Lecture 4.

Randomized algorithms

Input: usual input + n random bits

Las Vegas: always right, E(running time) polynomial ZPP

Monte Carlo two-sided errors YES-instance $\longrightarrow P \geq 2/3$ BPP
NO-instance

$\boxed{\text{one-sided errors}}$

RP: NO-instance $\longrightarrow$ answer NO
Yes-instance $\longrightarrow P(\text{answer YES}) \geq p > 0$

$(1-p)^t$ $\forall x, 1+x \leq e^x$

$\leq e^{-pt}$ $t = O\left(\frac{1}{p}\right)$

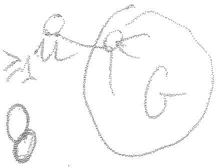
$\boxed{\text{polynomial algorithm}}$ with some success prob $\frac{1}{n^{O(1)}}$

US: FPT algorithm, prob $\frac{1}{f(k)n^{O(1)}}$

Prime numbers $\in$ BPP $\rightarrow$ 2002 $\in$P
PIT

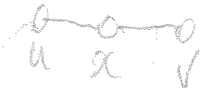
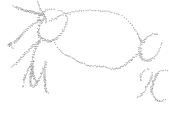
Feedback Vertex Set

Reduction rules   G graph

R1)  → remove u from the graph

R2)  → more than 2 parallel edges 

R3)  → remove u from G, include u in X_0, $k--$

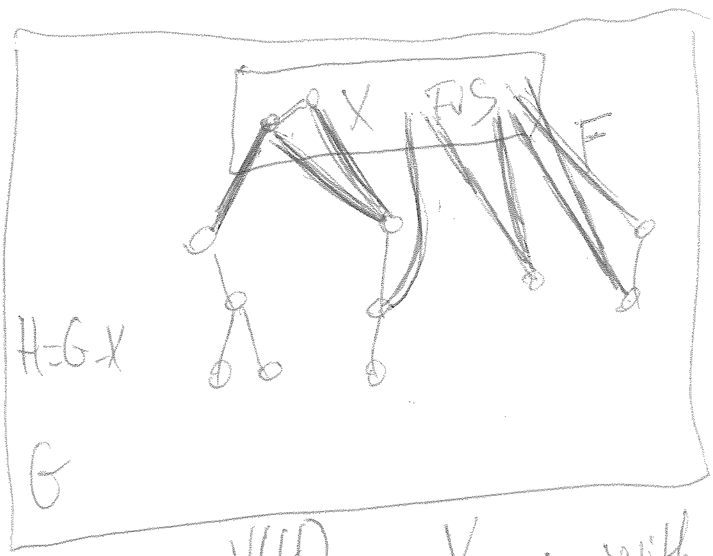
R4)  →     → 

R5) If $k<0$ → answer "No"

↳ (G^{**}, k^{**}) min degree 3 

F: edges going from X to V(H)

Claim: $|F| > |E(H)|$



$|E(H)| < |V(H)|$

$|F| \geq |V(H)|$

V(H) $V_{\leq 1}$: with degree at most 1 in H

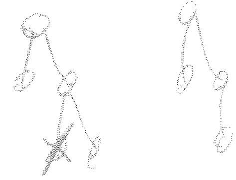
V_2: ——— 2 ——— in H

$V_{\geq 3}$: ——— ≥ 3 ——— in H

$|F| \geq 2|V_{\leq 1}| + |V_2| \geq 2|V_{\leq 1}| + |V_2| + |V_{\geq 3}| \geq |V(H)| > |E(H)|$

≥ 2 red edges   1 red edge

$|V_{\leq 1}| \geq |V_{\geq 3}|$?   induction



So? Pick a vertex

Take an edge uv u.a.r. $E(G) \longrightarrow P(uv \in F) \geq \frac{1}{2}$

x: Flip a coin and decide u or v $\longrightarrow P(x \in X) \geq \frac{1}{4}$

Algorithm: Apply R1–5) $\longrightarrow$ No

$\downarrow$

X_0

$\downarrow$

x Partial solution $S := X_0 \cup \{x\}$

k times $P \geq \frac{1}{4}$

Total success probability 4^{-k}

$t = O(4^k) \longrightarrow 4^k n^{O(1)}$

□

Color coding. Put random colors to your instance
And try to see if it's easier to solve

k-Path: Input: G (graph), integer k
Output: If there is a ~~path~~ simple path $v_1 v_2 \dots v_k$ in G

$k = |V(G)| \to$ Hamiltonian Path NP-complete

$k^{O(k)} n^{O(1)}$ was known $\to 2^{\log^2 n}$

Papadimitriou & Yannakakis

$\log n$ - path

$\hookrightarrow$ Alon, Yuster, Zwick '95

Color with k colors u.a.r

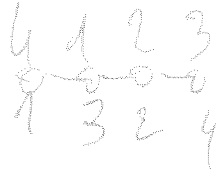
$\hookrightarrow$ solution multicolored S has each color of $[k] = \{1, \dots k\}$ exactly once

Prop: If you color $\chi : V(G) \longrightarrow [k]$.

