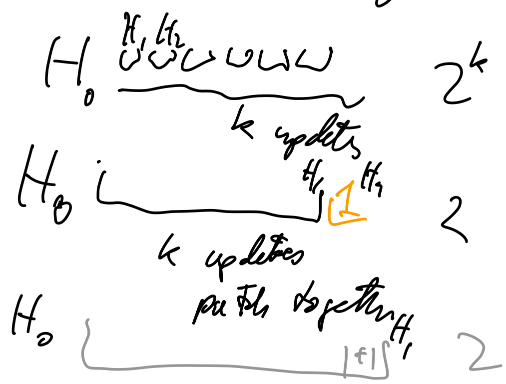
REMARK: you have to be careful about representation size of updates



---

proof ideas

1) one-shot patching



2) $n^{1.5}$ algo w. batching

3) $n^{1+o(1)}$

One-Shot Pruning Informed Chain

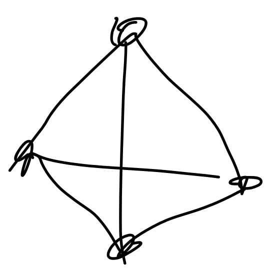
Given G and a spanner H of G
and k deletions/vertex splits to apply to $G \to$ results G'

we can get a spanner H' of G'

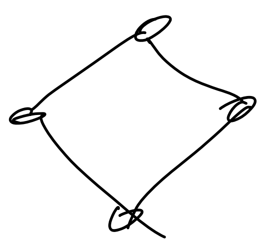
s.t. if H has stretch γ
then H' has stretch $\tilde{O}(\gamma)$

AND #changes $H \to H'$ is $\tilde{O}(k)$

G



H



Apply 1-shot patching ℓ times in sequence:

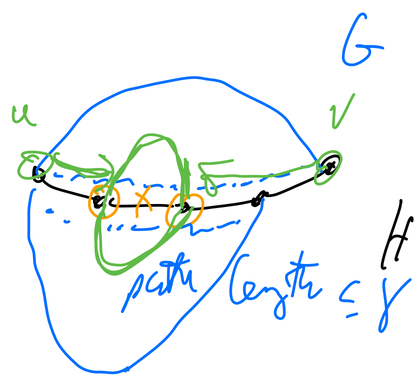
stretch $\left(\tilde{O}(\tfrac{1}{\epsilon})\right)^{\ell}$

Pf sketch

M = set of vertices incident to deletions of edges & vertex splits.

