Restricted Types of Matchings

Dieter Rautenbach

Universität Ulm

Restricted Types of Matchings

Dieter Rautenbach

Universität Ulm

Joint with Baste, Fürst, Leichter, Lima, Sau, Souza, Szwarcfiter

Matchings in Graphs

Matchings in Graphs

Definition

A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.

Matchings in Graphs

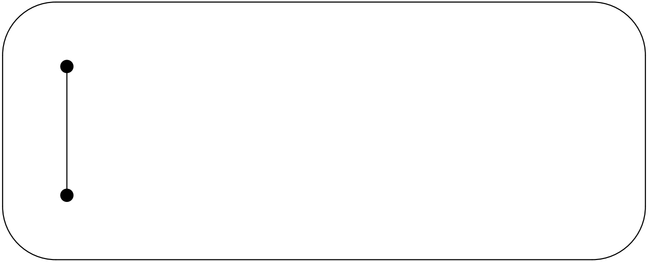
Definition

A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.

Matchings in Graphs

Definition

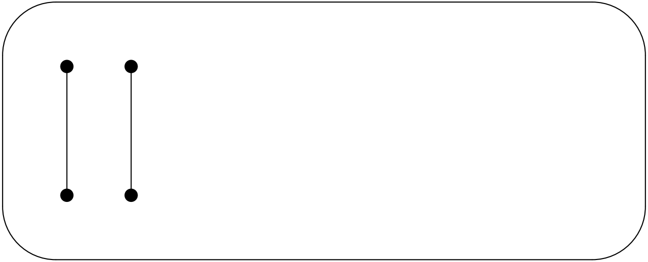
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

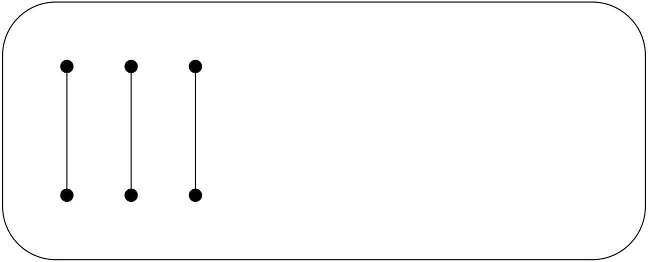
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

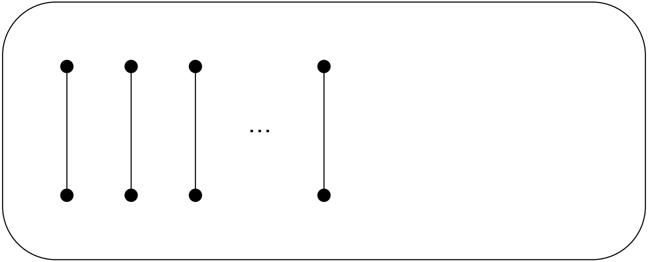
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

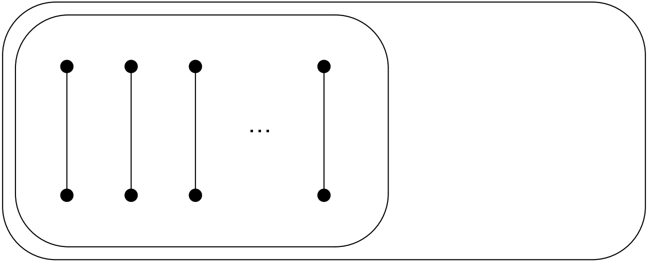
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

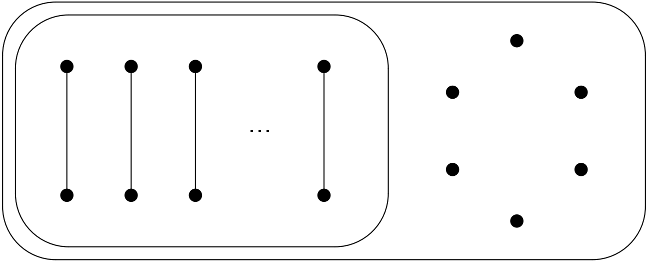
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

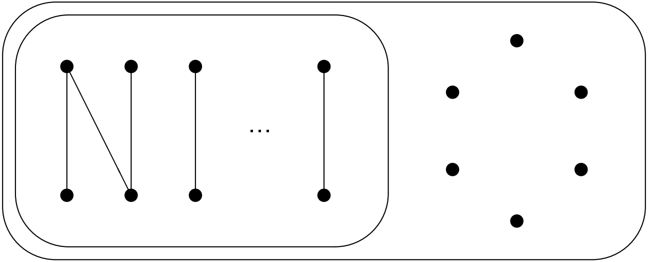
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

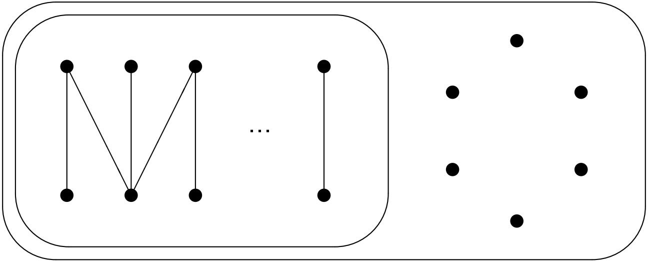
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

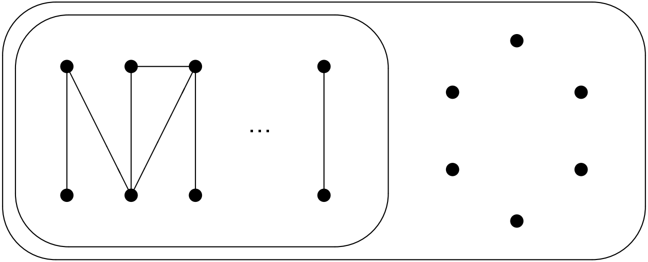
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

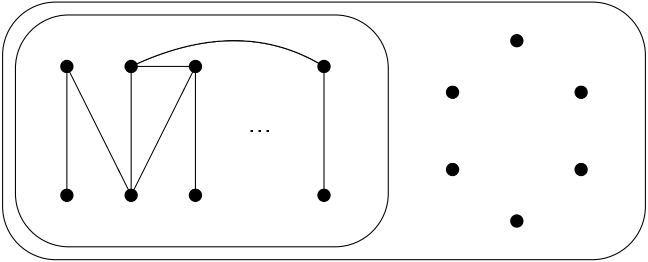
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

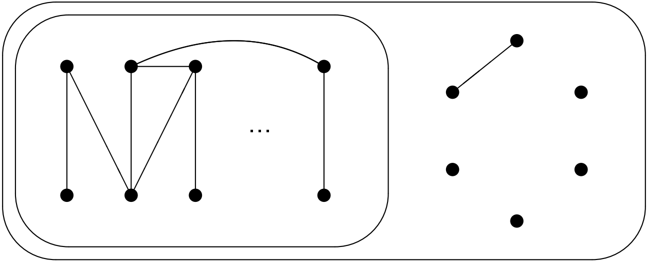
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

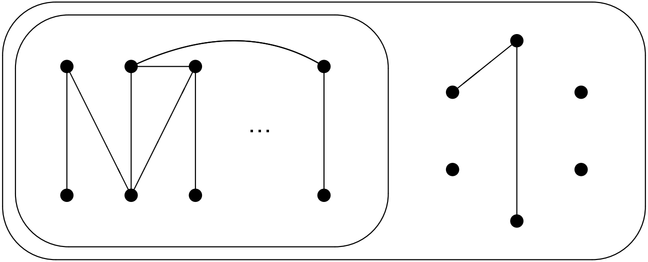
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

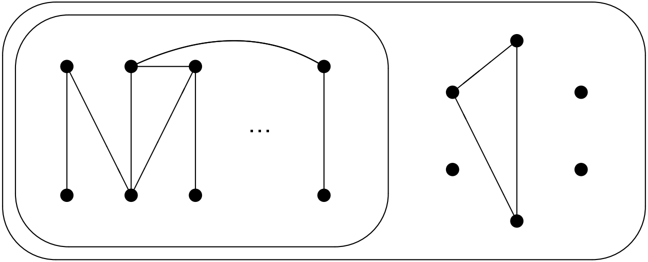
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

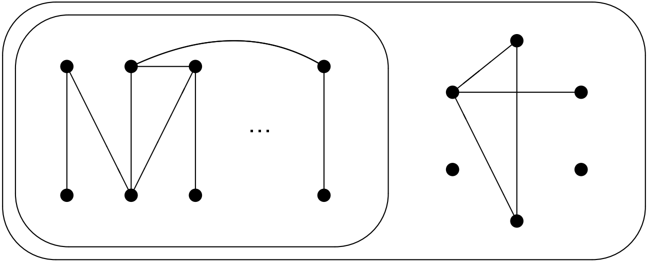
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

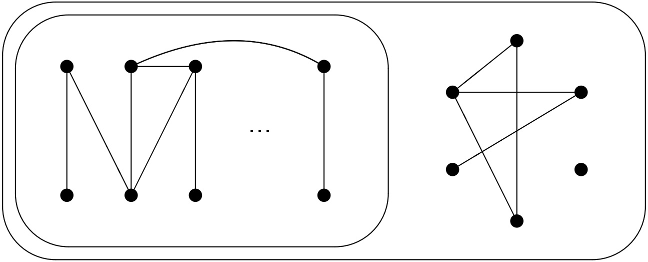
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

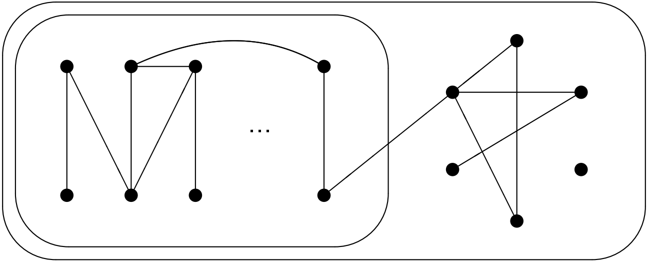
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

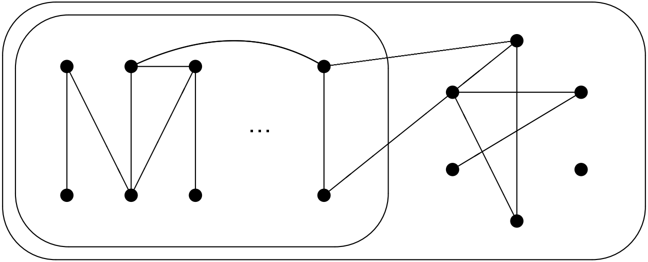
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

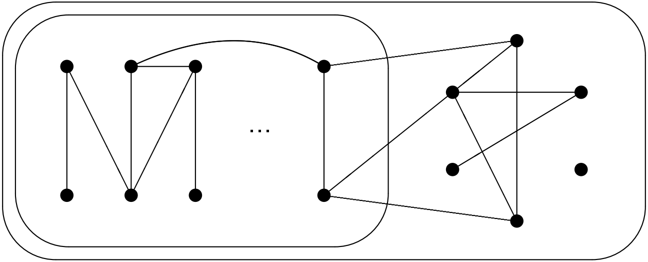
A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

A matching in a graph G is a set M of pairwise disjoint edges. A matching M in G is perfect if the set $V(M)$ of vertices of G incident with an edge in M equals $V(G)$. The matching number $\nu(G)$ of G is the maximum order of a matching in G.



