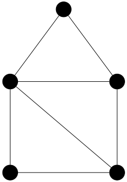
Colouring Probe H-Free Graphs

Daniël Paulusma[1] Johannes Rauch[2] **Erik Jan van Leeuwen[3]**

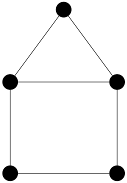
[1]Durham University, United Kingdom
[2]Ulm University, Germany
[3]Utrecht University, The Netherlands

March 11, 2026

H-Free Graphs



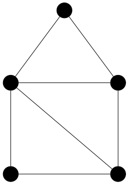
C_4-free.



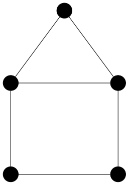
Not C_4-free.

Let H be a graph. A graph G is *H-free* if G does not contain an induced copy of H.

H-Free Graphs



C_4-free.



Not C_4-free.

Let H be a graph. A graph G is *H-free* if G does not contain an induced copy of H.

Examples:

- C_t: cycle on t vertices.
- P_t: path on t vertices.
- $H_1 + H_2$: disjoint union of H_1 and H_2.

Colouring H-Free Graphs

COLOURING: minimum number of colours to colour a graph.

Theorem (Král et al. 2001)

On H-free graphs, COLOURING *is polynomial-time solvable if $H \subseteq P_4$ or $H \subseteq P_3 + P_1$ and NP-hard otherwise.*

Colouring H-Free Graphs

COLOURING: minimum number of colours to colour a graph.

Theorem (Král et al. 2001)

On H-free graphs, COLOURING is polynomial-time solvable if $H \subseteq P_4$ or $H \subseteq P_3 + P_1$ and NP-hard otherwise.

k-COLOURING: can a graph be coloured by k colours?

Full dichotomy (poly vs. NPc, depending on H) known for all $k \geq 4$.

Colouring H-Free Graphs

COLOURING: minimum number of colours to colour a graph.

Theorem (Král et al. 2001)

On H-free graphs, COLOURING *is polynomial-time solvable if $H \subseteq P_4$ or $H \subseteq P_3 + P_1$ and NP-hard otherwise.*

k-COLOURING: can a graph be coloured by k colours?

Full dichotomy (poly vs. NPc, depending on H) known for all $k \geq 4$.

3-COLOURING:

Colouring H-Free Graphs

COLOURING: minimum number of colours to colour a graph.

Theorem (Král et al. 2001)

On H-free graphs, COLOURING is polynomial-time solvable if $H \subseteq P_4$ or $H \subseteq P_3 + P_1$ and NP-hard otherwise.

k-COLOURING: can a graph be coloured by k colours?

Full dichotomy (poly vs. NPc, depending on H) known for all $k \geq 4$.

3-COLOURING:

- NP-complete when H contains a cycle (Emden-Weinert et al. 1998) or a claw (Holyer 1981 / Leven & Galil 1983).

Colouring H-Free Graphs

COLOURING: minimum number of colours to colour a graph.

Theorem (Král et al. 2001)

On H-free graphs, COLOURING is polynomial-time solvable if $H \subseteq P_4$ or $H \subseteq P_3 + P_1$ and NP-hard otherwise.

k-COLOURING: can a graph be coloured by k colours?

Full dichotomy (poly vs. NPc, depending on H) known for all $k \geq 4$.

3-COLOURING:

- NP-complete when H contains a cycle (Emden-Weinert et al. 1998) or a claw (Holyer 1981 / Leven & Galil 1983).
- polynomial on P_7-free graphs (Bonomo et al. 2018).

Colouring H-Free Graphs

COLOURING: minimum number of colours to colour a graph.

Theorem (Král et al. 2001)

On H-free graphs, COLOURING is polynomial-time solvable if $H \subseteq P_4$ or $H \subseteq P_3 + P_1$ and NP-hard otherwise.

k-COLOURING: can a graph be coloured by k colours?

Full dichotomy (poly vs. NPc, depending on H) known for all $k \geq 4$.

3-COLOURING:

- NP-complete when H contains a cycle (Emden-Weinert et al. 1998) or a claw (Holyer 1981 / Leven & Galil 1983).
- polynomial on P_7-free graphs (Bonomo et al. 2018).
- quasi-polynomial on P_t-free graphs for $t \geq 8$ (Pilipczuk et al. 2021).

Probe H-Free Graphs

Model for incomplete adjacency information: know only neighbourhoods of *probes*.

Probe H-Free Graphs

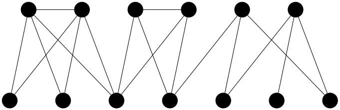
Model for incomplete adjacency information: know only neighbourhoods of *probes*.

Let H be a graph. A graph G is *probe H-free* if G can be made H-free by adding some edges between an independent set N of G.

Probe H-Free Graphs

Model for incomplete adjacency information: know only neighbourhoods of *probes*.

Let H be a graph. A graph G is *probe H-free* if G can be made H-free by adding some edges between an independent set N of G.

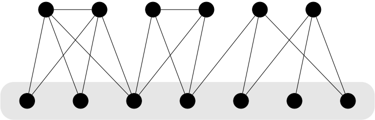


A probe P_5-free graph.

Probe H-Free Graphs

Model for incomplete adjacency information: know only neighbourhoods of *probes*.

Let H be a graph. A graph G is *probe H-free* if G can be made H-free by adding some edges between an independent set N of G.



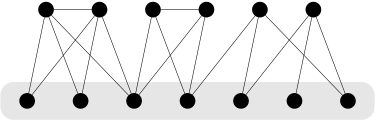
A probe P_5-free graph.

Probe H-Free Graphs

Model for incomplete adjacency information: know only neighbourhoods of *probes*.

Let H be a graph. A graph G is *probe H-free* if G can be made H-free by adding some edges between an independent set N of G.

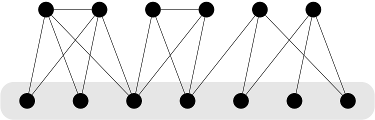
Any H-free graph is trivially probe H-free.



A probe P_5-free graph.

Probe H-Free Graphs

Model for incomplete adjacency information: know only neighbourhoods of *probes*.



A probe P_5-free graph.

