Decomposition of ordered graphs

Romain Bourneuf
Based on joint works with Édouard Bonnet, Julien Cocquet,
Colin Geniet, Chaoliang Tang and Stéphan Thomassé

Workshop ANR GODASse, Sète, 2025

Perfect graphs

G is *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Perfect graphs

G is *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G.
G is *Berge* if both G and $\overline{G}$ have no induced odd cycle.

Perfect graphs

G is *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G.
G is *Berge* if both G and $\overline{G}$ have no induced odd cycle.

Theorem [Chudnovsky, Robertson, Seymour, Thomas '06]

G is perfect $\Leftrightarrow$ G is Berge.

Perfect graphs

G is *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G.
G is *Berge* if both G and $\overline{G}$ have no induced odd cycle.

Theorem [Chudnovsky, Robertson, Seymour, Thomas '06]

G is perfect $\Leftrightarrow$ G is Berge.

Proof strategy: every Berge graph can be constructed from some basic graphs using some set of operations, such that:

Perfect graphs

G is *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G.
G is *Berge* if both G and $\overline{G}$ have no induced odd cycle.

Theorem [Chudnovsky, Robertson, Seymour, Thomas '06]

G is perfect $\Leftrightarrow$ G is Berge.

Proof strategy: every Berge graph can be constructed from some basic graphs using some set of operations, such that:

- The basic graphs are perfect.

Perfect graphs

G is *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G.
G is *Berge* if both G and $\overline{G}$ have no induced odd cycle.

Theorem [Chudnovsky, Robertson, Seymour, Thomas '06]

G is perfect $\Leftrightarrow$ G is Berge.

Proof strategy: every Berge graph can be constructed from some basic graphs using some set of operations, such that:

- The basic graphs are perfect.
- These operations preserve perfection.

More examples

Theorem [Wagner '37]

Every edge-maximal K_5-minor-free graph can be constructed from plane triangulations and the Wagner graph by recursively gluing along K_2's and K_3's.

More examples

Theorem [Wagner '37]

Every edge-maximal K_5-minor-free graph can be constructed from plane triangulations and the Wagner graph by recursively gluing along K_2's and K_3's.

Theorem [Seymour '80]

Every Totally Unimodular matrix can be constructed from network matrices and two basic TU matrices by recursively applying k-sum operations.

More examples

Theorem [Wagner '37]

Every edge-maximal K_5-minor-free graph can be constructed from plane triangulations and the Wagner graph by recursively gluing along K_2's and K_3's.

Theorem [Seymour '80]

Every Totally Unimodular matrix can be constructed from network matrices and two basic TU matrices by recursively applying k-sum operations.

Theorem [Bonnet, B., Geniet, Thomassé '23]

Every pattern-avoiding permutation is the product of a bounded number of separable permutations.

Tame operations

Want: cannot create too complex class using our operations.

Tame operations

Want: cannot create too complex class using our operations.
If $\mathcal{C}_1$ and $\mathcal{C}_2$ are nice, so is $\mathcal{C}_1 \mathbin{/\!\!\!\wedge} \mathcal{C}_2$.

Tame operations

Want: cannot create too complex class using our operations.
If $\mathcal{C}_1$ and $\mathcal{C}_2$ are nice, so is $\mathcal{C}_1 \mathbin{\wedge\!\!\!\wedge} \mathcal{C}_2$.
Running example: graph coloring.

Tame operations

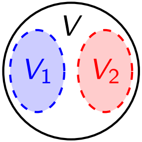
Want: cannot create too complex class using our operations.
If $\mathcal{C}_1$ and $\mathcal{C}_2$ are nice, so is $\mathcal{C}_1 \mathbin{/\!\!\Lambda\!\!\backslash} \mathcal{C}_2$.
Running example: graph coloring.

Definition (χ-bounded)

$\mathcal{C}$ – hereditary class of graphs, is χ-*bounded* if there is a function f such that $\forall G \in \mathcal{C}, \chi(G) \leq f(\omega(G))$.

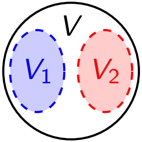
Vertex union

$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.

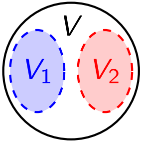


Vertex union

$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.



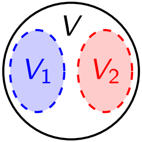
Vertex union



$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

Order between the two sets?

Vertex union

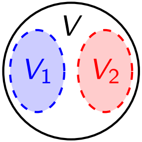


$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

Order between the two sets?

- Arbitrary.

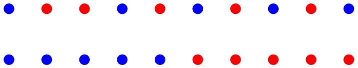
Vertex union



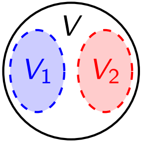
$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

Order between the two sets?

- Arbitrary.
- One before the other.



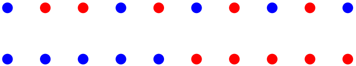
Vertex union



$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

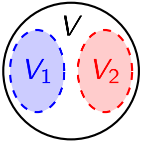
Order between the two sets?

- Arbitrary.
- One before the other.



Edges between the two sets?