Matchings in Graphs

Definition

An edge coloring of a graph G is a partition of its edge set $E(G)$ into matchings. The chromatic index $\chi'(G)$ of G is the minimum number of matchings/colors needed for such a partition.

$$\nu(G)\chi'(G) \geq m(G)$$

Matchings in Graphs

- Petersen 1891
  - cubic bridgeless graphs have a perfect matching
- Kőnig and Egerváry 1930ies
  - $\nu(G) \overset{G\ bip.}{=} \tau(G) \rightsquigarrow$ LP duality, tum matrices, integral polyhedra
  - Hungarian method $\rightsquigarrow$ Ford-Fulkerson 1956
- Tutte 1947
  - $2\nu(G) = n(G) \Leftrightarrow \forall S \subseteq V(G) : q(S) \leq |S| \rightsquigarrow$ good characterization
- Vizing 1964
  - $\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1$
- Edmonds 1965
  - maximum matching algorithm
  - Gallai-Edmonds structure theorem
  - ...
- Holyer 1981
- ...

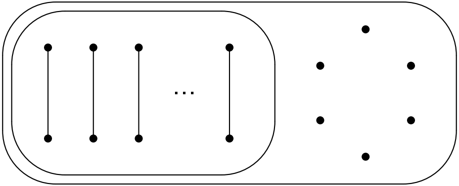
Restricted Types of Matchings

Restricted Types of Matchings

A matching M in a graph G is strong/induced if $G[V(M)]$ is 1-regular.

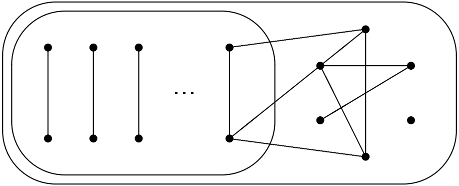
Restricted Types of Matchings

A matching M in a graph G is strong/induced if $G[V(M)]$ is 1-regular.



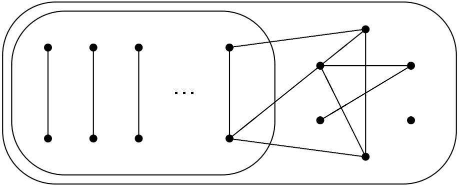
Restricted Types of Matchings

A matching M in a graph G is strong/induced if $G[V(M)]$ is 1-regular.



Restricted Types of Matchings

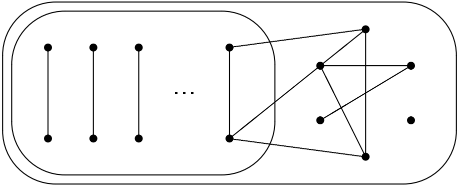
A matching M in a graph G is strong/induced if $G[V(M)]$ is 1-regular.



- strong/induced matching number $\nu_s(G)$
  Stockmeyer and Vazirani 1982, Cameron 1989
  Faudree, Gyárfás, Schelp, and Tuza 1989

Restricted Types of Matchings

A matching M in a graph G is strong/induced if $G[V(M)]$ is 1-regular.



- strong/induced matching number $\nu_s(G)$
  Stockmeyer and Vazirani 1982, Cameron 1989
  Faudree, Gyárfás, Schelp, and Tuza 1989
- strong/induced chromatic index $\chi'_s(G)$
  Erdős and Nešetřil 1985: $\chi'_s(G) \overset{?!}{\leq} \frac{5}{4}\Delta(G)^2$
  Molloy and Reed 1997: $\chi'_s(G) \leq \Big(1.998 + o(\Delta(G))\Big)\Delta(G)^2$

Restricted Types of Matchings

M is uniquely restricted if $V(M) = V(M') \Rightarrow M = M'$.

Restricted Types of Matchings

M is uniquely restricted if $V(M) = V(M') \Rightarrow M = M'$.

$$(a_{i,j}) \to (a_{\pi(i),\sigma(j)}) = \begin{pmatrix} 1 & * & * & \dots \\ 0 & 1 & * & \dots \\ 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

Restricted Types of Matchings

M is uniquely restricted if $V(M) = V(M') \Rightarrow M = M'$.

$$(a_{i,j}) \to (a_{\pi(i),\sigma(j)}) = \begin{pmatrix} 1 & * & * & \dots \\ 0 & 1 & * & \dots \\ 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

- uniquely restricted matching number $\nu_{ur}(G)$
  (Hershkowitz and Schneider 1993)
  Golumbic, Hirst, and Lewenstein 2001

Restricted Types of Matchings

M is uniquely restricted if $V(M) = V(M') \Rightarrow M = M'$.

$$(a_{i,j}) \to (a_{\pi(i),\sigma(j)}) = \begin{pmatrix} 1 & * & * & \dots \\ 0 & 1 & * & \dots \\ 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

- uniquely restricted matching number $\nu_{ur}(G)$
  (Hershkowitz and Schneider 1993)
  Golumbic, Hirst, and Lewenstein 2001
  Levit and Mandrescu 2003
  Mishra 2011
  Francis, Jacob, and Jana 2016

Restricted Types of Matchings

M is uniquely restricted if $V(M) = V(M') \Rightarrow M = M'$.

$$(a_{i,j}) \to (a_{\pi(i),\sigma(j)}) = \begin{pmatrix} 1 & * & * & \dots \\ 0 & 1 & * & \dots \\ 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

- uniquely restricted matching number $\nu_{ur}(G)$
  (Hershkowitz and Schneider 1993)
  Golumbic, Hirst, and Lewenstein 2001
  Levit and Mandrescu 2003
  Mishra 2011
  Francis, Jacob, and Jana 2016
- uniquely restricted chromatic index $\chi'_{ur}(G)$
  Baste, R, and Sau 2016

Restricted Types of Matchings

M is acyclic if $G[V(M)]$ is a forest.

Restricted Types of Matchings

M is acyclic if $G[V(M)]$ is a forest.

- acyclic matching number $\nu_1(G)$
  Goddard, Hedetniemi, Hedetniemi, and Laskar 2005

Restricted Types of Matchings

M is acyclic if $G[V(M)]$ is a forest.

- acyclic matching number $\nu_1(G)$
  Goddard, Hedetniemi, Hedetniemi, and Laskar 2005

M is r-degenerate if $G[V(M)]$ is r-degenerate, that is,

$$H \subseteq G[V(M)] \text{ and } n(H) > 0 \Rightarrow \delta(H) \leq r.$$

Restricted Types of Matchings

M is acyclic if $G[V(M)]$ is a forest.

- acyclic matching number $\nu_1(G)$
  Goddard, Hedetniemi, Hedetniemi, and Laskar 2005

M is r-degenerate if $G[V(M)]$ is r-degenerate, that is,

$$H \subseteq G[V(M)] \text{ and } n(H) > 0 \Rightarrow \delta(H) \leq r.$$

- r-degenerate matching number $\nu_r(G)$
  r-degenerate chromatic index $\chi'_r(G)$
  Baste and R 2017

Restricted Types of Matchings

induced $\Rightarrow$ acyclic $\Rightarrow$ uniquely restricted

Restricted Types of Matchings

induced $\Rightarrow$ acyclic $\Rightarrow$ uniquely restricted

$$\nu(G) \geq \nu_{ur}(G) \geq \nu_1(G) \geq \nu_s(G)$$

Restricted Types of Matchings

induced $\Rightarrow$ acyclic $\Rightarrow$ uniquely restricted

$$\nu(G) \geq \nu_{ur}(G) \geq \nu_1(G) \geq \nu_s(G)$$

$$\chi'(G) \leq \chi'_{ur}(G) \leq \chi'_1(G) \leq \chi'_s(G).$$

Restricted Types of Matchings

induced $\Rightarrow$ acyclic $\Rightarrow$ uniquely restricted

$$\nu(G) \geq \nu_{ur}(G) \geq \nu_1(G) \geq \nu_s(G)$$

$$\chi'(G) \leq \chi'_{ur}(G) \leq \chi'_1(G) \leq \chi'_s(G).$$

r-degenerate $\Rightarrow$ $(r+1)$-degenerate

Restricted Types of Matchings

induced $\Rightarrow$ acyclic $\Rightarrow$ uniquely restricted

$$\nu(G) \geq \nu_{ur}(G) \geq \nu_1(G) \geq \nu_s(G)$$

$$\chi'(G) \leq \chi'_{ur}(G) \leq \chi'_1(G) \leq \chi'_s(G).$$

r-degenerate $\Rightarrow$ $(r+1)$-degenerate

$$\nu_1(G) \leq \nu_2(G) \leq \nu_3(G) \leq \ldots \leq \nu_{\Delta(G)}(G) = \nu(G)$$

Restricted Types of Matchings

Given a graph G.

Restricted Types of Matchings

Given a graph G.

Does G have a matching M such that $G - M$ is a forest?

Restricted Types of Matchings

Given a graph G.

Does G have a matching M such that $G - M$ is a forest?

Does G have a matching M such that $G - M$ is bipartite?

Today's menu

Today's menu

Fürst, Leichter, and R 2017

- Locally searching for large induced matchings

Today's menu

Fürst, Leichter, and R 2017

- Locally searching for large induced matchings

Baste, R, and Sau 2016

- Approximation algorithm for $\nu_{ur}(G)$ in bipartite graphs
- Upper bounds on $\chi'_{ur}(G)$

Today's menu

Fürst, Leichter, and R 2017

- Locally searching for large induced matchings

Baste, R, and Sau 2016

- Approximation algorithm for $\nu_{ur}(G)$ in bipartite graphs
- Upper bounds on $\chi'_{ur}(G)$

Baste and R 2017

- Upper bounds on $\chi'_{r}(G)$
- Efficient algorithm for $\nu_{r}(G)$ in chordal graphs

Today's menu

Fürst, Leichter, and R 2017

- Locally searching for large induced matchings

Baste, R, and Sau 2016

- Approximation algorithm for $\nu_{ur}(G)$ in bipartite graphs
- Upper bounds on $\chi'_{ur}(G)$

Baste and R 2017

- Upper bounds on $\chi'_{r}(G)$
- Efficient algorithm for $\nu_{r}(G)$ in chordal graphs

Lima, R, Souza, and Szwarcfiter 2016

- Decycling with a matching
- Bipartizing with a matching

Approximating $\nu_s(G)$

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem is APX-hard for d-regular graphs for every $d \geq 3$.

