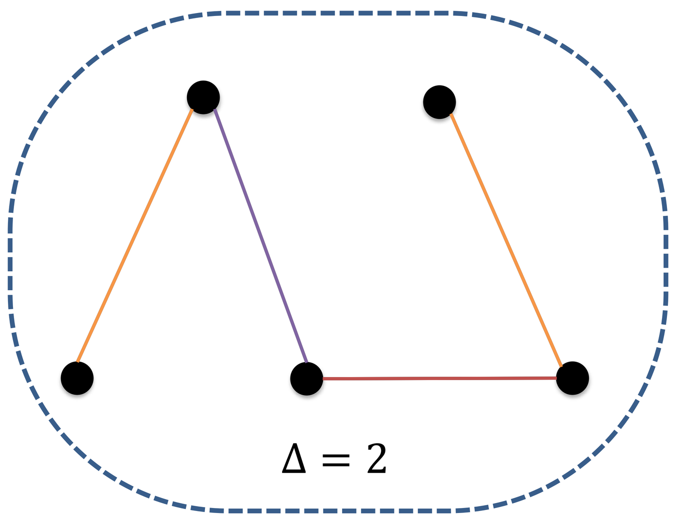
Online Edge Coloring via Tree Recurrences and Correlation Decay

Janardhan Kulkarni, MSR
Yang P. Liu, Stanford
Mehtaab Sawhney, MIT
Ashwin Sah, MIT
Jakub Tarnawski, MSR

Online edge coloring

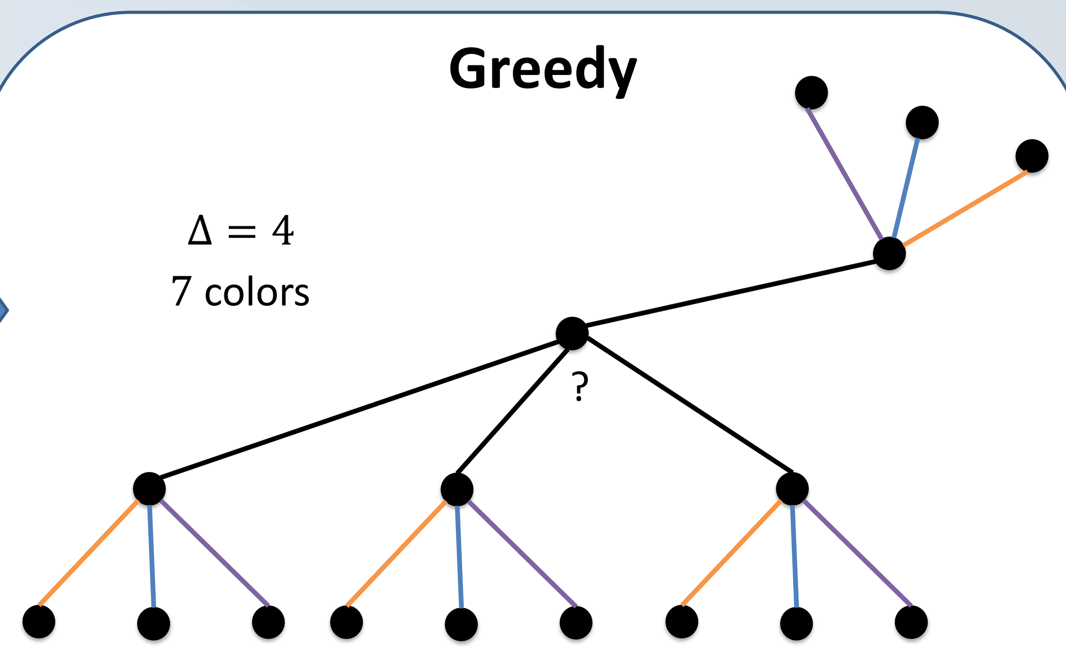
Given: graph with maximum degree Δ, edges arrive online (adversarial order)

Assign color to each edge to get a proper coloring and **minimize** number of colors



