Algorithm Graph Theory: How hard is your combinatorial optimization problem?

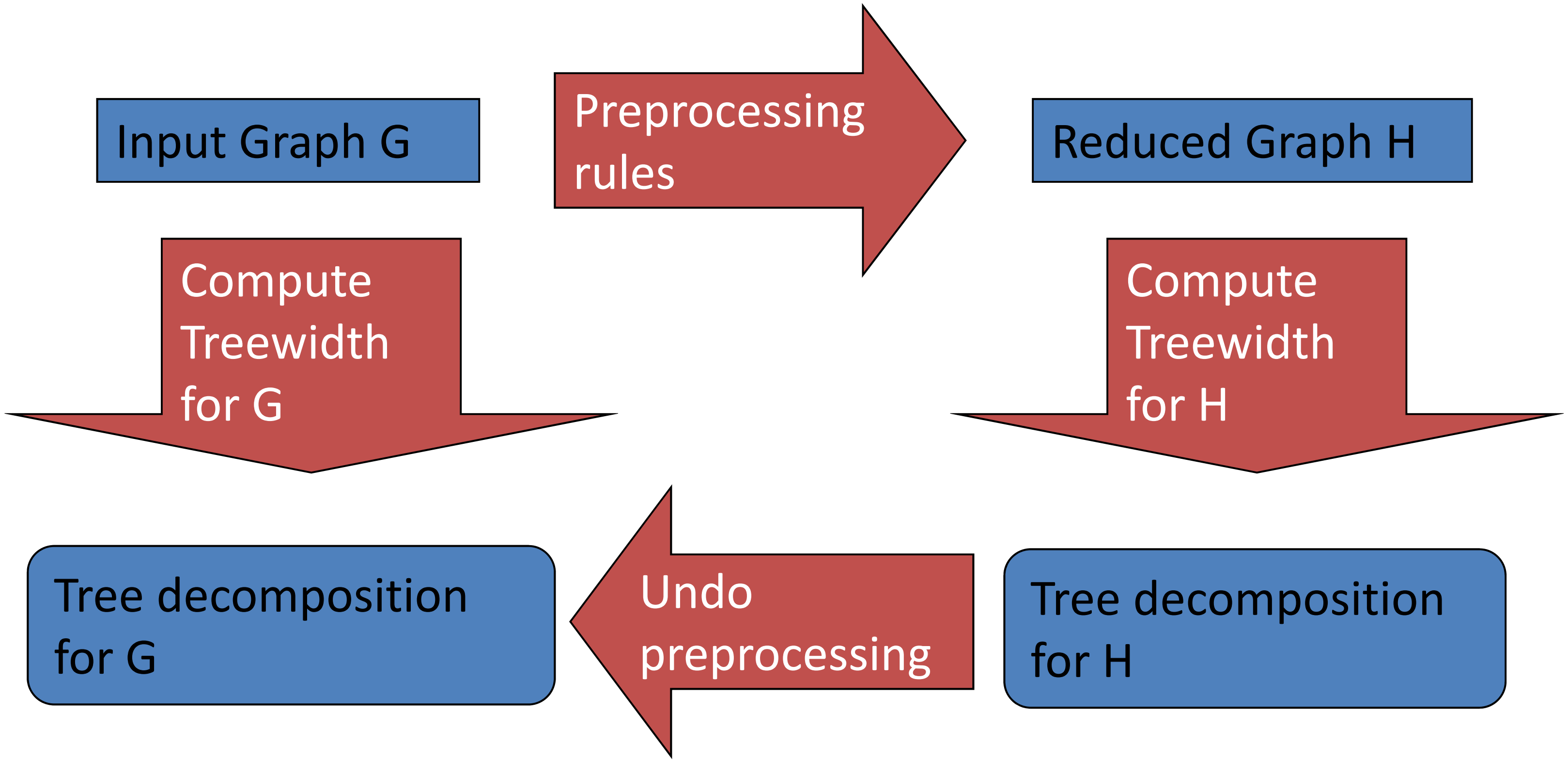
Short Course – Lecture 6
June 9, 2017

Slides available at:
https://www.math2.rwth-aachen.de/de/mitarbeiter/koster/agtclemson

Two types of preprocessing

- Reduction rules (*Simplification*) [39]
  - Rules that change G into a smaller `equivalent' graph
  - Maintains a lower bound variable for treewidth *low*
- Safe separators (*Divide and Conquer*) [32]
  - Splits the graph into two or more smaller parts with help of a separator that is made to a clique

