Decomposition techniques applied to the Clique-Stable set Separation problem

Aurélie Lagoutte

LIMOS, University Clermont Auvergne, France

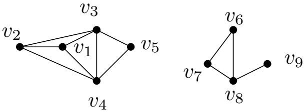
A tribute to Frédéric Maffray

Grenoble, September 2019

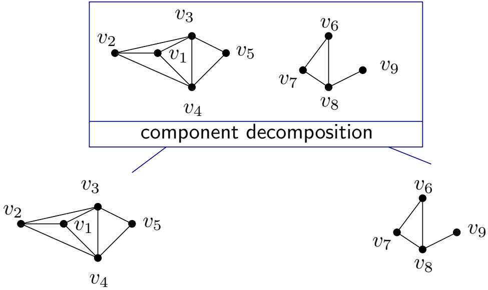
*Joint work with N. Bousquet, **F. Maffray** and L. Pastor*

Decomposing a graph using structure

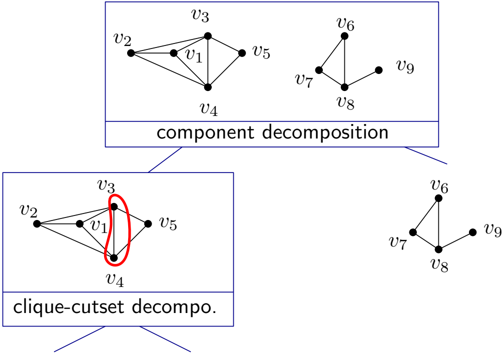
Decomposing a graph using structure



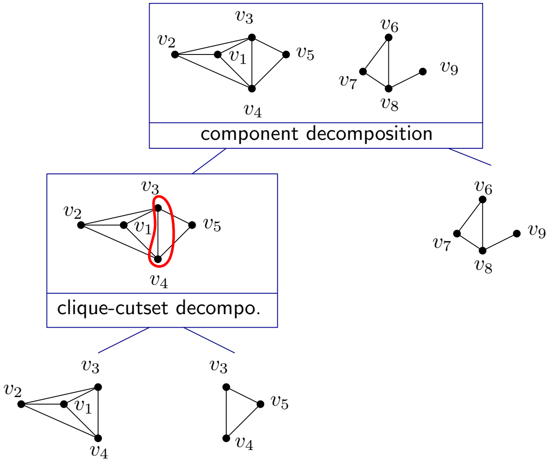
Decomposing a graph using structure



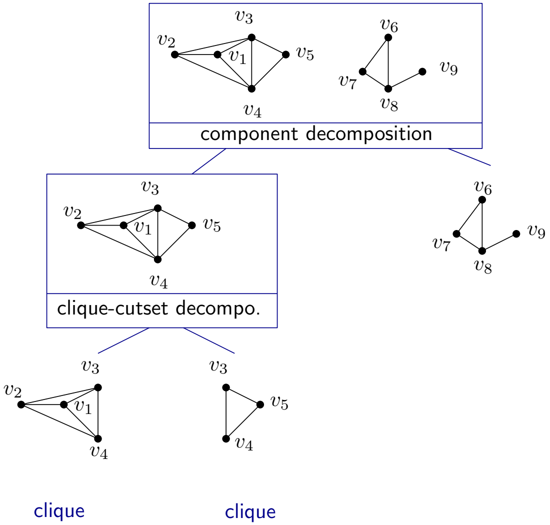
Decomposing a graph using structure



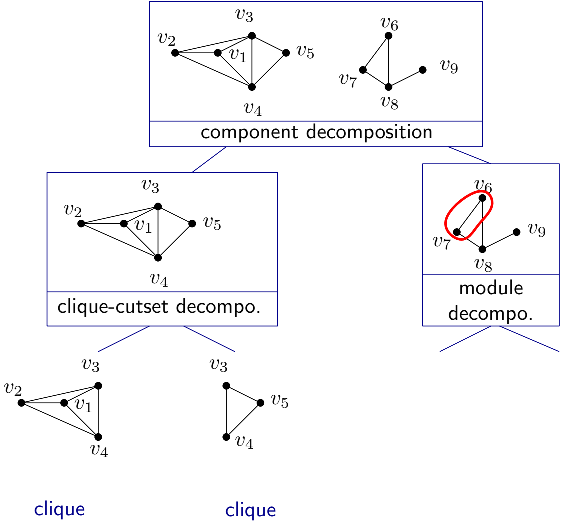
Decomposing a graph using structure



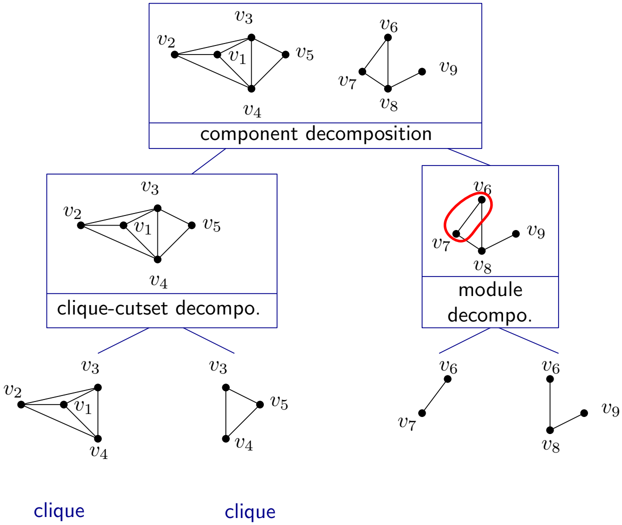
Decomposing a graph using structure



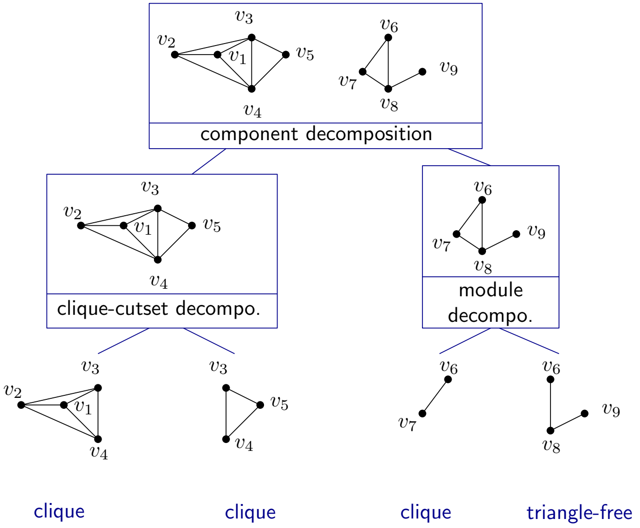
Decomposing a graph using structure



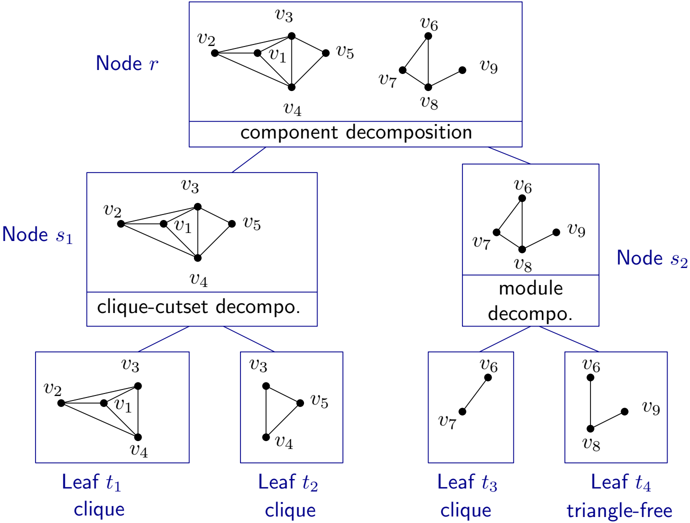
Decomposing a graph using structure



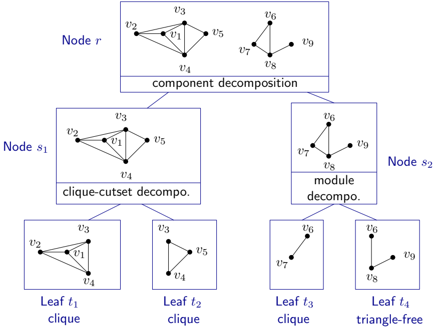
Decomposing a graph using structure



Decomposing a graph using structure



Decomposing a graph using structure



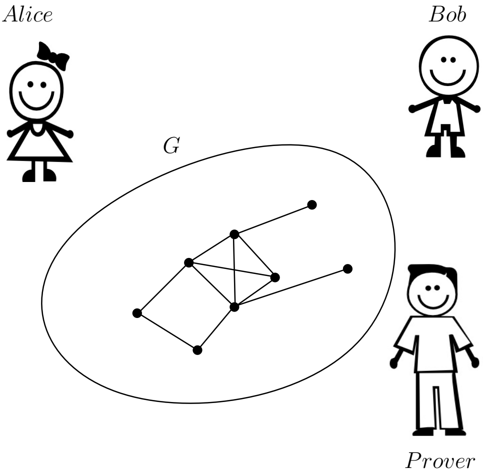
G is decomposed along:

- connected components
- clique-cutset
- modules

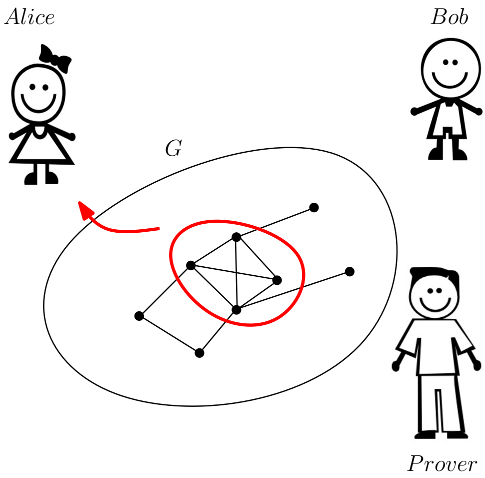
into leaves which are:

- cliques, or
- triangle-free graphs.