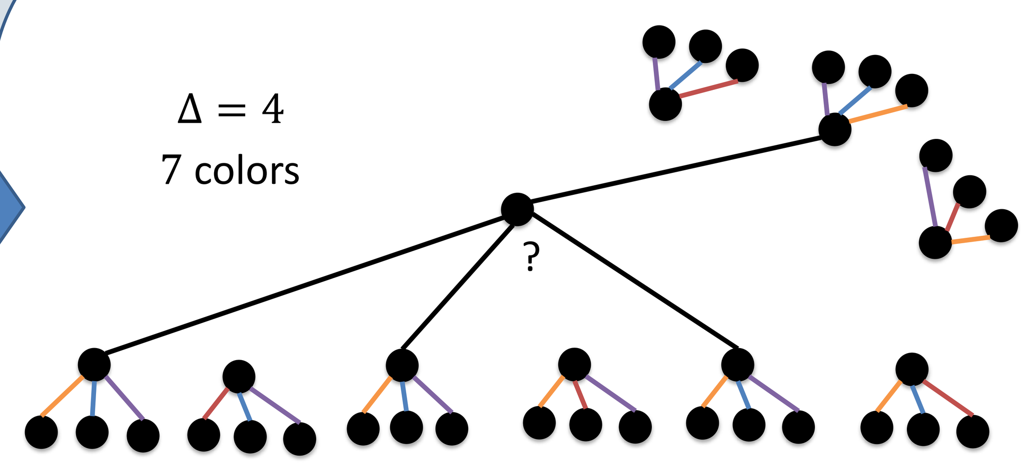
Offline:
optimum is Δ or $\Delta + 1$ colors [Vizing 1964]
NP-hard to distinguish [Holyer 1981]

Greedy



- When edge arrives, assign smallest free color
- Uses $2\Delta - 1$ colors
- This is tight

Bar-Noy, Motwani, Naor (1992)



Lower bound: impossible to beat $2\Delta - 1$

Proof:
make lots of $(\Delta - 1)$-stars, until there are Δ identical ones

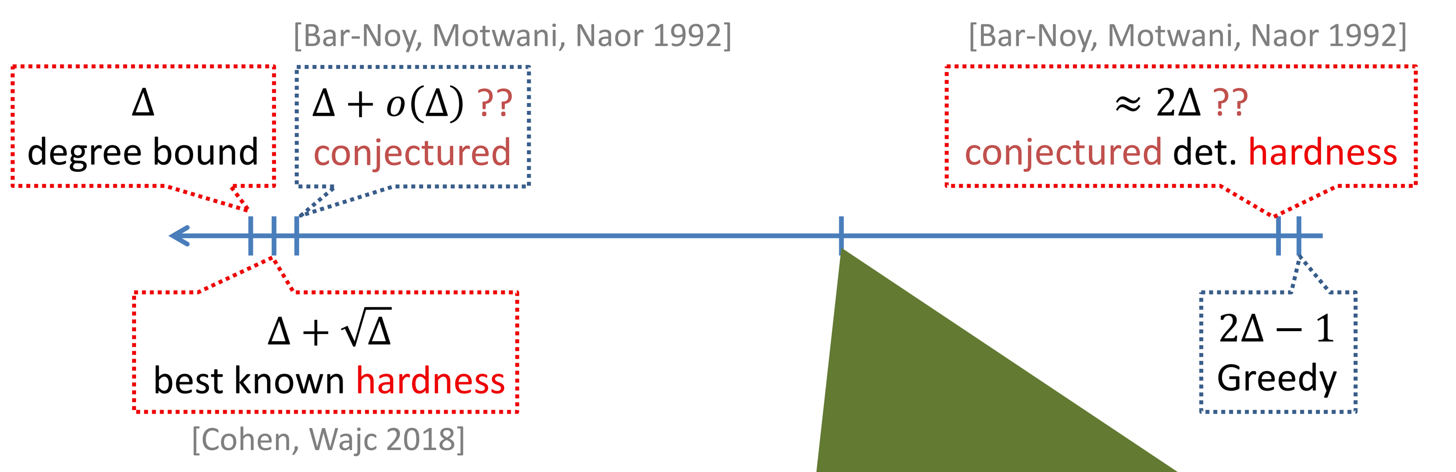
However: need $\exp(\Delta)$ vertices for this, so $\Delta \le O(\log n)$

What about $\Delta > \omega(\log n)$?
Question: can better bounds be achieved?