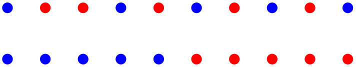
Vertex union



$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

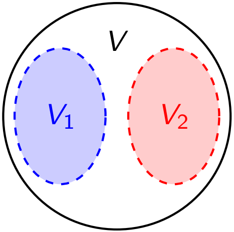
Order between the two sets?

- Arbitrary.
- One before the other.

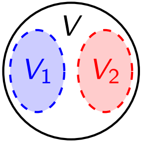


Edges between the two sets?

- No (disjoint union).



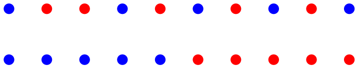
Vertex union



$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

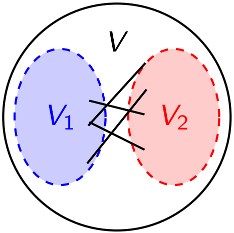
Order between the two sets?

- Arbitrary.
- One before the other.

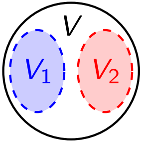


Edges between the two sets?

- No (disjoint union).
- Arbitrary.



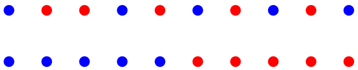
Vertex union



$V(G) = V_1 \cup V_2$, consider $G[V_1]$ and $G[V_2]$.
$\chi(G) \leq \chi(G[V_1]) + \chi(G[V_2])$.

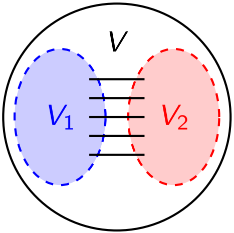
Order between the two sets?

- Arbitrary.
- One before the other.



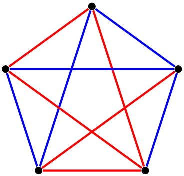
Edges between the two sets?

- No (disjoint union).
- Arbitrary.
- Yes but controlled.



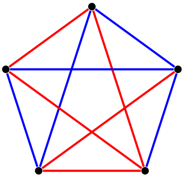
Edge union

$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.



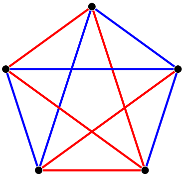
Edge union

$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.
$\chi(G) \leq \chi(G[E_1]) \cdot \chi(G[E_2])$.



Edge union

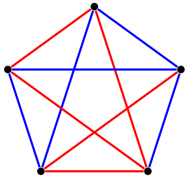
$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.
$\chi(G) \leq \chi(G[E_1]) \cdot \chi(G[E_2])$.



Partition or union?

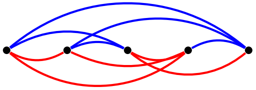
Edge union

$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.
$\chi(G) \leq \chi(G[E_1]) \cdot \chi(G[E_2])$.



Partition or union?

Order of the graphs?



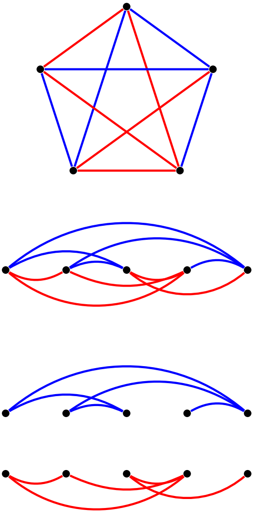
Edge union

$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.
$\chi(G) \leq \chi(G[E_1]) \cdot \chi(G[E_2])$.

Partition or union?

Order of the graphs?

- Always the same.



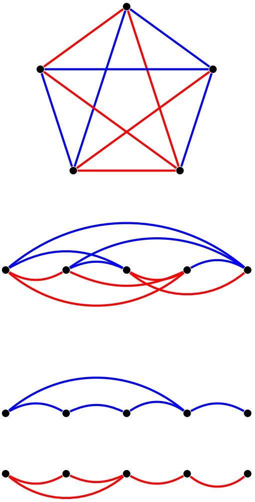
Edge union

$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.
$\chi(G) \leq \chi(G[E_1]) \cdot \chi(G[E_2])$.

Partition or union?

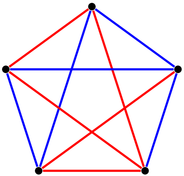
Order of the graphs?

- Always the same.
- Arbitrary.



Edge union

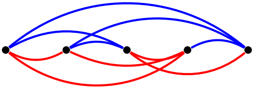
$E(G) = E_1 \cup E_2$, consider $G[E_1]$ and $G[E_2]$.
$\chi(G) \le \chi(G[E_1]) \cdot \chi(G[E_2])$.



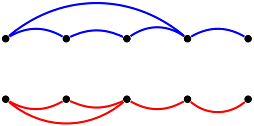
Partition or union?

Order of the graphs?

- Always the same.
- Arbitrary.

