

Shorter Implicit Representation for Planar Graphs

Cyril Gavoille

LABORATOIRE
BORDELAIS
DE RECHERCHE
EN INFORMATIQUE

LaBRI - *University of Bordeaux*

Joint with Marthe Bonamy (U. Bordeaux)

and Michał Pilipczuk (U. Warsaw)

9th Slovenian International Conference on Graph Theory, Bled 2019

BLÉD, SLOVENIA, JUNE 23 – JUNE 29, 2019

Representation of a Graph



adjacency list

1	2,5
2	1,3
3	2,4,5
4	3
5	1,3

matrix

	1	2	3	4	5
1	0	1	0	0	1
2	1	0	1	0	0
3	0	1	0	1	1
4	0	0	1	0	0
5	1	0	1	0	0

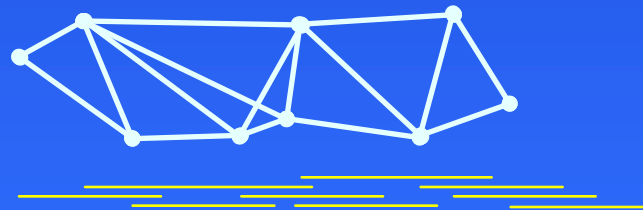
1 node = 1 pointer in the data-structure
(it does not carry any specific information)

Implicit Representation

To associate with the nodes more information, typically adjacency, and to remove the data-structure.

Interval graphs: $u \mapsto I(u) \subseteq [1, 2n]$

Edges: $u-v \Leftrightarrow I(u) \cap I(v) \neq \emptyset$



Compact representation: $O(\log n)$ bits/node

Possibly time $O(n)$ algorithms vs. $O(n+m)$

Labelling Schemes

P = a graph property defined on pairs of nodes

F = a graph family

A P -labelling scheme for F is a pair (λ, f) such that $\forall G \in F, \forall u, v \in V(G)$:

- [labelling] $\lambda(u, G)$ is a binary string
- [decoder] $f(\lambda(u, G), \lambda(v, G)) = P(u, v, G)$

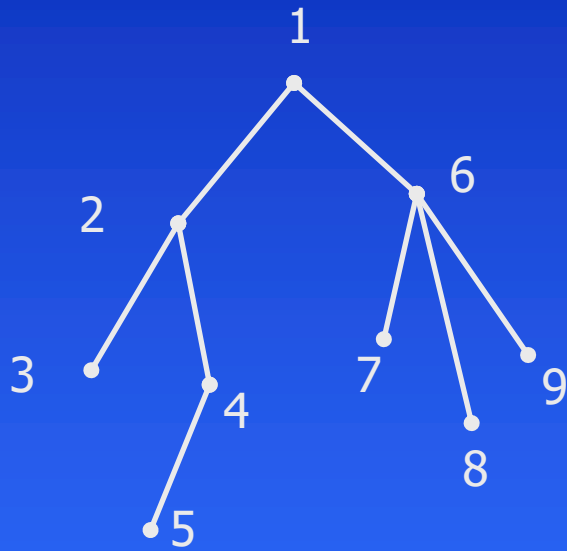
Goal: to minimize the maximum label size

