Oriented trees in digraphs of high chromatic number

Amadeus Reinald
LIRMM, Université de Montpellier, CNRS

joint work with
Stéphane Bessy, Daniel Gonçalves

Journées Graphes et Algorithmes
21 Novembre 2023, Lyon

Preliminaries

Question

What structures necessarily appear in (di)graphs of "high" chromatic number?

Preliminaries

Question

What structures necessarily appear in (di)graphs of "high" chromatic number?

- minors, Hadwiger's conjecture: $\chi \geq t \rightarrow K_t$

Preliminaries

Question

What structures necessarily appear in (di)graphs of "high" chromatic number?

- minors, Hadwiger's conjecture: $\chi \geq t \rightarrow K_t$
- induced subgraphs, Gyárfás-Sumner conjecture: $\chi \geq f(t, T) \rightarrow K_t$ or T

Preliminaries

Question

What structures necessarily appear in (di)graphs of "high" chromatic number?

- minors, Hadwiger's conjecture: $\chi \geq t \to K_t$
- induced subgraphs, Gyárfás-Sumner conjecture: $\chi \geq f(t, T) \to K_t$ or T
- subgraphs: solved, trivial (next slide)

Preliminaries

Question

What structures necessarily appear in (di)graphs of "high" chromatic number?

- minors, Hadwiger's conjecture: $\chi \geq t \to K_t$
- induced subgraphs, Gyárfás-Sumner conjecture: $\chi \geq f(t, T) \to K_t$ or T
- subgraphs: solved, trivial (next slide)
- sub**di**graphs: **this talk!**

Subtrees in graphs of large chromatic number

Question

Given H, is there some k s.t. $\chi(G) \geq k \implies G$ contains H as a subgraph?

Subtrees in graphs of large chromatic number

Question

Given H, is there some k s.t. $\chi(G) \geq k \implies G$ contains H as a subgraph?

Theorem (Erdős, Hajnal '1966)

There exist graphs of arbitrarily large girth and χ.

Subtrees in graphs of large chromatic number

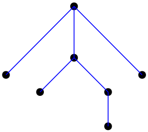
Question
Given H, is there some k s.t. $\chi(G) \geq k \implies G$ contains H as a subgraph?

Theorem (Erdős, Hajnal '1966)
There exist graphs of arbitrarily large girth and χ.

$\hookrightarrow$ if H is not a tree, **no**.

χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.



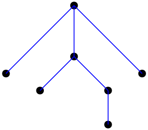
,

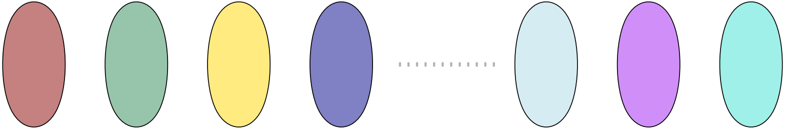
χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

- start with any colouring

,

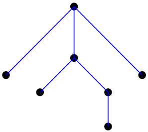


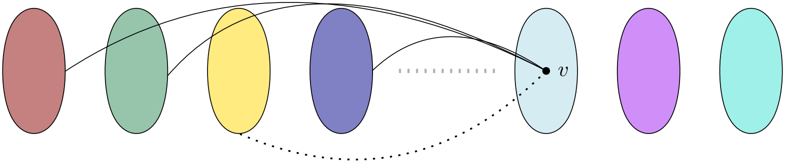


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

- start with any colouring
- stack left,

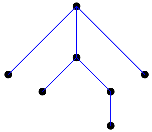


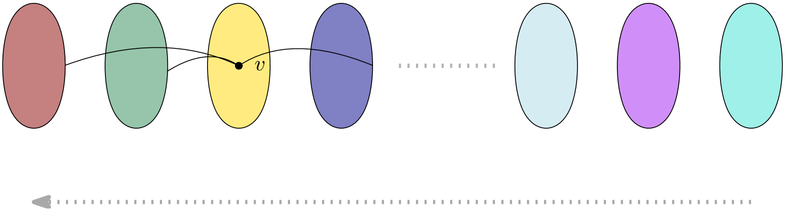


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

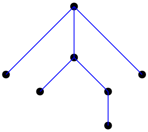
- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*