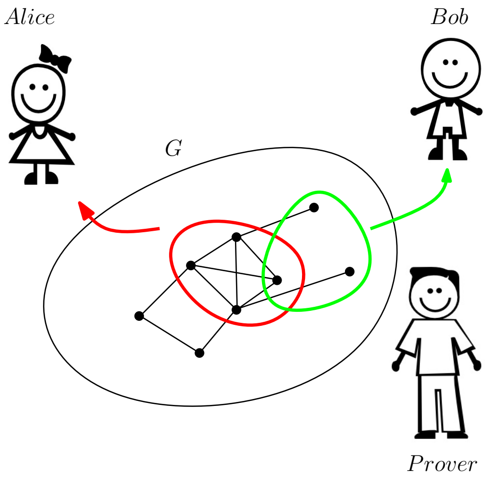
Clique vs Independent Set Problem



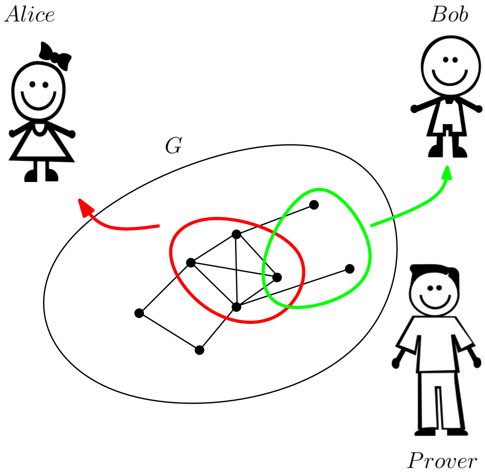
Clique vs Independent Set Problem



Clique vs Independent Set Problem

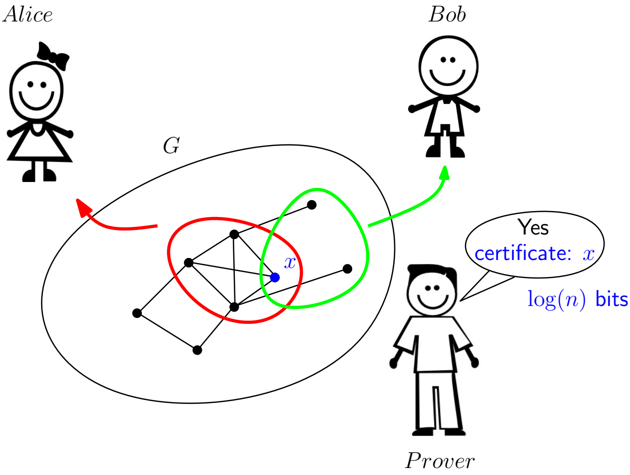


Clique vs Independent Set Problem



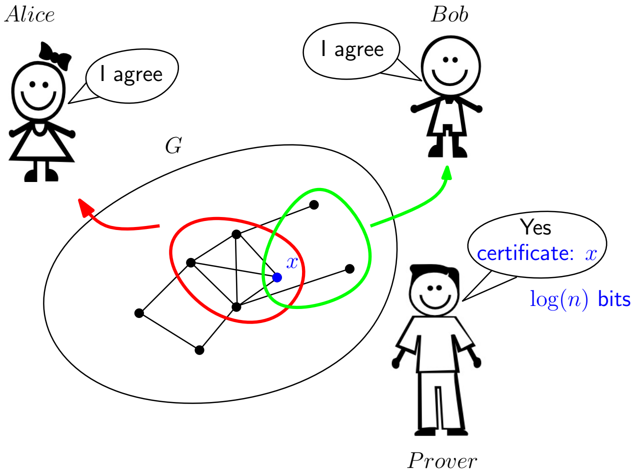
Do the clique and the stable set intersect?

Clique vs Independent Set Problem



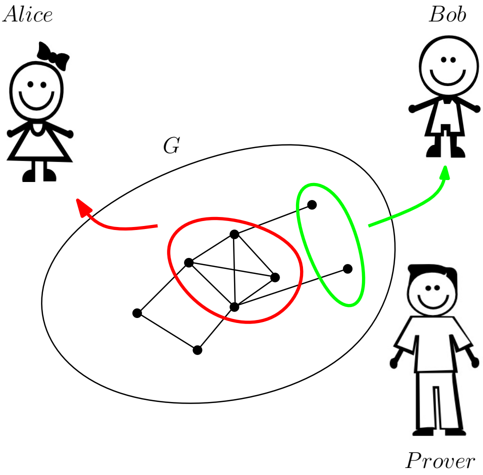
Do the clique and the stable set intersect?

Clique vs Independent Set Problem



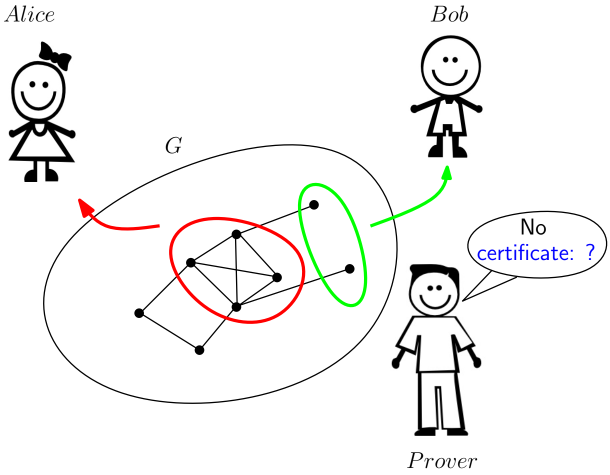
Do the clique and the stable set intersect?

Clique vs Independent Set Problem



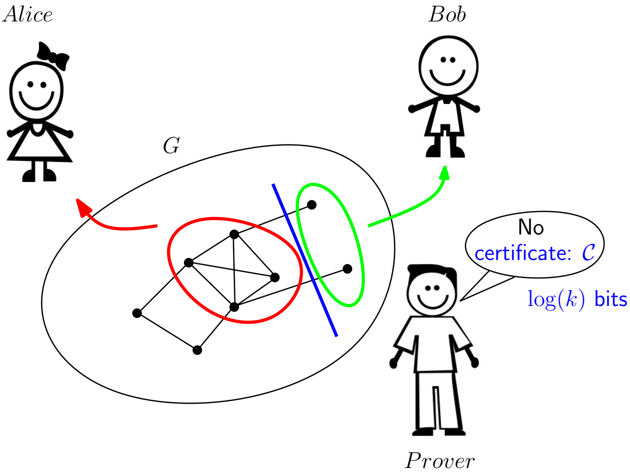
Do the clique and the stable set intersect?

Clique vs Independent Set Problem



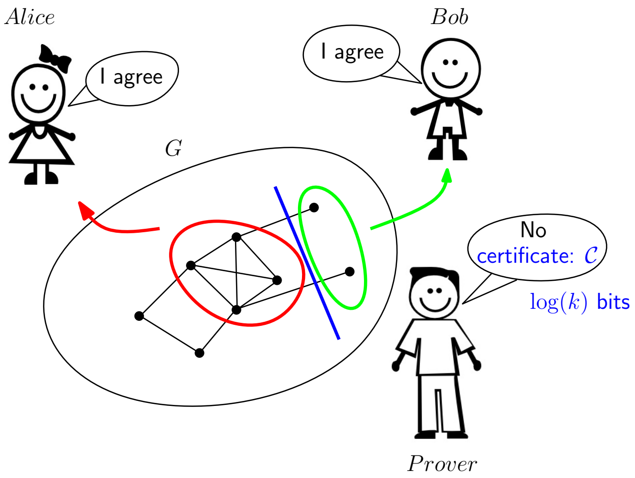
Do the clique and the stable set intersect?

Clique vs Independent Set Problem



Do the clique and the stable set intersect?

Clique vs Independent Set Problem



Do the clique and the stable set intersect?

Clique vs Independent Set Problem

Goal (Yannakakis 1991)

Find a *CS-separator* : a family of cuts that can separate all the pairs Clique-Stable set.

Non-det communication complexity $\leftrightarrow$ min. size of a CS-Separator.

Clique vs Independent Set Problem

Goal (Yannakakis 1991)

Find a *CS-separator* : a family of cuts that can separate all the pairs Clique-Stable set.

Non-det communication complexity $\leftrightarrow$ min. size of a CS-Separator.

Main question (Yannakakis 1991)

Do perfect graphs admit polynomial-size CS-Separator?

Clique vs Independent Set Problem

Goal (Yannakakis 1991)

Find a *CS-separator* : a family of cuts that can separate all the pairs Clique-Stable set.

Non-det communication complexity $\leftrightarrow$ min. size of a CS-Separator.

Main question (Yannakakis 1991)

Do perfect graphs admit polynomial-size CS-Separator?

Upper Bound: there exists a CS-separator of size $\mathcal{O}(n^{\log n})$.

Clique vs Independent Set Problem

Goal (Yannakakis 1991)

Find a *CS-separator* : a family of cuts that can separate all the pairs Clique-Stable set.

Non-det communication complexity $\leftrightarrow$ min. size of a CS-Separator.

Main question (Yannakakis 1991)

Do perfect graphs admit polynomial-size CS-Separator?

Upper Bound: there exists a CS-separator of size $\mathcal{O}(n^{\log n})$.
Lower bound in perfect graphs?

Clique vs Independent Set Problem

Goal (Yannakakis 1991)

Find a *CS-separator* : a family of cuts that can separate all the pairs Clique-Stable set.

Non-det communication complexity $\leftrightarrow$ min. size of a CS-Separator.

Main question (Yannakakis 1991)

Do perfect graphs admit polynomial-size CS-Separator?

Upper Bound: there exists a CS-separator of size $\mathcal{O}(n^{\log n})$.
Lower bound in perfect graphs? Lower bound in general?

Clique vs Independent Set Problem

Goal (Yannakakis 1991)

Find a *CS-separator* : a family of cuts that can separate all the pairs Clique-Stable set.

Non-det communication complexity $\leftrightarrow$ min. size of a CS-Separator.

Main question (Yannakakis 1991)

Do perfect graphs admit polynomial-size CS-Separator?

Upper Bound: there exists a CS-separator of size $\mathcal{O}(n^{\log n})$.
Lower bound in perfect graphs? Lower bound in general?
Does there exist for all graph G on n vertices a CS-separator of size poly(n)? Or for which classes of graphs does it exist?