In this talk: $P(u, v, G)$ is TRUE $\Leftrightarrow uv \in E(G)$

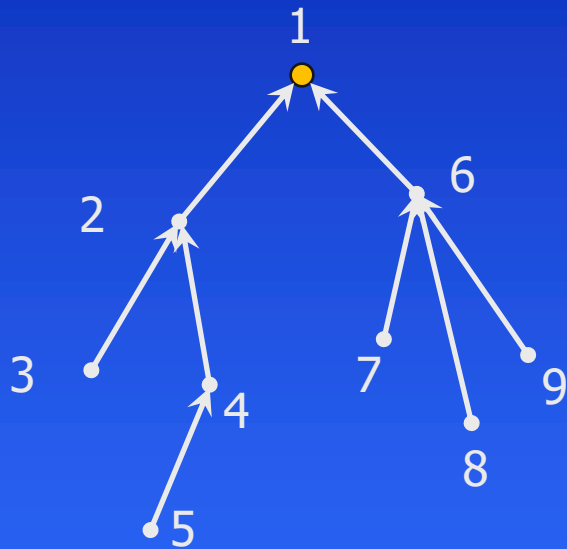
Basic Example: Trees



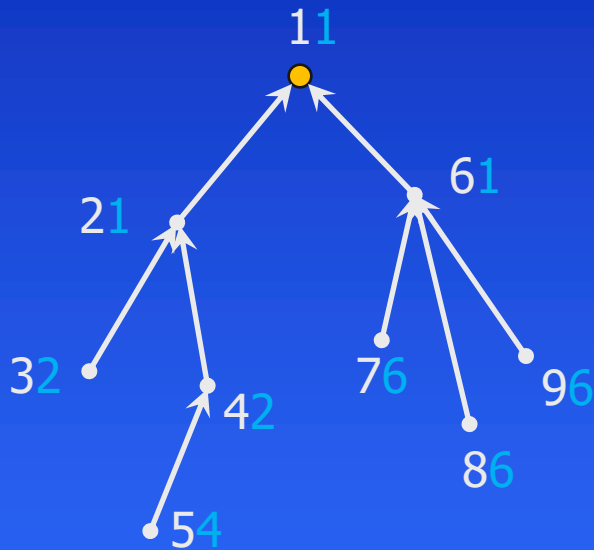
Basic Example: Trees



Basic Example: Trees



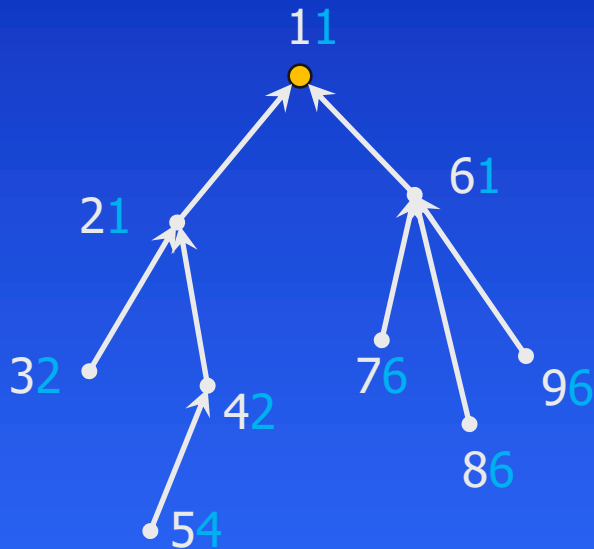
Basic Example: Trees



$\lambda(u, T) = (u, \text{parent}(u))$ or (u, u)

$f(uv, xy) = (v = x$ or $u = y)$

Basic Example: Trees



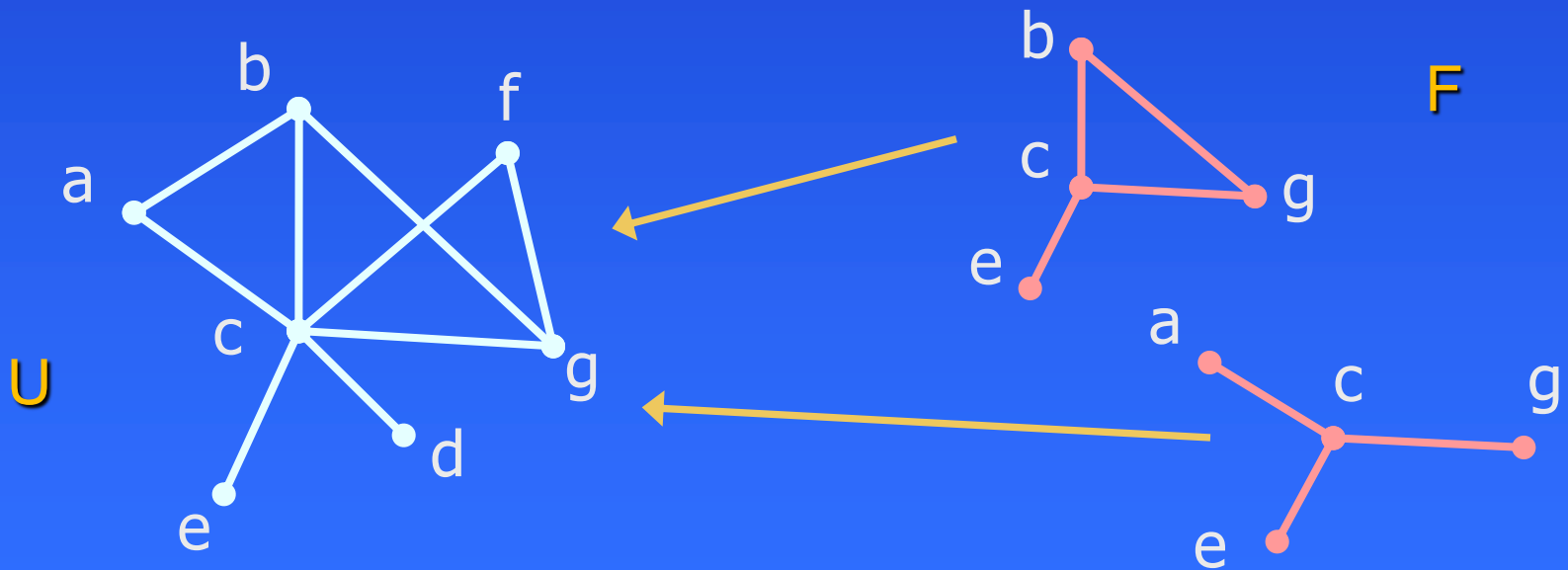
$$\lambda(u, T) = (u, \text{parent}(u)) \text{ or } (u, u)$$
$$f(uv, xy) = (v=x \text{ or } u=y)$$