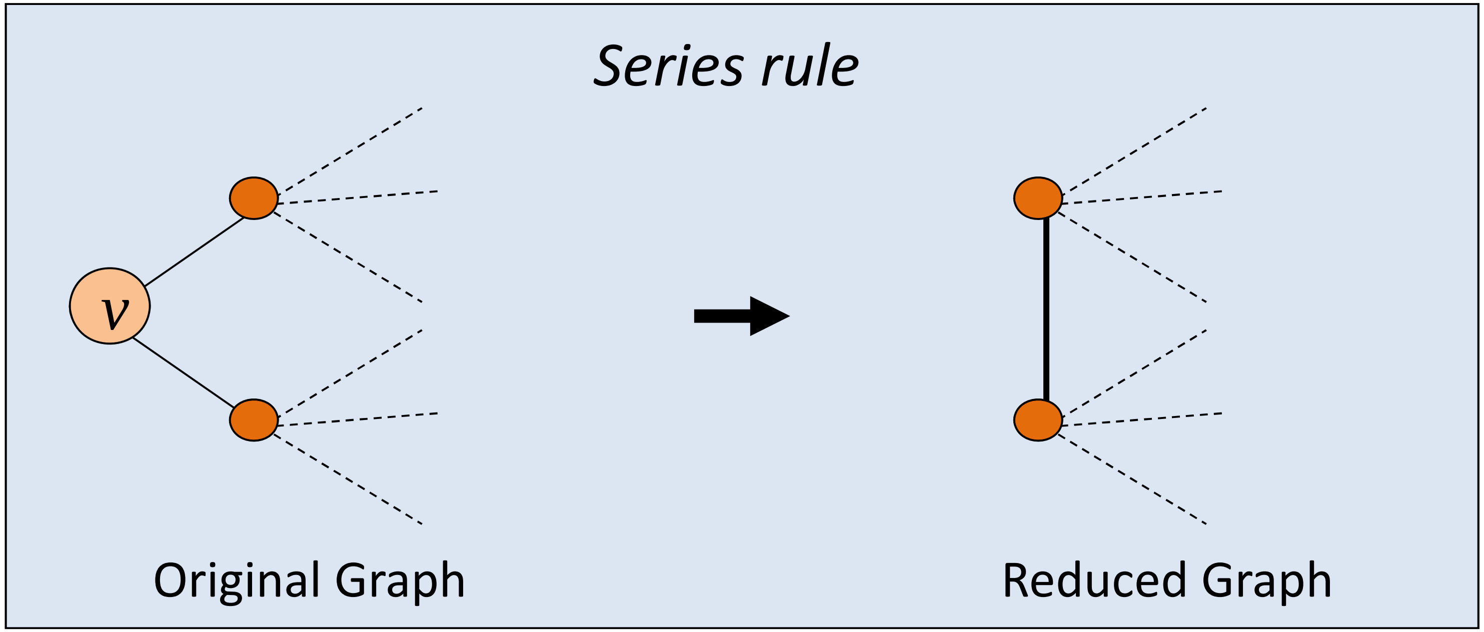
Reduction



- Safe rules that
  - Make *G* smaller
  - Maintain optimality…
- Use for preprocessing graphs when computing treewidth

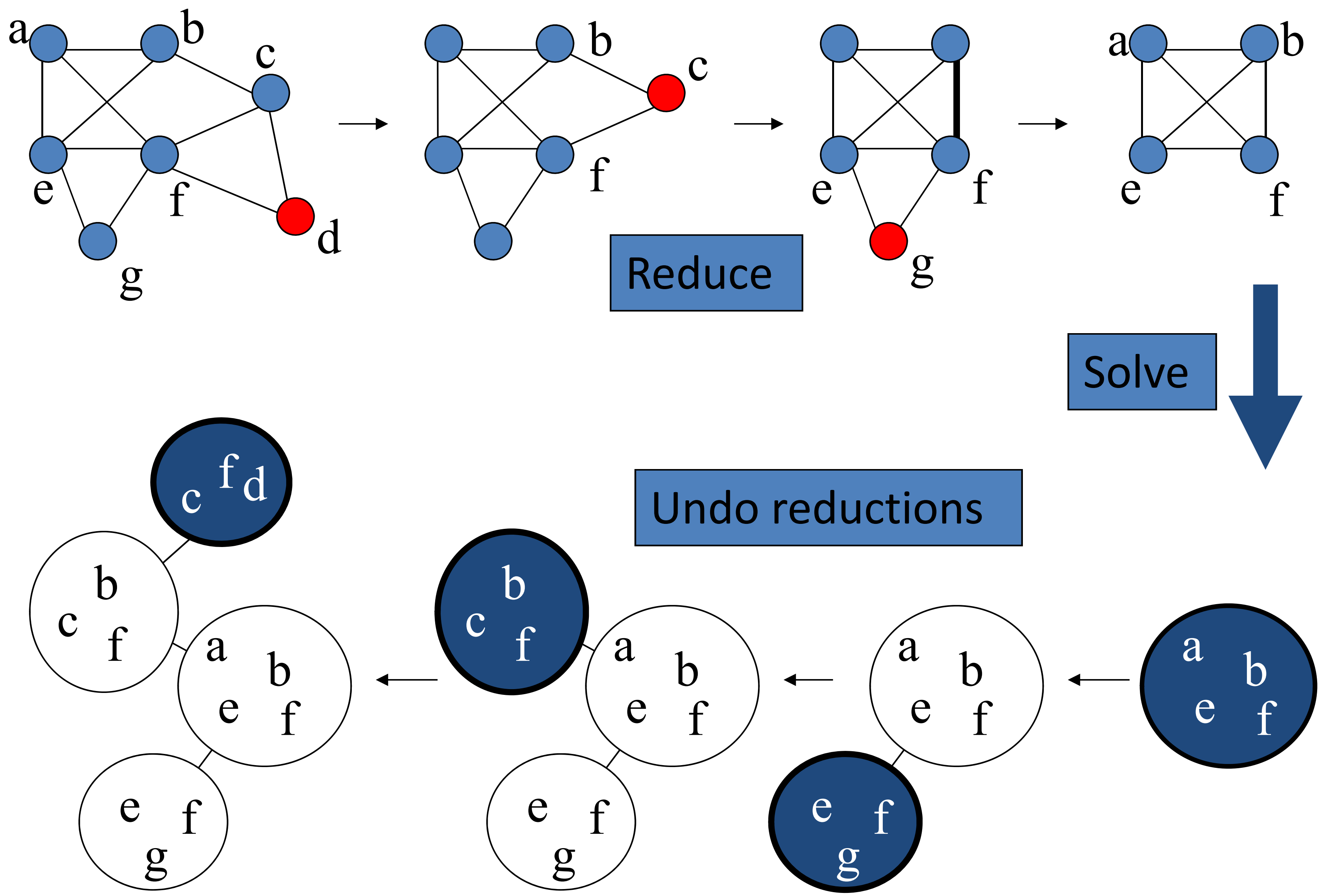
Reduction rules

- Uses and generalizes ideas and rules from algorithm to recognize graphs of treewidth ≤ 3 from Arnborg and Proskurowski
- Example: *Series rule:* remove a vertex of degree 2 and connect its neighbors