Does there exist for all graph G on n vertices a CS-separator of size poly(n)?

Does there exist for all graph G on n vertices a CS-separator of size poly(n)?

What about random graphs?

Does there exist for all graph G on n vertices a CS-separator of size poly(n)?

What about random graphs?

Theorem [Bousquet, L., Thomassé, 2014]

Random graphs admit a CS-Separator of size $\mathcal{O}(n^7)$ asymptotically almost surely.

Does there exist for all graph G on n vertices a CS-separator of size poly(n)?

What about random graphs?

Theorem [Bousquet, L., Thomassé, 2014]

Random graphs admit a CS-Separator of size $\mathcal{O}(n^7)$ asymptotically almost surely.

Lower Bound:

- (Huang, Sudakov 2012): we need $\Omega(n^{\frac{6}{5}})$ cuts for some graphs.

Does there exist for all graph G on n vertices a CS-separator of size poly(n)?

What about random graphs?

Theorem [Bousquet, L., Thomassé, 2014]

Random graphs admit a CS-Separator of size $\mathcal{O}(n^7)$ asymptotically almost surely.

Lower Bound:

- (Huang, Sudakov 2012): we need $\Omega(n^{\frac{6}{5}})$ cuts for some graphs.
- (Amano, Shigeta 2013): we need $\Omega(n^{2-\varepsilon})$ for some graphs.

~~**Does there exist for all graph G on n vertices a CS-separator of size poly(n)?**~~ **No!**

What about random graphs?

Theorem [Bousquet, L., Thomassé, 2014]

Random graphs admit a CS-Separator of size $\mathcal{O}(n^7)$ asymptotically almost surely.

Lower Bound:

- (Huang, Sudakov 2012): we need $\Omega(n^{\frac{6}{5}})$ cuts for some graphs.
- (Amano, Shigeta 2013): we need $\Omega(n^{2-\varepsilon})$ for some graphs.
- (Göös 2015): we need $n^{\Omega(\log^{0.128} n)}$ cuts for some graphs.

~~**Does there exist for all graph G on n vertices a CS-separator of size poly(n)?**~~ **No!**

What about random graphs?

Theorem [Bousquet, L., Thomassé, 2014]

Random graphs admit a CS-Separator of size $\mathcal{O}(n^7)$ asymptotically almost surely.

Lower Bound:

- (Huang, Sudakov 2012): we need $\Omega(n^{\frac{6}{5}})$ cuts for some graphs.
- (Amano, Shigeta 2013): we need $\Omega(n^{2-\varepsilon})$ for some graphs.
- (Göös 2015): we need $n^{\Omega(\log^{0.128} n)}$ cuts for some graphs.

~~Does there exist for all graph G on n vertices a CS-separator of size poly(n)?~~ No!

What about random graphs?

Theorem [Bousquet, L., Thomassé, 2014]

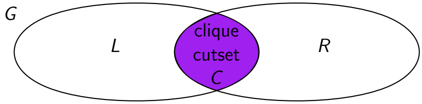
Random graphs admit a CS-Separator of size $\mathcal{O}(n^7)$ asymptotically almost surely.

Lower Bound:

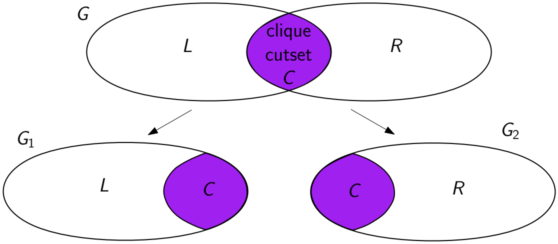
- (Huang, Sudakov 2012): we need $\Omega(n^{\frac{6}{5}})$ cuts for some graphs.
- (Amano, Shigeta 2013): we need $\Omega(n^{2-\varepsilon})$ for some graphs.
- (Göös 2015): we need $n^{\Omega(\log^{0.128} n)}$ cuts for some graphs.

For which classes of graphs does there exist a polynomial CS-Separator?

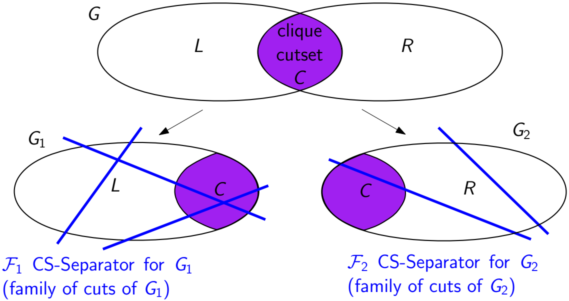
Valid decomposition



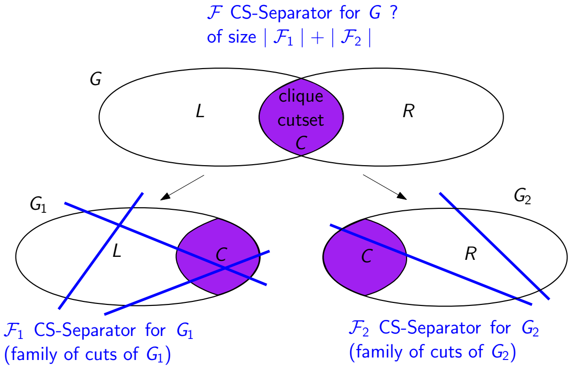
Valid decomposition



Valid decomposition

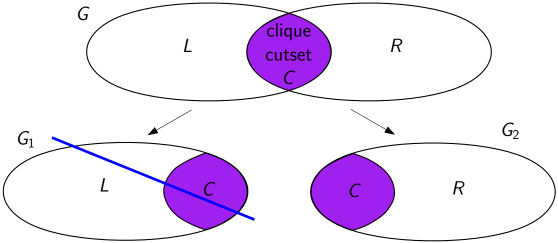


Valid decomposition

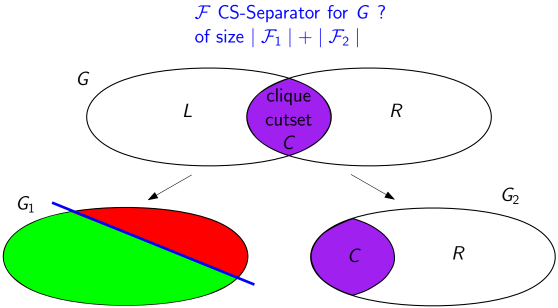


Valid decomposition

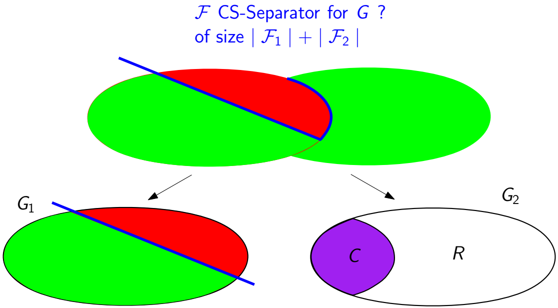
$\mathcal{F}$ CS-Separator for G ?
of size $|\ \mathcal{F}_1\ | + |\ \mathcal{F}_2\ |$



Valid decomposition

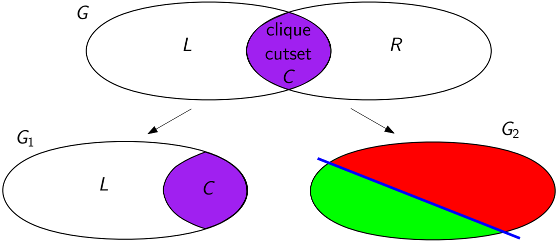


Valid decomposition



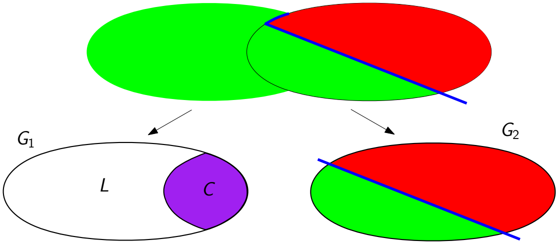
Valid decomposition

$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$



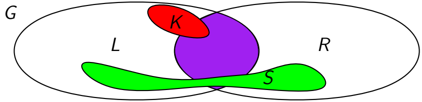
Valid decomposition

$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$



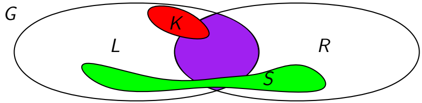
Valid decomposition

$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$



Valid decomposition

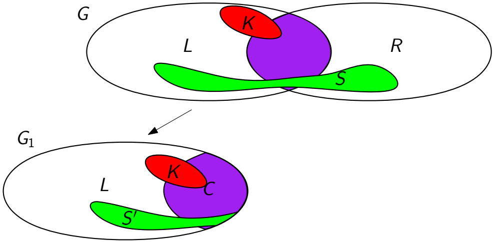
$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$



Clique K cannot intersect both L and R!