For trees with n nodes: $\sim 2 \log n$ bits/node
(the constant does matter [Abiteboul et al. - SICOMP '06])

Induced-Universal Graphs

[Babai, Chung, Erdős, Graham, Spencer '82]

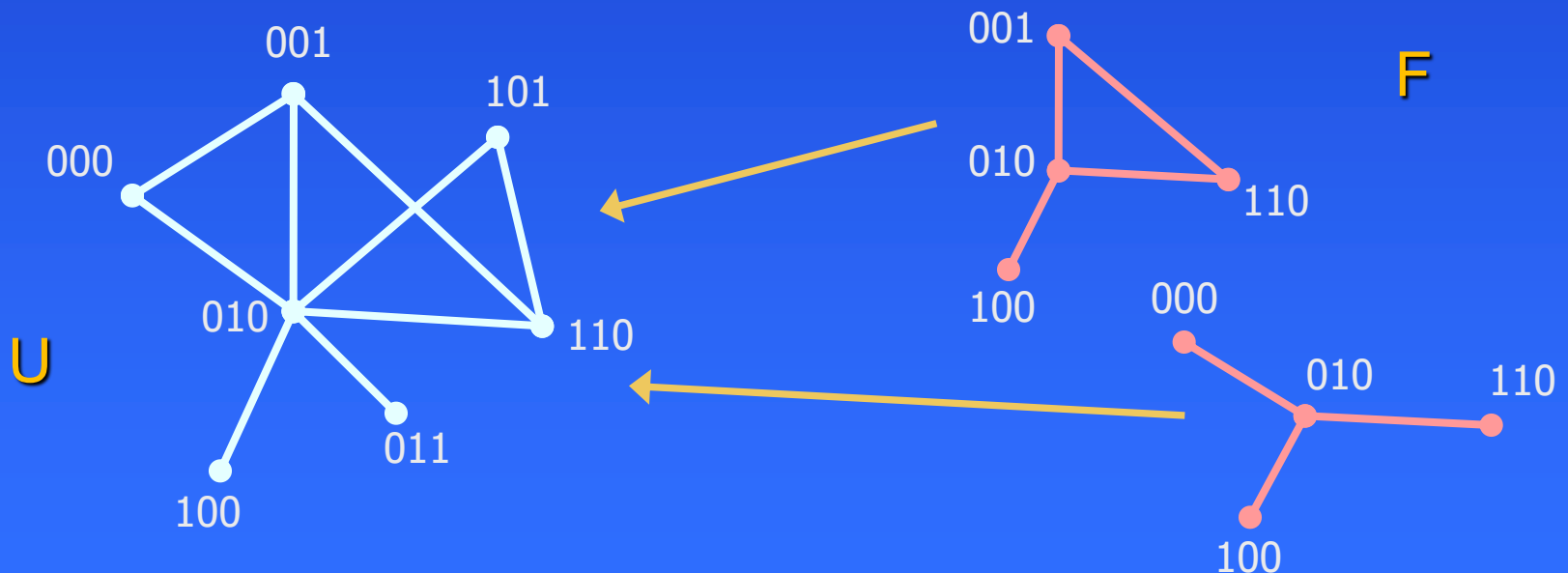
A graph U is an **induced-universal** graph for the family F if every graph of F is isomorphic to an *induced* subgraph of U .