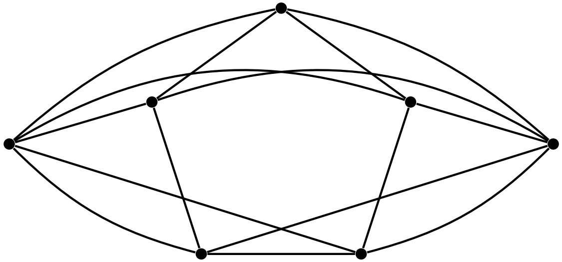


(Very powerful: all cubic graphs from ordered matchings)

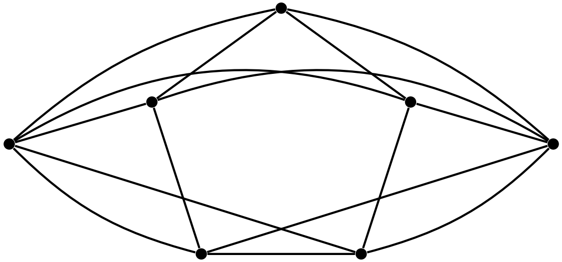


Substitutions

Substitutions



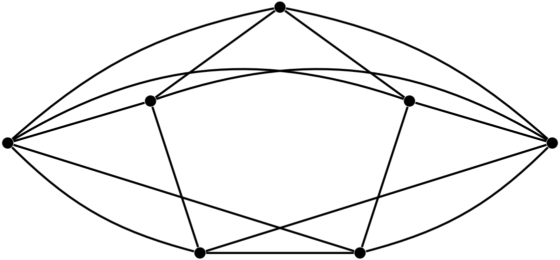
Substitutions



Theorem [Chudnovsky, Penev, Scott, Trotignon '13]

If $\mathcal{C}$ is χ-bounded, so is $\overline{\mathcal{C}}$.

Substitutions

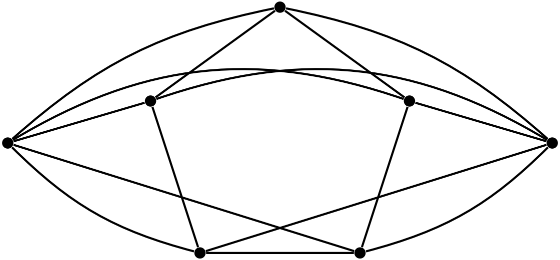


Theorem [Chudnovsky, Penev, Scott, Trotignon '13]

If $\mathcal{C}$ is χ-bounded, so is $\overline{\mathcal{C}}$.

$\mathcal{C}$ – class of graphs $\longrightarrow \overline{\mathcal{C}} :=$ all graphs which can be obtained by iterating substitutions from $\mathcal{C}$.

Substitutions



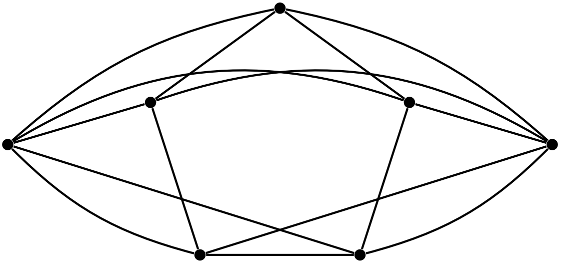
Theorem [Chudnovsky, Penev, Scott, Trotignon '13]

If $\mathcal{C}$ is χ-bounded, so is $\overline{\mathcal{C}}$.

$\mathcal{C}$ – class of graphs $\longrightarrow \overline{\mathcal{C}} :=$ all graphs which can be obtained by iterating substitutions from $\mathcal{C}$.

What about the order?

Substitutions



Theorem [Chudnovsky, Penev, Scott, Trotignon '13]

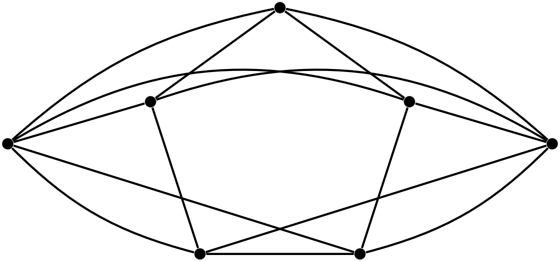
If $\mathcal{C}$ is χ-bounded, so is $\overline{\mathcal{C}}$.

$\mathcal{C}$ – class of graphs $\longrightarrow \overline{\mathcal{C}} :=$ all graphs which can be obtained by iterating substitutions from $\mathcal{C}$.

What about the order?

- Arbitrary.

Substitutions



Theorem [Chudnovsky, Penev, Scott, Trotignon '13]

If $\mathcal{C}$ is χ-bounded, so is $\overline{\mathcal{C}}$.

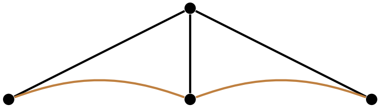
$\mathcal{C}$ – class of graphs $\longrightarrow \overline{\mathcal{C}} :=$ all graphs which can be obtained by iterating substitutions from $\mathcal{C}$.

What about the order?

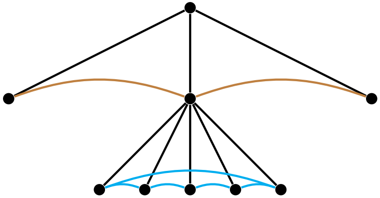
- Arbitrary.
- Consistent.