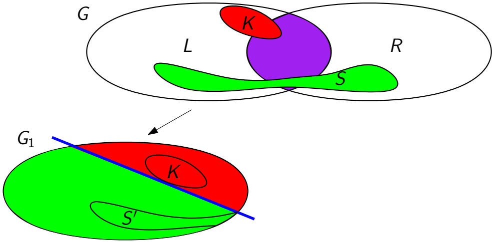
Valid decomposition

$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$

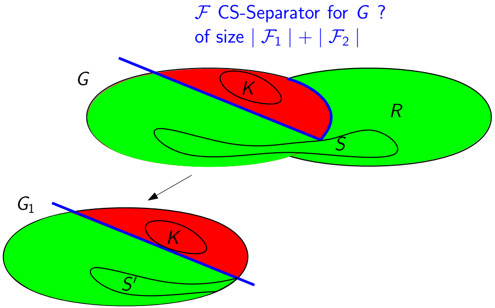


Valid decomposition

$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$



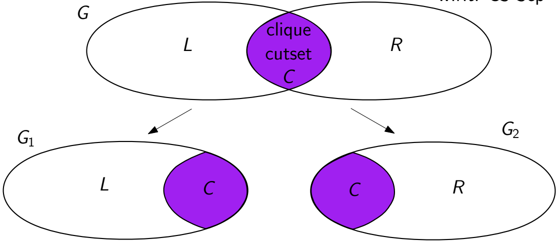
Valid decomposition



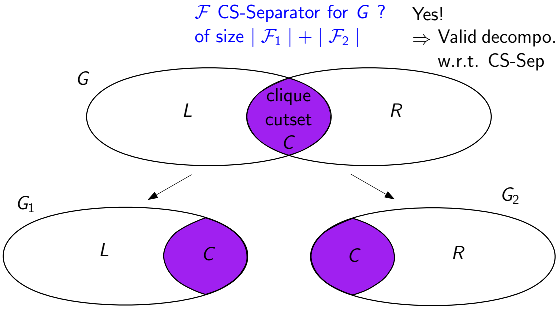
Valid decomposition

$\mathcal{F}$ CS-Separator for G ?
of size $|\mathcal{F}_1| + |\mathcal{F}_2|$

Yes!
$\Rightarrow$ Valid decompo.
w.r.t. CS-Sep

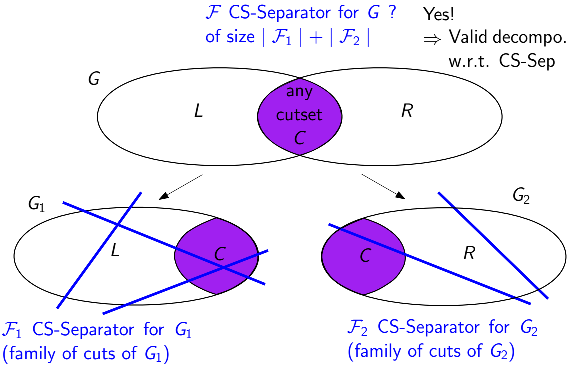


Valid decomposition



Clique-cutset?

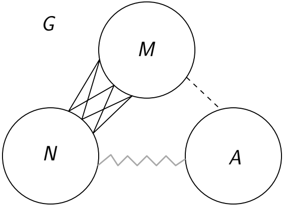
Valid decomposition



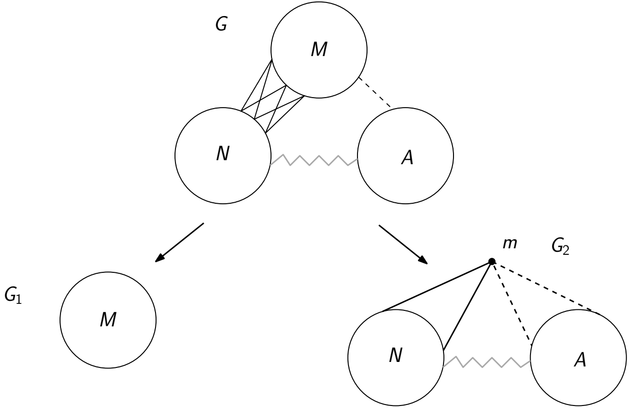
Clique-cutset?

Any cutset gives a valid decomposition into two children!

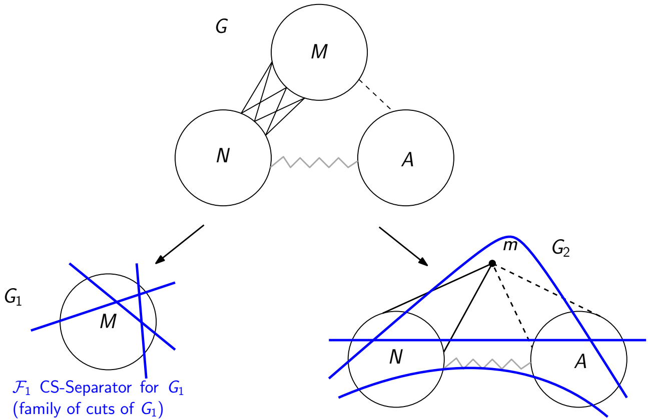
Modules gives valid decomposition w.r.t CS-Sep



Modules gives valid decomposition w.r.t CS-Sep



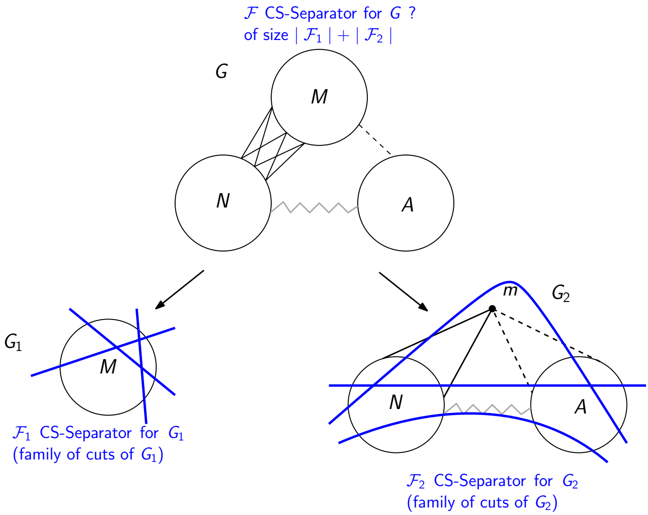
Modules gives valid decomposition w.r.t CS-Sep



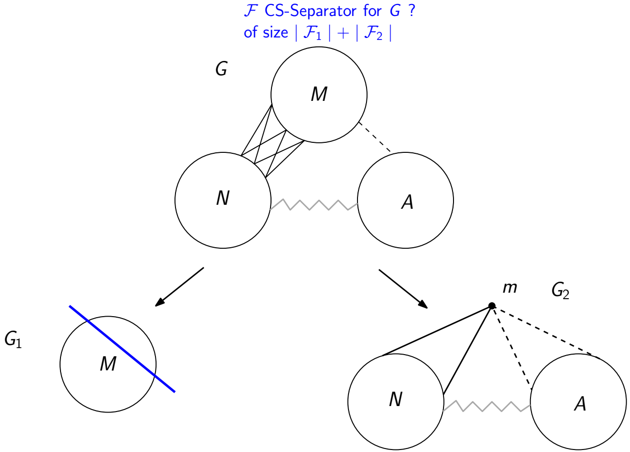
$\mathcal{F}_1$ CS-Separator for G_1
(family of cuts of G_1)

$\mathcal{F}_2$ CS-Separator for G_2
(family of cuts of G_2)

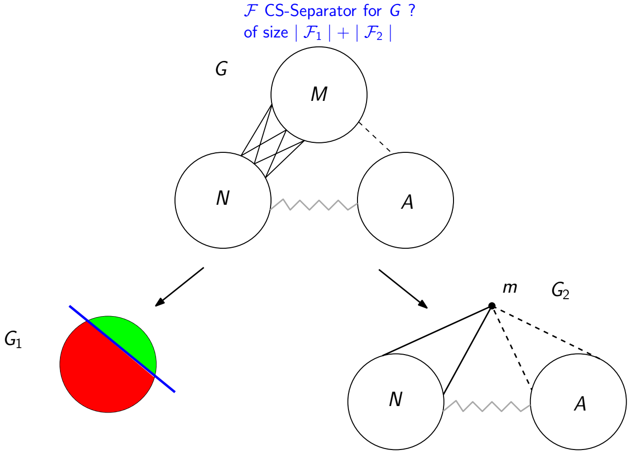
Modules gives valid decomposition w.r.t CS-Sep



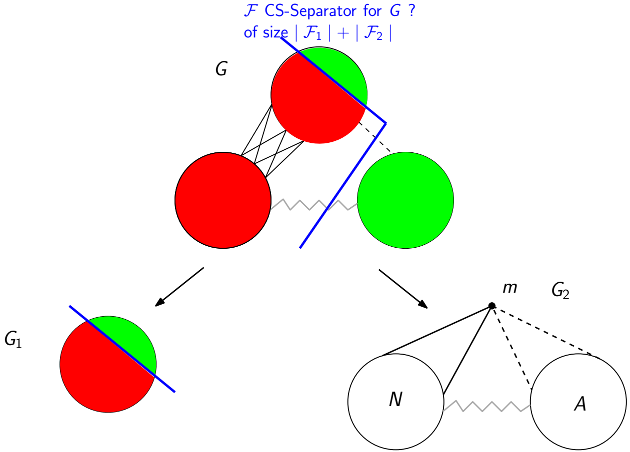
Modules gives valid decomposition w.r.t CS-Sep



Modules gives valid decomposition w.r.t CS-Sep

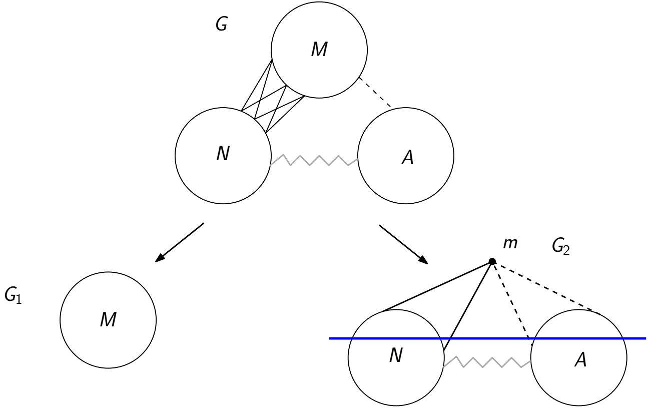


Modules gives valid decomposition w.r.t CS-Sep

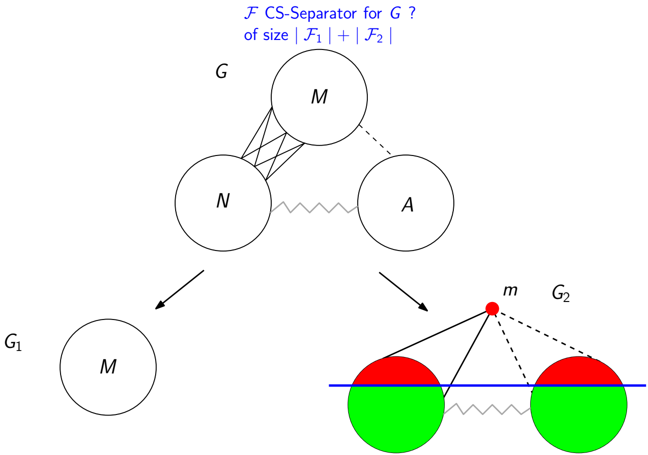


Modules gives valid decomposition w.r.t CS-Sep

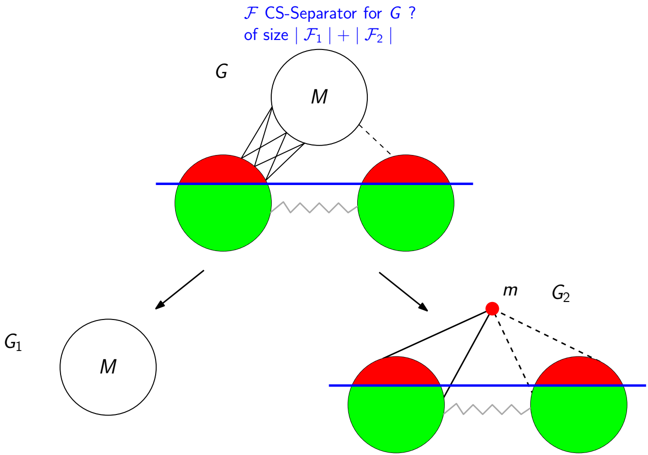
$\mathcal{F}$ CS-Separator for G ?
of size $\mid \mathcal{F}_1 \mid + \mid \mathcal{F}_2 \mid$