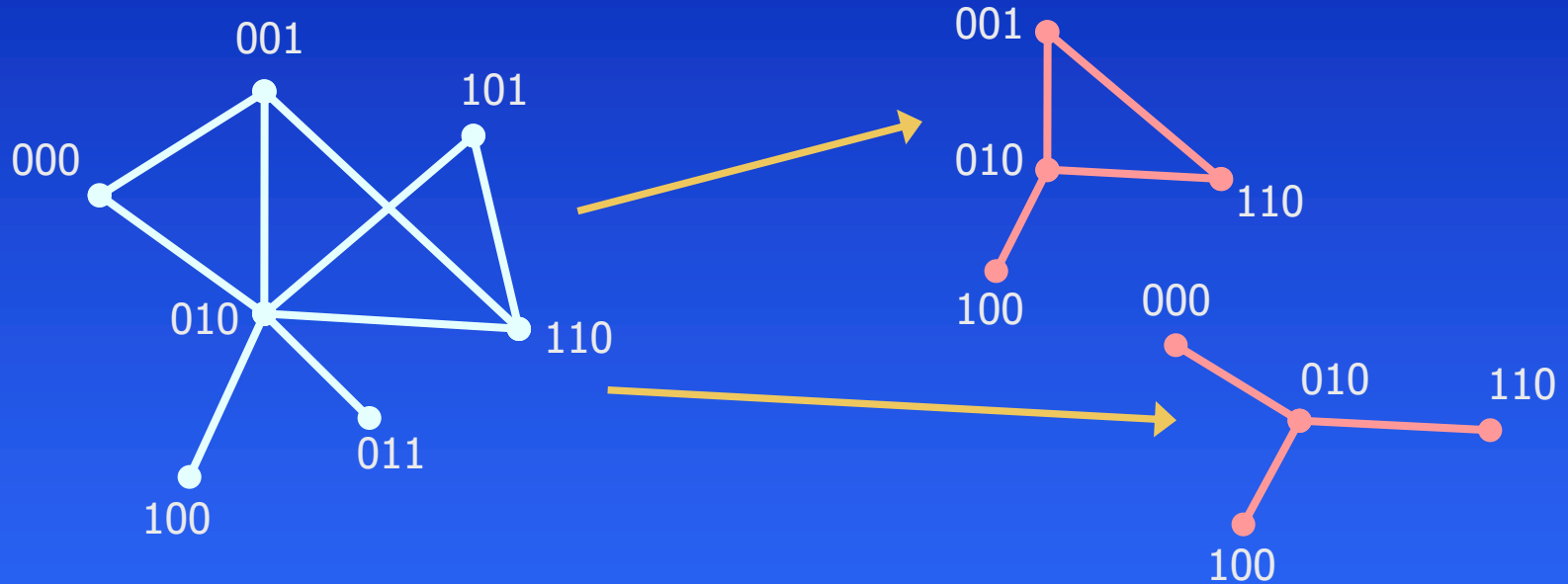
Induced-Universal Graphs

[Babai, Chung, Erdős, Graham, Spencer '82]

A graph U is an **induced-universal** graph for the family \mathcal{F} if every graph of \mathcal{F} is isomorphic to an *induced* subgraph of U .