- Safe for graphs of treewidth ≥ 2

Example



Type of rules

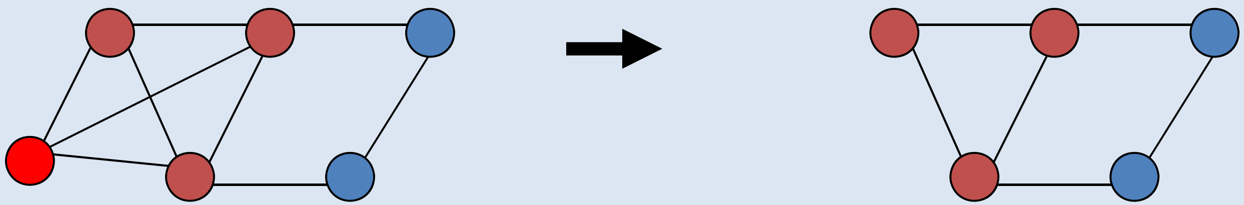
- Variable: **low** (integer, lower bound on treewidth)
- Graph G
- Invariant: value of **max(low, treewidth(G))**
- Rules
  - Locally rewrite G to a graph with fewer vertices
  - Possibly update or check **low**
- We say a rule is *safe*, when it maintains the invariant.
- Use only safe rules.

Rule 1: Simplicial rule

- Let *v* be a simplicial vertex in *G*
- Remove *v*.
- Set *low := max (low, degree(v))*

Simplicial =
Neighbors form a clique

Simplicial rule



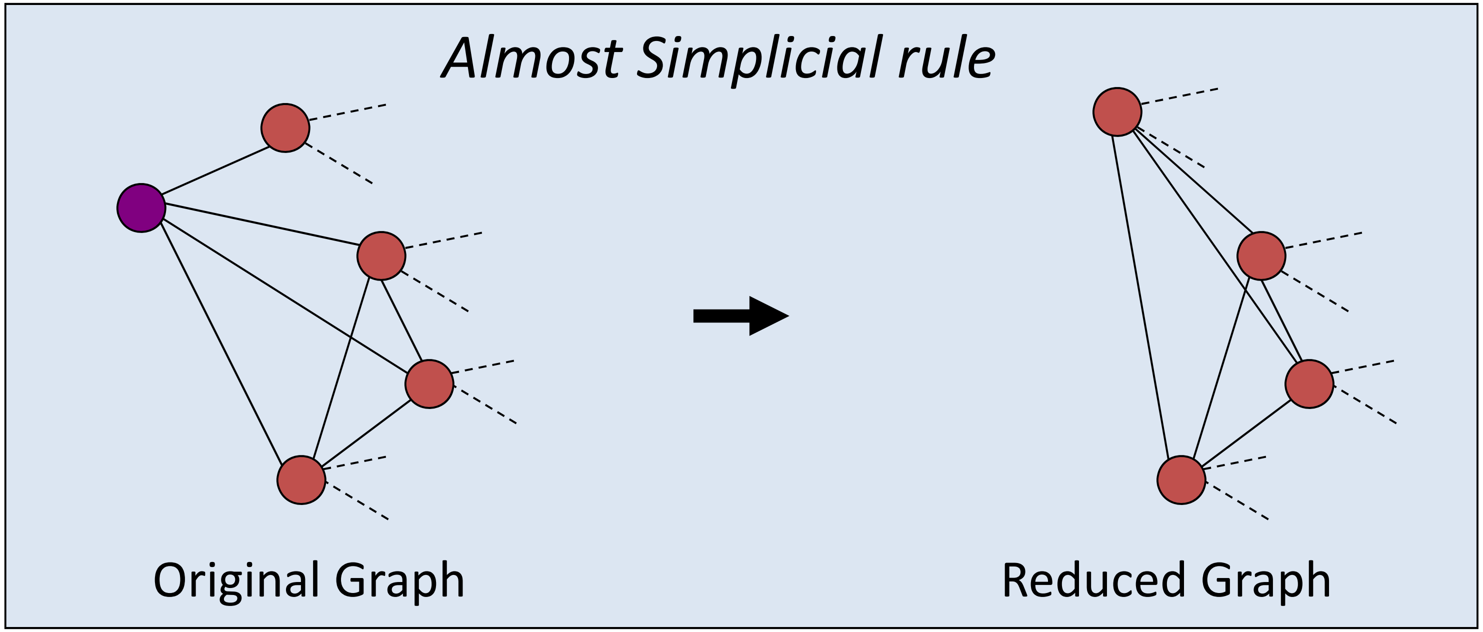
Original Graph   Reduced Graph

- Simplicial rule is safe.
- Special cases: islet rule (singletons), twig rule (*degree(v)* = 1)

Rule 2: Almost Simplicial rule

- Let v be a almost simplicial vertex in G and $low \geq degree(v)$
- Remove v,
- turn neighbors into clique

Almost Simplicial = Neighbors except one form a clique



- Almost Simplicial rule is safe.