Conjecture: there is a randomized algorithm using $(1 + o(1))\Delta$ colors

State of the art

(everything is for $\Delta > \omega(\log n)$)



Our result: randomized algorithm that uses $\frac{e}{e-1}\Delta \approx 1.58\Delta$ colors

Oblivious adversary: adversary fixes graph and edge order, then we flip our coins

Partial progress on special cases:

random-order arrival:
- 1.26Δ colors [Bahmani, Mehta, Motwani 2012]
- $(1 + o(1))\Delta$ colors [Bhattacharya, Grandoni, Wajc 2021]

vertex arrival:
- bipartite one-sided: $(1 + o(1))\Delta$ colors [Cohen, Peng, Wajc 2019]
- general: 1.9Δ colors [Saberi, Wajc 2021]

Reduction 1 (to matching)

online edge coloring using C colors (i.e. partition graph into matchings)

Reduction:
While $G \neq \emptyset$:
$M :=$ matching(G)
color M with new color
remove M from G

[Cohen, Peng, Wajc 2019] [Saberi, Wajc 2021]

a version of online matching where every edge must be matched with prob $\ge 1/C$

If $\Delta > \omega(\log n)$, enough concentration to finish in $C + o(C)$ iterations

Reduction 2 (subsampling)

same online matching, but in locally treelike graphs

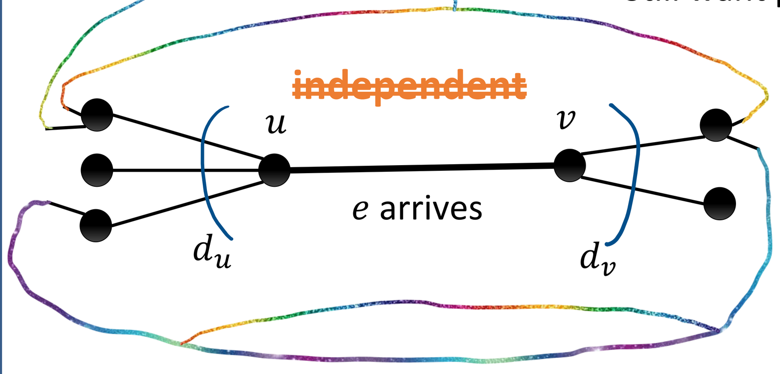
Reduction: Keep each edge with prob. Δ'/Δ

Effects:
- Each vertex has degree $\le \Delta'$ w.h.p.
- Most edges have no short cycles

Why?
of g-cycles containing some edge in org graph $\le \Delta^{g-2}$
Probability that such a cycle survives subsampling $\le (\Delta'/\Delta)^g$
So expected number of surviving cycles $\le (\Delta')^g/\Delta^2$:
very small if $(\Delta')^g < o(\log n)$; think $g, \Delta' = \omega(1)$

General treelike case – challenge

Still want prob. $\approx 1/C$ for each edge



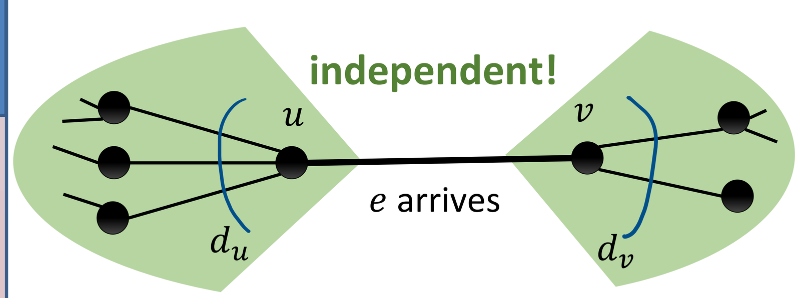
Obstacle for analysis: events "u already matched", "v already matched" (upon arrival of e) are **not** independent

But correlation is over long paths, and we show that it decays!