Induced-Universal Graphs



induced-universal graph U

graphs of F

$c \log n$ -bit labelling \Leftrightarrow induced-universal graph of n^c nodes

Universal Graphs for Trees (for $n=6$ nodes)

From the $(u, \text{parent}(u))$ labelling
 \Rightarrow induced-universal graph of $n^2=36$ nodes



Using DFS for T: (u, v)
 $\Rightarrow u > v$ or $u = v = 1$

Universal Graphs for Trees (for $n=6$ nodes)

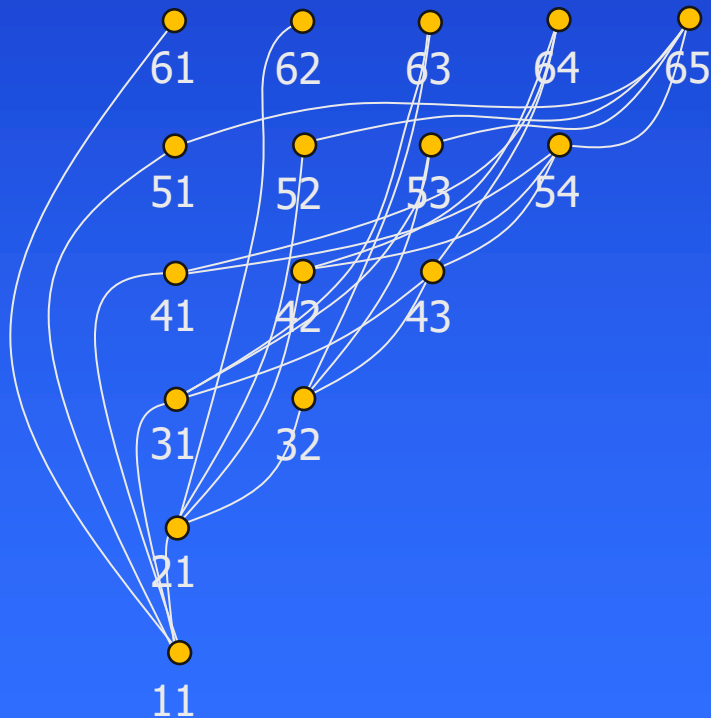
From the $(u, \text{parent}(u))$ labelling
 \Rightarrow induced-universal graph of $n^2=36$ nodes



Using DFS for T: (u, v)
 $\Rightarrow u > v$ or $u = v = 1$
 $\Rightarrow n(n-1)/2 + 1 = 16$ nodes

Universal Graphs for Trees (for $n=6$ nodes)

From the $(u, \text{parent}(u))$ labelling
 \Rightarrow induced-universal graph of $n^2=36$ nodes



Using DFS for $T: (u, v)$
 $\Rightarrow u > v$ or $u = v = 1$
 $\Rightarrow n(n-1)/2 + 1 = 16$ nodes

Universal Graphs for Trees (universal trees)

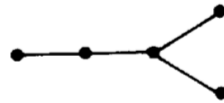
$n = 1$:



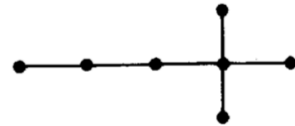
$n = 2$:



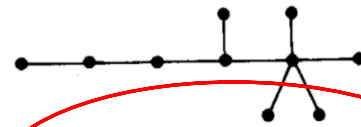
$n = 3$:



$n = 4$:



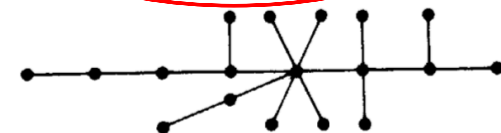
$n = 5$:



$n = 6$:



$n = 7$:

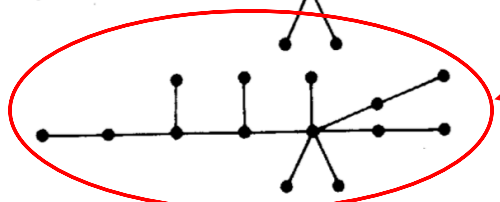


ON GRAPHS WHICH CONTAIN ALL SMALL TREES, II.