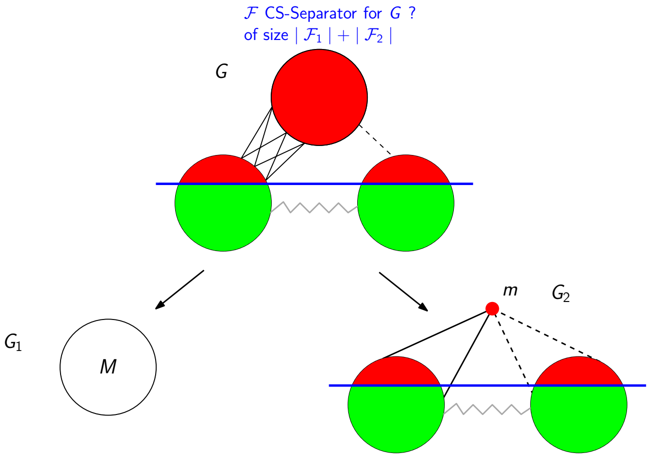
Modules gives valid decomposition w.r.t CS-Sep



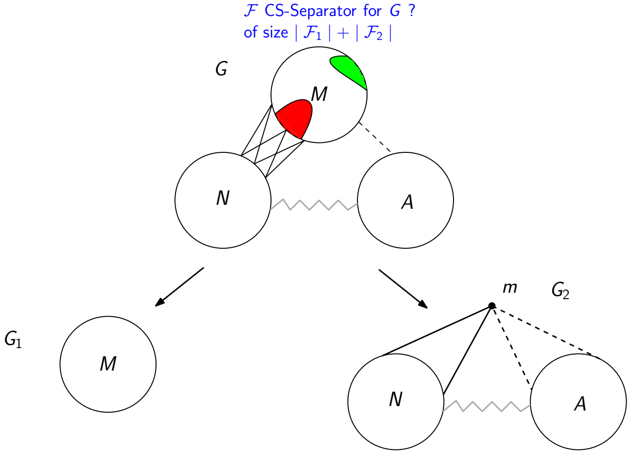
Modules gives valid decomposition w.r.t CS-Sep



Modules gives valid decomposition w.r.t CS-Sep

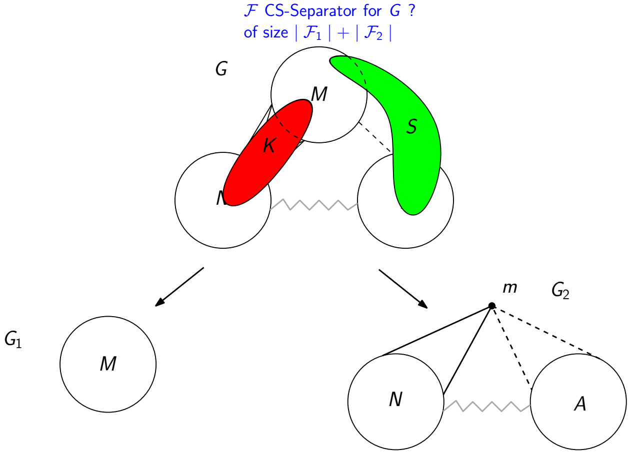


Modules gives valid decomposition w.r.t CS-Sep



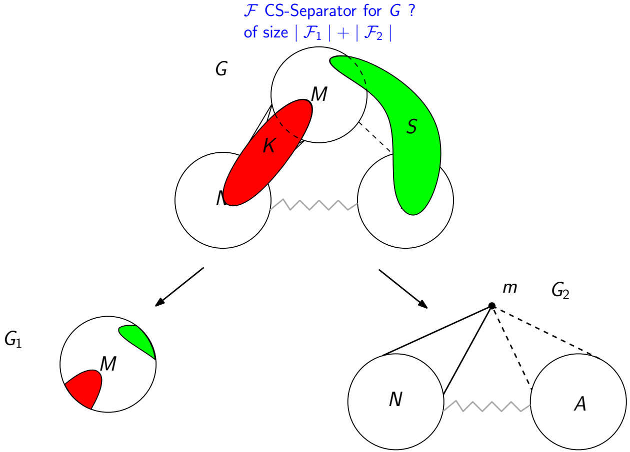
Case 1: Both K and S intersect M

Modules gives valid decomposition w.r.t CS-Sep



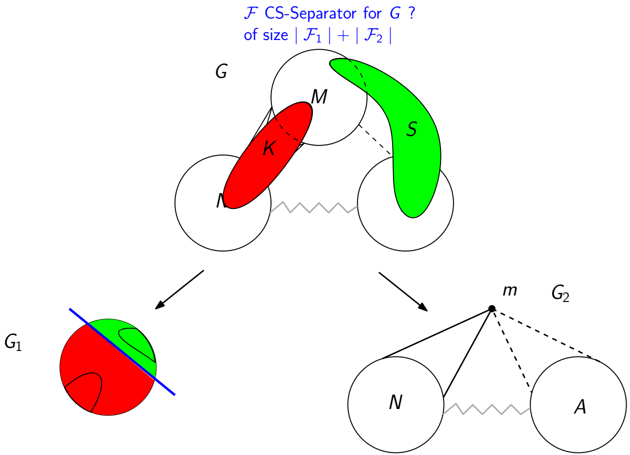
Case 1: Both K and S intersect M

Modules gives valid decomposition w.r.t CS-Sep



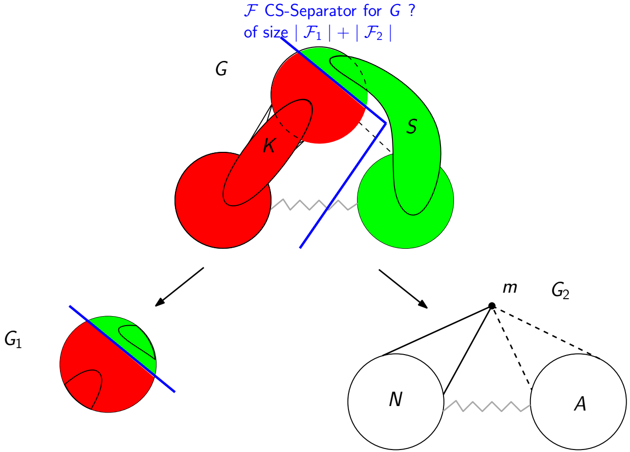
Case 1: Both K and S intersect M

Modules gives valid decomposition w.r.t CS-Sep



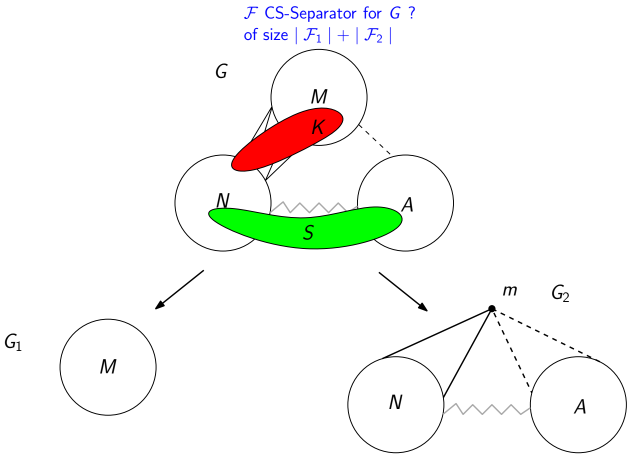
Case 1: Both K and S intersect M

Modules gives valid decomposition w.r.t CS-Sep



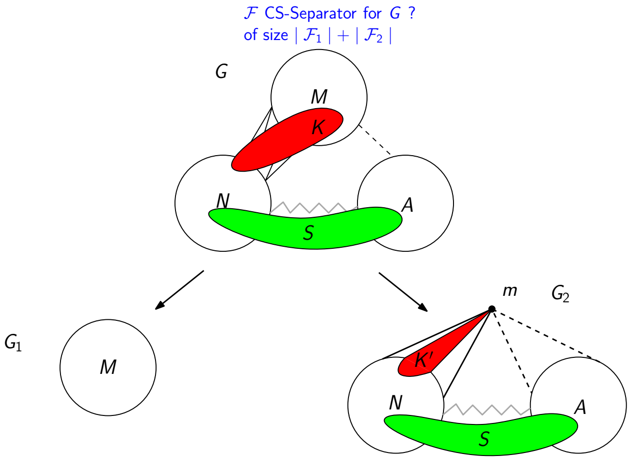
Case 1: Both K and S intersect M

Modules gives valid decomposition w.r.t CS-Sep



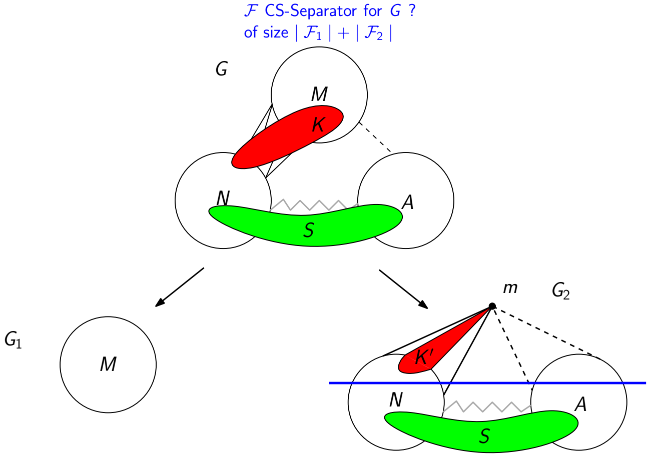
Case 2: At least one of K or S does not intersect M