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem is APX-hard for d-regular graphs for every $d \geq 3$.

Theorem (Dabrowski, Demange, and Lozin 2013)

...this remains true even restricted to bipartite graphs...

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem is APX-hard for d-regular graphs for every $d \geq 3$.

Theorem (Dabrowski, Demange, and Lozin 2013)

...this remains true even restricted to bipartite graphs...

Theorem (Kobler and Rotics 2003)

The maximum induced matching problem is NP-hard for line graphs.

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem restricted to d-regular graphs has an $(d + O(1))$-factor approximation algorithm.

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem restricted to d-regular graphs has an $(d + O(1))$-factor approximation algorithm.

If M is an induced matching of a d-regular graph G, then

$$|M| \leq \frac{m(G)}{2d-1},$$

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem restricted to d-regular graphs has an $(d + O(1))$-factor approximation algorithm.

If M is an induced matching of a d-regular graph G, then

$$|M| \leq \frac{m(G)}{2d-1},$$

and, if M is maximal,

$$|M| \geq \frac{m(G)}{2d^2-2d+1}.$$

Approximating $\nu_s(G)$

Theorem (Duckworth, Manlove, and Zito 2005)

The maximum induced matching problem restricted to d-regular graphs has an $(d + O(1))$-factor approximation algorithm.

If M is an induced matching of a d-regular graph G, then

$$|M| \leq \frac{m(G)}{2d-1},$$

and, if M is maximal,

$$|M| \geq \frac{m(G)}{2d^2-2d+1}.$$

$$\frac{2d^2-2d+1}{2d-1} = d - \frac{1}{2} + \frac{1}{4d-2}$$

Approximating $\nu_s(G)$

Theorem (Gotthilf and Lewenstein 2006)

The maximum induced matching problem restricted to d-regular graphs has an $(0.75d + 0.15)$*-factor approximation algorithm.*

Approximating $\nu_s(G)$

Theorem (Gotthilf and Lewenstein 2006)

The maximum induced matching problem restricted to d-regular graphs has an $(0.75d + 0.15)$-factor approximation algorithm.

Theorem (R 2016)

...for $\{C_3, C_5\}$-free graphs the factor can be improved to $0.708\bar{3}d + 0.425$.

Approximating $\nu_s(G)$

Definition (Conflict edges)

For an edge e of a graph G, let

$$C_G(e) = \{e\} \cup N_{L(G)^2}(e) = \{f \in E(G) : \mathrm{dist}_{L(G)}(e, f) \leq 2\},$$

and let $c_G(e) = |C_G(e)|$.

Approximating $\nu_s(G)$

Definition (Conflict edges)

For an edge e of a graph G, let

$$C_G(e) = \{e\} \cup N_{L(G)^2}(e) = \{f \in E(G) : \mathrm{dist}_{L(G)}(e, f) \leq 2\},$$

and let $c_G(e) = |C_G(e)|$.

Definition (Private conflict edges)

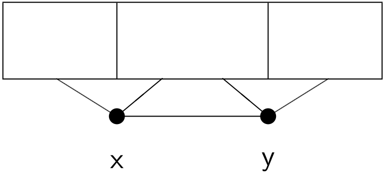
For a set M of edges of G and an edge e in M, let

$$PC_G(M, e) = C_G(e) \setminus \bigcup_{f \in M \setminus \{e\}} C_G(f),$$

and let $pc_G(M, e) = |PC_G(M, e)|$.

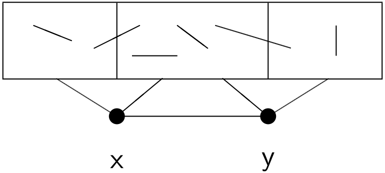
Approximating $\nu_s(G)$

If $n_{xy} = |N_G(x) \cap N_G(y)|$ and $m_{xy} = m_G((N_G(x) \cup N_G(y)) \setminus \{x, y\})$ for an edge xy of G, then



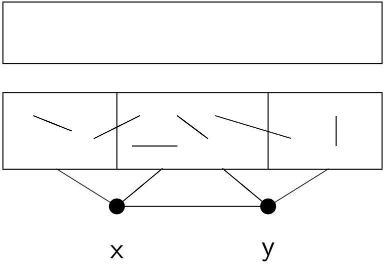
Approximating $\nu_s(G)$

If $n_{xy} = |N_G(x) \cap N_G(y)|$ and $m_{xy} = m_G((N_G(x) \cup N_G(y)) \setminus \{x, y\})$ for an edge xy of G, then



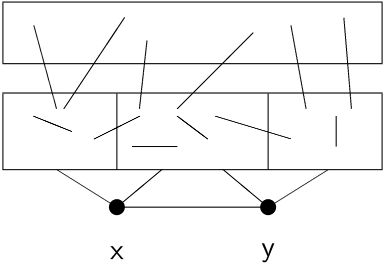
Approximating $\nu_s(G)$

If $n_{xy} = |N_G(x) \cap N_G(y)|$ and $m_{xy} = m_G((N_G(x) \cup N_G(y)) \setminus \{x, y\})$ for an edge xy of G, then



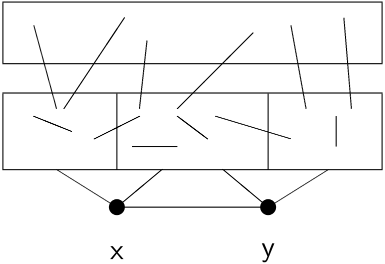
Approximating $\nu_s(G)$

If $n_{xy} = |N_G(x) \cap N_G(y)|$ and $m_{xy} = m_G((N_G(x) \cup N_G(y)) \setminus \{x, y\})$ for an edge xy of G, then



Approximating $\nu_s(G)$

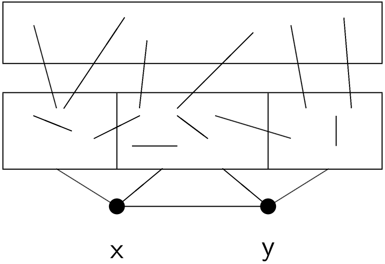
If $n_{xy} = |N_G(x) \cap N_G(y)|$ and $m_{xy} = m_G((N_G(x) \cup N_G(y)) \setminus \{x, y\})$ for an edge xy of G, then



$$c_G(xy) \quad \leq \quad |\{xy\}| + d\Big|(N_G(x) \cup N_G(y)) \setminus \{x, y\}\Big| - m_{xy}$$

Approximating $\nu_s(G)$

If $n_{xy} = |N_G(x) \cap N_G(y)|$ and $m_{xy} = m_G((N_G(x) \cup N_G(y)) \setminus \{x, y\})$ for an edge xy of G, then



$$\begin{aligned} c_G(xy) &\le |\{xy\}| + d\Big|(N_G(x) \cup N_G(y)) \setminus \{x, y\}\Big| - m_{xy} \\ &= 2d^2 - 2d + 1 - \Big(dn_{xy} + m_{xy}\Big). \end{aligned}$$

Approximating $\nu_s(G)$

$\textsc{Greedy}(\alpha)$
Input: A graph G.
Output: A pair (M, G') such that M is an induced matching of G, and G' is a subgraph of G.

Approximating $\nu_s(G)$

$\textsc{Greedy}(\alpha)$
Input: A graph G.
Output: A pair (M, G') such that M is an induced matching of G, and G' is a subgraph of G.

$M \leftarrow \emptyset$;
$G_0 \leftarrow G$;
$i \leftarrow 1$;

Approximating $\nu_s(G)$

$\textsc{Greedy}(\alpha)$
Input: A graph G.
Output: A pair (M, G') such that M is an induced matching of G, and G' is a subgraph of G.

$M \leftarrow \emptyset$;
$G_0 \leftarrow G$;
$i \leftarrow 1$;
while $\min\{c_{G_{i-1}}(e) : e \in E(G_{i-1})\} \leq \alpha$ **do**

Approximating $\nu_s(G)$

$\textsc{Greedy}(\alpha)$
Input: A graph G.
Output: A pair (M, G') such that M is an induced matching of G, and G' is a subgraph of G.

$M \leftarrow \emptyset$;
$G_0 \leftarrow G$;
$i \leftarrow 1$;
while $\min\{c_{G_{i-1}}(e) : e \in E(G_{i-1})\} \leq \alpha$ **do**
  Choose an edge e_i of G_{i-1} with $c_{G_{i-1}}(e_i) \leq \alpha$;
  $M \leftarrow M \cup \{e_i\}$;
  $G_i \leftarrow G_{i-1} - C_{G_{i-1}}(e_i)$;
  $i \leftarrow i + 1$;
end

Approximating $\nu_s(G)$

$\textsc{Greedy}(\alpha)$
Input: A graph G.
Output: A pair (M, G') such that M is an induced matching of G, and G' is a subgraph of G.

$M \leftarrow \emptyset$;
$G_0 \leftarrow G$;
$i \leftarrow 1$;
while $\min\{c_{G_{i-1}}(e) : e \in E(G_{i-1})\} \leq \alpha$ **do**
  Choose an edge e_i of G_{i-1} with $c_{G_{i-1}}(e_i) \leq \alpha$;
  $M \leftarrow M \cup \{e_i\}$;
  $G_i \leftarrow G_{i-1} - C_{G_{i-1}}(e_i)$;
  $i \leftarrow i + 1$;
end
return (M, G_{i-1});

Approximating $\nu_s(G)$

Local Search

Input: A graph G.

Output: An induced matching M of G.

$M \leftarrow \emptyset$;

repeat

  if *$M \cup \{e\}$ is an induced matching of G for some edge $e \in E(G) \setminus M$* **then**

    $M \leftarrow M \cup \{e\}$;

  end

Approximating $\nu_s(G)$

LOCAL SEARCH
Input: A graph G.
Output: An induced matching M of G.