F.R.K. CHUNG — R.L. GRAHAM — N. PIPPENGER

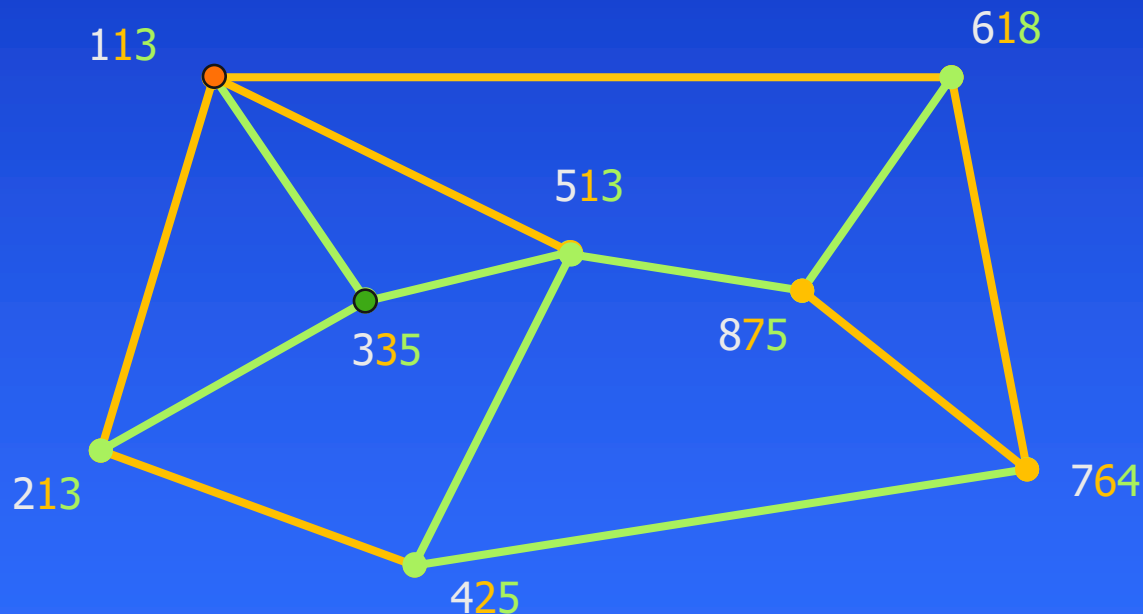
COLLOQUIA MATHEMATICA SOCIETATIS JÁNOS BOLYAI
18. COMBINATORICS, KESZTHELY (HUNGARY), 1976.

14 nodes



Labelling Schemes for Planar Graphs

Edge partition: combining labelling schemes



Arboricity- k graphs: $(k+1)\log n$ bits
 \Rightarrow Planar ($k=3$): **4** $\log n$ bits [KNR - STOC'88]

Better Labelling Schemes

For trees: $\log n + O(\log^* n)$

[Alstrup, Rauhe – FOCS'02]

\Rightarrow Arboricity- k : $k \log n + O(\log^* n)$

\Rightarrow Planar: **3** $\log n + O(\log^* n)$

For treewidth- k : $\log n + O(k \log \log n)$

[G., Labourel – ESA'07]

\Rightarrow Planar & Minor-free: **2** $\log n + O(\log \log n)$

For trees: $\log n + O(1)$

[Alstrup, Dahlgaard, B.T. Knudsen – FOCS'15 & JACM'17]

A New Bound

For planar & bounded genus: $\frac{4}{3} \log n + O(\log \log n)$

\Rightarrow induced-universal graph of $n^{4/3+o(1)}$ nodes for n -node planar graphs (and bounded genus graphs)