$|M| = O(k)$

Consider every edge in G whose embedding in H goes through M



H is a γ spanner of G

"Project" the edges of G, whose embeddings touch M, to M

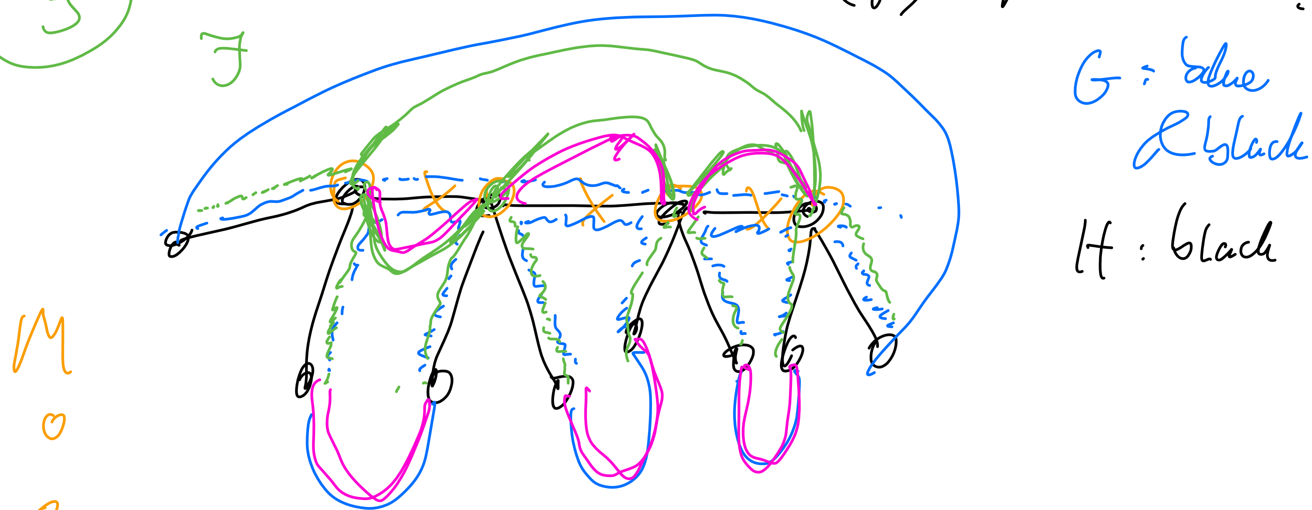
This creates a graph F

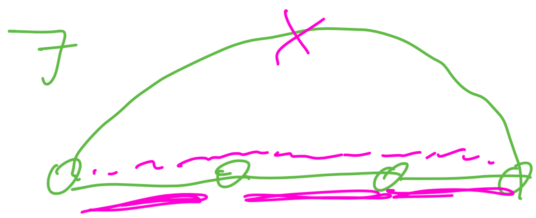
Notice: we can also "unproject" edges

Compute a spanner F' of F $\rightarrow$ $\tilde{O}(1)$ avg deg
$\tilde{O}(1)$ str.

<u>Claim</u> if we unproject the edges of F' into H + updates
then call this H'.
Now H' is a $\tilde{O}(\gamma)$ spanner of G'
stretch ✓

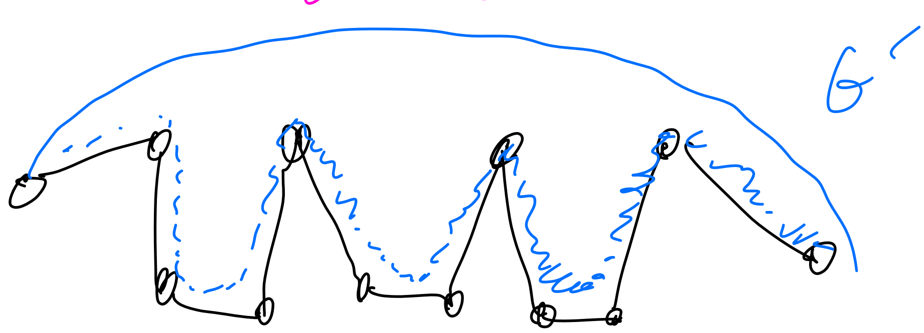
F





spanner F'

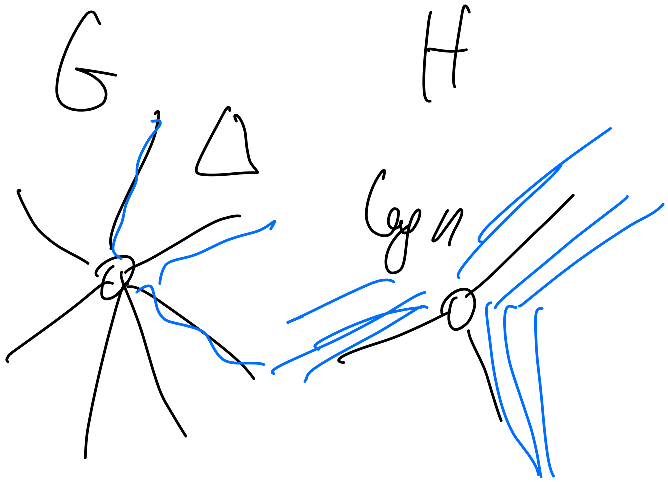
H'