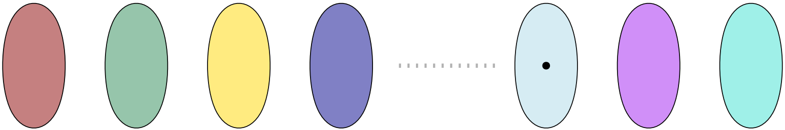


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

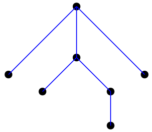


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$

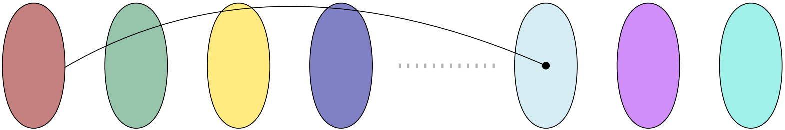


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

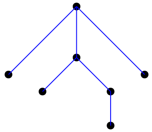


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$

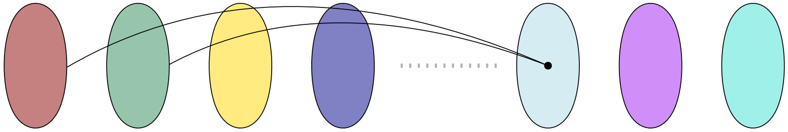


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

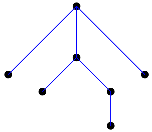


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$

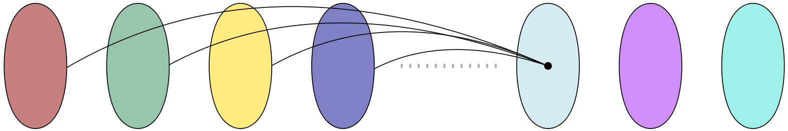


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.



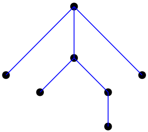
- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$

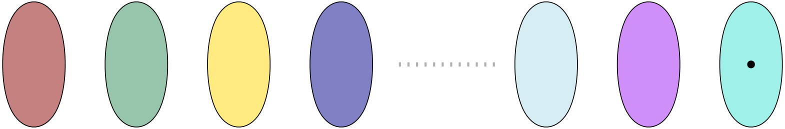


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

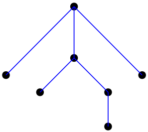
- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T



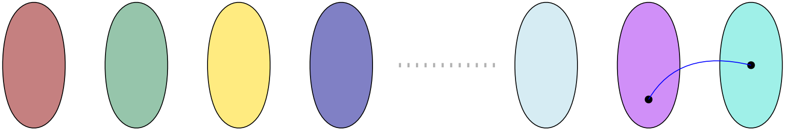


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

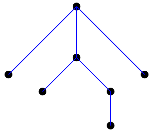


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

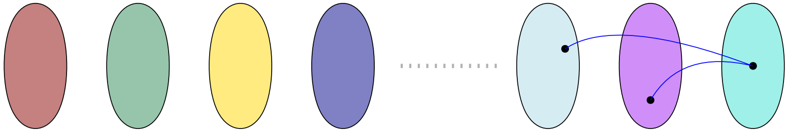


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

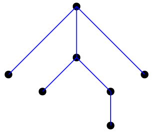


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

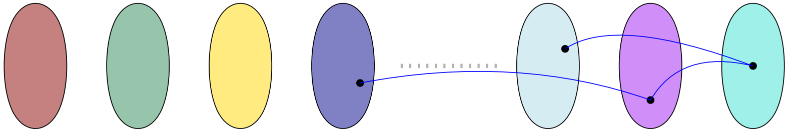


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

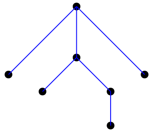


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

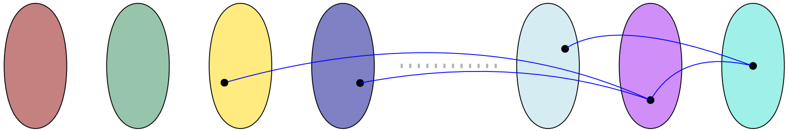


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.