Labelling the nodes is polynomial

Decoding adjacency takes constant time

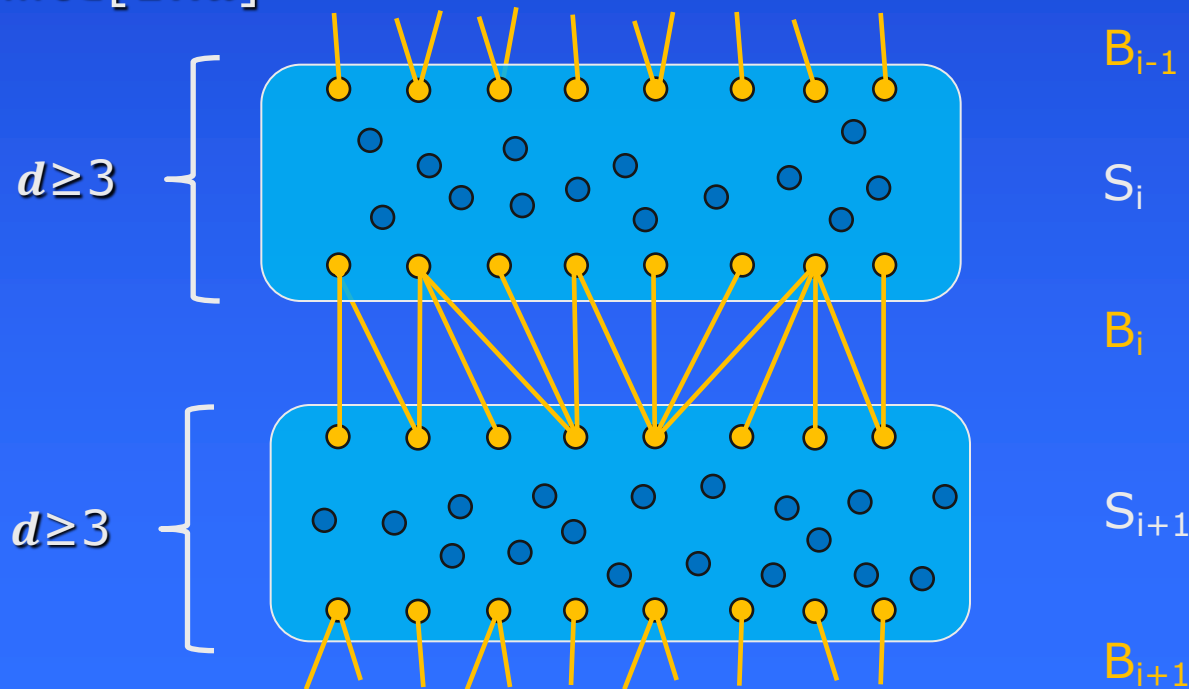
Sketch of Proof (1/2)

Edge partition: $G = S \cup B$ (Strips & Border)

S: components have depth $d \sim n^{1/3}$

B: has treewidth ≤ 5 and $n/d \sim n^{2/3}$ nodes

BFS & $\text{shift} \in [1..d]$



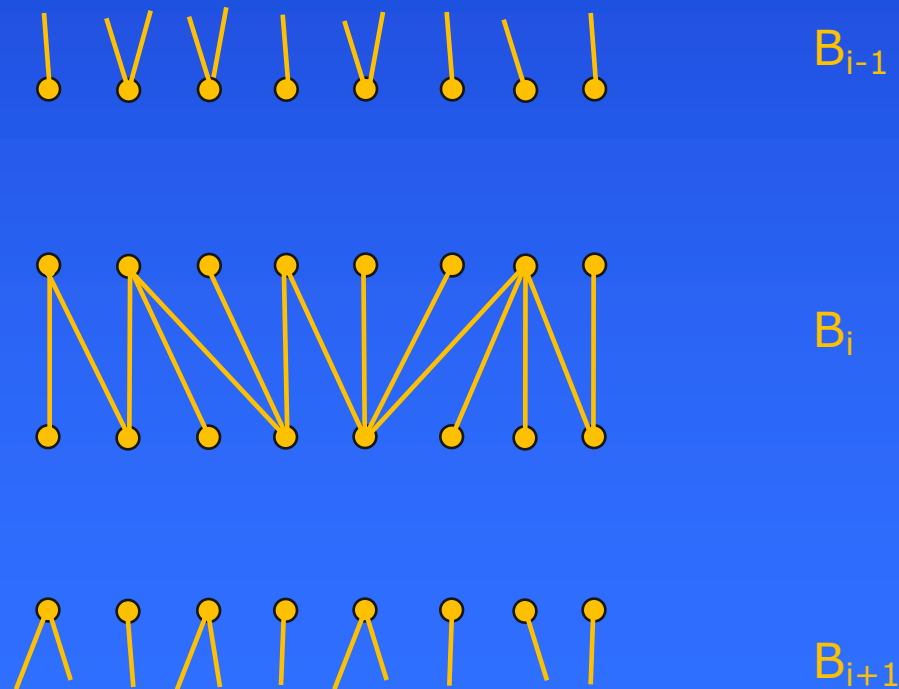
Sketch of Proof (1/2)

Edge partition: $G = S \cup B$ (Strips & Border)

S: components have depth $d \sim n^{1/3}$

B: has treewidth ≤ 5 and $n/d \sim n^{2/3}$ nodes

BFS & shift $\in [1..d]$



Sketch of Proof (2/2)

Labelling for B: $\log(n/d)$



Labelling for S: $\log n + \log d$

new!