$M \leftarrow \emptyset$;
repeat
    if $M \cup \{e\}$ *is an induced matching of* G *for some edge* $e \in E(G) \setminus M$ **then**
        $M \leftarrow M \cup \{e\}$;
    end
    if $(M \setminus \{e\}) \cup \{e', e''\}$ *is an induced matching of* G *for some three distinct edges* $e \in M$ *and* $e', e'' \in E(G) \setminus M$ **then**
        $M \leftarrow (M \setminus \{e\}) \cup \{e', e''\}$;
    end

Approximating $\nu_s(G)$

LOCAL SEARCH
Input: A graph G.
Output: An induced matching M of G.

$M \leftarrow \emptyset$;
repeat
  if $M \cup \{e\}$ *is an induced matching of* G *for some edge* $e \in E(G) \setminus M$ **then**
    $M \leftarrow M \cup \{e\}$;
  end
  if $(M \setminus \{e\}) \cup \{e', e''\}$ *is an induced matching of* G *for some three distinct edges* $e \in M$ *and* $e', e'' \in E(G) \setminus M$ **then**
    $M \leftarrow (M \setminus \{e\}) \cup \{e', e''\}$;
  end
until $|M|$ *does not increase during one iteration*;
return M;

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G,

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left| \left\{ (e, f) : e \in M \text{ and } f \in C_G(e) \right\} \right|,$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$p$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$p \leq$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$p \leq \sum_{e \in M} c_G(e)$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$\leq p \leq \sum_{e \in M} c_G(e)$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$m(G) \qquad\qquad\qquad \leq p \leq \sum_{e \in M} c_G(e)$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$2m(G) \qquad\qquad\qquad \leq p \leq \sum_{e \in M} c_G(e)$$

Approximating $\nu_s(G)$

If (M, G') is the output of $\textsc{Greedy}(\alpha)$ applied to G, then

$$c_{G'}(e) > \alpha \text{ for every edge } e \text{ of } G'.$$

If M is the output of $\textsc{Local Search}$ applied to G, and

$$p = \left|\left\{(e, f) : e \in M \text{ and } f \in C_G(e)\right\}\right|,$$

then

$$2m(G) - \sum_{e \in M} pc_G(M, e) \leq p \leq \sum_{e \in M} c_G(e)$$

Approximating $\nu_s(G)$

Theorem (Fürst, Leichter, and R 2017)

Local Search is an approximation algorithm for the maximum induced matching problem that has approximation factor

- $\frac{9}{16}d + \frac{33}{80}$ *for d-regular C_4-free graphs,*
- $\frac{1}{2}d + \frac{1}{4} + \frac{1}{8d-4}$ *for d-regular $\{C_3, C_4\}$-free graphs,*
- $\frac{3}{4}d - \frac{1}{8} + \frac{3}{16d-8}$ *d-regular C_5-free graphs,*

Approximating $\nu_s(G)$

Theorem (Fürst, Leichter, and R 2017)

Local Search *is an approximation algorithm for the maximum induced matching problem that has approximation factor*

- $\frac{9}{16}d + \frac{33}{80}$ *for* d*-regular* C_4*-free graphs,*
- $\frac{1}{2}d + \frac{1}{4} + \frac{1}{8d-4}$ *for* d*-regular* $\{C_3, C_4\}$*-free graphs,*
- $\frac{3}{4}d - \frac{1}{8} + \frac{3}{16d-8}$ d*-regular* C_5*-free graphs, and*
- $\frac{1}{2}d + \frac{3}{4} - \frac{1}{8d-4}$ *for* d*-regular claw-free graphs.*

Approximating $\nu_s(G)$

Theorem (Fürst, Leichter, and R 2017)

Local Search *is an approximation algorithm for the maximum induced matching problem that has approximation factor*

- $\frac{9}{16}d + \frac{33}{80}$ *for d-regular C_4-free graphs,*
- $\frac{1}{2}d + \frac{1}{4} + \frac{1}{8d-4}$ *for d-regular $\{C_3, C_4\}$-free graphs,*
- $\frac{3}{4}d - \frac{1}{8} + \frac{3}{16d-8}$ *d-regular C_5-free graphs, and*
- $\frac{1}{2}d + \frac{3}{4} - \frac{1}{8d-4}$ *for d-regular claw-free graphs.*

Theorem (Fürst, Leichter, and R 2017)

For claw-free d-regular graphs, choosing any maximal induced matching yields a $\left(\frac{7}{12}d + \frac{31}{24} + \frac{55}{48d-24}\right)$-factor approximation algorithm.

Approximating $\nu_s(G)$

Approximating $\nu_{ur}(G)$

Approximating $\nu_{ur}(G)$ in bipartite graphs

Approximating $\nu_{ur}(G)$ in bipartite graphs

Theorem (Golumbic, Hirst, and Lewenstein 2001)

$\nu_{ur}(G)$ is hard for bipartite graphs and split graphs.

Approximating $\nu_{ur}(G)$ in bipartite graphs

Theorem (Golumbic, Hirst, and Lewenstein 2001)

$\nu_{ur}(G)$ is hard for bipartite graphs and split graphs.

Theorem (Mishra 2011)

$\nu_{ur}(G)$ is $\mathtt{APX}$-complete for subcubic bipartite graphs but can be approximated within a factor of 2 for cubic bipartite graphs.

Approximating $\nu_{ur}(G)$ in bipartite graphs

Theorem (Baste, R, and Sau 2016)

For a given connected subcubic bipartite graph G, one can find in polynomial time a uniquely restricted matching of G of size at least

$$\frac{5}{9}\nu_{ur}(G).$$

Approximating $\nu_{ur}(G)$ in bipartite graphs

Theorem (Baste, R, and Sau 2016)

For a given connected subcubic bipartite graph G, one can find in polynomial time a uniquely restricted matching of G of size at least

$$\frac{5}{9}\nu_{ur}(G).$$

Theorem (Baste, R, and Sau 2016)

Let $\Delta \geq 3$ be an integer.
For a given connected C_4-free bipartite graph G of maximum degree at most Δ, one can find in polynomial time a uniquely restricted matching M of G of size at least

$$\frac{(\Delta-1)^2+(\Delta-2)}{(\Delta-1)^3+(\Delta-2)}\nu_{ur}(G).$$

Approximating $\nu_{ur}(G)$ in bipartite graphs

Lemma (Baste, R, and Sau 2016)

Let $\Delta \geq 3$ be an integer.
If G is a connected C_4-free bipartite graph of maximum degree at most Δ with partite sets A and B

Approximating $\nu_{ur}(G)$ in bipartite graphs

Lemma (Baste, R, and Sau 2016)

Let $\Delta \geq 3$ be an integer.
If G is a connected C_4-free bipartite graph of maximum degree at most Δ with partite sets A and B such that

- *every vertex in A has degree at least 2, and*
- *some vertex in B has degree less than Δ,*

Approximating $\nu_{ur}(G)$ in bipartite graphs

Lemma (Baste, R, and Sau 2016)

Let $\Delta \geq 3$ be an integer.
If G is a connected C_4-free bipartite graph of maximum degree at most Δ with partite sets A and B such that

- *every vertex in A has degree at least 2, and*
- *some vertex in B has degree less than Δ,*

then G has a uniquely restricted matching M of size at least

$$\frac{(\Delta-1)^2+(\Delta-2)}{(\Delta-1)^3+(\Delta-2)}|A|.$$

Furthermore, such a matching can be found in polynomial time.

Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof:

Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...

Approximating $\nu_{ur}(G)$ in bipartite graphs

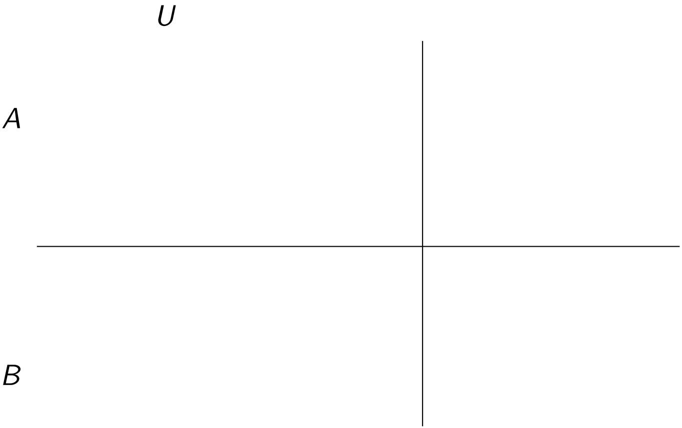
Proof: Given G, we maintain pairs (U, M) such that...

A

B

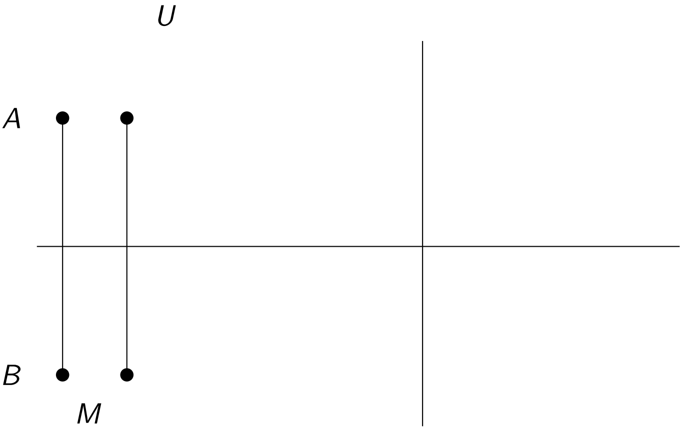
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



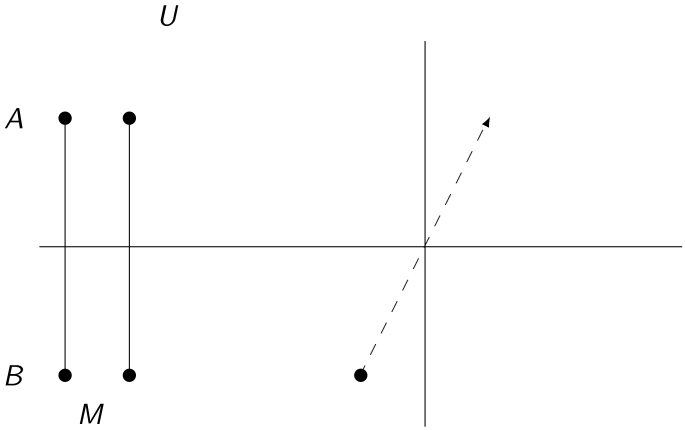
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