Increasing *low*

Further rules: buddy/buddies rule, (extended) cube rule

Arnborg and Proskurowski [12]:

- tw(G)=1 if and only if G is reduced to the empty graph by islet rule (vertices of degree 0) and twig rule (vertices of degree 1)
- tw(G)=2 if and only if G is reduced to the empty graph by islet, twig, and series rule (vertices of degree 2)
- tw(G)=3 if and only if G is reduced to the empty graph by islet, twig, series, triangle, buddy, and cube rule

Low can be increased to 2, 3, and 4 respectively if these rules cannot be applied anymore and graph is not empty yet.

Results for probabilistic networks

	original		preprocessed		
instance	**|V|**	**|E|**	**|V|**	**|E|**	**low**
alarm	37	65	**0**	**0**	**4**
barley	48	126	26	78	4
boblo	221	328	**0**	**0**	**3**
diabetes	413	819	116	276	4
link	724	1738	308	1158	4
mildew	35	80	**0**	**0**	**4**
munin1	189	366	66	188	4
munin2	1003	1662	165	451	4
munin3	1044	1745	96	313	4
munin4	1041	1843	215	642	4
munin-kgo	1066	1730	**0**	**0**	**5**

	original		preprocessed		
instance	**|V|**	**|E|**	**|V|**	**|E|**	**low**
oesoca+	67	208	14	75	9
oesoca	39	67	**0**	**0**	**3**
oesoca42	42	72	**0**	**0**	**3**
oow-bas	27	54	**0**	**0**	**4**
oow-solo	40	87	27	63	4
oow-trad	33	72	23	54	4
pignet2	3032	7264	1002	3730	4
pigs	441	806	48	137	4
ship-ship	50	114	24	65	4
vsd	38	62	**0**	**0**	**4**
water	32	123	22	96	5
wilson	21	27	**0**	**0**	**3**

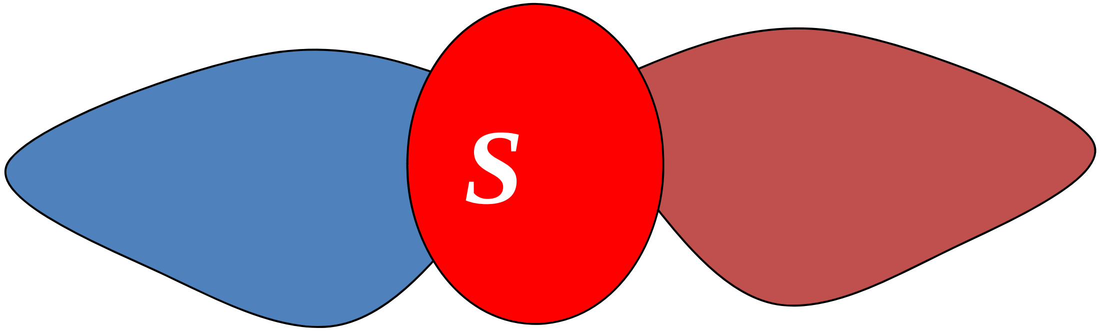
→ Some cases could be solved with preprocessing to optimality

→ Often substantial reductions obtained

→ Time needed for preprocessing is small (never more than a few seconds)

Graph separators

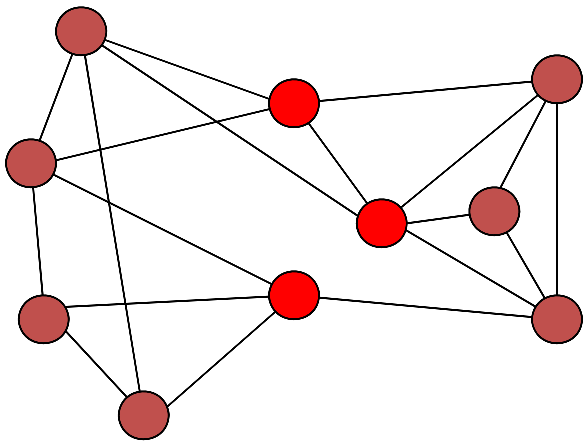
- $S \subset V$ is a *separator* of G, if $G - S$ has more than one connected component
- S is a *minimal separator*, if S is a separator and S does not contain another separator as proper subset



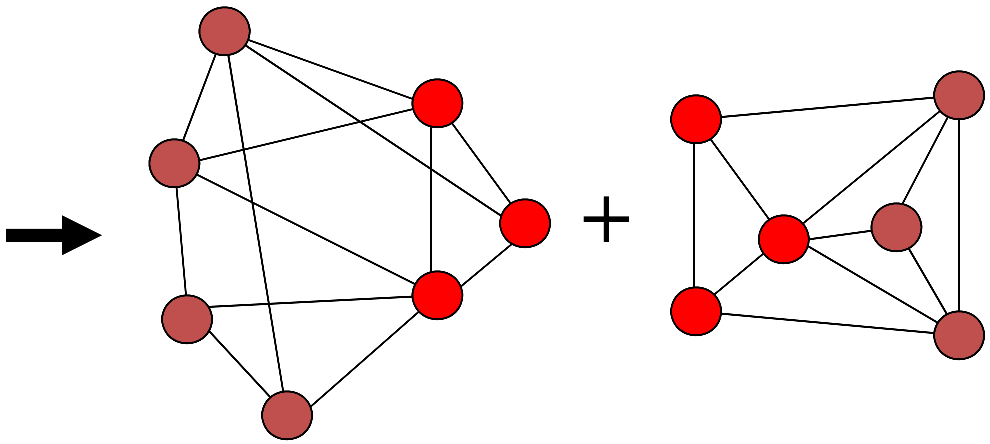
Safe separator

S is safe for treewidth, or *a* safe separator if and only if the treewidth of *G* equals the maximum over the treewidth of all graphs obtained by