Modules gives valid decomposition w.r.t CS-Sep



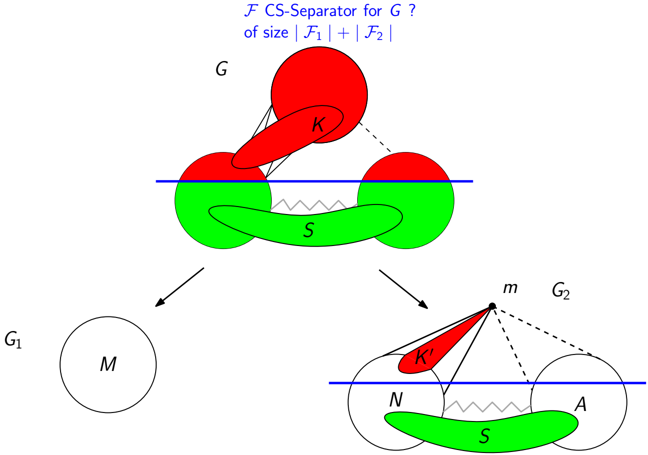
Case 2: At least one of K or S does not intersect M

Modules gives valid decomposition w.r.t CS-Sep



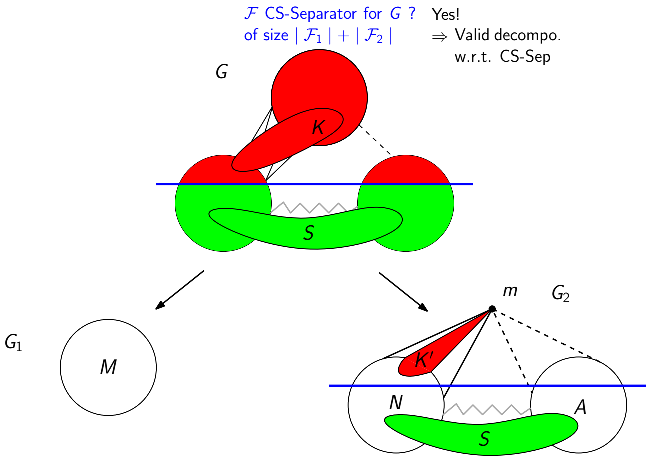
Case 2: At least one of K or S does not intersect M

Modules gives valid decomposition w.r.t CS-Sep



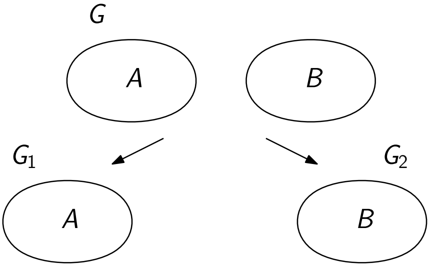
Case 2: At least one of K or S does not intersect M

Modules gives valid decomposition w.r.t CS-Sep



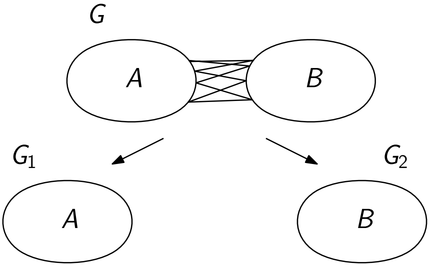
List of valid decompositions w.r.t CS-Sep

- component decomposition



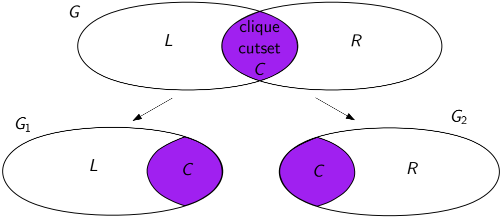
List of valid decompositions w.r.t CS-Sep

- component decomposition
- anticomponent decomposition



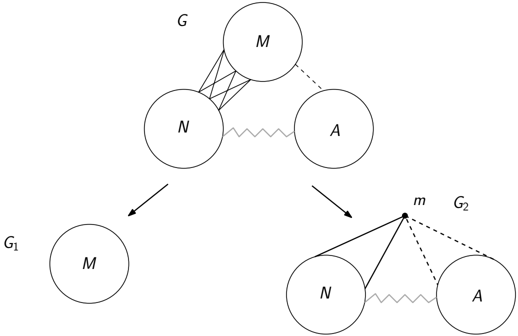
List of valid decompositions w.r.t CS-Sep

- component decomposition
- anticomponent decomposition
- cutset decomposition



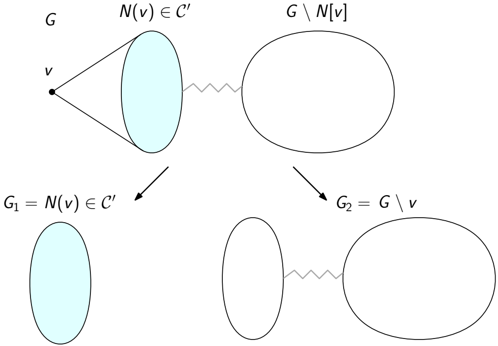
List of valid decompositions w.r.t CS-Sep

- component decomposition
- anticomponent decomposition
- cutset decomposition
- module decomposition



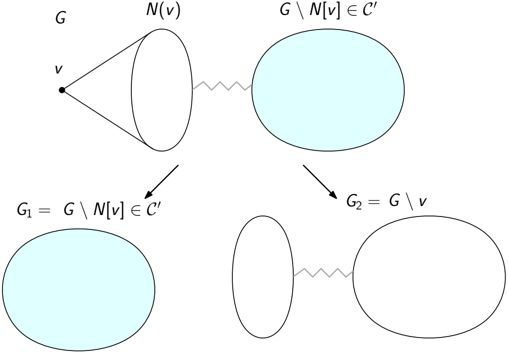
List of valid decompositions w.r.t CS-Sep

- component decomposition
- anticomponent decomposition
- cutset decomposition
- module decomposition
- $\mathcal{C}'$-neighborhood decomposition



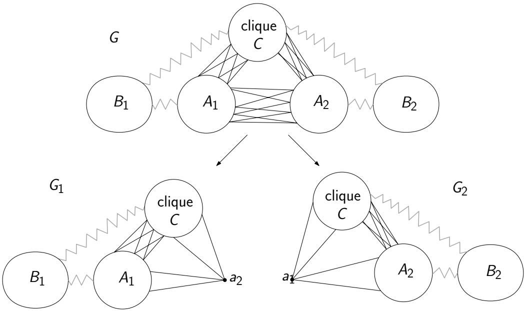
List of valid decompositions w.r.t CS-Sep

- component decomposition
- anticomponent decomposition
- cutset decomposition
- module decomposition
- $\mathcal{C}'$-neighborhood decomposition
- $\mathcal{C}'$-antineighborhood decomposition



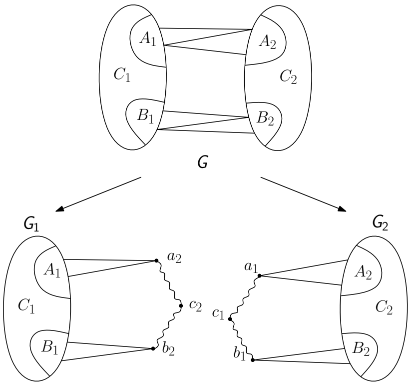
List of valid decompositions w.r.t CS-Sep

- component decomposition
- anticomponent decomposition
- cutset decomposition
- module decomposition
- $\mathcal{C}'$-neighborhood decomposition
- $\mathcal{C}'$-antineighborhood decomposition
- amalgam decomposition

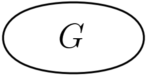


List of valid decompositions w.r.t CS-Sep

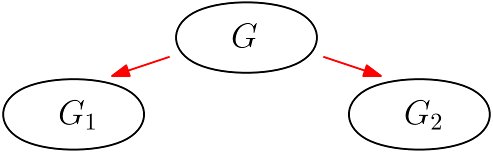
- component decomposition
- anticomponent decomposition
- cutset decomposition
- module decomposition
- $\mathcal{C}'$-neighborhood decomposition
- $\mathcal{C}'$-antineighborhood decomposition
- amalgam decomposition
- $\approx$ 2-join decomposition