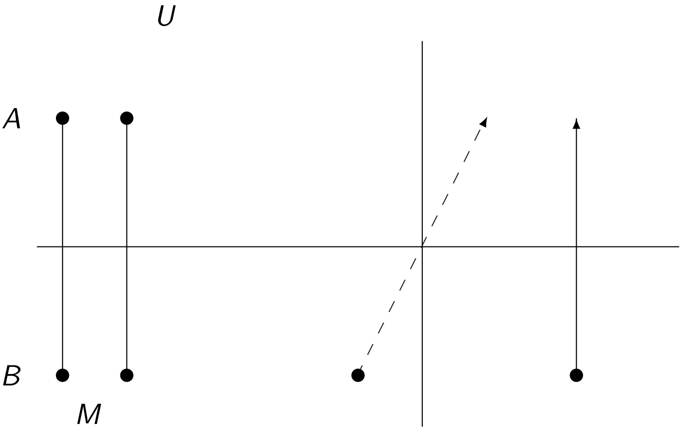
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



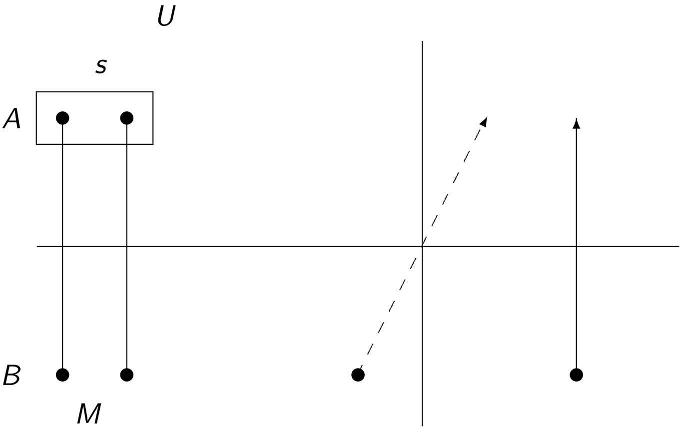
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



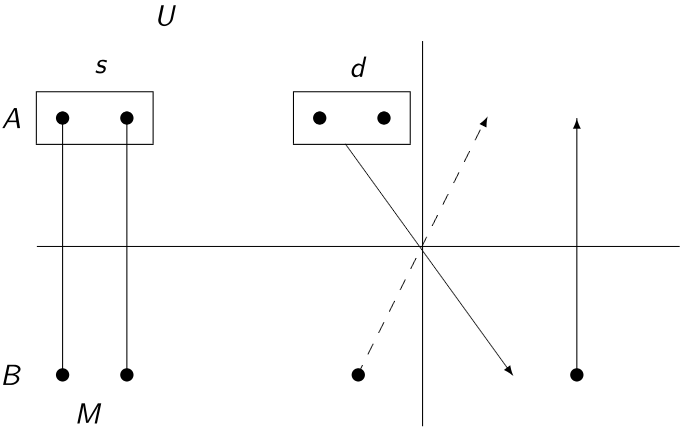
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



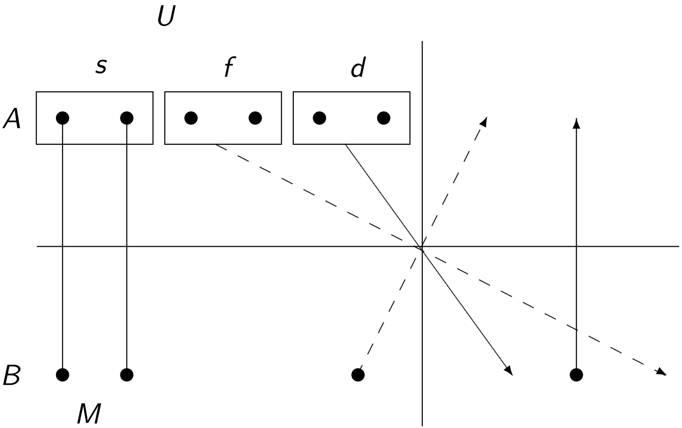
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



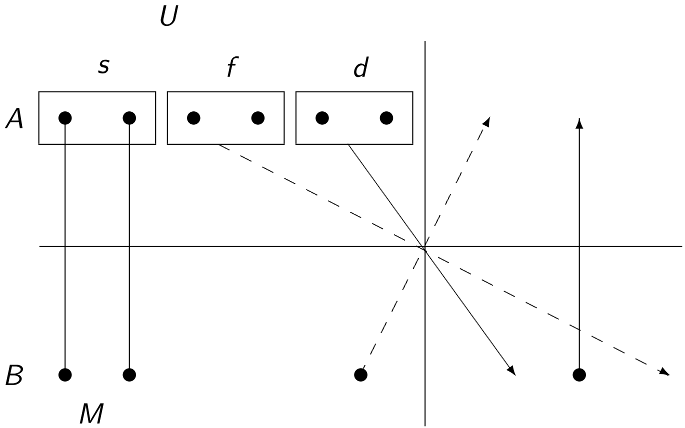
Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



Approximating $\nu_{ur}(G)$ in bipartite graphs

Proof: Given G, we maintain pairs (U, M) such that...



$$(\Delta - 1)^2 \Big((\Delta - 2)s - (d + f) \Big) \geq (\Delta - 2)f$$

Approximating $\nu_{ur}(G)$ in bipartite graphs

Initially, $(U, M) = (\emptyset, \emptyset)$.

Approximating $\nu_{ur}(G)$ in bipartite graphs

Initially, $(U, M) = (\emptyset, \emptyset)$.

While $U \neq V(G)$, we extend (U, M) to (U', M').

Approximating $\nu_{ur}(G)$ in bipartite graphs

Initially, $(U, M) = (\emptyset, \emptyset)$.

While $U \neq V(G)$, we extend (U, M) to (U', M').

Once $U = V(G)$, the statement follows using

$$(\Delta - 1)^2 \Big((\Delta - 2) \underbrace{|M|}_{s} - (\underbrace{|A| - |M|}_{d+f}) \Big) \geq (\Delta - 2)(\underbrace{|A| - |M|}_{f}).$$

Approximating $\nu_{ur}(G)$ in bipartite graphs

Initially, $(U, M) = (\emptyset, \emptyset)$.

While $U \neq V(G)$, we extend (U, M) to (U', M').

Once $U = V(G)$, the statement follows using

$$(\Delta - 1)^2\Big((\Delta - 2)\underbrace{|M|}_{s} - (\underbrace{|A| - |M|}_{d+f})\Big) \geq (\Delta - 2)(\underbrace{|A| - |M|}_{f}).$$

Suppose $U \neq V(G)$.

Approximating $\nu_{ur}(G)$ in bipartite graphs

Initially, $(U, M) = (\emptyset, \emptyset)$.

While $U \neq V(G)$, we extend (U, M) to (U', M').

Once $U = V(G)$, the statement follows using

$$(\Delta - 1)^2 \Big((\Delta - 2) \underbrace{|M|}_{s} - (\underbrace{|A| - |M|}_{d+f}) \Big) \geq (\Delta - 2)(\underbrace{|A| - |M|}_{f}).$$

Suppose $U \neq V(G)$. There is some vertex u in $B \setminus U$ with

$$1 \leq d_{A \setminus U}(u) \leq \Delta - 1.$$

Approximating $\nu_{ur}(G)$ in bipartite graphs

Initially, $(U, M) = (\emptyset, \emptyset)$.

While $U \neq V(G)$, we extend (U, M) to (U', M').

Once $U = V(G)$, the statement follows using

$$(\Delta - 1)^2 \Big((\Delta - 2) \underbrace{|M|}_{s} - (\underbrace{|A| - |M|}_{d+f}) \Big) \geq (\Delta - 2)(\underbrace{|A| - |M|}_{f}).$$

Suppose $U \neq V(G)$. There is some vertex u in $B \setminus U$ with

$$1 \leq d_{A \setminus U}(u) \leq \Delta - 1.$$

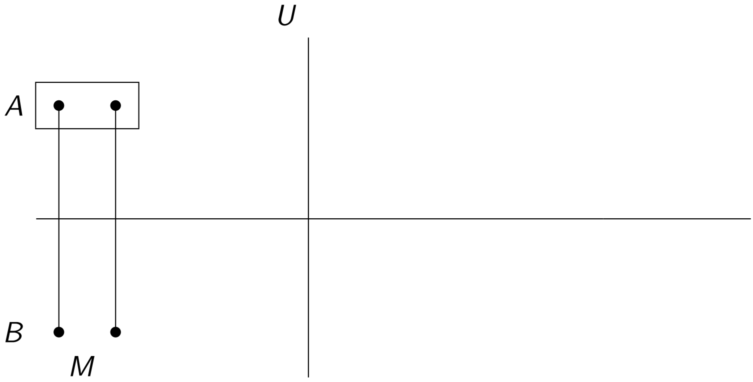
Choose u minimizing $d_{A \setminus U}(u)$.

Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 1 $d_{A \setminus U}(u) = 1$.

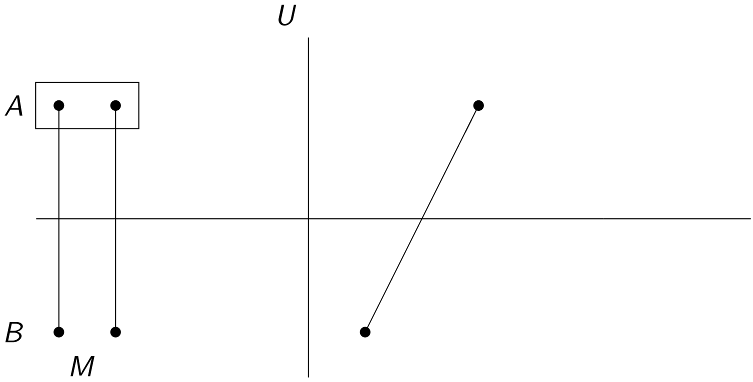
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 1 $d_{A \setminus U}(u) = 1.$



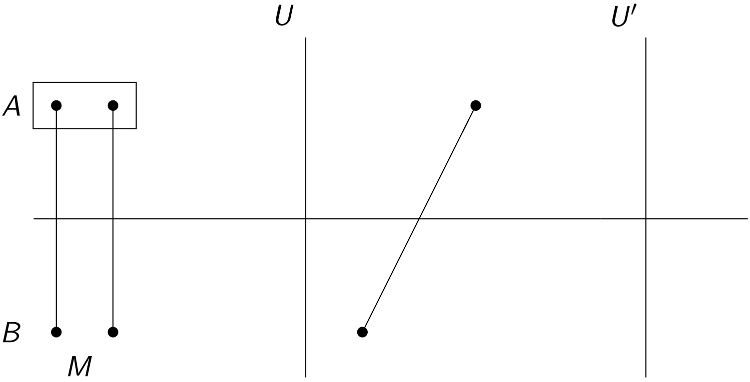
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 1 $d_{A \setminus U}(u) = 1$.