- Taking a connected component *W* of *G* -*S*
- Take the graph, induced by $W \cup S$
- Make *S* into a clique in that graph



Original Graph



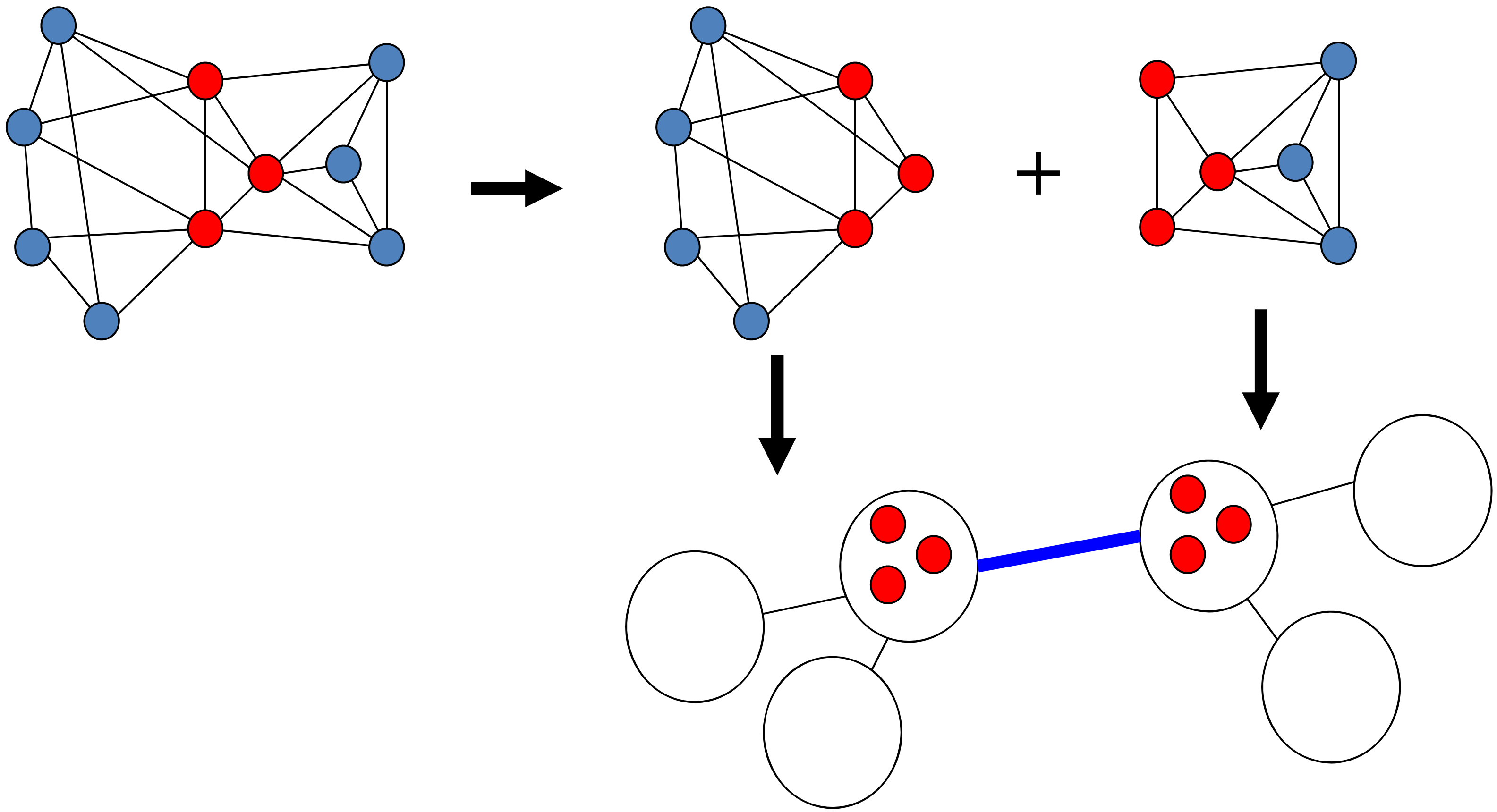
Components

Using safe separators

- Splitting the graph for divide and conquer preprocessing
- Until no safe separators can be found
- Slower but more powerful compared to reduction
  - Most or all reduction rules can be obtained as special cases of the use of safe separators
- Look for sufficient conditions for separators to be safe