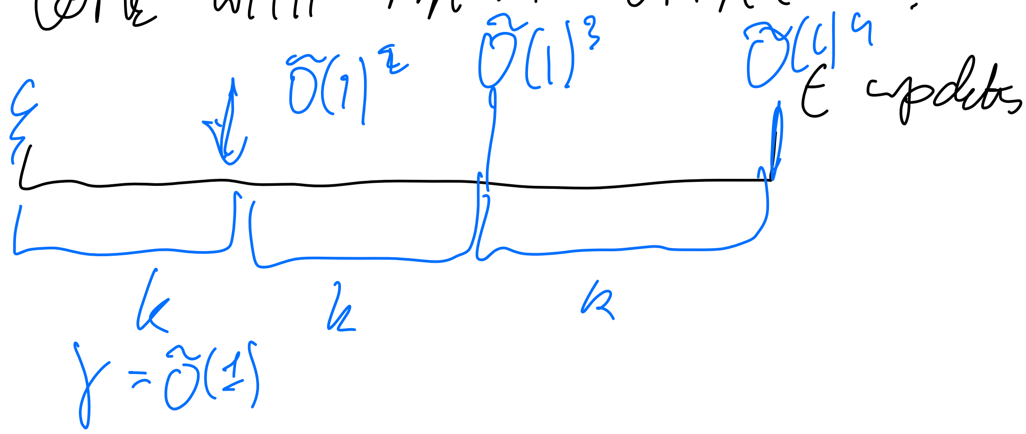
WHY does H' only add $\tilde{O}(k)$ edges to H + updates $\rightarrow$ # of deletions & vertex splits.

$|M| = O(k)$

$|E_{H'}| = \tilde{O}(k)$



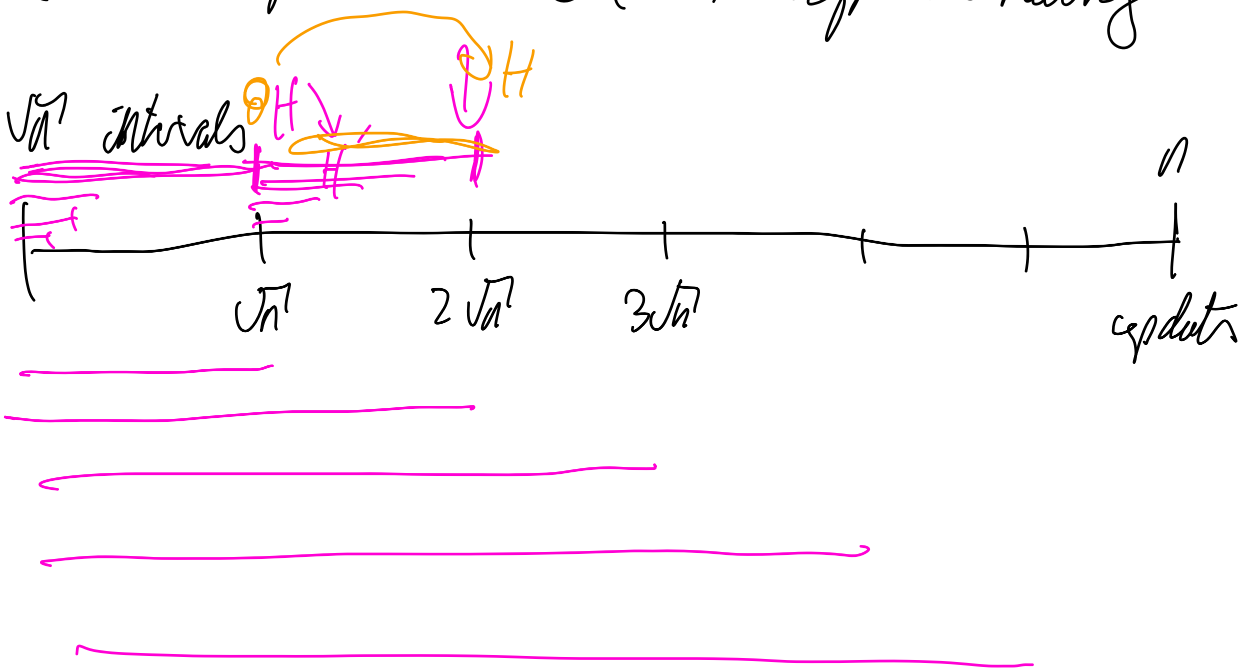
HOW CAN WE USE THIS PATCHING to COPE WITH MANY UPDATES?