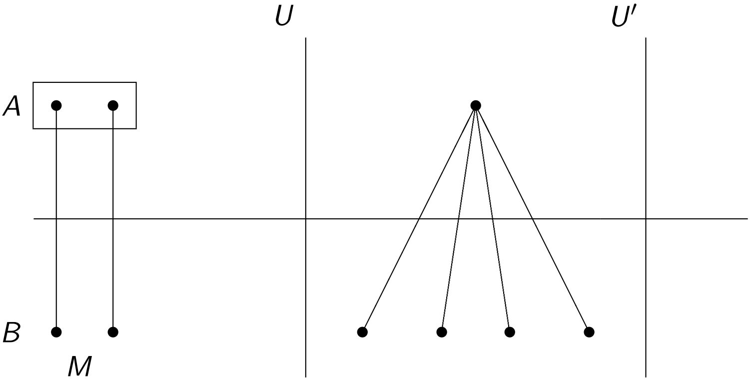
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 1 $d_{A \setminus U}(u) = 1$.



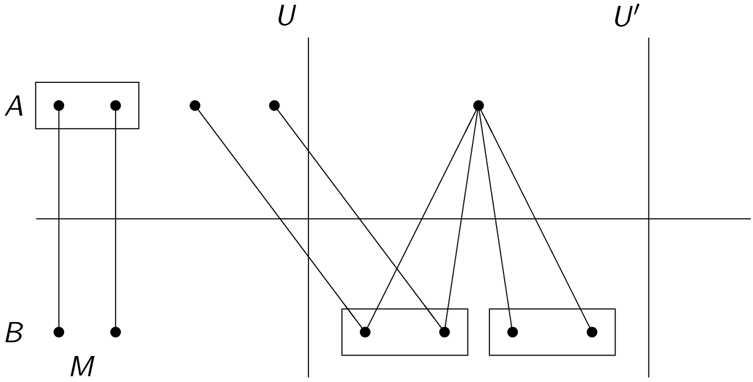
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 1 $d_{A\setminus U}(u) = 1$.



Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 1 $d_{A\setminus U}(u) = 1$.

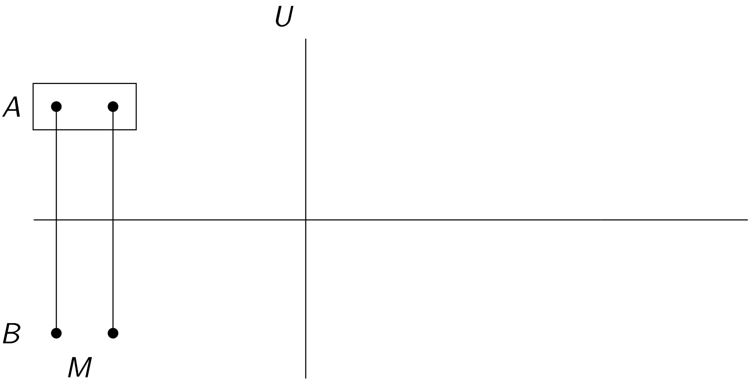


Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 2 $d_{A \setminus U}(u) \geq 2$.

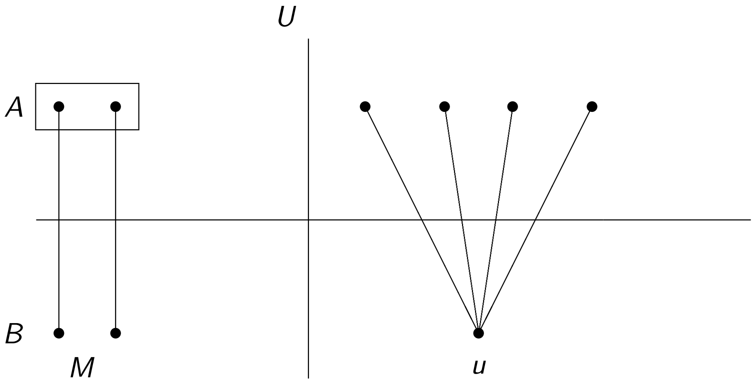
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 2 $d_{A\setminus U}(u) \geq 2$.



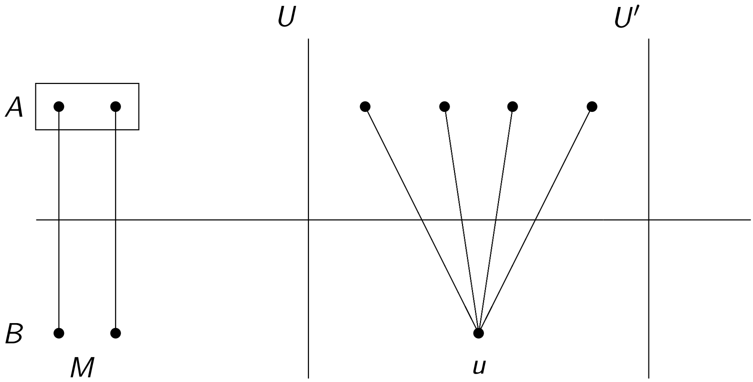
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 2 $d_{A \setminus U}(u) \geq 2$.



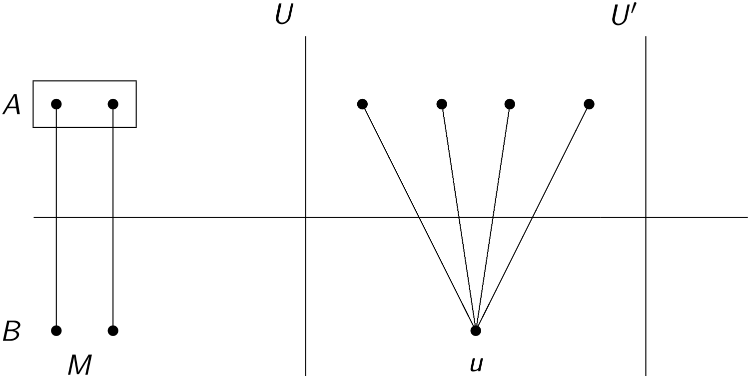
Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 2 $d_{A \setminus U}(u) \geq 2$.



Approximating $\nu_{ur}(G)$ in bipartite graphs

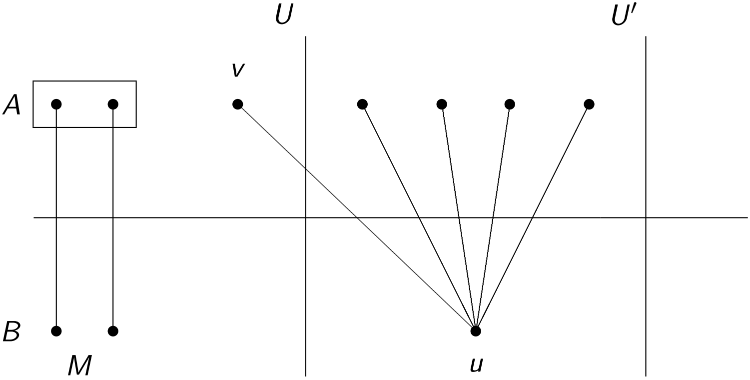
Case 2 $d_{A \setminus U}(u) \geq 2$.



If u has a neighbor v in U and no neighbor of u in U is incident with M, then...

Approximating $\nu_{ur}(G)$ in bipartite graphs

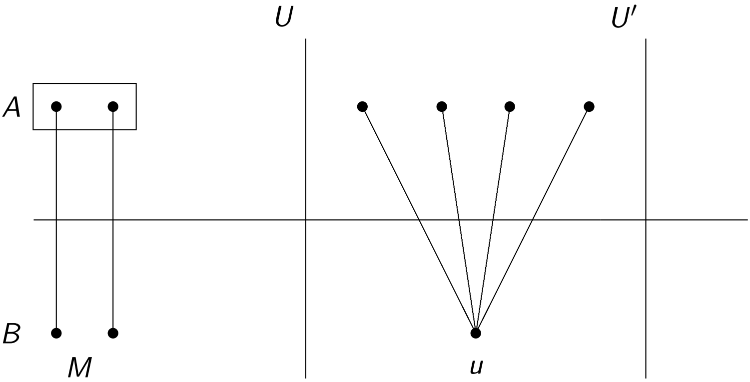
Case 2 $d_{A \setminus U}(u) \geq 2$.



If u has a neighbor v in U and no neighbor of u in U is incident with M, then...

Approximating $\nu_{ur}(G)$ in bipartite graphs

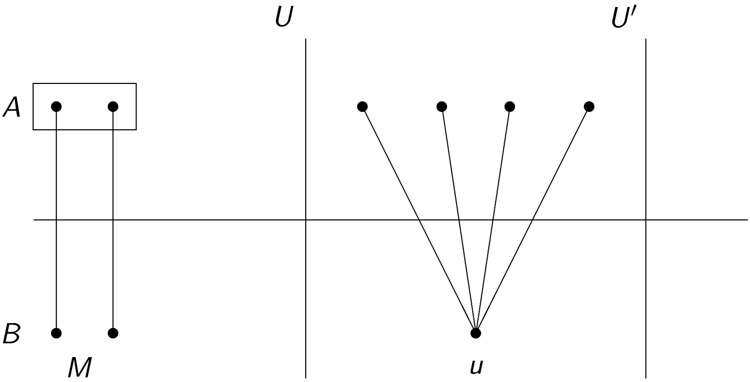
Case 2 $d_{A\setminus U}(u) \geq 2$.



If u has a neighbor v in U and no neighbor of u in U is incident with M, then...otherwise...

Approximating $\nu_{ur}(G)$ in bipartite graphs

Case 2 $d_{A \setminus U}(u) \geq 2$.



If u has a neighbor v in U and no neighbor of u in U is incident with M, then...otherwise...$\square$

Upper bounds on $\chi'_{ur}(G)$

Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$.

Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored,

Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and

Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.