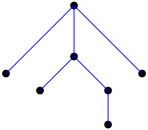
- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

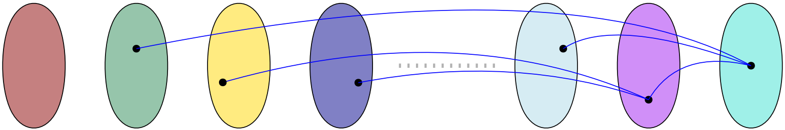


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

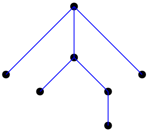
- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T



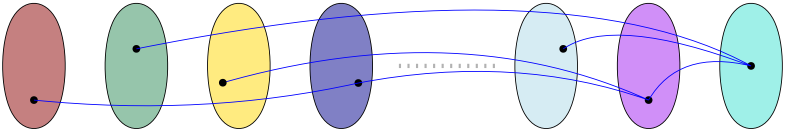


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

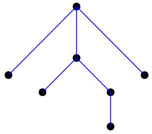


- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

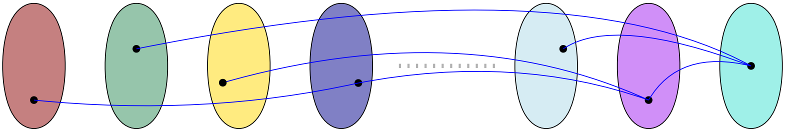


χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.



- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

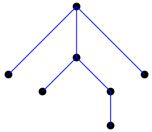


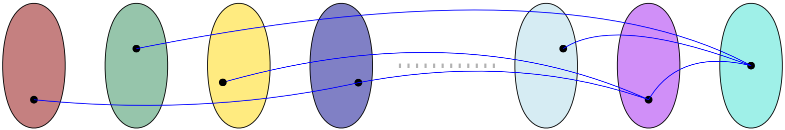
- What if we **orient** G?

χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.

- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T

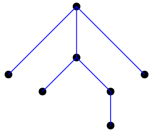




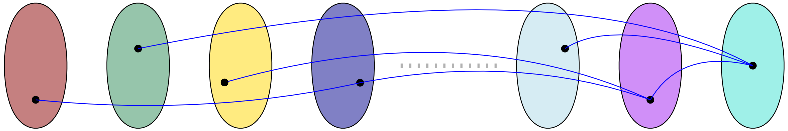
- What if we **orient** G? No guarantees on directions of arcs...

χ-vertex trees in simple graphs

Graph G: find trees on $\chi(G)$ vertices as subgraphs, **tight**.



- start with any colouring
- stack left, still $\geq \chi$ classes *(independent sets)*
- class i: at least one neighbour in $j < i$ build T



- What if we **orient** G? No guarantees on directions of arcs...
- *This talk*: find k-vertex oriented subtrees when $\chi \geq f(k)$.

Burr's conjecture

Conjecture (Burr '80)
If a digraph D satisfies $\chi(D) \geq 2k-2$, then D contains every oriented tree on k vertices as a subgraph.

Burr's conjecture

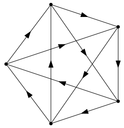
Conjecture (Burr '80)
If a digraph D satisfies $\chi(D) \geq 2k-2$, then D contains every oriented tree on k vertices as a subgraph.

- **Tight** : diregular K_{2k-3} has no S_k^+ *(ex: $k=4$)*,

Burr's conjecture

Conjecture (Burr '80)
If a digraph D satisfies $\chi(D) \geq 2k-2$, then D contains every oriented tree on k vertices as a subgraph.

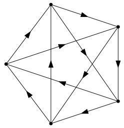
- **Tight** : diregular K_{2k-3} has no S_k^+ *(ex: $k=4$)*,
- $(k-1)^2$ suffices for k-vertex trees [Burr '80].



Burr's conjecture

Conjecture (Burr '80)
If a digraph D satisfies $\chi(D) \geq 2k-2$, then D contains every oriented tree on k vertices as a subgraph.

- **Tight** : diregular K_{2k-3} has no S_k^+ *(ex: $k = 4$)*,
- $(k-1)^2$ suffices for k-vertex trees [Burr '80].

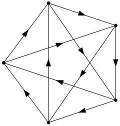


Theorem (Addario-Berry, Havet, Linhares Sales, Reed, Thomassé '13)
If a digraph D satisfies $\chi(D) \geq \frac{k^2}{2} - \frac{k}{2} + 1$, then D contains every oriented tree on k vertices.