Let H be a graph. A graph G is *probe H-free* if G can be made H-free by adding some edges between an independent set N of G.

Any H-free graph is trivially probe H-free.

A *partitioned* probe H-free graph (G, P, N):

- a graph G,
- a set $P \subseteq V(G)$,
- an independent set $N = V(G) \setminus P$,

such that $G + F$ is H-free for some $F \subseteq \binom{N}{2}$.

Optimisation Problems on Probe *H*-Free Graphs

Existing studies:

- VERTEX COVER (Brettell et al. 2025)
- MATCHING CUT (Dabrowski et al. 2025)

Optimisation Problems on Probe H-Free Graphs

Existing studies:

- Vertex Cover (Brettell et al. 2025)
- Matching Cut (Dabrowski et al. 2025)

Complexity boundary at probe P_5-free graphs?

	P_5-free	probe P_5-free
Vertex Cover	poly (Lokshtanov et al. 2014)	?
Matching Cut	poly (Feghali 2023)	NP-complete (Dabrowski et al. 2025)

(Vertex Cover becomes NP-complete on probe P_7-free graphs (Brettell et al. 2025).)

Optimisation Problems on Probe H-Free Graphs

Existing studies:

- Vertex Cover (Brettell et al. 2025)
- Matching Cut (Dabrowski et al. 2025)

Complexity boundary at probe P_5-free graphs?

	P_5-free	probe P_5-free
Vertex Cover	poly (Lokshtanov et al. 2014)	?
Matching Cut	poly (Feghali 2023)	NP-complete (Dabrowski et al. 2025)

(Vertex Cover becomes NP-complete on probe P_7-free graphs (Brettell et al. 2025).)

Our Research Question

What is the complexity of Colouring and k-Colouring on probe H-free graphs? Particularly when $H = P_t$ for some t.

Our Results

Theorem

For a graph H, COLOURING is polynomial-time solvable for probe H-free graphs if $H \subseteq_i P_4$, and else it is NP-hard even for partitioned probe H-free graphs.

Contrast: COLOURING is poly-time solvable on $3P_1$-free graphs (Král et al. 2001).

Our Results

Theorem

For a graph H, COLOURING is polynomial-time solvable for probe H-free graphs if $H \subseteq_i P_4$, and else it is NP-hard even for partitioned probe H-free graphs.

Contrast: COLOURING is poly-time solvable on $3P_1$-free graphs (Král et al. 2001).

Theorem

For an integer $t \geq 1$, 3-COLOURING on partitioned probe P_t-free graphs is polynomial-time solvable if $t \leq 5$ and NP-complete if $t \geq 6$.

Our Results

Theorem

For a graph H, COLOURING is polynomial-time solvable for probe H-free graphs if $H \subseteq_i P_4$, and else it is NP-hard even for partitioned probe H-free graphs.

Contrast: COLOURING is poly-time solvable on $3P_1$-free graphs (Král et al. 2001).

Theorem

For an integer $t \geq 1$, 3-COLOURING on partitioned probe P_t-free graphs is polynomial-time solvable if $t \leq 5$ and NP-complete if $t \geq 6$.

Main Result

3-COLOURING on partitioned probe P_5-free graphs is polynomial-time solvable.

Our Results

Theorem

For a graph H, Colouring *is polynomial-time solvable for probe H-free graphs if $H \subseteq_i P_4$, and else it is NP-hard even for partitioned probe H-free graphs.*

Contrast: Colouring is poly-time solvable on $3P_1$-free graphs (Král et al. 2001).

Theorem

For an integer $t \geq 1$, 3-Colouring *on partitioned probe P_t-free graphs is polynomial-time solvable if $t \leq 5$ and NP-complete if $t \geq 6$.*

Main Result

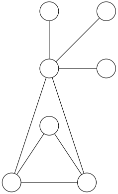
3-Colouring on partitioned probe P_5-free graphs is polynomial-time solvable.

- First problem for which polynomial-time solvability on P_5-free graphs is lifted to probe P_5-free graphs.
- Known algorithms for 3-Colouring on P_5-free graphs (Brettell et al. 2022 / Randerath et al. 2002 / Woeginger & Sgall 2001) do not directly generalize.

Subroutines for 3-COLOURING on Probe P_5-Free Graphs

Two important subroutines to the algorithm

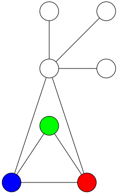
Partial colour propagation



Subroutines for 3-Colouring on Probe P_5-Free Graphs

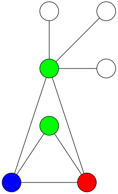
Two important subroutines to the algorithm

Partial colour propagation



Two important subroutines to the algorithm

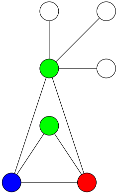
Partial colour propagation



Subroutines for 3-Colouring on Probe P_5-Free Graphs

Two important subroutines to the algorithm

Partial colour propagation

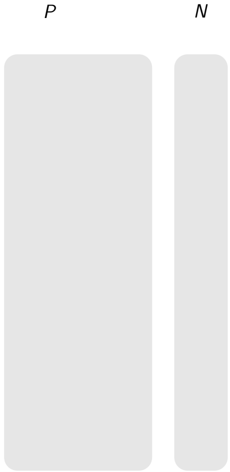


Pre-colouring extension

Theorem (Edwards 1986)

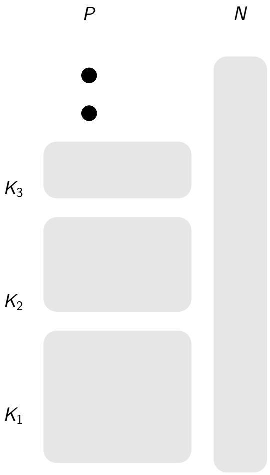
If, in a partial colouring of G, every vertex neighbours a pre-coloured vertex, then we can decide in polynomial time whether G admits a 3-colouring extending the partial colouring.

3-Colouring on Probe P_5-Free Graphs



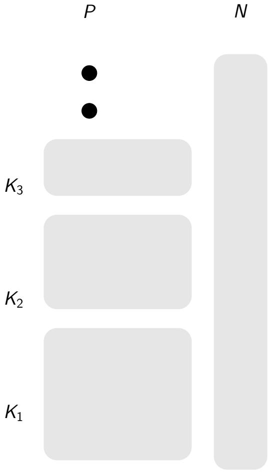
- Recall: N is an independent set in G.