Substitution tree

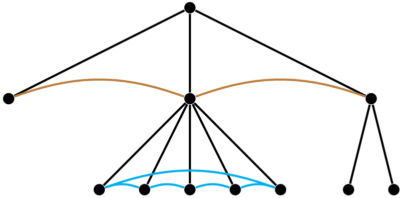
Substitution tree



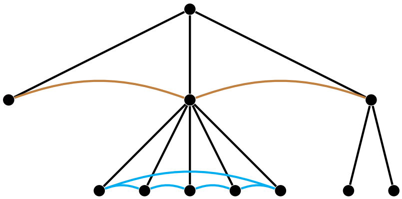
Substitution tree



Substitution tree

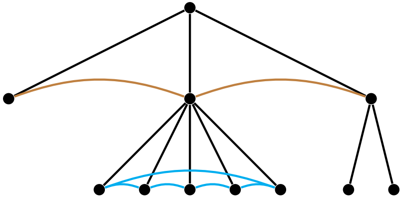


Substitution tree



Well-suited for ordered graphs.

Substitution tree

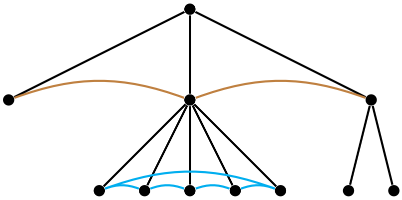


Well-suited for ordered graphs.

Theorem [Cournier, Habib '94], [McConnell, Spinrad '94]

The modular decomposition can be computed in linear time.

Substitution tree



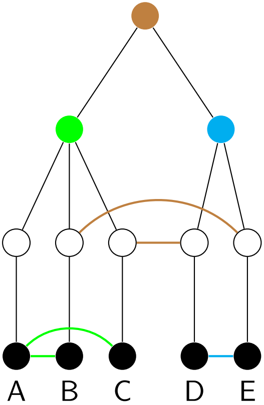
Well-suited for ordered graphs.

Theorem [Cournier, Habib '94], [McConnell, Spinrad '94]

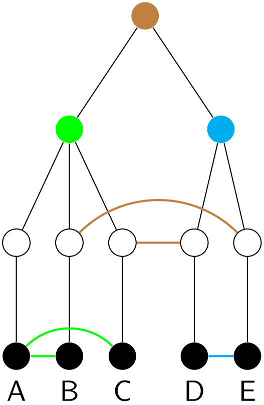
The modular decomposition can be computed in linear time.

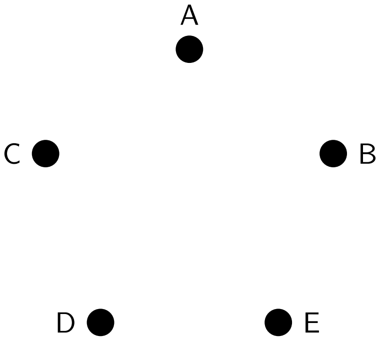
The substitution tree is often trivial...