[up to $+O(\log \log n)$ terms]

Problem: nodes in $V(B)$ pay both labels

$$\Rightarrow \log(n/d) + \log n + \log d = 2 \log n \quad \text{😞}$$

Improved labelling for S: nodes in $V(B)$ pay only $\log|B| = \log(n/d)$ bits!

$$\Rightarrow \text{nodes in } S \setminus V(B): \log n + \log d = \frac{4}{3} \log n \quad \text{😊}$$

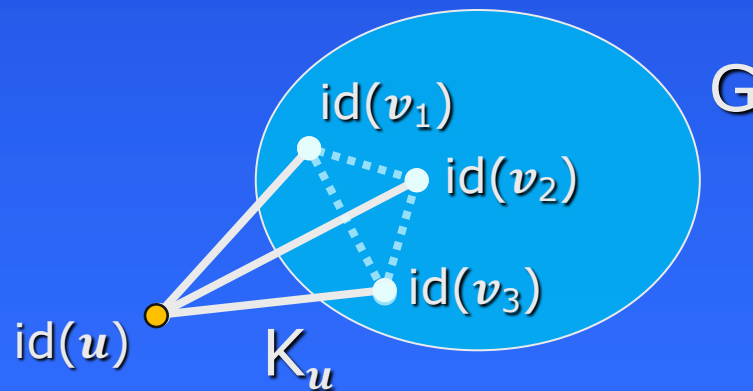
$$\Rightarrow \text{nodes in } V(B): \log(n/d) + \log(n/d) = \frac{4}{3} \log n \quad \text{😊}$$

Improved Scheme for Treewidth- k

$G = \text{treewidth-}k$, $V(G) = V_1 \cup V_2$, $K_u = \text{simplicial complex of } u$, $|K_u| \leq k+1$.

Lemma. G has a scheme providing, for each u , $\text{id}(u)$ and $\lambda(u)$ st. $\text{id}(v)$ can be extracted $\forall v \in K_u$ from $\lambda(u)$. Moreover, for $u \in V_i$

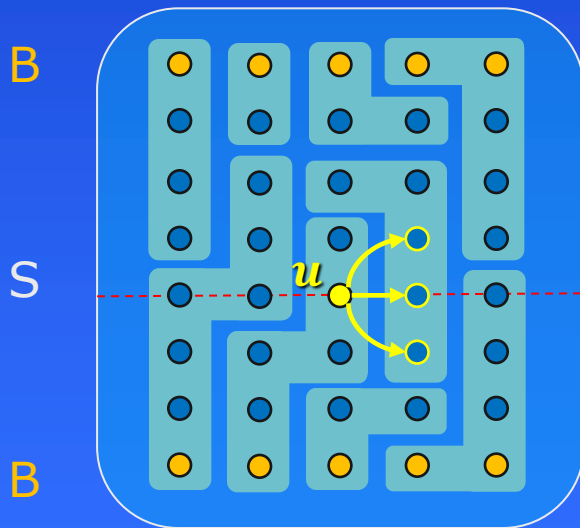
$$|\lambda(u)| = \log |V_i| + O(k \log \log |V(G)|).$$



$$u-v \Leftrightarrow \text{id}(v) \in \{\text{id}(K_u)\} \text{ or } \text{id}(u) \in \{\text{id}(K_v)\}$$

Labelling Scheme for S

Key lemma. [2018,2019] If G is planar, there is a node partition into monotone paths taken from any given BFS such that contracting each one leads to a treewidth-8 graph.



Label $\lambda(u)$ consists of:

- the treewidth-8 scheme
- the depth of u in S (unless if $u \in B$)
- 3 bits/path in the treewidth-8 scheme, i.e., $3 \times 8 = 24$ extra bits

[Extend to genus- g graphs]

Open Problems

1. Improve to $c \log n$ with $c < 4/3$ for planar
2. Extend to minor-free graphs
3. Improve to $\log n + \theta(k)$ for treewidth- k
4. Prove lower bounds for planar or minor-free

Best lower bound for planar: $\log n + \Omega(1)$ ☹️

No hereditary family with $n! 2^{O(n)}$ labelled graphs (trees, planar, bounded genus, bounded treewidth, minor-free ...) is known to require labels of $\log n + \omega(1)$ bits.

THANK YOU!