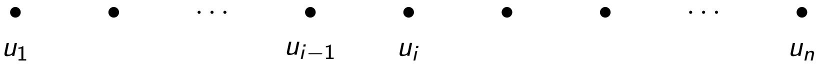
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



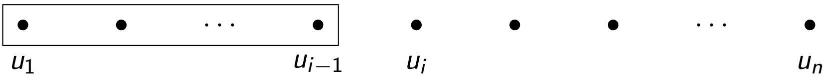
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



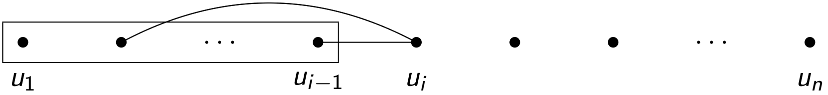
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



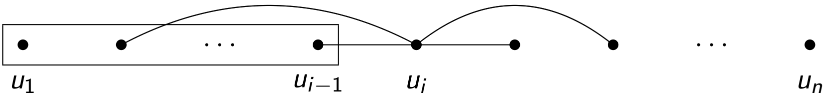
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \le \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



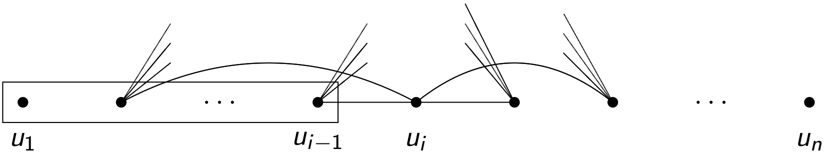
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



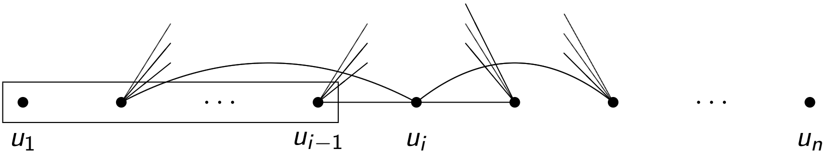
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



This requires at most $(\Delta - d)\Delta + d(\Delta - 1) + d$

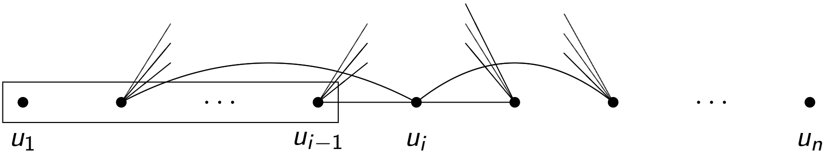
Upper bounds on $\chi'_{ur}(G)$

Theorem (Baste, R, and Sau 2016)

If G is a connected graph of maximum degree at most Δ, then $\chi'_{ur}(G) \leq \Delta^2$ with equality if and only if G is $K_{\Delta,\Delta}$.

Proof: Let $u_1, \ldots, u_n$ be any linear ordering of $V(G)$. For i from 1 up to $n-1$, assume that the edges incident with vertices in $\{u_1, \ldots, u_{i-1}\}$ are already colored, and color the d edges between u_i and $\{u_{i+1}, \ldots, u_n\}$

- using distinct colors, and
- avoiding any color that has already been used on a previously colored edge incident with some neighbor of u_i.



This requires at most $(\Delta - d)\Delta + d(\Delta - 1) + d = \Delta^2$ many colors.

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \dots u_{r_k} u_{s_k} u_{r_1}.$$

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \dots u_{r_k} u_{s_k} u_{r_1}.$$

Let r_1 be the smallest index, and let $u_{r_1} u_{s_k} \in M$, that is, $r_1 < s_1, r_2$.

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \dots u_{r_k} u_{s_k} u_{r_1}.$$

Let r_1 be the smallest index, and let $u_{r_1} u_{s_k} \in M$, that is, $r_1 < s_1, r_2$. Since $u_{s_1} u_{r_2} \in M$, the coloring rule implies

$$r_1 < r_2 < s_1.$$

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \dots u_{r_k} u_{s_k} u_{r_1}.$$

Let r_1 be the smallest index, and let $u_{r_1} u_{s_k} \in M$, that is, $r_1 < s_1, r_2$. Since $u_{s_1} u_{r_2} \in M$, the coloring rule implies

$$r_1 < r_2 < s_1.$$

Again exploiting the coloring rule, we obtain

$$r_i < r_{i+1} < s_i \quad \Rightarrow \quad r_{i+1} < r_{i+2} < s_{i+1},$$

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \dots u_{r_k} u_{s_k} u_{r_1}.$$

Let r_1 be the smallest index, and let $u_{r_1} u_{s_k} \in M$, that is, $r_1 < s_1, r_2$. Since $u_{s_1} u_{r_2} \in M$, the coloring rule implies

$$r_1 < r_2 < s_1.$$

Again exploiting the coloring rule, we obtain

$$r_i < r_{i+1} < s_i \quad \Rightarrow \quad r_{i+1} < r_{i+2} < s_{i+1},$$

which implies the contradiction

$$r_1 < r_2 < \cdots < r_k < r_1.$$

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \ldots u_{r_k} u_{s_k} u_{r_1}.$$

Let r_1 be the smallest index, and let $u_{r_1} u_{s_k} \in M$, that is, $r_1 < s_1, r_2$. Since $u_{s_1} u_{r_2} \in M$, the coloring rule implies

$$r_1 < r_2 < s_1.$$

Again exploiting the coloring rule, we obtain

$$r_i < r_{i+1} < s_i \quad \Rightarrow \quad r_{i+1} < r_{i+2} < s_{i+1},$$

which implies the contradiction

$$r_1 < r_2 < \cdots < r_k < r_1.$$

For the characterization of the extremal graphs, consider a uniquely restricted edge coloring

- using Δ^2 colors, and
- minimizing the number of edges colored Δ^2.

Upper bounds on $\chi'_{ur}(G)$

If some color class M is not a uniquely restricted matching, then there is an M-alternating cycle

$$C : u_{r_1} u_{s_1} u_{r_2} u_{s_2} \dots u_{r_k} u_{s_k} u_{r_1}.$$

Let r_1 be the smallest index, and let $u_{r_1} u_{s_k} \in M$, that is, $r_1 < s_1, r_2$. Since $u_{s_1} u_{r_2} \in M$, the coloring rule implies

$$r_1 < r_2 < s_1.$$

Again exploiting the coloring rule, we obtain

$$r_i < r_{i+1} < s_i \quad \Rightarrow \quad r_{i+1} < r_{i+2} < s_{i+1},$$

which implies the contradiction

$$r_1 < r_2 < \cdots < r_k < r_1.$$

For the characterization of the extremal graphs, consider a uniquely restricted edge coloring

- using Δ^2 colors, and
- minimizing the number of edges colored Δ^2.

□

Upper bounds on $\chi'_{ur}(G)$

The following is inspired by Lovász's elegant proof of Brooks' Theorem.

Upper bounds on $\chi'_{ur}(G)$

The following is inspired by Lovász's elegant proof of Brooks' Theorem.

Lemma (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, and M is a matching in G, then M can be partitioned into at most $\Delta - 1$ uniquely restricted matchings in G.

Upper bounds on $\chi'_{ur}(G)$

The following is inspired by Lovász's elegant proof of Brooks' Theorem.

Lemma (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, and M is a matching in G, then M can be partitioned into at most $\Delta - 1$ uniquely restricted matchings in G.

Theorem (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, then $\chi'_{ur}(G) \leq \Delta^2 - \Delta$.

Upper bounds on $\chi'_{ur}(G)$

The following is inspired by Lovász's elegant proof of Brooks' Theorem.

Lemma (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, and M is a matching in G, then M can be partitioned into at most $\Delta - 1$ uniquely restricted matchings in G.

Theorem (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, then $\chi'_{ur}(G) \leq \Delta^2 - \Delta$.

Proof: Lemma + Kőnig's theorem. □

Upper bounds on $\chi'_{ur}(G)$

The following is inspired by Lovász's elegant proof of Brooks' Theorem.

Lemma (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, and M is a matching in G, then M can be partitioned into at most $\Delta - 1$ uniquely restricted matchings in G.

Theorem (Baste, R, and Sau 2016)

If G is a connected bipartite graph of maximum degree at most $\Delta \geq 4$ that is distinct from $K_{\Delta,\Delta}$, then $\chi'_{ur}(G) \leq \Delta^2 - \Delta$.

Proof: Lemma + Kőnig's theorem. $\square$

For $\Delta = 3$ or G non-bipartite, the lemma fails.

Upper bounds on $\chi'_r(G)$

Upper bounds on $\chi'_r(G)$

Theorem (Baste and R 2017)

If r is a positive integer and G is a graph of maximum degree at most Δ, then

$$\chi'_r(G) \leq \frac{2(\Delta-1)^2}{r+1} + 2(\Delta-1) + 1. \tag{1}$$

Upper bounds on $\chi'_r(G)$

Theorem (Baste and R 2017)

If r is a positive integer and G is a graph of maximum degree at most Δ, then

$$\chi'_r(G) \leq \frac{2(\Delta-1)^2}{r+1} + 2(\Delta-1) + 1. \tag{1}$$

For $r = 1$, G satisfies (1) with equality if and only if G is $K_{\Delta,\Delta}$

Upper bounds on $\chi'_r(G)$

Theorem (Baste and R 2017)

If r is a positive integer and G is a graph of maximum degree at most Δ, then

$$\chi'_r(G) \leq \frac{2(\Delta-1)^2}{r+1} + 2(\Delta-1) + 1. \tag{1}$$

For $r = 1$, G satisfies (1) with equality if and only if G is $K_{\Delta,\Delta}$

Proposition (Baste and R 2017)

If r is an integer at least 2, then no graph G of maximum degree at most Δ satisfies (1) with equality.

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Goddard, Hedetniemi, Hedetniemi, and Laskar 2005 ask for an efficient algorithm to determine the acyclic matching number of interval graphs.

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Goddard, Hedetniemi, Hedetniemi, and Laskar 2005 ask for an efficient algorithm to determine the acyclic matching number of interval graphs. Panda and Pradhan 2012 describe such algorithms for chain graphs and bipartite permutation graphs.

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Goddard, Hedetniemi, Hedetniemi, and Laskar 2005 ask for an efficient algorithm to determine the acyclic matching number of interval graphs. Panda and Pradhan 2012 describe such algorithms for chain graphs and bipartite permutation graphs.

Let $r \in \mathbb{N}$ be fixed.

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Goddard, Hedetniemi, Hedetniemi, and Laskar 2005 ask for an efficient algorithm to determine the acyclic matching number of interval graphs. Panda and Pradhan 2012 describe such algorithms for chain graphs and bipartite permutation graphs.

Let $r \in \mathbb{N}$ be fixed.

Let G be a given chordal graph.

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Goddard, Hedetniemi, Hedetniemi, and Laskar 2005 ask for an efficient algorithm to determine the acyclic matching number of interval graphs. Panda and Pradhan 2012 describe such algorithms for chain graphs and bipartite permutation graphs.

Let $r \in \mathbb{N}$ be fixed.

Let G be a given chordal graph.

Consider a nice tree decomposition

$$(T, (X_t)_{t \in V(T)})$$

of G with complete bags X_t.