Burr's conjecture

Conjecture (Burr '80)
If a digraph D satisfies $\chi(D) \geq 2k-2$, then D contains every oriented tree on k vertices as a subgraph.

- **Tight** : diregular K_{2k-3} has no S_k^+ *(ex: $k=4$)*,
- $(k-1)^2$ suffices for k-vertex trees [Burr '80].



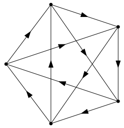
Theorem (Addario-Berry, Havet, Linhares Sales, Reed, Thomassé '13)
If a digraph D satisfies $\chi(D) \geq \frac{k^2}{2} - \frac{k}{2} + 1$, then D contains every oriented tree on k vertices.

- $\geq 2k-2 \implies$ paths, two-blocks paths*, diameter 3 trees

Burr's conjecture

Conjecture (Burr '80)
If a digraph D satisfies $\chi(D) \geq 2k-2$, then D contains every oriented tree on k vertices as a subgraph.

- **Tight** : diregular K_{2k-3} has no S_k^+ *(ex: $k=4$)*,
- $(k-1)^2$ suffices for k-vertex trees [Burr '80].



Theorem (Addario-Berry, Havet, Linhares Sales, Reed, Thomassé '13)
If a digraph D satisfies $\chi(D) \geq \frac{k^2}{2} - \frac{k}{2} + 1$, then D contains every oriented tree on k vertices.

- $\geq 2k-2 \implies$ paths, two-blocks paths*, diameter 3 trees
- $\geq 10k \implies$ antidirected trees, ≤ 4 blocks paths

Our results

Theorem (Bessy, Gonçalves, R. '23+)
If a digraph D satisfies $\chi(D) \geq 5k\sqrt{k}$, then D contains every oriented tree on k vertices.

Our results

Theorem (Bessy, Gonçalves, R. '23+)
If a digraph D satisfies $\chi(D) \geq 5k\sqrt{k}$, then D contains every oriented tree on k vertices.

In particular,

- $\chi(D) \geq (b-1)k \implies D$ contains k-paths with b blocks $(b \geq 2)$.

Our results

Theorem (Bessy, Gonçalves, R. '23+)
If a digraph D satisfies $\chi(D) \geq 5k\sqrt{k}$, then D contains every oriented tree on k vertices.

In particular,

- $\chi(D) \geq (b-1)k \implies D$ contains k-paths with b blocks ($b \geq 2$).
- $\chi(D) \geq 3k\sqrt{k} \implies D$ contains k-arborescences.

Our results

Theorem (Bessy, Gonçalves, R. '23+)
If a digraph D satisfies $\chi(D) \geq 5k\sqrt{k}$, then D contains every oriented tree on k vertices.

In particular,

- $\chi(D) \geq (b-1)k \implies D$ contains k-paths with b blocks $(b \geq 2)$.
- $\chi(D) \geq 3k\sqrt{k} \implies D$ contains k-arborescences. **This talk!**

Our results

Theorem (Bessy, Gonçalves, R. '23+)
If a digraph D satisfies $\chi(D) \geq 5k\sqrt{k}$, then D contains every oriented tree on k vertices.

In particular,

- $\chi(D) \geq (b-1)k \implies D$ contains k-paths with b blocks $(b \geq 2)$.
- $\chi(D) \geq 3k\sqrt{k} \implies D$ contains k-arborescences. **This talk!**

Definition *An **out-arborescence** is an oriented tree with:*

- *unique source u, **root***
- *$\forall v \neq u$, one path from u to v.*

Our results

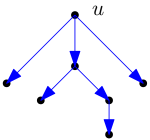
Theorem (Bessy, Gonçalves, R. '23+)
If a digraph D satisfies $\chi(D) \geq 5k\sqrt{k}$, then D contains every oriented tree on k vertices.