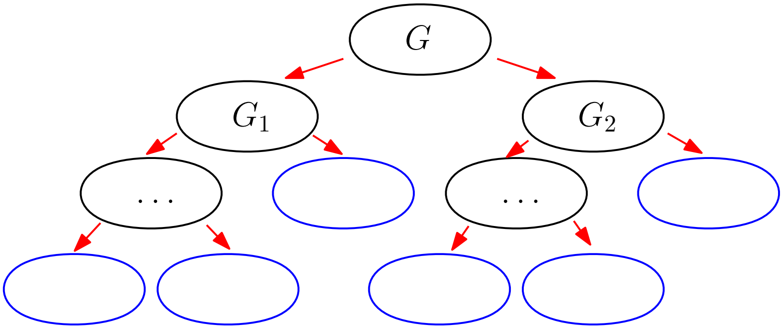
Too good to be true



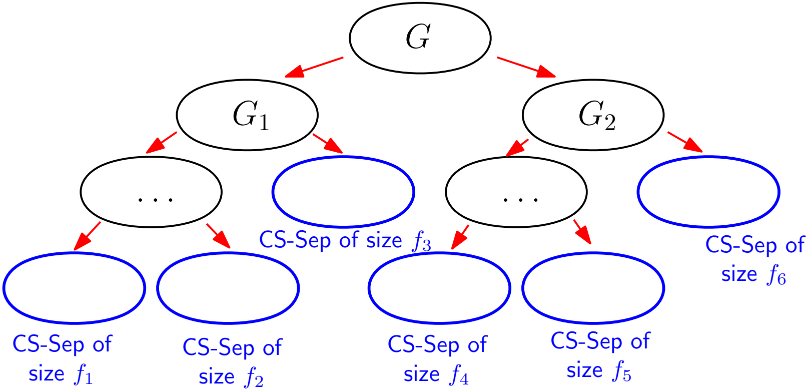
Too good to be true



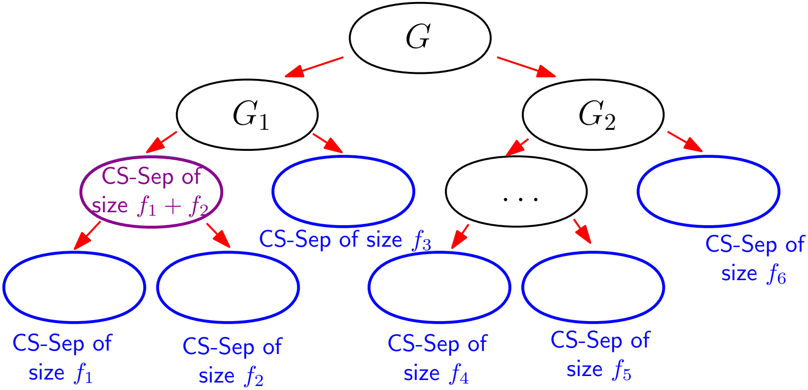
Too good to be true



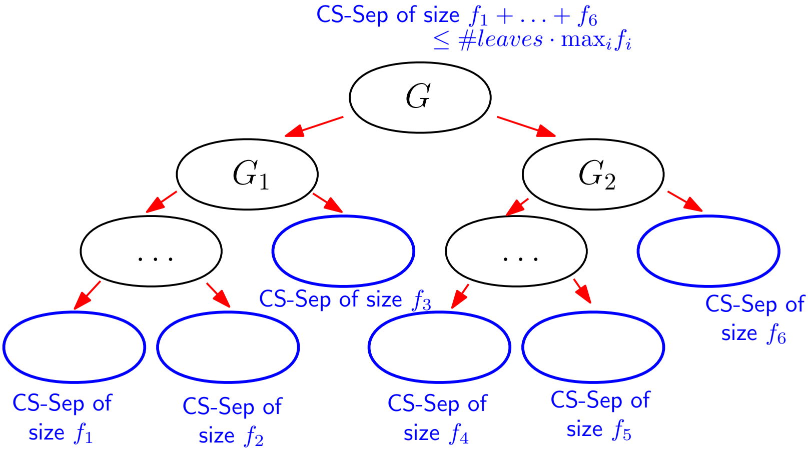
Too good to be true



Too good to be true

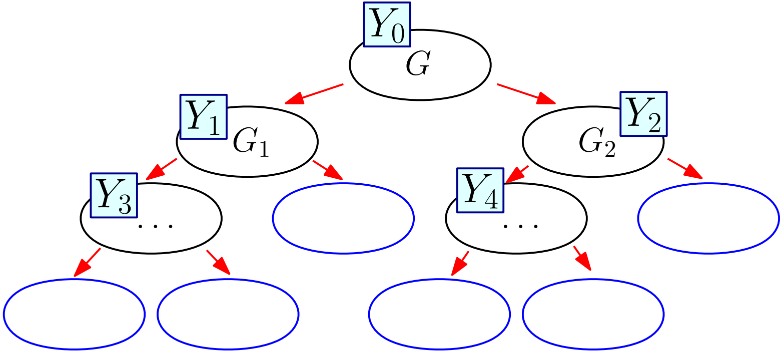


Too good to be true



Need to control the size of the tree!

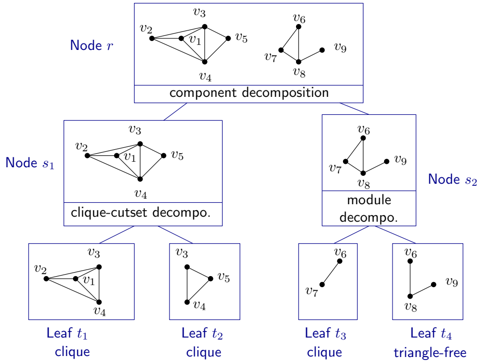
Too good to be true



Need to control the size of the tree!

$\Rightarrow$ Choose a *unique* label for each (internal) node among a poly. number of subsets of $V(G)$ (e.g. non-edges, triples, squares, ...)

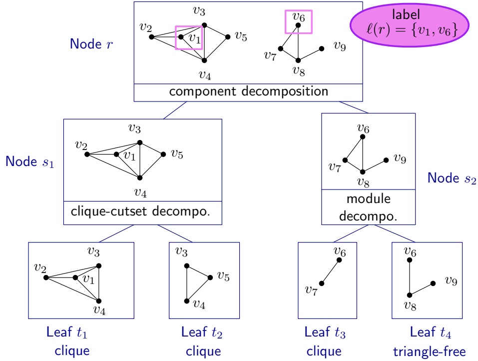
Label the tree : an example



Label with **non-edges** because:

- $O(n^2)$ of them
- $\geq$ one non-edge is "broken" (does not survive in any child) $\Rightarrow$ use it to label the node
- no non-edge can survive in both children

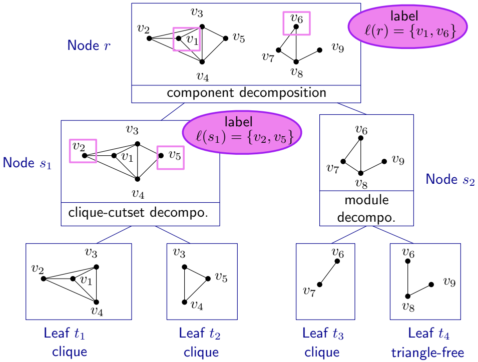
Label the tree : an example



Label with **non-edges** because:

- $O(n^2)$ of them
- $\geq$ one non-edge is "broken" (does not survive in any child) $\Rightarrow$ use it to label the node
- no non-edge can survive in both children

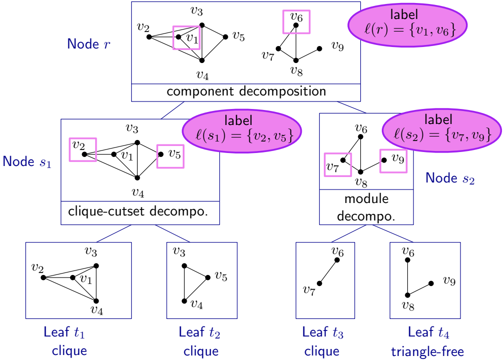
Label the tree : an example



Label with **non-edges** because:

- $O(n^2)$ of them
- $\geq$ one non-edge is "broken" (does not survive in any child)
  $\Rightarrow$ use it to label the node
- no non-edge can survive in both children

Label the tree : an example



Label with **non-edges** because:

- $O(n^2)$ of them
- $\geq$ one non-edge is "broken" (does not survive in any child)
  $\Rightarrow$ use it to label the node
- no non-edge can survive in both children

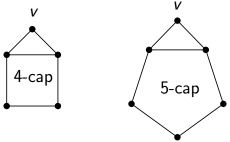
The framework

Sufficient conditions to have a polynomial CS-Sep for a graph G:

- Find a valid decomposition tree for G
- Prove that every leaf has a polynomial CS-Separator
- Bound the size of the tree by $\mathrm{poly}(n)$:
  Find a labeling as follows:
  - polynomial number of label candidates
  - the label of each node is "broken" (does not survive in any child)
  - no label candidate can survive in both children

  $\Rightarrow$ Injective labeling

Cap-free graphs



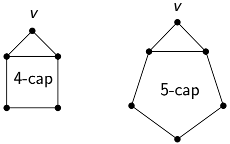
A *cap* $=$ a hole (length ≥ 4) $+$ a vertex v incident to 2 consecutive vertices

Theorem [Conforti, Cornuéjols, Kappor, Vušković 99]

Every connected cap-free graph either has an amalgam or is a basic cap-free graph.

A *basic* cap-free graph is either chordal or *almost triangle-free*.

Cap-free graphs



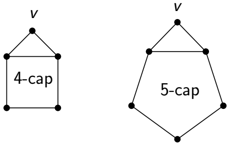
A *cap* = a hole (length ≥ 4) + a vertex v incident to 2 consecutive vertices

Theorem [Conforti, Cornuéjols, Kappor, Vušković 99]

Every connected cap-free graph either has an amalgam or is a basic cap-free graph.

A *basic* cap-free graph is either chordal or *almost triangle-free*.
$\Rightarrow$ it has a quadratic CS-Separator.

Cap-free graphs



A *cap* = a hole (length ≥ 4) + a vertex v incident to 2 consecutive vertices

Theorem [Conforti, Cornuéjols, Kappor, Vušković 99]

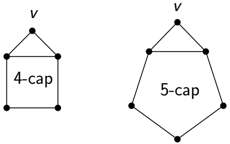
Every connected cap-free graph either has an amalgam or is a basic cap-free graph.

A *basic* cap-free graph is either chordal or *almost triangle-free*.
$\Rightarrow$ it has a quadratic CS-Separator.

---

1. Decompose using component, anticomponent and amalgam.

Cap-free graphs



A *cap* = a hole (length ≥ 4) + a vertex v incident to 2 consecutive vertices

Theorem [Conforti, Cornuéjols, Kappor, Vušković 99]

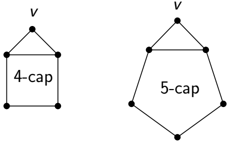
Every connected cap-free graph either has an amalgam or is a basic cap-free graph.

A *basic* cap-free graph is either chordal or *almost triangle-free*.
$\Rightarrow$ it has a quadratic CS-Separator.

---

1. Decompose using component, anticomponent and amalgam.
2. Label each node with a *trio*: at most three vertices containing a non-edge.