For every $X \in \binom{V(G)}{k}$, $P(X \text{ is multicolored}) \geq e^{-k}$

Proof: $\dfrac{k!\, k^{n-k}}{k^{n}} = k!\, k^{-k} \geq e^{-k}$ □

$V(G) \setminus X$



Uniformly Randomly color χ $V(G)$ with k colors

Focus on a situation where a fixed solution P is multicolored

Multicolored k-Path.

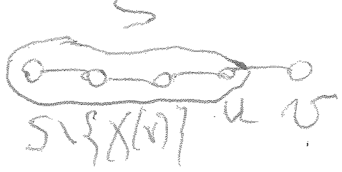
Design a dynamic programming $2^k \cdot n$

↑ Size of the DP table.

$T[S \subseteq [k], v \in V(G)] = \exists$ path P' rooted at v and P' is using the colors of S

$\bigvee_{v \in V(G)} T[[k], v]$ = Solution

- $T[\{c\}, v] = [\chi(v) = c]$
- $T[S, v] = \begin{cases} \text{False} & \chi(v) \notin S \\ \bigvee_{u \in V(G)} T[S \setminus \chi(v), u] \wedge uv \in E(G) & \end{cases}$



$\longrightarrow 2^k n^{O(1)}$

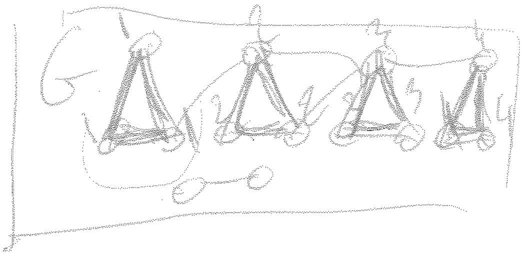
$P \geq e^{-k}$ success

$\hookrightarrow (2e)^k \, n^{O(1)}$-time randomized algorithm for k-Path

5.4^k

<u>Triangle Packing</u> Input: Graph G, integer k

Output: k vertex-disjoint triangles G



Question: Solve Triangle Packing in randomized

$2^{O(k)} \, n^{O(1)}$-time (using color coding)

only $3k$ vertices involved

$\hookrightarrow$ each $G[V_i]$

$V_i = \{ S \subseteq V(G)$, with color $i \}$

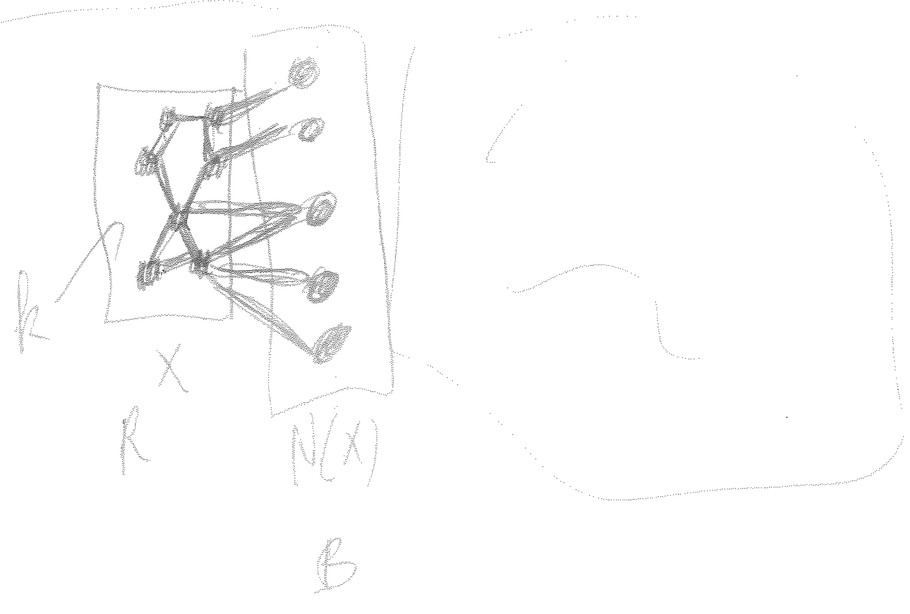
$P(X \text{ is well-colored}) \geq c^{-k}$?