3-Colouring on Probe P_5-Free Graphs



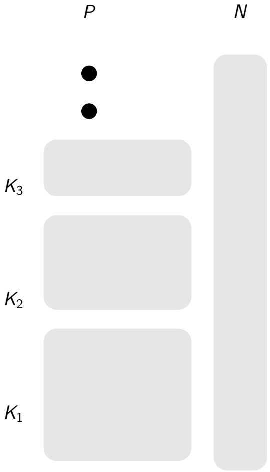
- Recall: N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.

3-Colouring on Probe P_5-Free Graphs



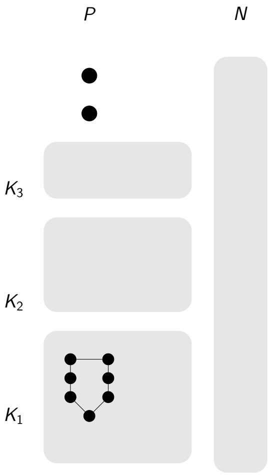
- Recall: N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- $K_1, \ldots, K_t$ bipartite: 3-colouring exists.

3-COLOURING on Probe P_5-Free Graphs



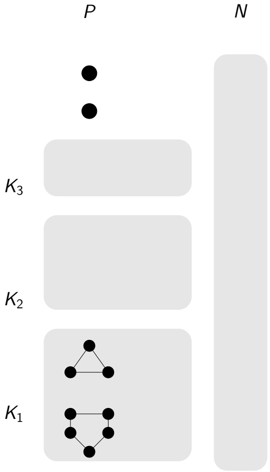
- Recall: N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- $K_1, \ldots, K_t$ bipartite: 3-colouring exists.
- Assume K_1 is not bipartite.

3-Colouring on Probe P_5-Free Graphs



- Recall: N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- $K_1, \ldots, K_t$ bipartite: 3-colouring exists.
- Assume K_1 is not bipartite.

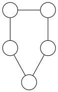
3-COLOURING on Probe P_5-Free Graphs



- Recall: N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- $K_1, \ldots, K_t$ bipartite: 3-colouring exists.
- Assume K_1 is not bipartite.
- K_1 contains a C_3 or a C_5.

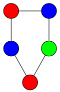
K_1 Contains a C_5

- $G[P]$ contains an induced 5-cycle C.



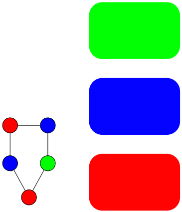
K_1 Contains a C_5

- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.

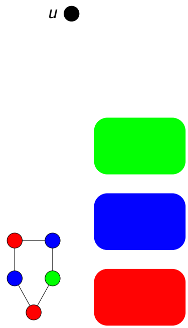


K_1 Contains a C_5

- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.



K_1 Contains a C_5

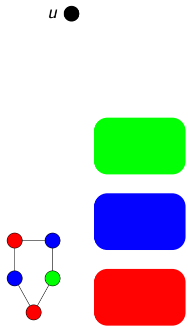


- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

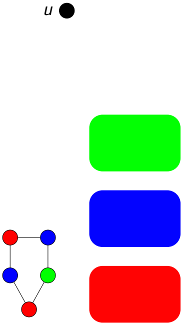
Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

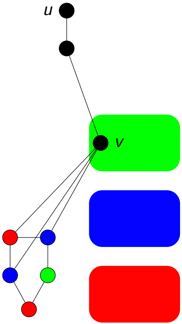
Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.
- Let $F \subseteq \binom{N}{2}$ s.t. $G + F$ is P_5-free.
- Find a shortest u-C path Q in $G + F$.
- Let v be the vertex of Q preceeding C.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

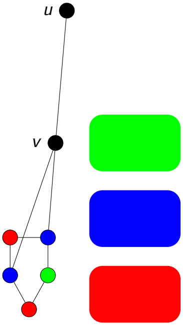
Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.
- Let $F \subseteq \binom{N}{2}$ s.t. $G + F$ is P_5-free.
- Find a shortest u-C path Q in $G + F$.
- Let v be the vertex of Q preceeding C.
- v is not complete to C.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

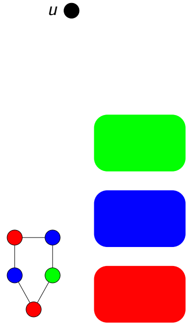
Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.
- Let $F \subseteq \binom{N}{2}$ s.t. $G + F$ is P_5-free.
- Find a shortest u-C path Q in $G + F$.
- Let v be the vertex of Q preceeding C.
- v sees only one colour.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

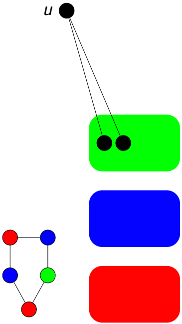
Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.
- Let $F \subseteq \binom{N}{2}$ s.t. $G + F$ is P_5-free.
- Find a shortest u-C path Q in $G + F$.
- Let v be the vertex of Q preceeding C.
- $G + F$ contains an induced P_5, contradiction.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

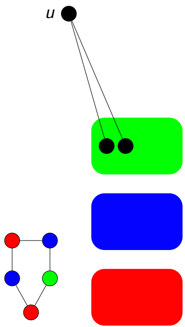
Lemma

After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.
- Let $F \subseteq \binom{N}{2}$ s.t. $G + F$ is P_5-free.
- Find a shortest u-C path Q in $G + F$.
- Let v be the vertex of Q preceeding C.
- $G + F$ contains an induced P_5, contradiction.

K_1 Contains a C_5



- $G[P]$ contains an induced 5-cycle C.
- Branch on all valid 3-colourings of C.
- For each branch, propagate the colours.

Lemma

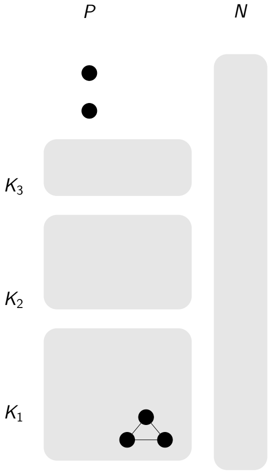
After propogation, every vertex u that is still uncoloured has a coloured neighbour.