In particular,

- $\chi(D) \geq (b-1)k \implies D$ contains k-paths with b blocks ($b \geq 2$).
- $\chi(D) \geq 3k\sqrt{k} \implies D$ contains k-arborescences. **This talk!**

Definition *An **out-arborescence** is an oriented tree with:*

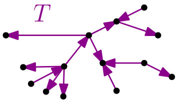
- *unique source u, **root***
- *$\forall v \neq u$, one path from u to v.*



Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

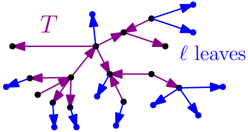
- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,



Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

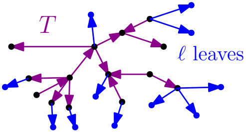
- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,
- *then* $\chi(D) \geq c + 2(k+\ell) \implies T$ *with* ℓ *out (/in) leaves glued to* T.

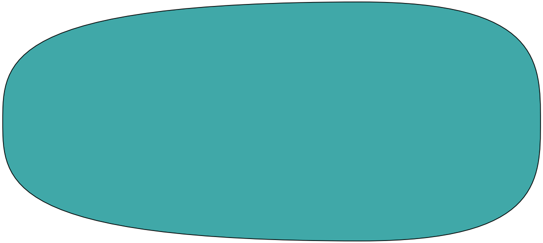


Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,
- *then* $\chi(D) \geq c + 2(k + \ell) \implies T$ *with* ℓ *out (/in) leaves glued to* T.



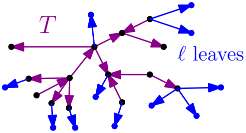


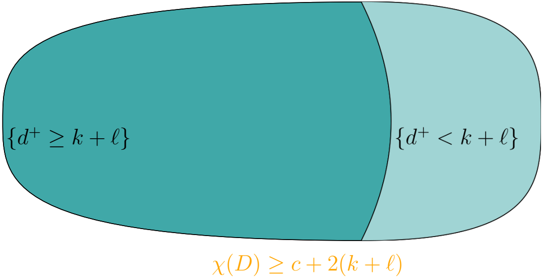
$\chi(D) \geq c + 2(k + \ell)$

Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,
- *then* $\chi(D) \geq c + 2(k + \ell) \implies T$ *with* ℓ *out (/in) leaves glued to* T.

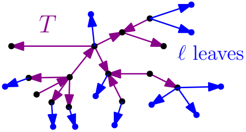


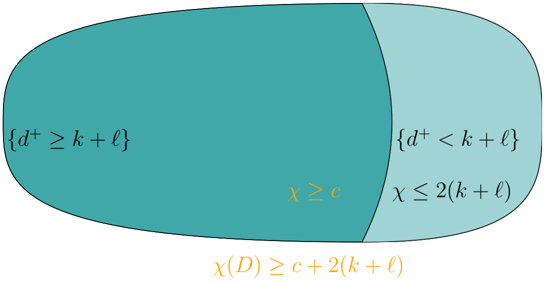


Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex T,*
- *then* $\chi(D) \geq c + 2(k+\ell) \implies$ *T with* ℓ *out (/in) leaves glued to T.*

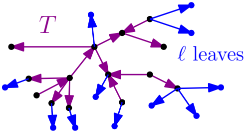


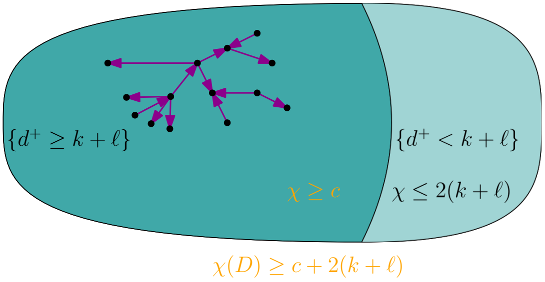


Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,
- *then* $\chi(D) \geq c + 2(k+\ell) \implies T$ *with* ℓ *out (/in) leaves glued to* T.

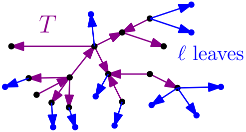


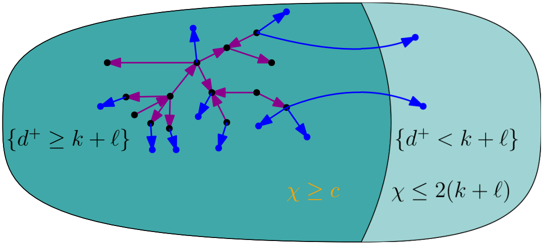


Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,
- *then* $\chi(D) \geq c + 2(k+\ell) \implies T$ *with* ℓ *out (/in) leaves glued to* T.