Analysis inspired by [Weitz 2006]

Tree case

How to solve online matching on *really* treelike instances: trees?
We want to match each edge with prob. $= 1/C$
Let's satisfy this inductively!



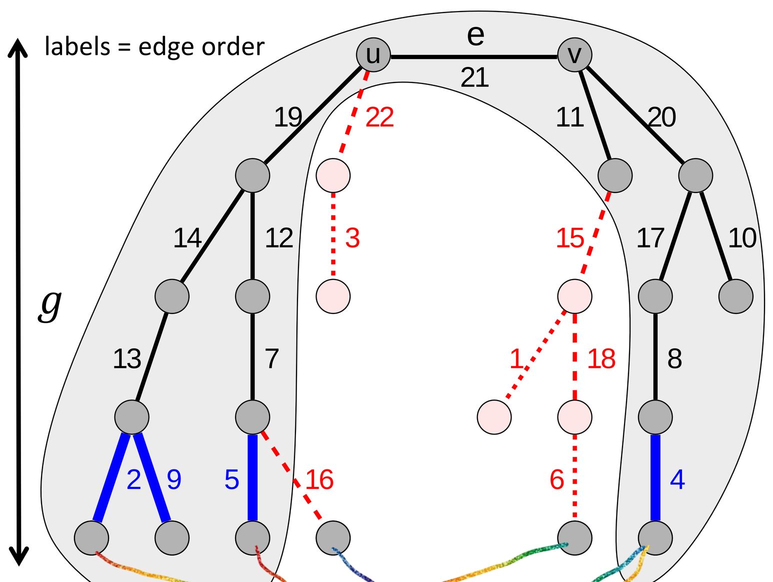
Algorithm:
If u or v matched, we can't match e.
If **u and v unmatched**, we match e with prob. p

$$P(u \text{ unmatched}) = 1 - \frac{d_u}{C} \qquad P(v \text{ unmatched}) = 1 - \frac{d_v}{C}$$
$$P(u \text{ unmatched}) \cdot P(v \text{ unmatched}) \cdot p = 1/C$$
$$p := \frac{C}{(C - d_u)(C - d_v)}$$

We just keep this algorithm for the general treelike case!

Edge matching game

Objective: show that $e = (u, v)$ gets matched with prob. $\gtrsim 1/C$



Up to distance g, we have a tree T. Beyond that, no control.

Worst-case analysis: we cede control of **boundary** to an adversary!

Edge matching game played on T:
- when **boundary edge** arrives, adversary matches or not (can randomize)
- when internal edge arrives, we follow the algorithm
- adversary plays to minimize $P(e \text{ matched})$

Observation 1: true probability $\ge$ game probability (that e gets matched)

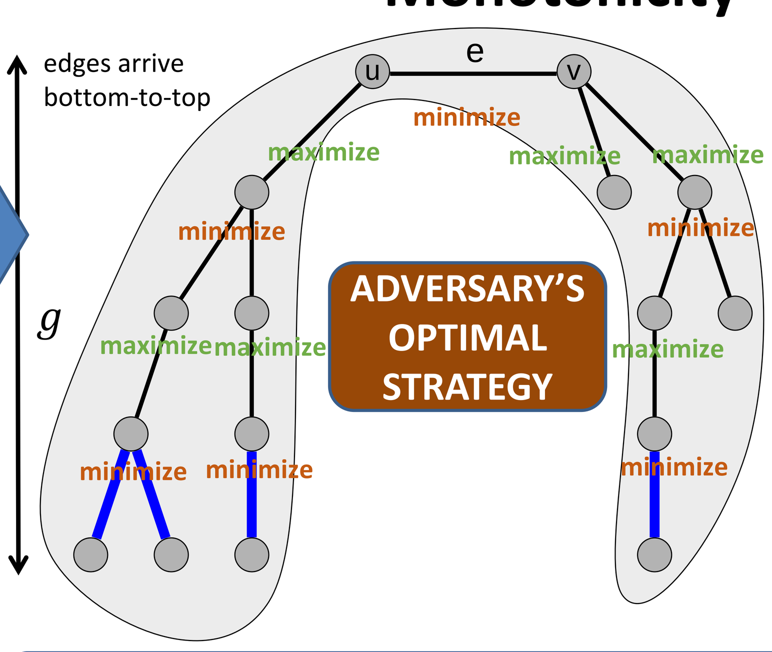
New objective: $P(e \text{ matched in edge matching game}) \ge 1/C$