Proof

- Suppose u has no coloured neighbours.
- Let $F \subseteq \binom{N}{2}$ s.t. $G + F$ is P_5-free.
- Find a shortest u-C path Q in $G + F$.
- Let v be the vertex of Q preceeding C.
- $G + F$ contains an induced P_5, contradiction.

Apply pre-colouring extension subroutine.

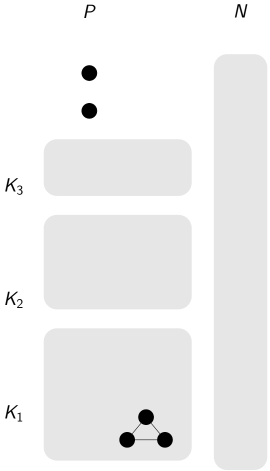
K_1 Contains a C_3



Recall:

- N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- K_1 contains no C_5, but a C_3.

K_1 Contains a C_3



Recall:

- N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- K_1 contains no C_5, but a C_3.

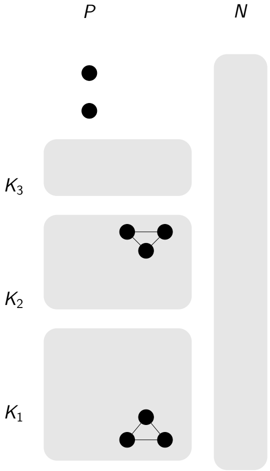
Proposition

$K_2, \ldots, K_t$ are bipartite.

Proof.

□

K_1 Contains a C_3



Recall:

- N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- K_1 contains no C_5, but a C_3.

Proposition

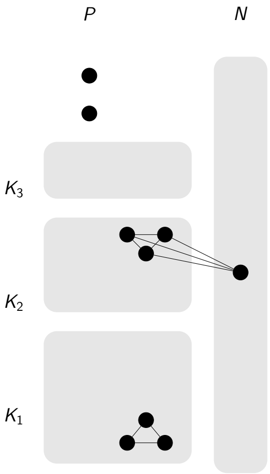
$K_2, \ldots, K_t$ are bipartite.

Proof.

- Suppose K_2 contains a C_3.

□

K_1 Contains a C_3



Recall:

- N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- K_1 contains no C_5, but a C_3.

Proposition

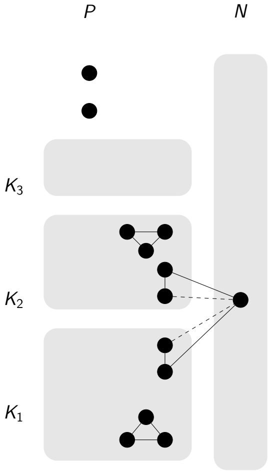
$K_2, \ldots, K_t$ are bipartite.

Proof.

- Suppose K_2 contains a C_3.
- Any vertex in N is not complete to K_1 nor K_2.

□

K_1 Contains a C_3



Recall:

- N is an independent set in G.
- $K_1, \ldots, K_t$: non-trivial comps of $G[P]$.
- K_1 contains no C_5, but a C_3.

Proposition

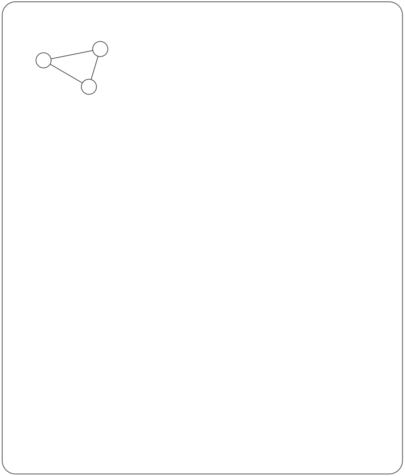
$K_2, \ldots, K_t$ are bipartite.

Proof.

- Suppose K_2 contains a C_3.
- Any vertex in N is not complete to K_1 nor K_2.
- K_1, K_2 have a neighbour in N.