$\geq k^{-2k} \rightarrow k^{O(k)} n^{O(1)}$

$k!$ good colorings

k^{3k} ← total number of colorings

Random Separation. random 2-colouring



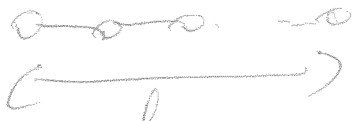
bounded degree $\leq d$ Δ parameter. $\boxed{k+\Delta}$ $f(k,\Delta)$ $g(k+\Delta)$

Bounded degree Subgraph Isomorphism: Input: Graphs G, H

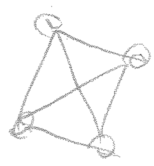
Output: Is H isomorphic to a subgraph of G?

$k := |V(H)|$

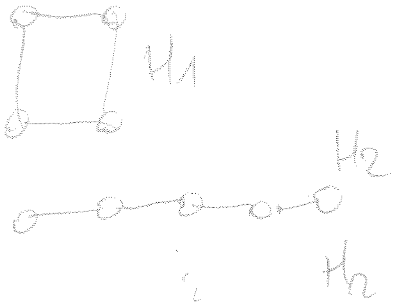
If $H = \underbrace{\circ\!-\!\circ\!-\!\circ \cdots -\!\circ}_{k}$