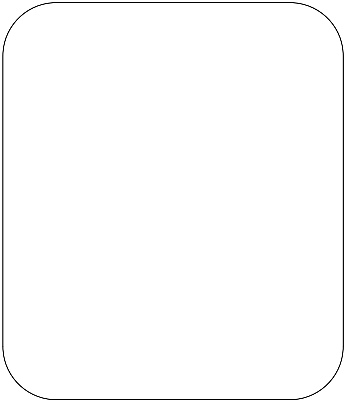
Efficient algorithm for $\nu_r(G)$ in chordal graphs

For every node t of T, generate the set $\mathcal{R}_t$ of all triples (S, N, k) such that...

Efficient algorithm for $\nu_r(G)$ in chordal graphs

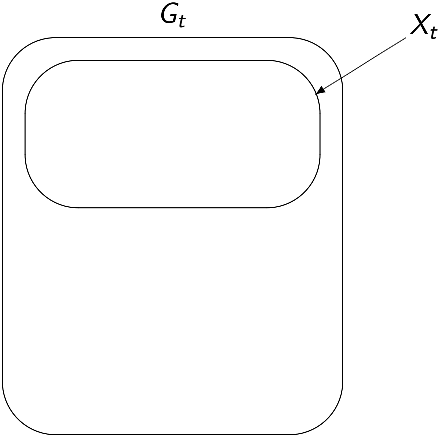
For every node t of T, generate the set $\mathcal{R}_t$ of all triples (S, N, k) such that...

G_t



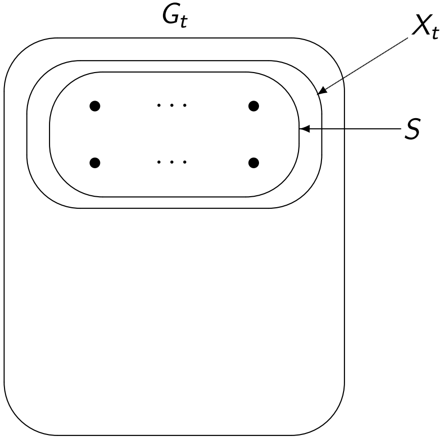
Efficient algorithm for $\nu_r(G)$ in chordal graphs

For every node t of T, generate the set $\mathcal{R}_t$ of all triples (S, N, k) such that...



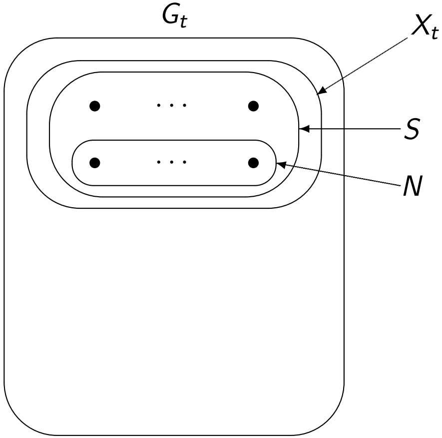
Efficient algorithm for $\nu_r(G)$ in chordal graphs

For every node t of T, generate the set $\mathcal{R}_t$ of all triples (S, N, k) such that...



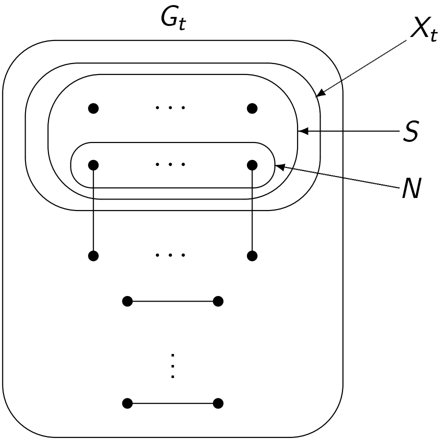
Efficient algorithm for $\nu_r(G)$ in chordal graphs

For every node t of T, generate the set $\mathcal{R}_t$ of all triples (S, N, k) such that...



Efficient algorithm for $\nu_r(G)$ in chordal graphs

For every node t of T, generate the set $\mathcal{R}_t$ of all triples (S, N, k) such that...



...where M has size k and $G[V(M) \cup S]$ is r-degenerate.

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Lemma (Baste and R 2017)

Let G, $(T, (X_t)_{t\in V(T)})$, and $(\mathcal{R}_t)_{t\in V(T)}$ be as above.

(a) *If t is a leaf of T, then $\mathcal{R}_t = \{(\emptyset, \emptyset, 0)\}$.*

(b) *If t is an introduce node, t' is the child of t, and $\{x\} = X_t \setminus X_{t'}$, then $(S, N, k) \in \mathcal{R}_t$ if and only if*

- *either $(S, N, k) \in \mathcal{R}_{t'}$*
- *or $(S, N, k) = (S' \cup \{x\}, N, k)$ for some $(S', N, k) \in \mathcal{R}_{t'}$ with $|S'| \leq r$.*

(c) *If t is a forget node, t' is the child of t, and $\{x\} = X_{t'} \setminus X_t$, then $(S, N, k) \in \mathcal{R}_t$ if and only if*

- *either $(S, N, k) \in \mathcal{R}_{t'}$ and $x \notin S$,*
- *or $(S, N, k) = (S' \setminus \{x\}, N' \cup \{y\}, k' + 1)$ for some $(S', N', k') \in \mathcal{R}_{t'}$ with $x \in S' \setminus N'$ and some $y \in S' \setminus (N' \cup \{x\})$,*
- *or $(S, N, k) = (S' \setminus \{x\}, N' \setminus \{x\}, k')$ for some $(S', N', k') \in \mathcal{R}_{t'}$ with $x \in N'$.*

(d) *If t is a join node, and t' and t'' are the children of t, then $(S, N, k) \in \mathcal{R}_t$ if and only if $(S, N, k) = (S, N' \cup N'', k' + k'')$ for some $(S, N', k') \in \mathcal{R}_{t'}$ and $(S, N'', k'') \in \mathcal{R}_{t''}$ with $N' \cap N'' = \emptyset$.*

Efficient algorithm for $\nu_r(G)$ in chordal graphs

Theorem (Baste and R 2017)

For a fixed positive integer r, and a given chordal graph G, the maximum size of an r-degenerate matching can be determined in polynomial time.

Decycling/bipartizing matchings

Decycling/bipartizing matchings

Let $\mathcal{FM}$ be the set of all graphs G that have a matching M such that $G - M$ is a forest.

Decycling/bipartizing matchings

Let $\mathcal{FM}$ be the set of all graphs G that have a matching M such that $G - M$ is a forest. Let $\mathcal{BM}$ ... is bipartite.

Decycling/bipartizing matchings

Let $\mathcal{FM}$ be the set of all graphs G that have a matching M such that $G - M$ is a forest. Let $\mathcal{BM}$ ... is bipartite.

Lemma (Lima, R, Souza, and Szwarcfiter 2016)

Let G be a graph.

(i) *If $G \in \mathcal{FM}$ is connected, then G has a matching M for which $G - M$ is a tree.*

Decycling/bipartizing matchings

Let $\mathcal{FM}$ be the set of all graphs G that have a matching M such that $G - M$ is a forest. Let $\mathcal{BM}$ ... is bipartite.

Lemma (Lima, R, Souza, and Szwarcfiter 2016)

Let G be a graph.

(i) *If $G \in \mathcal{FM}$ is connected, then G has a matching M for which $G - M$ is a tree.*

(ii) *If $G \in \mathcal{FM}$, then $m(H) \leq \left\lfloor \frac{3n(H)}{2} \right\rfloor - 1$ for every subgraph H of G.*

Decycling/bipartizing matchings

Let $\mathcal{FM}$ be the set of all graphs G that have a matching M such that $G - M$ is a forest. Let $\mathcal{BM}$ ... is bipartite.

Lemma (Lima, R, Souza, and Szwarcfiter 2016)

Let G be a graph.

(i) *If $G \in \mathcal{FM}$ is connected, then G has a matching M for which $G - M$ is a tree.*

(ii) *If $G \in \mathcal{FM}$, then $m(H) \leq \left\lfloor \frac{3n(H)}{2} \right\rfloor - 1$ for every subgraph H of G.*

(iii) *If G is subcubic and connected, then $G \in \mathcal{FM}$ if and only if G has a spanning tree T such that all endvertices of T are of degree at most 2 in G.*

Decycling/bipartizing matchings

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

For a given 2-connected planar subcubic graph G, it is NP-complete to decide whether $G \in \mathcal{FM}$.

Decycling/bipartizing matchings

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

For a given 2-connected planar subcubic graph G, it is NP-complete to decide whether $G \in \mathcal{FM}$.

Proof: HAMILTONIAN CYCLE is NP-complete for 3-connected planar cubic graphs (Garey, Johnson, and Tarjan 1976).

Decycling/bipartizing matchings

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

For a given 2-connected planar subcubic graph G, it is NP-complete to decide whether $G \in \mathcal{FM}$.

Proof: HAMILTONIAN CYCLE is NP-complete for 3-connected planar cubic graphs (Garey, Johnson, and Tarjan 1976). Remove an edge from their construction that belongs to every Hamiltonian cycle. □

Decycling/bipartizing matchings

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

For a given 2-connected planar subcubic graph G, it is NP-complete to decide whether $G \in \mathcal{FM}$.

Proof: HAMILTONIAN CYCLE is NP-complete for 3-connected planar cubic graphs (Garey, Johnson, and Tarjan 1976). Remove an edge from their construction that belongs to every Hamiltonian cycle. □

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

Deciding whether a given graph belongs to $\mathcal{FM}$ can be done in polynomial time for

- *{claw,paw}-free graphs,*
- *P_5-free graphs,*
- *chordal graphs, and*
- *C_4-free distance hereditary graphs.*

Decycling/bipartizing matchings

Fact (Lima, R, Souza, and Szwarcfiter 2016)

Every subcubic graphs belongs to $\mathcal{BM}$.

Decycling/bipartizing matchings

Fact (Lima, R, Souza, and Szwarcfiter 2016)

Every subcubic graphs belongs to $\mathcal{BM}$.

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

Deciding whether a given graph of maximum degree 4 *or a given planar graph of maximum degree* 5 *belongs to* $\mathcal{BM}$ *is NP-complete.*

Decycling/bipartizing matchings

Fact (Lima, R, Souza, and Szwarcfiter 2016)

Every subcubic graphs belongs to $\mathcal{BM}$.

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

Deciding whether a given graph of maximum degree 4 *or a given planar graph of maximum degree* 5 *belongs to* $\mathcal{BM}$ *is NP-complete.*

Theorem (Lima, R, Souza, and Szwarcfiter 2016)

Deciding whether a given graph belongs to $\mathcal{BM}$ *can be done in polynomial time for*

- $\{$*claw,paw*$\}$*-free graphs, and*
- P_5*-free graphs.*

Thank you for the attention!