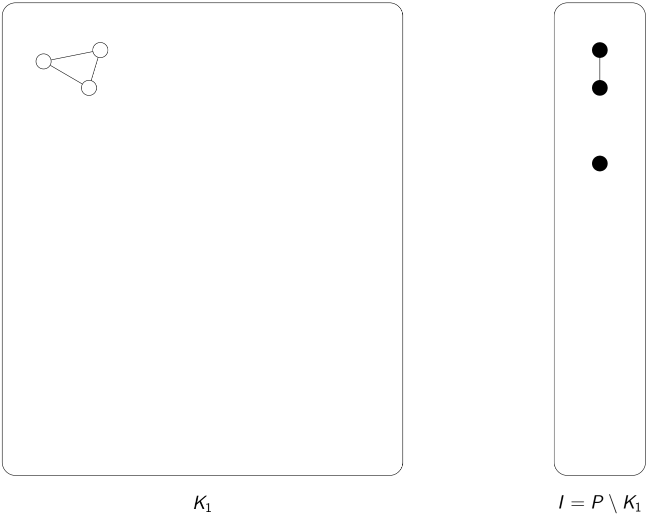
□

K_1 Contains a C_3

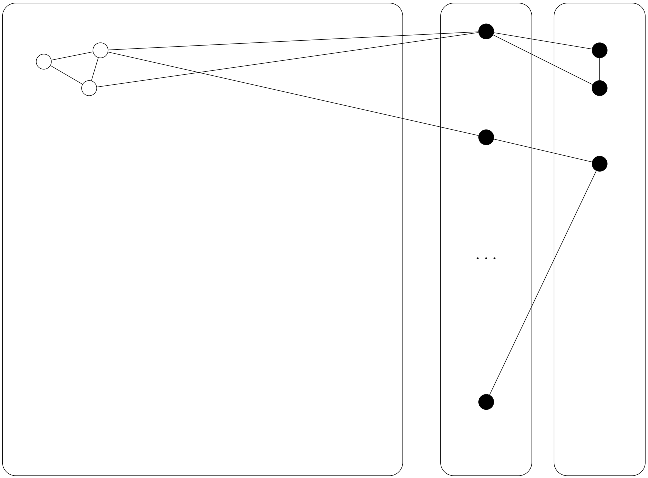


K_1

K_1 Contains a C_3

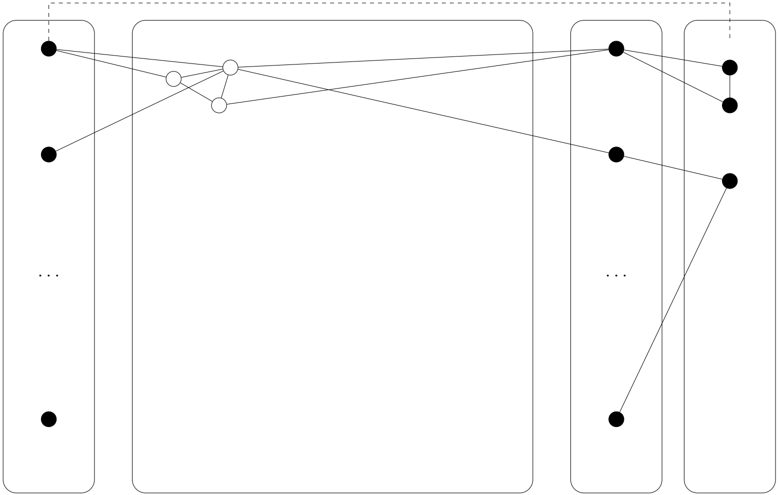


K_1 Contains a C_3



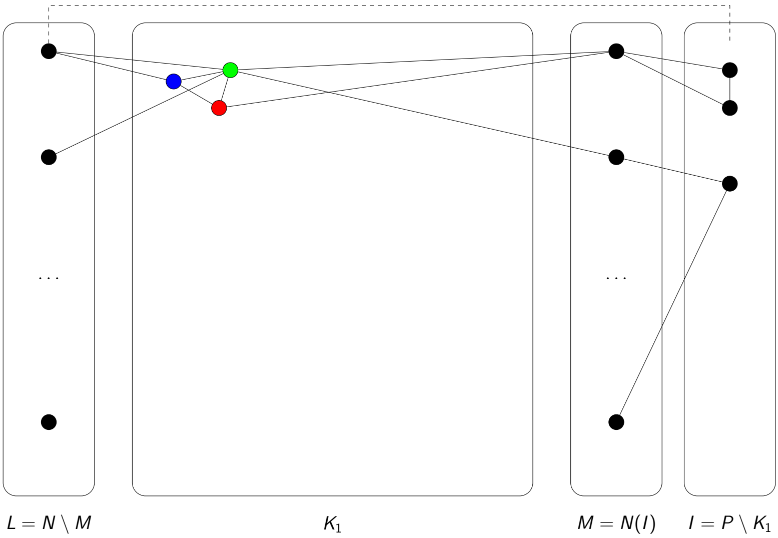
K_1 $M = N(I)$ $I = P \setminus K_1$

K_1 Contains a C_3

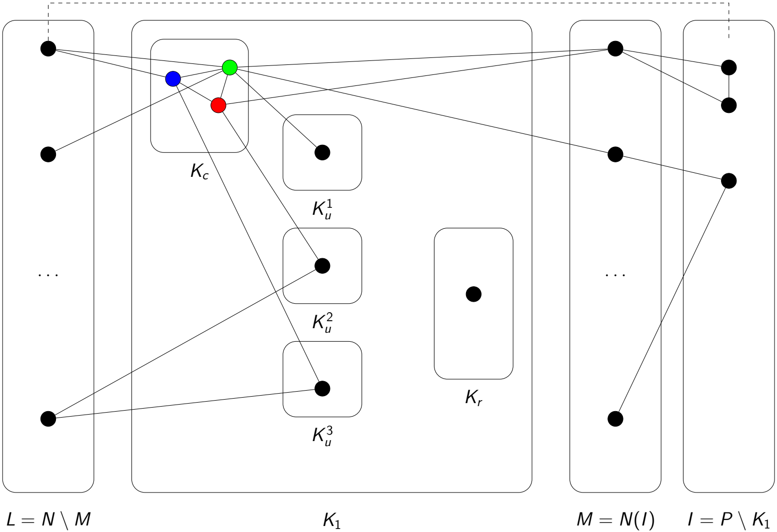


$L = N \setminus M$ K_1 $M = N(I)$ $I = P \setminus K_1$

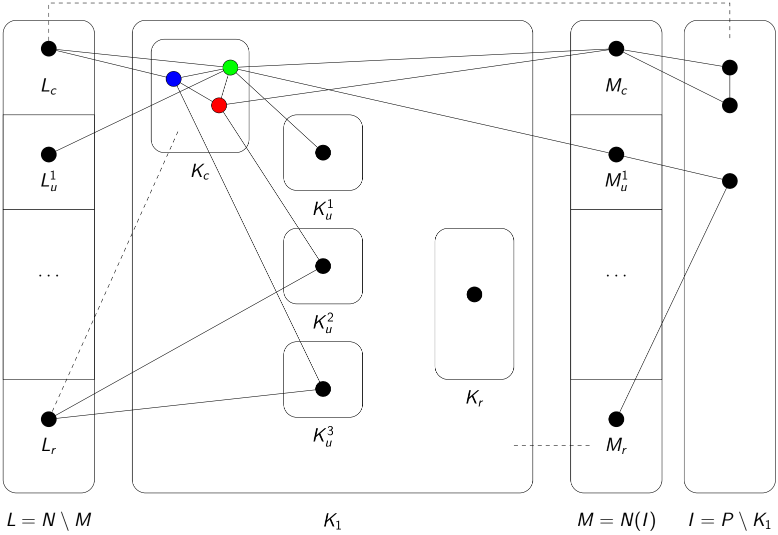
K_1 Contains a C_3



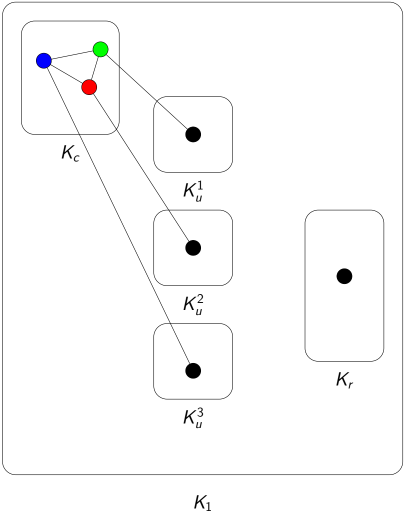
K_1 Contains a C_3



K_1 Contains a C_3

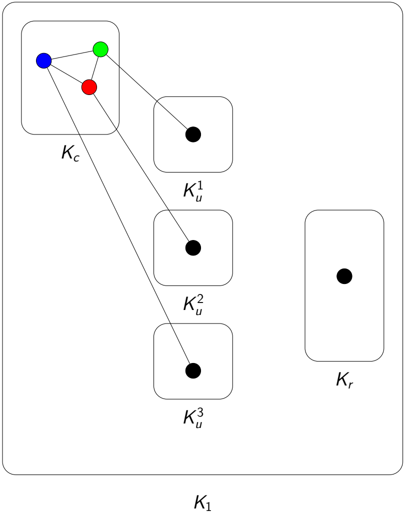


A Single Vertex Dominates K_r



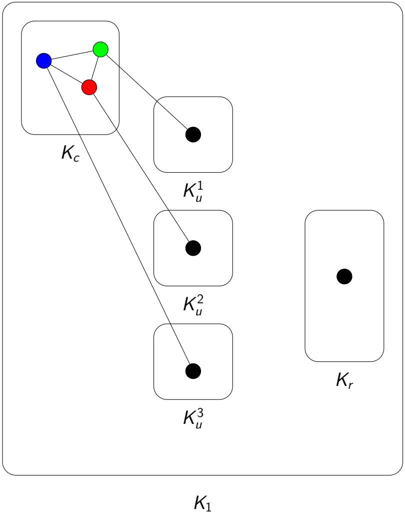
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique or C_5.

A Single Vertex Dominates K_r



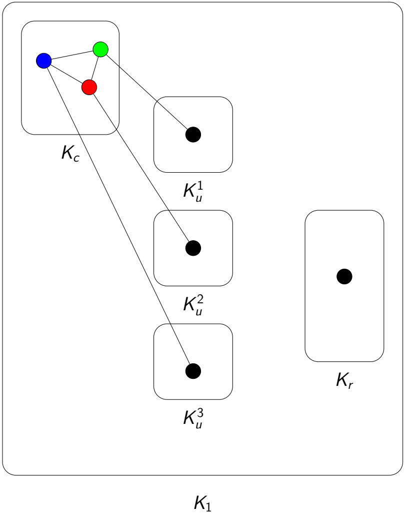
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.

A Single Vertex Dominates K_r