Delayed substitution tree

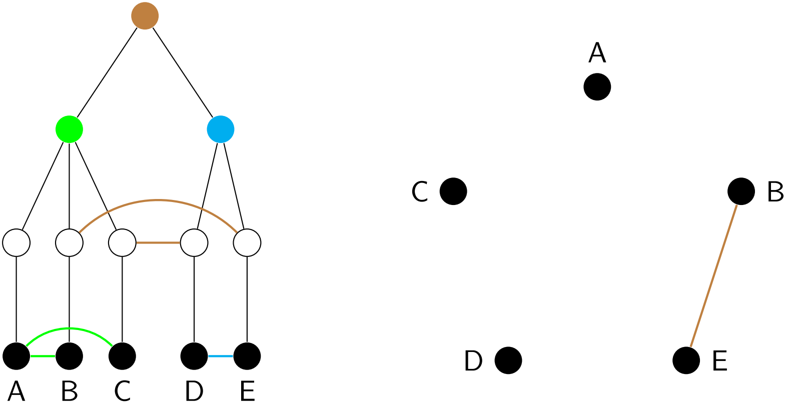


Delayed substitution tree

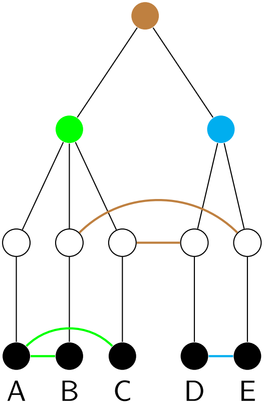


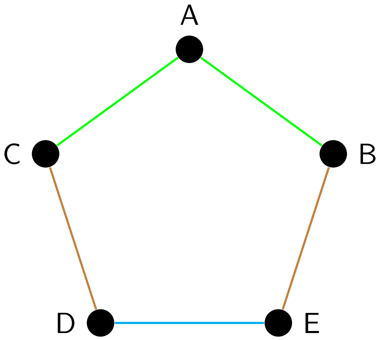


Delayed substitution tree

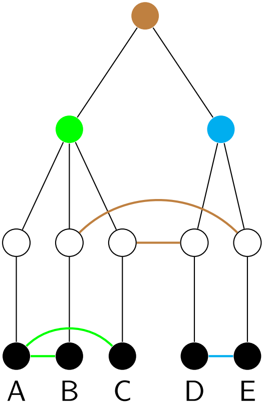


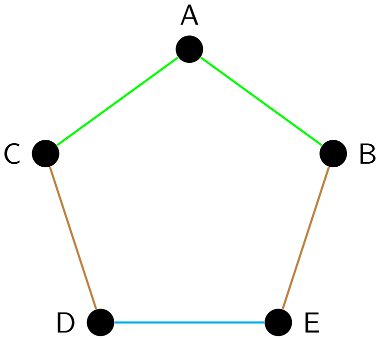
Delayed substitution tree





Delayed substitution tree





Edges between siblings are irrelevant.

Delayed extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_d :=$ all graphs with a delayed substitution tree with all quotient graphs in $\mathcal{C}$.

Delayed extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_d :=$ all graphs with a delayed substitution tree with all quotient graphs in $\mathcal{C}$.

Theorem

Every graph in $\mathcal{C}_d$ is the edge union of two graphs in $\overline{\mathcal{C}}$.

Delayed extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_d :=$ all graphs with a delayed substitution tree with all quotient graphs in $\mathcal{C}$.

Theorem

Every graph in $\mathcal{C}_d$ is the edge union of two graphs in $\overline{\mathcal{C}}$.

Theorem

If $\mathcal{C}$ is χ-bounded, so is $\mathcal{C}_d$.

Delayed extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_d :=$ all graphs with a delayed substitution tree with all quotient graphs in $\mathcal{C}$.

Theorem

Every graph in $\mathcal{C}_d$ is the edge union of two graphs in $\overline{\mathcal{C}}$.

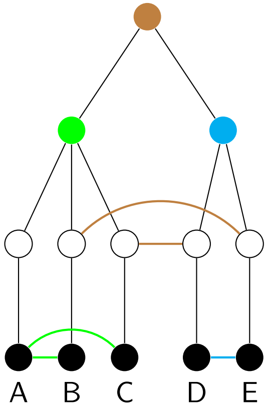
Theorem

If $\mathcal{C}$ is χ-bounded, so is $\mathcal{C}_d$.

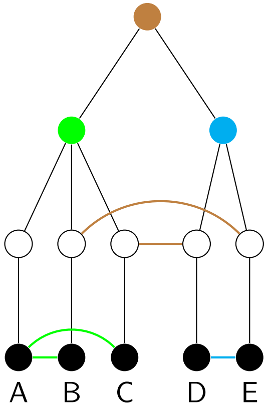
Theorem [Bonnet, B., Geniet, Thomassé '23]

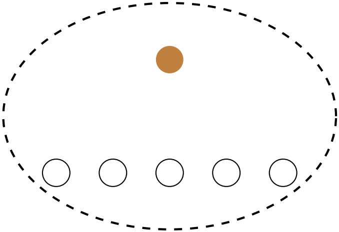
A nontrivial delayed decomposition can be computed in time $O(n+m)$.