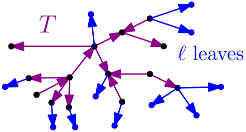


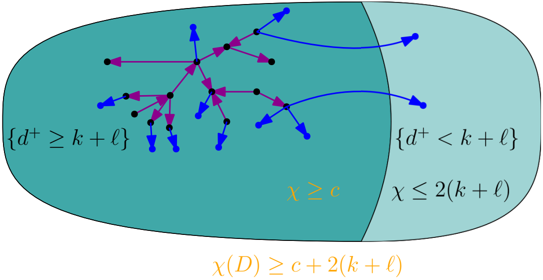
$\chi(D) \geq c + 2(k+\ell)$

Gluing leaves

Lemma (B '80; A-B,H,L-S,R,T '13)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex* T,
- *then* $\chi(D) \geq c + 2(k+\ell) \implies T$ *with* ℓ *out (/in) leaves glued to* T.





$\hookrightarrow O(k^2)$.

Gluing a path

- Last technique: good for lots of leaves,

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree* T.

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

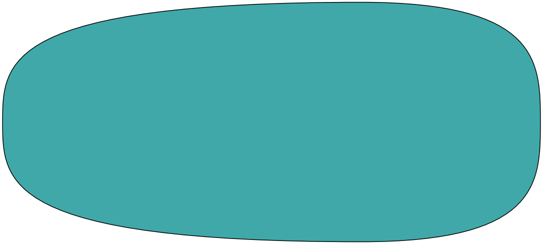
- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree T.*
- *Then* $\chi(D) \geq c + k + 2\ell \implies$ *T glued with a $\overrightarrow{P}_\ell$.*

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree* T.
- *Then* $\chi(D) \geq c + k + 2\ell \implies$ T *glued with a* $\overrightarrow{P}_\ell$.

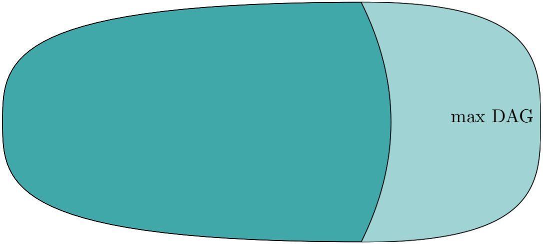


Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree* T.
- *Then* $\chi(D) \geq c + k + 2\ell \implies$ T *glued with a* $\overrightarrow{P}_\ell$.



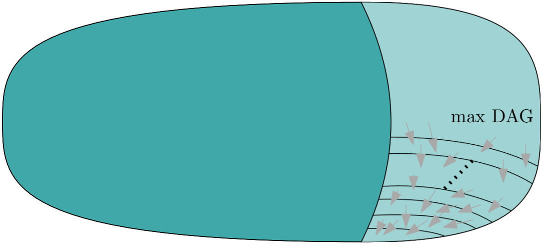
$\chi(D) \geq c + k + 2\ell$

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree* T.
- *Then* $\chi(D) \geq c + k + 2\ell \implies$ T *glued with a* $\overrightarrow{P}_\ell$.



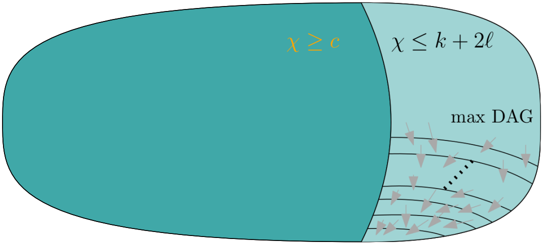
$\chi(D) \geq c + k + 2\ell$

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree T.*
- *Then* $\chi(D) \geq c + k + 2\ell \implies$ *T glued with a $\overrightarrow{P}_\ell$.*



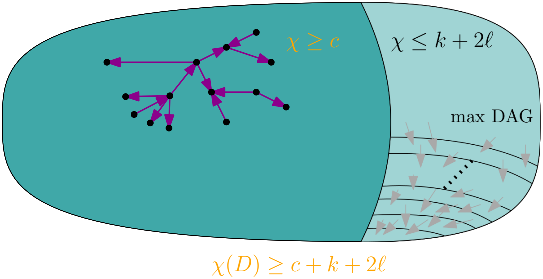
$\chi(D) \geq c + k + 2\ell$

Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. '23+)