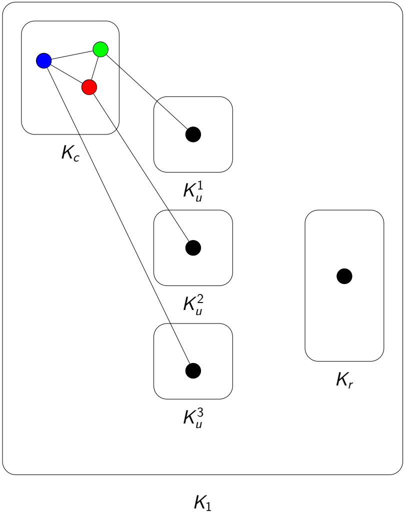
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.

A Single Vertex Dominates K_r



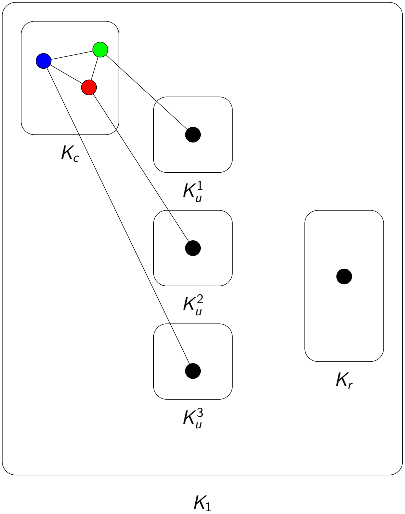
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.
- $|D| = 3$, assume $C = D$. $K_r = \emptyset$.

A Single Vertex Dominates K_r



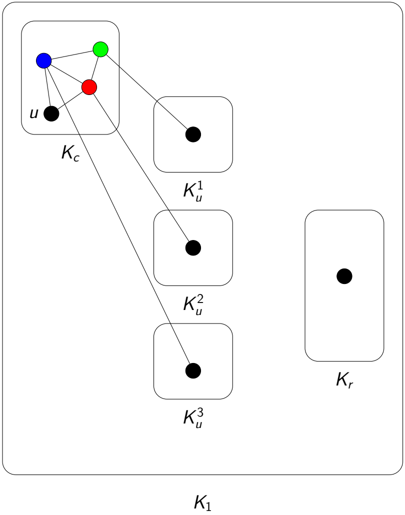
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.
- $|D| = 3$, assume $C = D$. $K_r = \emptyset$.
- $|D| = 1$, either:
  - $D \subseteq C$, then $K_r = \emptyset$,
  - $D \cap C = \emptyset$, then no 3-colouring.

A Single Vertex Dominates K_r



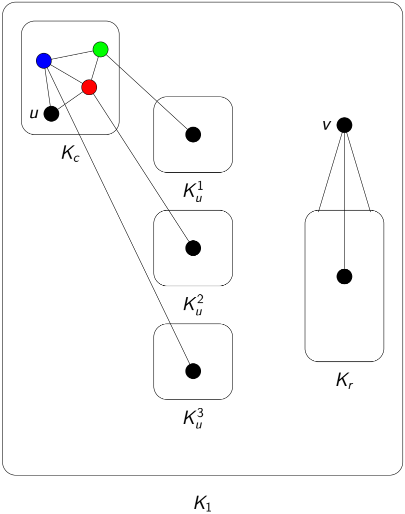
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.
- $|D| = 3$, assume $C = D$. $K_r = \emptyset$.
- $|D| = 1$, either:
  - $D \subseteq C$, then $K_r = \emptyset$,
  - $D \cap C = \emptyset$, then no 3-colouring.
- Let $D = \{u, v\}$. $N(u) \cap N(v) = \emptyset$.