BATCHING SCHEME to get $n^{1.5}$ total changes to H'

$t \approx n$ updates $\tilde{O}(n^2)$ updates naive

$\sqrt{n}$ intervals



$\sqrt{n}$ updates
$\downarrow$
patching on $\sim \sqrt{n}$

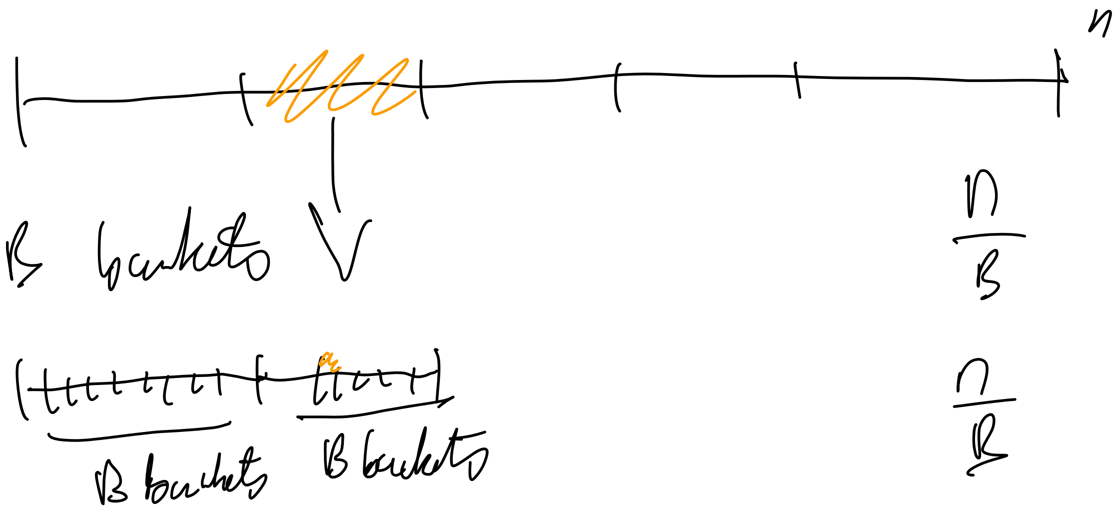
$\cdot \sqrt{n}$ intervals

$n^{1.5}$

$\sqrt{n}$ interval endpoints
each time $\sim n$ updates

$n^{1.5}$

$\tilde{O}(n^{1.5})$



$$\frac{n}{B^L} = 1$$

L levels

$$B = n^{1/L} \qquad \text{Try } L = \log^{0.2} n$$

$$= 2^{\frac{\log n}{\log^{0.2} n}} = 2^{\log^{0.8} n}$$

Updates at level i

$$\frac{n}{B^{i-1}} \text{ size} \cdot \underset{\text{buckets}}{B^i} = \epsilon B$$

<u>L levels</u>: total changes to H is $\tilde{O}(nB \cdot L)$

$$\tilde{O}(n\, n^{1/L} L)$$

$L = \log^{0.2} n$

$\tilde{O}\left(n \cdot 2^{\log^{0.8}} \cdot \log^{0.2} n\right)$

$\left(\tilde{O}(1)\right)^L$ stretch

$(\log^{10} n)^{\log^{0.2} n} = 2^{O(\log^{0.2} n \cdot \log\log n)}$

$n^{o(1)}$