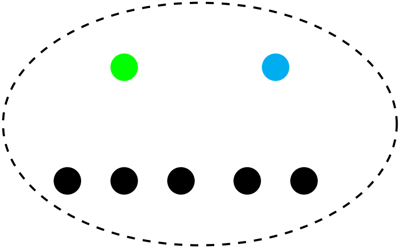
Delayed substitutions & substitutions



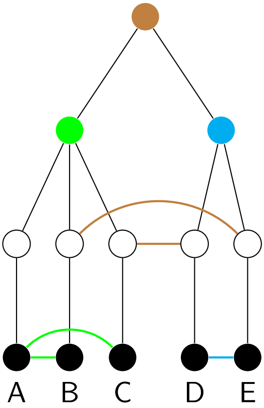
Delayed substitutions & substitutions

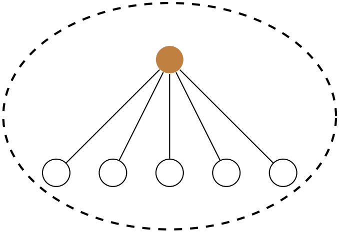


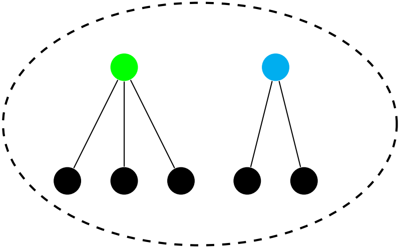




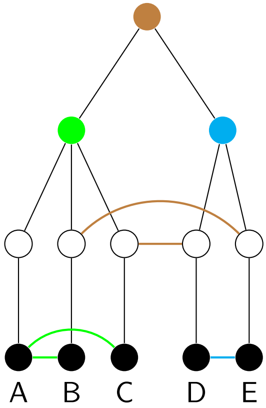
Delayed substitutions & substitutions

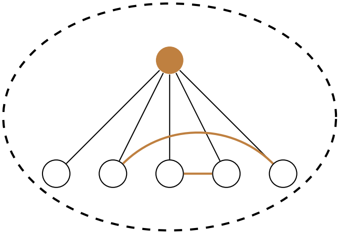


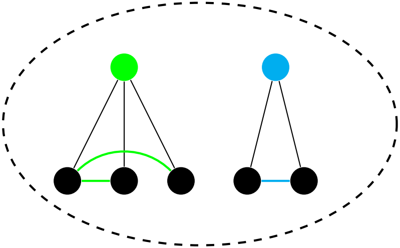




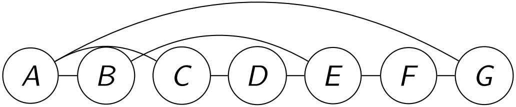
Delayed substitutions & substitutions



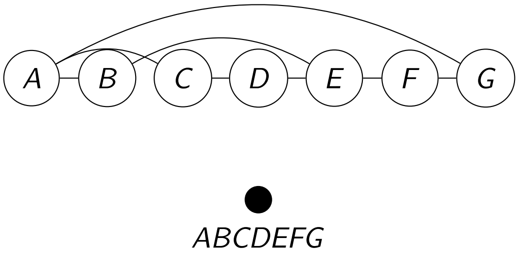




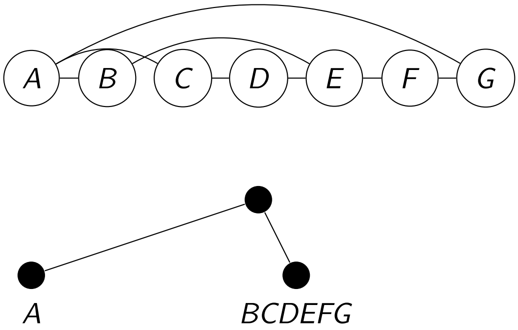
Computing a delayed decomposition



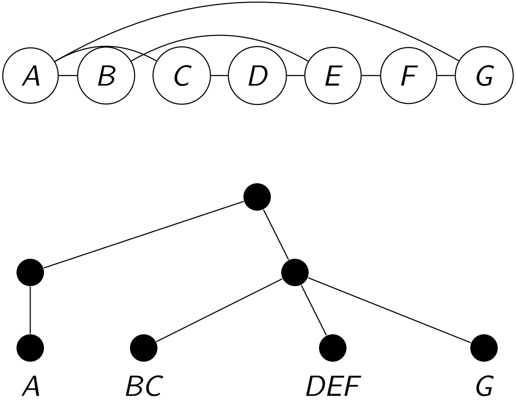
Computing a delayed decomposition



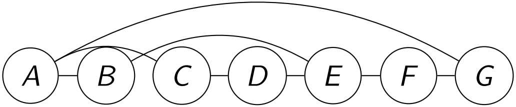
Computing a delayed decomposition

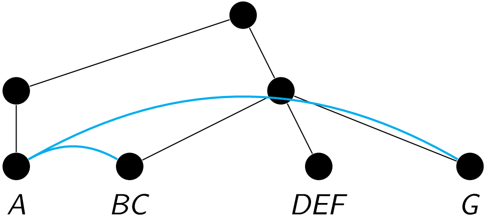


Computing a delayed decomposition

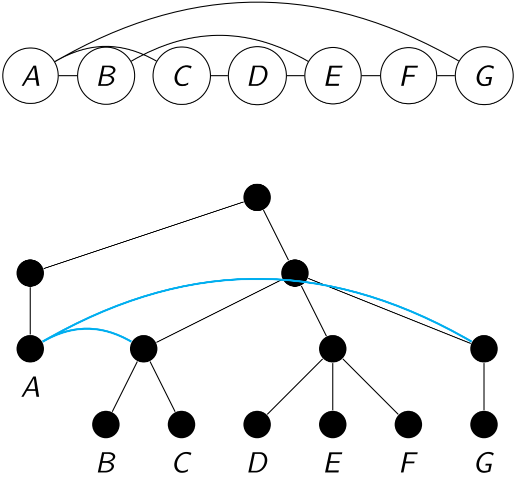


Computing a delayed decomposition

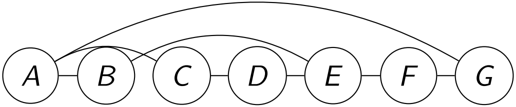


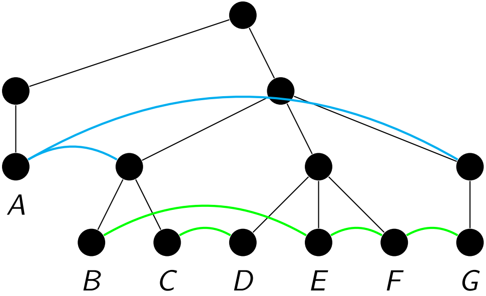


Computing a delayed decomposition

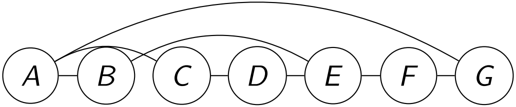


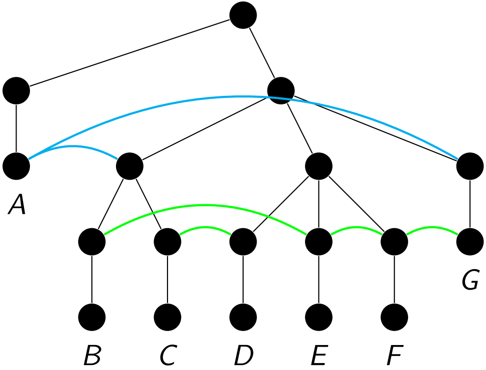
Computing a delayed decomposition



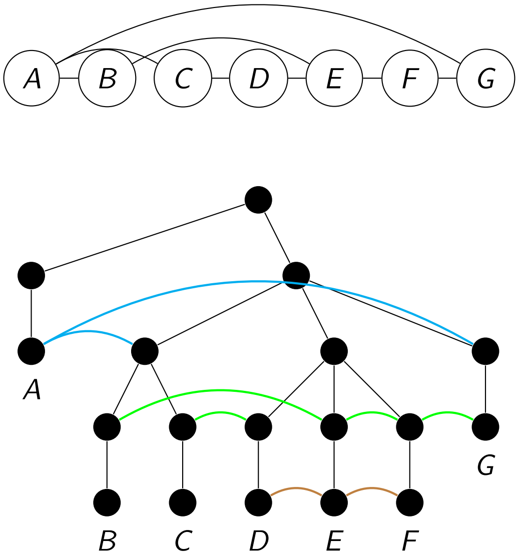


Computing a delayed decomposition

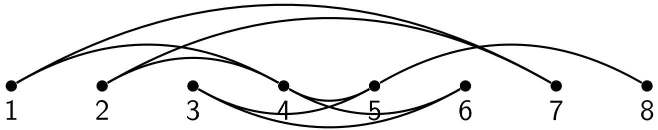




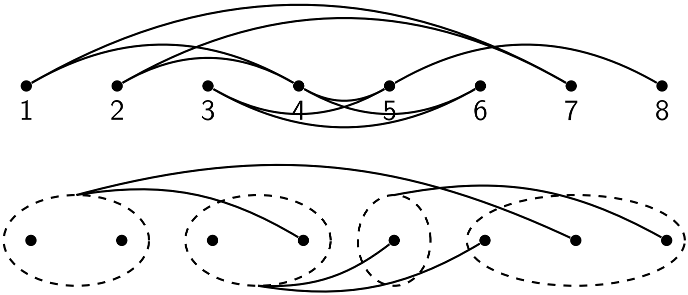
Computing a delayed decomposition



Right Module Partition



Right Module Partition



Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

With this definition, the right extension of the class of all forests is not χ-bounded.

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

With this definition, the right extension of the class of all forests is not χ-bounded.
Shift graphs:

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

With this definition, the right extension of the class of all forests is not χ-bounded.
Shift graphs: $V(S_n) = \{(i,j) : 1 \leq i < j \leq n\}$,

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

With this definition, the right extension of the class of all forests is not χ-bounded.
Shift graphs: $V(S_n) = \{(i,j) : 1 \leq i < j \leq n\}$, $E(S_n) = \{(i,j)(j,k)\}$.

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

With this definition, the right extension of the class of all forests is not χ-bounded.
Shift graphs: $V(S_n) = \{(i,j) : 1 \leq i < j \leq n\}$, $E(S_n) = \{(i,j)(j,k)\}$.

- Triangle-free.

Quotients?

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all "quotient graphs" in $\mathcal{C}$.

First try: all subgraphs with at most 1 point in each independent set.

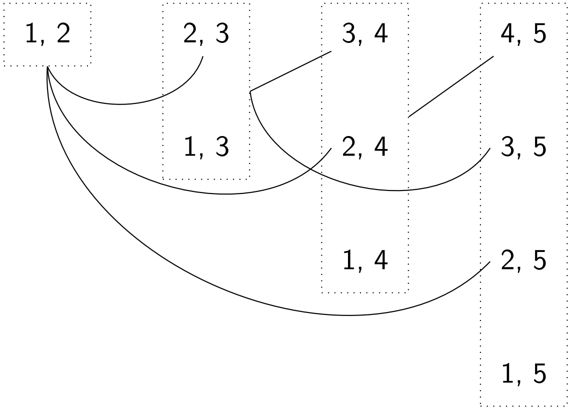
With this definition, the right extension of the class of all forests is not χ-bounded.
Shift graphs: $V(S_n) = \{(i,j) : 1 \leq i < j \leq n\}$, $E(S_n) = \{(i,j)(j,k)\}$.

- Triangle-free.
- Unbounded χ.