A Single Vertex Dominates K_r



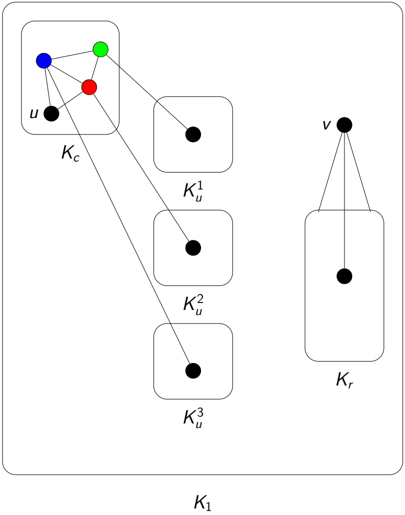
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.
- $|D| = 3$, assume $C = D$. $K_r = \emptyset$.
- $|D| = 1$, either:
  - $D \subseteq C$, then $K_r = \emptyset$,
  - $D \cap C = \emptyset$, then no 3-colouring.
- Let $D = \{u, v\}$. $N(u) \cap N(v) = \emptyset$.

A Single Vertex Dominates K_r



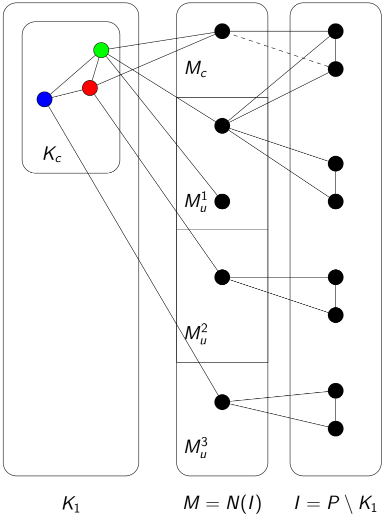
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.
- $|D| = 3$, assume $C = D$. $K_r = \emptyset$.
- $|D| = 1$, either:
  - $D \subseteq C$, then $K_r = \emptyset$,
  - $D \cap C = \emptyset$, then no 3-colouring.
- Let $D = \{u, v\}$. $N(u) \cap N(v) = \emptyset$.

A Single Vertex Dominates K_r



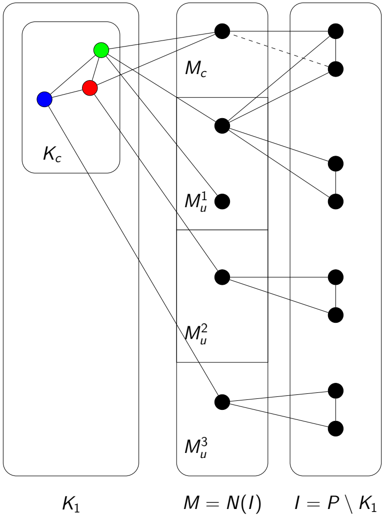
- $G[K_1]$ is a connected P_5-free graph.
- It has dominating clique D.
- $|D| \leq 3$, or no 3-colouring.
- $|D| = 3$, assume $C = D$. $K_r = \emptyset$.
- $|D| = 1$, either:
  - $D \subseteq C$, then $K_r = \emptyset$,
  - $D \cap C = \emptyset$, then no 3-colouring.
- Let $D = \{u, v\}$. $N(u) \cap N(v) = \emptyset$.
- $u \in K_c$, v is complete to K_r.
- Branch on the colour of v.

Dealing with I



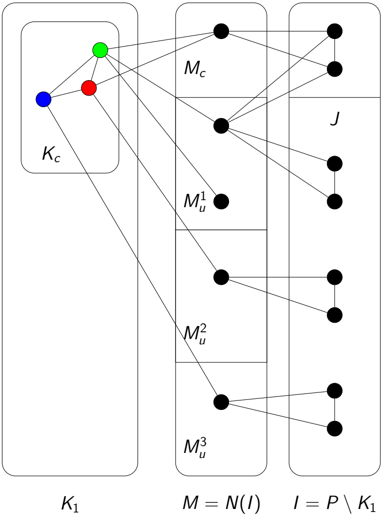
- Separate argument: wlog $M_r = \emptyset$.

Dealing with I



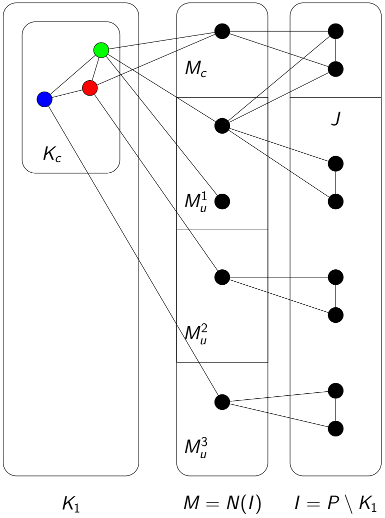
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c$.

Dealing with I



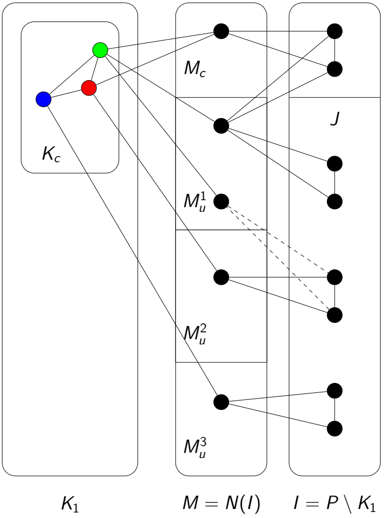
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.

Dealing with I



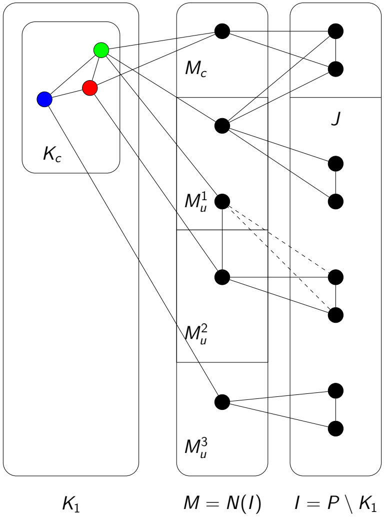
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.