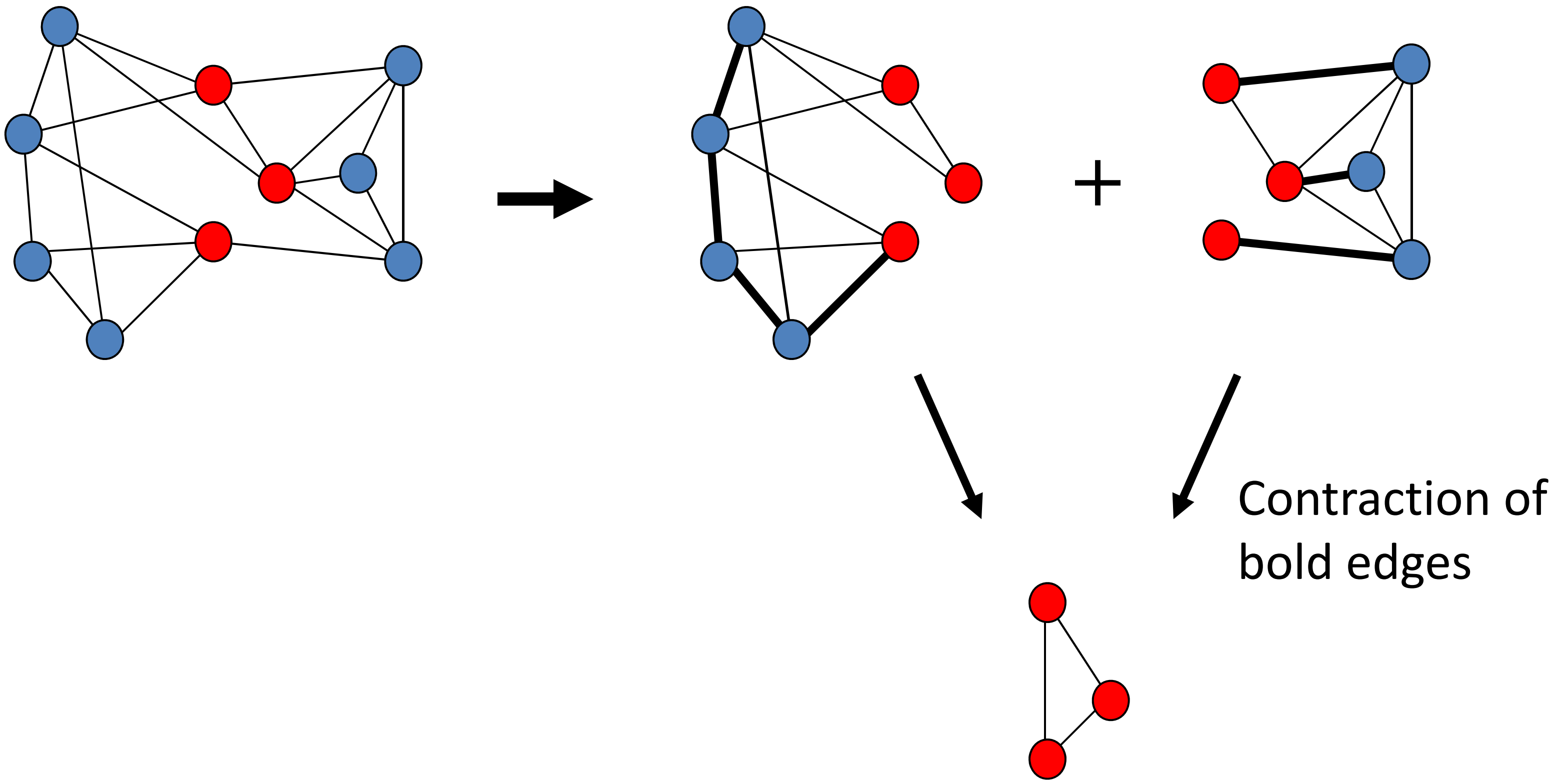
Lemma 1

Let S be a separator in G. The treewidth of G is at most the maximum over all connected components W of G of the treewidth of $G[W \cup S]$ + clique(S)



Lemma 2

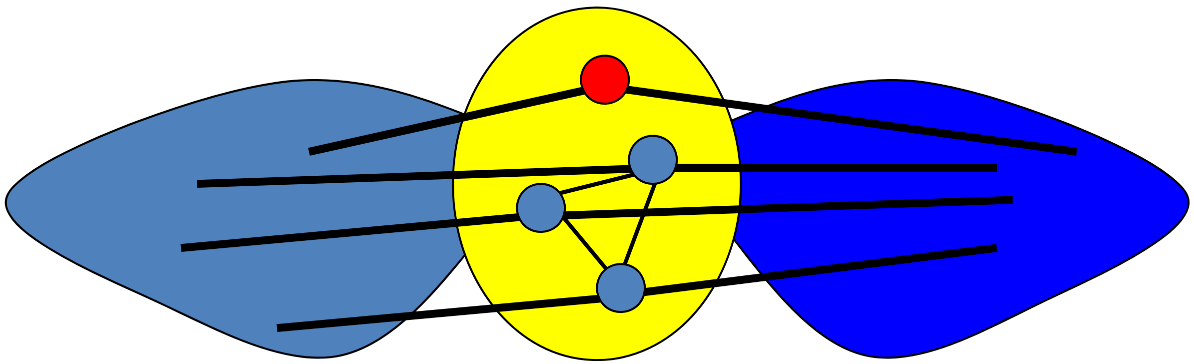
Let S be a separator. If for all components W of $G - S$, $G[W \cup S]$ contains a clique on S as a minor, then S is **safe**.



→ Clique separators are safe

→ Separators of size 0 and 1 are safe

Safeness of minimal almost clique separators



S is *almost clique* when *S* -*v* is a clique for some vertex *v*

- If one component is contracted to the red vertex, the separator turns into a clique: minimal almost clique separators are safe!

→ Minimal Separators of size 2 are safe

→ `Almost all’ minimal separators of size 3 are safe

  - only 3 independent vertices can be non-safe

- Minimal separators of size 3 that split off at least two vertices are safe

Why Lower Bounds?