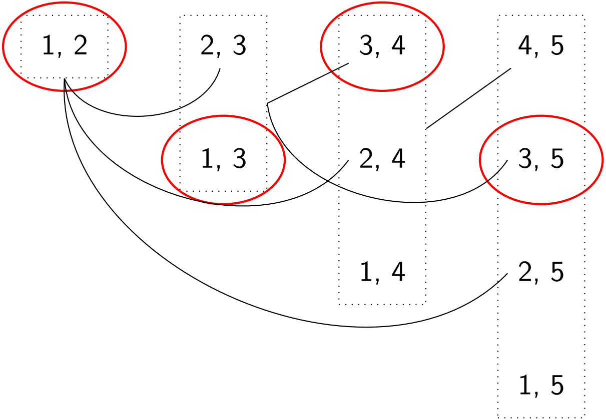
Shift graphs

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

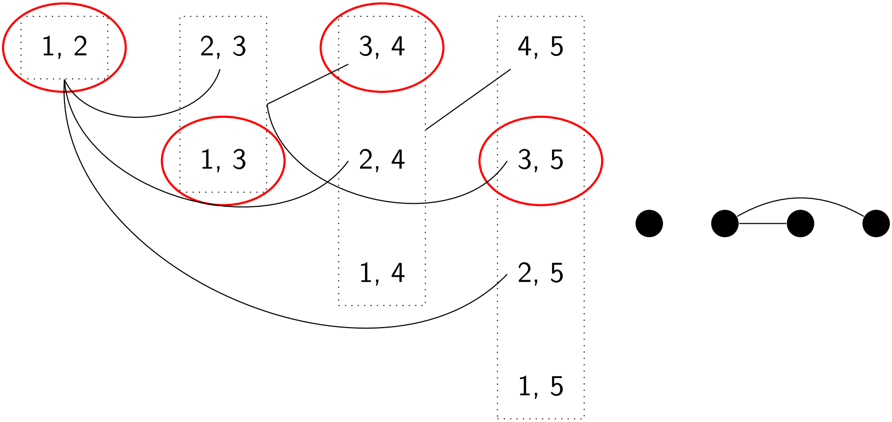
Shift graphs



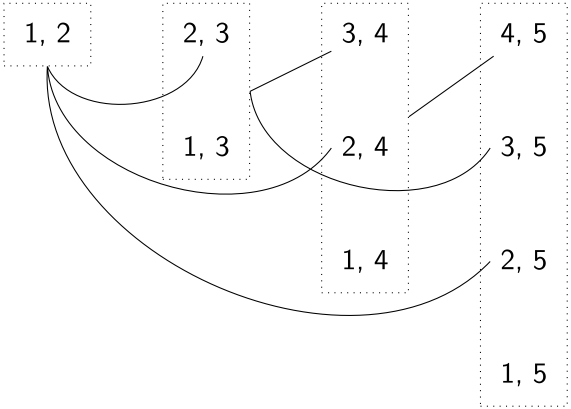
Shift graphs



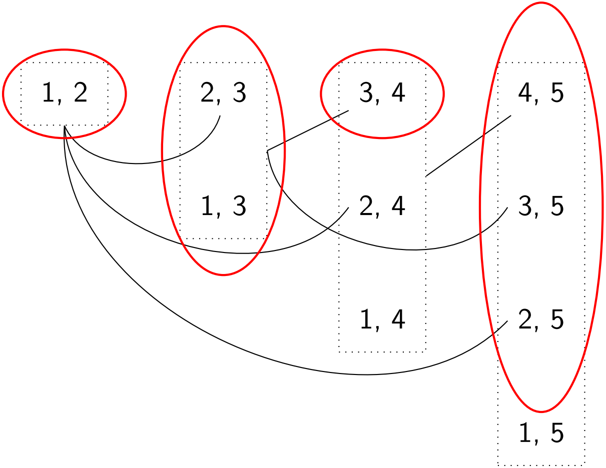
Shift graphs



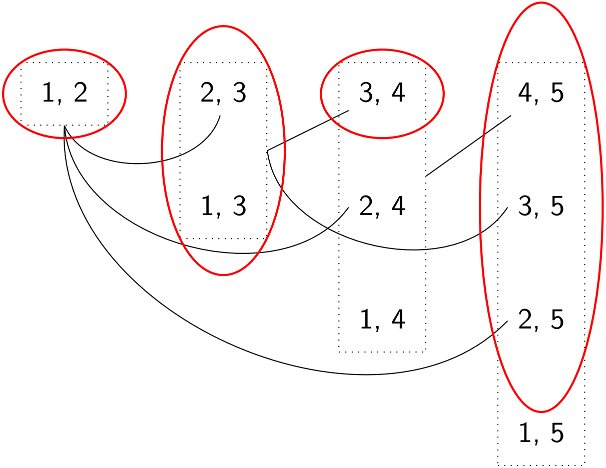
Transversal minor

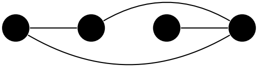


Transversal minor



Transversal minor





Right extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all transversal minors in $\mathcal{C}$.

Right extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all transversal minors in $\mathcal{C}$.

Theorem [B., Thomassé '23]

If $\mathcal{C}$ is χ-bounded, then so is $\mathcal{C}_{RMP}$.

Right extension

$\mathcal{C}$ – class of graphs $\longrightarrow \mathcal{C}_{RMP} :=$ all graphs with a right module partition with all transversal minors in $\mathcal{C}$.

Theorem [B., Thomassé '23]

If $\mathcal{C}$ is χ-bounded, then so is $\mathcal{C}_{RMP}$.

Open: does it preserve polynomial χ-boundedness?