Dealing with I



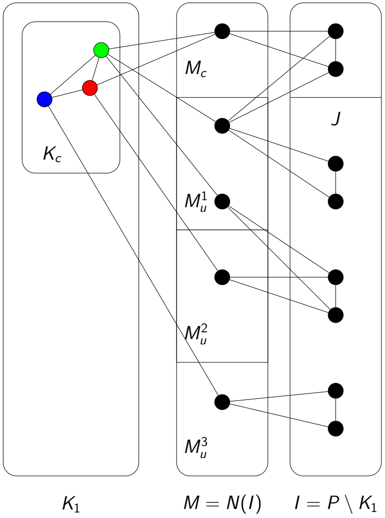
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.

Dealing with I



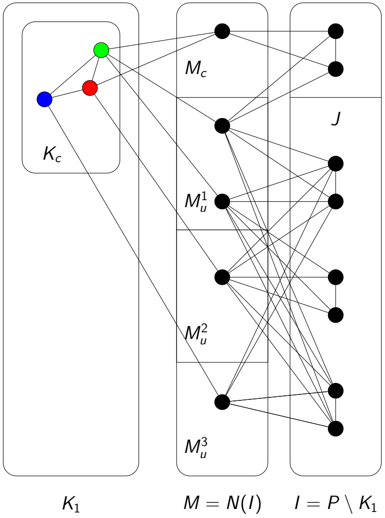
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.

Dealing with I



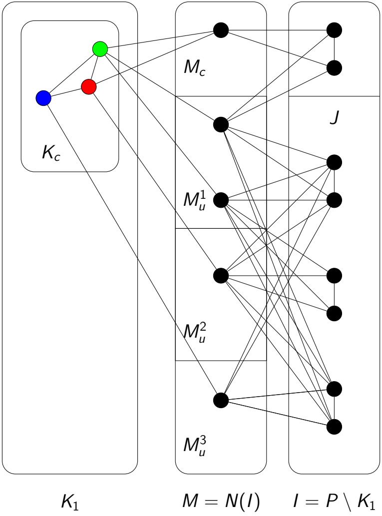
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.

Dealing with I



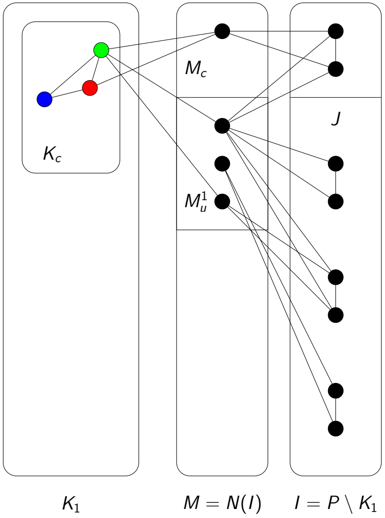
- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.

Dealing with I



- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.
  - Branch on the colour of v. Then every vertex has a coloured neighbour.

Dealing with I



- Separate argument: wlog $M_r = \emptyset$.
- Component in I is complete or anticompl. to $z \in M_c, M_u^1, M_u^2, M_u^3$.
- Let $M_u^i, M_u^j \neq \emptyset$.
  - Some $v \in M_u^1, M_u^2, M_u^3$ is complete to J.
  - Branch on the colour of v. Then every vertex has a coloured neighbour.
- Let $M_u^i, M_u^j = \emptyset$. Wlog $M_u^1 \neq \emptyset$.
  - In each $K' \in I$, some partite set gets colour 1.
  - $N(K')$ must receive same colour.
  - Add these constraints to 2-SAT, solve for $G - J$, and extend this colouring to J.

Discussion

Open Problems

- 3-COLOURING on probe H-free graphs when H is a linear forest. Five open cases, e.g. $H = P_5 + sP_1$, $P_4 + P_2 + sP_1$. We show $P_3 + sP_1$ is polynomial.

Discussion

Open Problems

- 3-Colouring on probe H-free graphs when H is a linear forest. Five open cases, e.g. $H = P_5 + sP_1$, $P_4 + P_2 + sP_1$. We show $P_3 + sP_1$ is polynomial.
- k-Colouring on probe P_5-free graphs for $k \geq 4$.

Discussion

Open Problems

- 3-Colouring on probe H-free graphs when H is a linear forest. Five open cases, e.g. $H = P_5 + sP_1$, $P_4 + P_2 + sP_1$. We show $P_3 + sP_1$ is polynomial.
- k-Colouring on probe P_5-free graphs for $k \geq 4$.
- Recognition of probe P_5-free graphs.

Discussion

Open Problems

- 3-Colouring on probe H-free graphs when H is a linear forest. Five open cases, e.g. $H = P_5 + sP_1$, $P_4 + P_2 + sP_1$. We show $P_3 + sP_1$ is polynomial.
- k-Colouring on probe P_5-free graphs for $k \geq 4$.
- Recognition of probe P_5-free graphs.

Results

- For a graph H, Colouring is polynomial-time solvable for probe H-free graphs if $H \subseteq_i P_4$; else, is NP-hard even for partitioned probe H-free graphs.
- For an integer $t \geq 1$, 3-Colouring on partitioned probe P_t-free graphs is polynomial-time solvable if $t \leq 5$ and NP-complete if $t \geq 6$.
- **Main Result:** 3-Colouring on partitioned probe P_5-free graphs is polynomial-time solvable.

Discussion

Open Problems

- 3-COLOURING on probe H-free graphs when H is a linear forest. Five open cases, e.g. $H = P_5 + sP_1$, $P_4 + P_2 + sP_1$. We show $P_3 + sP_1$ is polynomial.
- k-COLOURING on probe P_5-free graphs for $k \geq 4$.
- Recognition of probe P_5-free graphs.

Results

- For a graph H, COLOURING is polynomial-time solvable for probe H-free graphs if $H \subseteq_i P_4$; else, is NP-hard even for partitioned probe H-free graphs.
- For an integer $t \geq 1$, 3-COLOURING on partitioned probe P_t-free graphs is polynomial-time solvable if $t \leq 5$ and NP-complete if $t \geq 6$.
- **Main Result:** 3-COLOURING on partitioned probe P_5-free graphs is polynomial-time solvable.

More open problems and proofs: arXiv:2505.20784 and STACS proceedings.