- Benchmark quality of constructed tree decompositions (upper bounds)
- Speed up of branch & bound methods (e.g. Gogate & Dechter [63])
- Indicates expected performance of dynamic programming algorithms

Theorem *Let $V_1, V_2 \subseteq V$ induce a complete bipartite subgraph. Then $tw(G) \geq \min\{|V_1|, |V_2|\}$*

Induced subgraphs

Theorem *The treewidth of a graph can not increase by taking subgraphs*

H subgraph of *G*

$$\left.\begin{aligned} tw(H) &\leq tw(G) \\ LB(G) &\leq tw(G) \end{aligned}\right\} \quad LB(H) \leq tw(G)$$

Corollary *If the LB can increase by taking subgraphs, an improved lower bound can be found by taking the maximum over all subgraphs:*

$$\max_{H \subseteq G} LB(H) \leq tw(G)$$

Foundations II

Theorem *The treewidth of a graph can not increase by taking minors*

H minor of *G*

$$\left.\begin{array}{l} tw(H) \leq tw(G) \\ LB(G) \leq tw(G) \end{array}\right\} \quad LB(H) \leq tw(G)$$

Corollary *If the LB can increase by taking minors, an improved lower bound can be found by taking the maximum over all minors:*

$$\max_{H \prec G} LB(H) \leq tw(G)$$

Degree-Based Lower Bounds I

Lemma *The minimum degree of a graph is a lower bound for treewidth*

$$\delta(G) \leq tw(G)$$

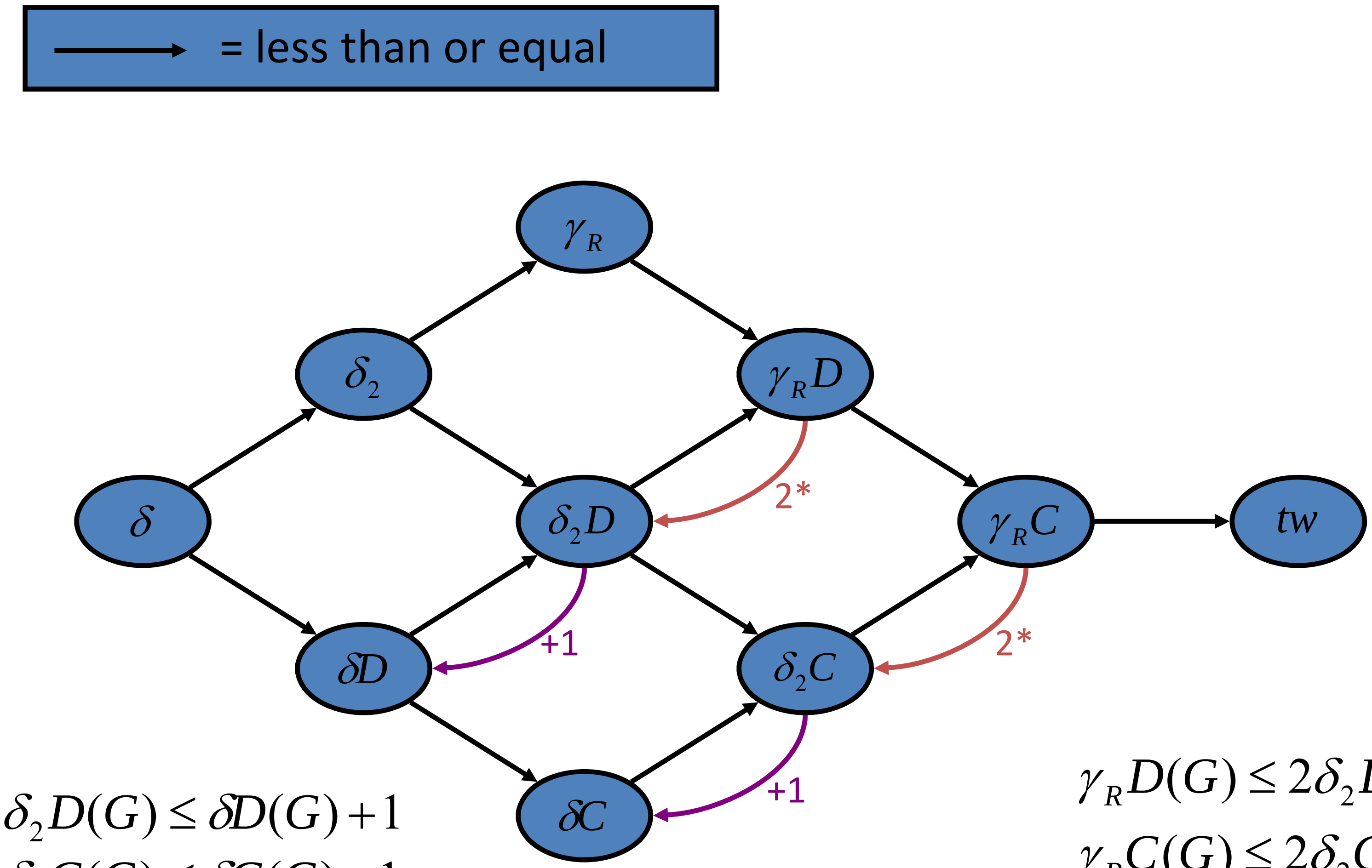
Corollary *The degeneracy of a graph is a lower bound for treewidth*