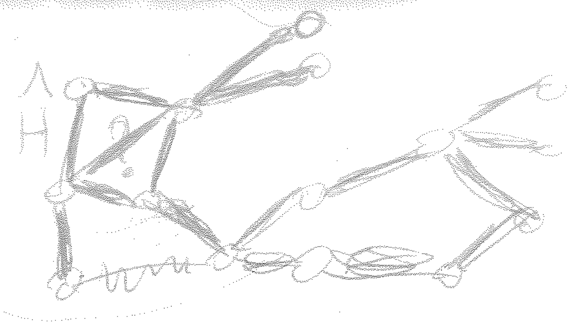


$H =$



k-Clique does not have an FPT algorithm

H = 



Red: edges of $\hat{H}$ isomorphic to H, $E(\hat{H})$

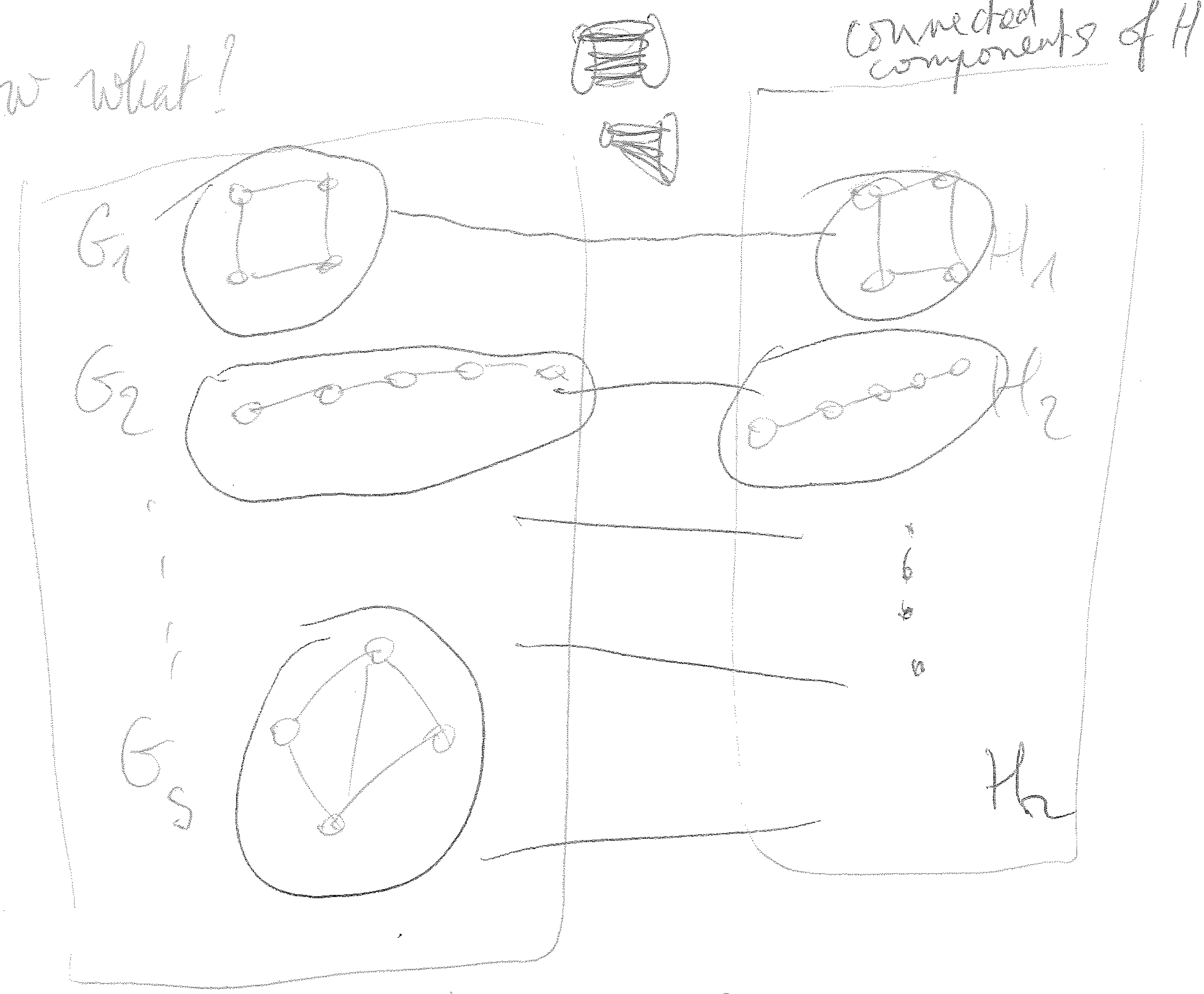
Blue (Γ): all the ~~other~~ edges incident to at least one vertex in $V(\hat{H})$

$|\Gamma \cup E(\hat{H})| \le k\Delta \qquad V(\hat{H}) = k$

$P(\text{success}) \ge 2^{-k\Delta}$

Q: Now what?

red connected components induced by G, red edges only

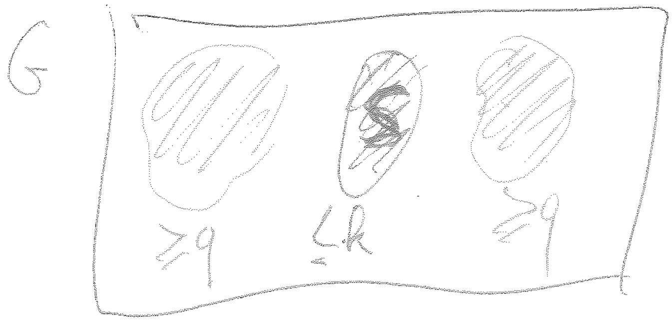


$k^{\Delta \log \Delta}\; 2^{k\Delta}\; n^{O(1)}$ - time

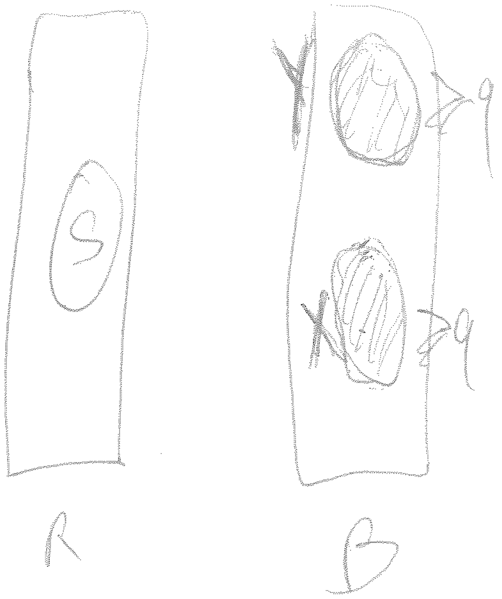
polynomial $k^{\Delta \log \Delta}$

Pb: delete (k) vertices, such that it disconnects the graph into ≥ 2 C.C of size at least (q). G, k, q.

Question: FPT algorithm $k+q$ for this problem.



on the vertices



$p \geq \frac{1}{2^{k+2q}}$

min X,Y - cut

G(B)