Observation 2: w.l.o.g., edges arrive bottom-to-top

So red edges can be discarded from T

What is the optimal strategy for the adversary?

Monotonicity



Adversary wants to minimize prob. of e, so intuitively it's best to maximize prob. of level-1 edges, minimize level-2 edges, etc.

Monotonicity:
when g even, it is optimal to **NOT MATCH** boundary edges

Note: this is oblivious! Can make all decisions before the game

Fixing this strategy, **new objective**: $P(e \text{ matched}) \ge 1/C$
when **boundary** is removed and edges arrive bottom-to-top

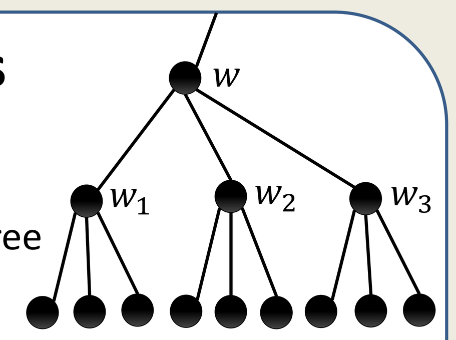
Events "u already matched", "v already matched" (upon arrival of e) are **again independent**

On the other hand, $P(u \text{ unmatched}) \neq 1 - \frac{d_u}{C}$
as we lost the $1/C$ probabilities for every edge:
e.g. for **blue** edges they are in fact 0.
But as we go up the tall tree, probs contract towards $1/C$...

Tree recurrences

$q_w := P(w \text{ is not matched from below})$

Imagine complete Δ-ary tree



$$q_w = \prod_{i=1}^{\Delta} P[(w, w_i) \text{ not matched} \mid (w, w_1), \ldots, (w, w_{i-1}) \text{ not matched}]$$
$$= \prod_{i=1}^{\Delta}\left(1 - q_{w_i} \cdot \frac{C}{(C - i + 1)(C - \Delta)}\right)$$

We like edge probabilities $\approx 1/C$
i.e. we like $q_w \approx 1 - \Delta/C$
so define error $\epsilon_w := 1 - \frac{q_w}{1 - \frac{\Delta}{C}}$

Some rewriting gives:

$$\epsilon_w = 1 - \prod_{i=1}^{\Delta}\left(1 + \frac{1}{C - i} \cdot \epsilon_{w_i}\right) \approx 1 - \exp\left(\sum_{i=1}^{\Delta} \frac{1}{C - i} \cdot \epsilon_{old}\right)$$
$$\approx -\log\left(\frac{C}{C - \Delta}\right) \cdot \epsilon_{old}$$

(ϵ_{w_i} → ϵ_{old}; $\approx \exp\left(\frac{1}{C - i} \cdot \epsilon_{old}\right)$)

So, as we go one level up:
$$\epsilon \to -\log\left(\frac{C}{C - \Delta}\right) \cdot \epsilon$$

$\log\left(\frac{C}{C - \Delta}\right) < 1$ iff $C > \frac{e}{e-1} \cdot \Delta$

If height g of tree large enough ($\omega(1)$), then $\epsilon_u, \epsilon_v \approx 0$ ∎