- *Assume $\chi(D) \geq c \implies$ k-vertex tree T.*
- *Then $\chi(D) \geq c + k + 2\ell \implies T$ glued with a $\overrightarrow{P}_\ell$.*

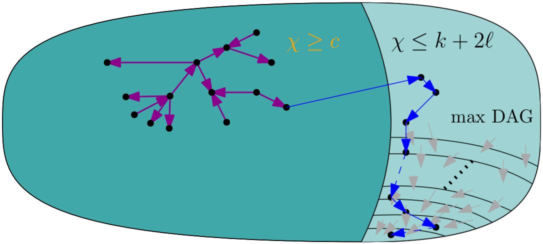


Gluing a path

- Last technique: good for lots of leaves,
- but adding a directed path $\overrightarrow{P_\ell}$ costs $O(k\ell)$...

Lemma (Bessy, Gonçalves, R. ’23+)

- *Assume* $\chi(D) \geq c \implies$ *k-vertex tree T.*
- *Then* $\chi(D) \geq c + k + 2\ell \implies$ *T glued with a* $\overrightarrow{P}_\ell$.



$\chi(D) \geq c + k + 2\ell$

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours
- Do this a minimal number of times...

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours
- Do this a minimal number of times...

Observation: an out-arborescence contains either $\geq \sqrt{k}$ out-leaves, or can be partitionned into $\leq \sqrt{k}$ paths.

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours
- Do this a minimal number of times...

Observation: an out-arborescence contains either $\geq \sqrt{k}$ out-leaves, or can be partitionned into $\leq \sqrt{k}$ paths.

Induction

- If $\leq \sqrt{k}$ leaves, **done**: pay $3k$ for each of the $\sqrt{k}$ paths

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours
- Do this a minimal number of times...

Observation: an out-arborescence contains either $\geq \sqrt{k}$ out-leaves, or can be partitionned into $\leq \sqrt{k}$ paths.

Induction

- If $\leq \sqrt{k}$ leaves, **done**: pay $3k$ for each of the $\sqrt{k}$ paths
- If $\geq \sqrt{k}$ out-leaves, pay $2k$ to add them,

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours
- Do this a minimal number of times...

Observation: an out-arborescence contains either $\geq \sqrt{k}$ out-leaves, or can be partitionned into $\leq \sqrt{k}$ paths.

Induction

- If $\leq \sqrt{k}$ leaves, **done**: pay $3k$ for each of the $\sqrt{k}$ paths
- If $\geq \sqrt{k}$ out-leaves, pay $2k$ to add them, induct...

Arborescences in digraphs

Tools for induction

- We can glue leaves or a path to T by paying $O(k)$ colours
- Do this a minimal number of times...

Observation: an out-arborescence contains either $\geq \sqrt{k}$ out-leaves, or can be partitionned into $\leq \sqrt{k}$ paths.

Induction

- If $\leq \sqrt{k}$ leaves, **done**: pay $3k$ for each of the $\sqrt{k}$ paths
- If $\geq \sqrt{k}$ out-leaves, pay $2k$ to add them, induct...

$\chi(D) \geq 3k\sqrt{k} \implies D$ contains k-arborescences

Conclusion

- same gluing: $\chi \geq (b-1)k \implies b$-blocks paths,

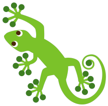
Conclusion

- same gluing: $\chi \geq (b-1)k \implies b$-blocks paths,
- General oriented trees: same approach, harder gluing...

Conclusion

- same gluing: $\chi \geq (b-1)k \implies b$-blocks paths,
- General oriented trees: same approach, harder gluing...

Merci!



References

- S. A. Burr, **Subtrees of directed graphs and hypergraphs**, *Proceedings of the Eleventh Southeastern Conference on Combinatorics, Graph Theory and Computing, Boca Raton, Congr. Numer.*, 28 (1980), 227–239.
- Louigi Addario-Berry, Frédéric Havet, Cláudia Linhares Sales, Bruce Reed, Stéphan Thomassé, **Oriented trees in digraphs**, *Discrete Mathematics*, Volume 313, Issue 8, 2013, 967-974.