Cap-free graphs



A *cap* = a hole (length ≥ 4) + a vertex v incident to 2 consecutive vertices

Theorem [Conforti, Cornuéjols, Kappor, Vušković 99]

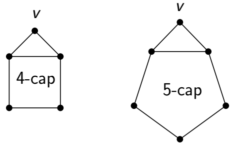
Every connected cap-free graph either has an amalgam or is a basic cap-free graph.

A *basic* cap-free graph is either chordal or *almost triangle-free*.
$\Rightarrow$ it has a quadratic CS-Separator.

---

1. Decompose using component, anticomponent and amalgam.
2. Label each node with a *trio*: at most three vertices containing a non-edge. $\Rightarrow O(n^3)$ different trios

Cap-free graphs



A *cap* = a hole (length ≥ 4) + a vertex v incident to 2 consecutive vertices

Theorem [Conforti, Cornuéjols, Kappor, Vušković 99]

Every connected cap-free graph either has an amalgam or is a basic cap-free graph.

A *basic* cap-free graph is either chordal or *almost triangle-free*.
$\Rightarrow$ it has a quadratic CS-Separator.

---

1. Decompose using component, anticomponent and amalgam.
2. Label each node with a *trio*: at most three vertices containing a non-edge. $\Rightarrow O(n^3)$ different trios

Theorem [Bousquet, L. Maffray, Pastor 18]

Every cap-free graph admits a $O(n^5)$ CS-Separator.

Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

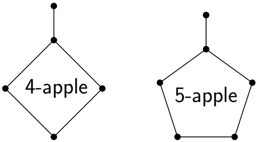
claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

Apple = a hole + a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

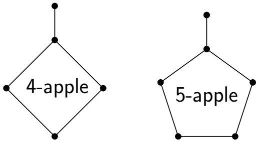
claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

Apple $=$ a hole $+$ a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

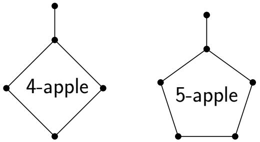
into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

C_6-free graph and in $\mathcal{C}'$

Apple = a hole + a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

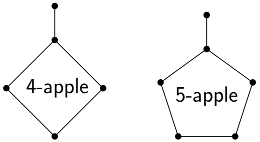
into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

C_6-free graph and in $\mathcal{C}'$

Apple $=$ a hole $+$ a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

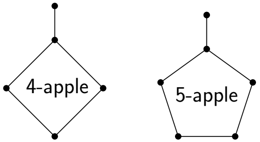
C_6-free graph and in $\mathcal{C}'$

Can be decomposed with:

- module
- clique-cutset
- $\{C_5$-*free*$\}$-antineighborhood

into cliques or (apple, C_5)-free

Apple = a hole + a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

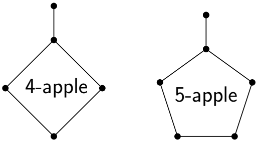
C_6-free graph and in $\mathcal{C}'$

Can be decomposed with:

- module
- clique-cutset
- $\{C_5\text{-}\mathit{free}\}$-antineighborhood

into cliques or (apple, C_5)-free

Apple = a hole + a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

C_6-free graph and in $\mathcal{C}'$

Can be decomposed with:

- module
- clique-cutset
- $\{C_5\text{-}\textit{free}\}$-antineighborhood

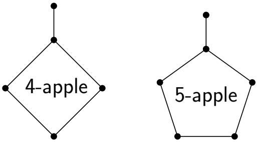
into cliques or (apple, C_5)-free graphs

Can be decomposed with:

- module
- clique-cutset
- chordal-antineighborhood

into chordal graphs

Apple = a hole + a leaf



Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

C_6-free graph and in $\mathcal{C}'$

Can be decomposed with:

- module
- clique-cutset
- $\{\mathcal{C}_5\text{-}\textit{free}\}$-antineighborhood

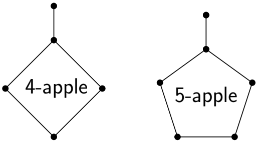
into cliques or (apple, C_5)-free

Can be decomposed with:

- module
- clique-cutset
- chordal-antineighborhood

into chordal graphs

Apple = a hole + a leaf



Chordal graphs : linear CS-Sep.
Claw-free graphs : $\mathcal{O}(n^4)$ CS-Sep.

Apple-free graphs

Theorem [Brandstädt, Lozin, Mosca 10]

Apple-free graph

or

claw-free graph

Can be decomposed with:

- module
- clique-cutset
- *easier* $\mathcal{C}'$-antineighborhood

into easier graphs (cliques or in $\mathcal{C}'$)

or

claw-free graph

C_6-free graph and in $\mathcal{C}'$

Can be decomposed with:

- module
- clique-cutset
- $\{C_5\text{-}free\}$-antineighborhood

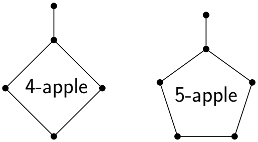
into cliques or (apple, C_5)-free

Can be decomposed with:

- module
- clique-cutset
- chordal-antineighborhood

into chordal graphs

Apple = a hole + a leaf



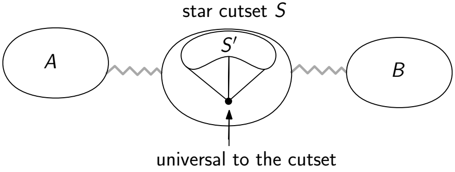
Chordal graphs : linear CS-Sep.
Claw-free graphs : $\mathcal{O}(n^4)$ CS-Sep.

Theorem [BLMP 18]

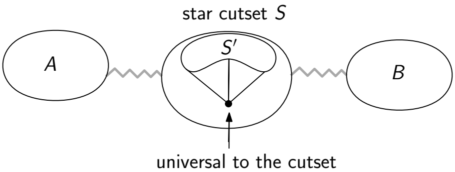
Every apple-free graphs has a CS-Separator of size $\mathcal{O}(n^{10})$.

(use trios again to label the trees)

Star-cutset: still not friendly enough



Star-cutset: still not friendly enough

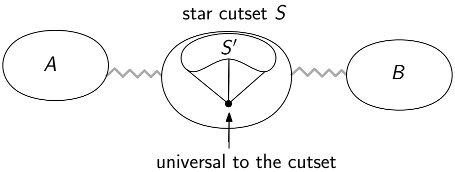


Theorem[Bousquet, L., Maffray, Pastor 18]

There exists a class $\mathcal{D}$ of graphs such that:

1. every graph of $\mathcal{D}$ is either a clique or admits a star-cutset decomposition (G_1, G_2) with $G_1, G_2 \in \mathcal{D}$, and
2. $\mathcal{D}$ does not have the polynomial CS-Separator property.

Star-cutset: still not friendly enough



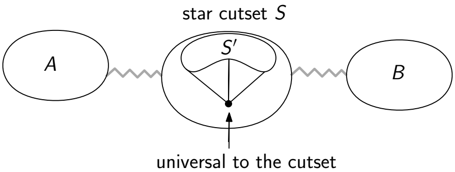
Theorem[Bousquet, L., Maffray, Pastor 18]

There exists a class $\mathcal{D}$ of graphs such that:

1. every graph of $\mathcal{D}$ is either a clique or admits a star-cutset decomposition (G_1, G_2) with $G_1, G_2 \in \mathcal{D}$, and
2. $\mathcal{D}$ does not have the polynomial CS-Separator property.

Let $\mathcal{C}$ be a class not having the polynomial CS-Separator property,

Star-cutset: still not friendly enough



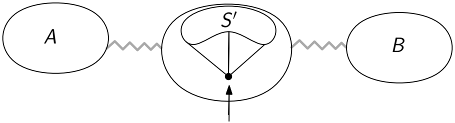
Theorem[Bousquet, L., Maffray, Pastor 18]

There exists a class $\mathcal{D}$ of graphs such that:

1. every graph of $\mathcal{D}$ is either a clique or admits a star-cutset decomposition (G_1, G_2) with $G_1, G_2 \in \mathcal{D}$, and
2. $\mathcal{D}$ does not have the polynomial CS-Separator property.

Let $\mathcal{C}$ be a class not having the polynomial CS-Separator property, $\mathcal{C}'$ its closure under taking induced subgraphs,

Star-cutset: still not friendly enough



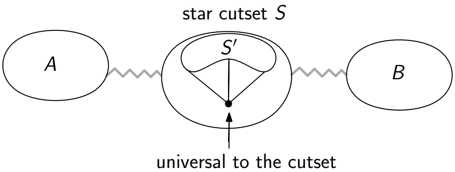
Theorem[Bousquet, L., Maffray, Pastor 18]

There exists a class $\mathcal{D}$ of graphs such that:

1. every graph of $\mathcal{D}$ is either a clique or admits a star-cutset decomposition (G_1, G_2) with $G_1, G_2 \in \mathcal{D}$, and
2. $\mathcal{D}$ does not have the polynomial CS-Separator property.

Let $\mathcal{C}$ be a class not having the polynomial CS-Separator property,
$\mathcal{C}'$ its closure under taking induced subgraphs,
$\mathcal{D}$ is obtained by adding a universal vertex to every graph of $\mathcal{C}'$

Star-cutset: still not friendly enough



Theorem[Bousquet, L., Maffray, Pastor 18]

There exists a class $\mathcal{D}$ of graphs such that:

1. every graph of $\mathcal{D}$ is either a clique or admits a star-cutset decomposition (G_1, G_2) with $G_1, G_2 \in \mathcal{D}$, and
2. $\mathcal{D}$ does not have the polynomial CS-Separator property.

Let $\mathcal{C}$ be a class not having the polynomial CS-Separator property,
$\mathcal{C}'$ its closure under taking induced subgraphs,
$\mathcal{D}$ is obtained by adding a universal vertex to every graph of $\mathcal{C}'$

⚠ $\mathcal{D}$ is not hereditary (so it is a bit of cheating..)

Conclusion

Looking for more decomposition theorems to exploit! :-)

Conclusion

Looking for more decomposition theorems to exploit! :-)

Open question

Do perfect graphs have polynomial CS-Separators?

Conclusion

Looking for more decomposition theorems to exploit! :-)

Open question

Do perfect graphs have polynomial CS-Separators?

Thank you for your attention!