$$\delta D(G) = \max_{H \subseteq G} \delta(H) \leq tw(G)$$

Corollary *The contraction degeneracy of a graph is a lower bound for treewidth*

$$\delta C(G) = \max_{H \prec G} \delta(H) \leq tw(G)$$

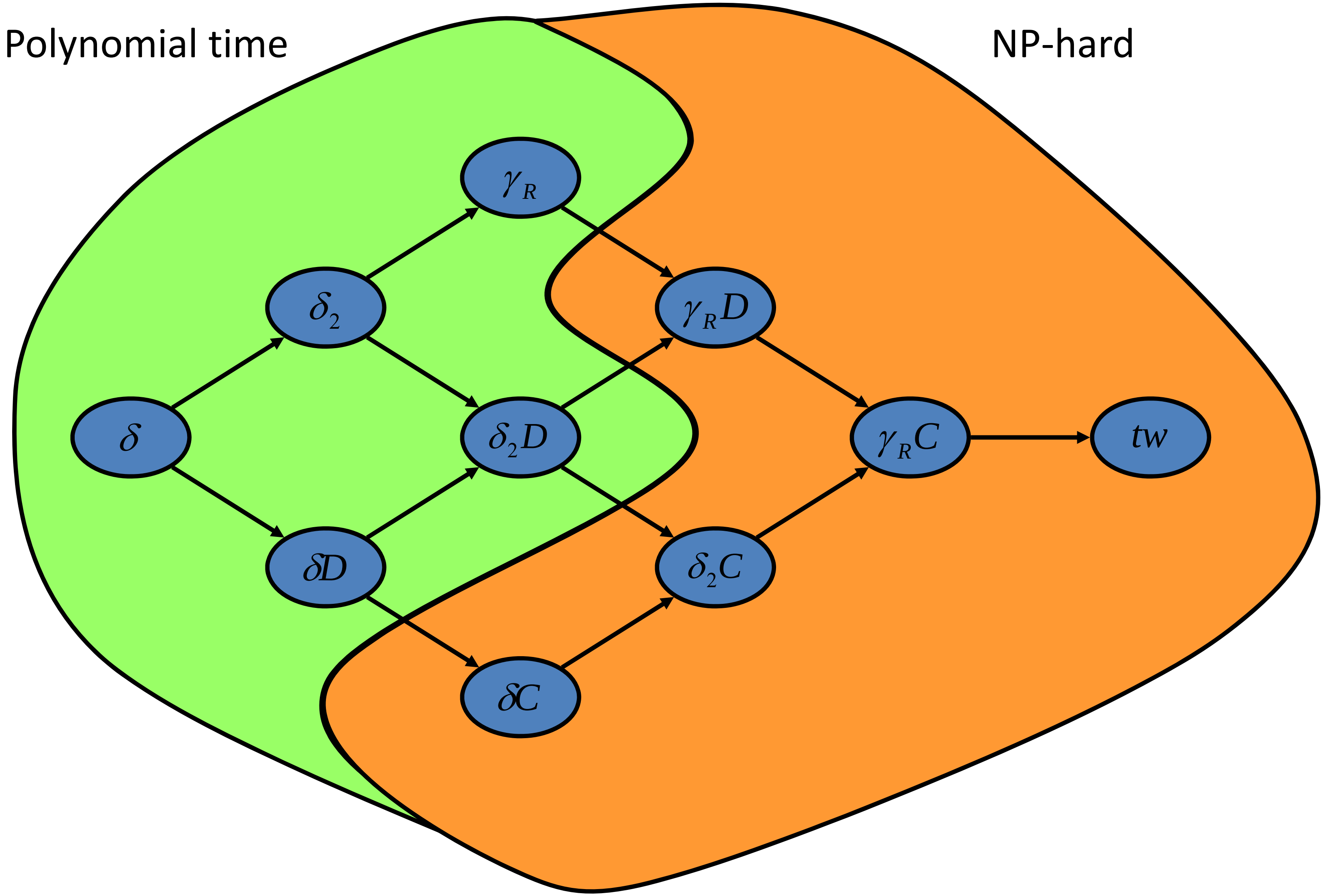
Relationships



$$\delta_2 D(G) \leq \delta D(G) + 1$$
$$\delta_2 C(G) \leq \delta C(G) + 1$$

$$\gamma_R D(G) \leq 2\delta_2 D(G)$$
$$\gamma_R C(G) \leq 2\delta_2 C(G)$$

Complexity


Polynomial time
NP-hard
γ_R
δ_2
$\gamma_R D$
δ
$\delta_2 D$
$\gamma_R C$
tw
δD
$\delta_2 C$
δC

More lower bounds

Theorem *Let $G=(V,E)$ and $tw(G) \leq k$. Then, $|E| \leq k|V| - \frac{1}{2}k(k+1)$*

Theorem *$tw(G) \geq n - \frac{1}{2} - \sqrt{(n^2-n-m+1)}$ with $n=|V|$ and $m=|E|$*

Planar Graphs

Theorem *Planarity is closed under taking minors*

G planar, *H* minor of *G*

$$\left.\begin{aligned} \delta C(G) &\le tw(G) \\ \delta(H) &\le 5 \end{aligned}\right\} \quad \delta C(G) \le 5$$

Theorem *The genus of G cannot increase by taking minors*

G graph of genus k, *H* minor of *G*

$$\left.\begin{aligned} \delta C(G) &\le tw(G) \\ \delta(H) &\le 5+k \end{aligned}\right\} \quad \delta C(G) \le 5+k$$

➡ Alternative lower bound by **Brambles** [36, ESA2005]