Canonisation and Definability

For Graphs of Bounded Rank Width

Paper from Martin Grohe and Daniel Neuen

Roadmap

A brief overview

Results

Machinery

- *Foundation*
- *Logic – Algorithm – Game Theory*
- *Split and Flip*

Proof Idea

Contributions

The main ones at least...

Isomorphism Test:

- *Runtime:* $n^{O(k)}$ *from* $n^{f(k)}$

k-fixed-point logic with counting:

- *Captures polynomial time on graphs* $rw(k)$

Contributions

The main ones at least...

Isomorphism Test:

- *Runtime:* $n^{O(k)}$ *from* $n^{f(k)}$

k-fixed-point logic with counting:

- *Captures polynomial time on graphs* $rw(k)$

Contributions

Detail

Isomorphism Test

- *General: quasi-polynomial*
- *Polynomial of graph classes:*
  - *Bounded degree, tree width, ... bounded rank width*

Rank width almost invariant to complement

- *Dense graphs can have small rank width*

Contributions

Detail

$(3k+4)$-dim Weisfeiler-Leman algorithm identifies

- *graphs rank width at most k*

Isomorphism in $O(n^{3k+5} \log n)$, $rw(G) \leq k$

Sentence from C^{3k+5} characterises, $rw(G) \leq k$

Canonisation algorithm in $O(n^{3k+7} \log n)$, $rw(G) \leq k$

Foundations

Foundations

Graphs

Isomorphism

- $\varphi : V(G) \to V(H)$, *bijective and*
- $vw \in E(G) \Leftrightarrow \varphi(v)\varphi(w) \in E(H)$

Coloured Graph

- (G, χ) *where* $\chi\colon V(G) \to \mathcal{C}$

Foundations

Colourings

Colouring of k-tuple:

- $\chi\colon V(G)^k \to \mathcal{C}$

For (G, χ) and $\bar{v} = (v_1, \ldots, v_k)$

- $\chi^{\bar{v}} : V \to \mathbb{N}$
- $v \mapsto i \;\; \text{if } v{=}u_i \wedge \forall\, j > i \; v \neq v_j$
- $v \mapsto \chi(v) + k$

Foundations

Colourings

Refinement $\chi_1 , \chi_2 : V(G)^k \to \mathcal{C}$:

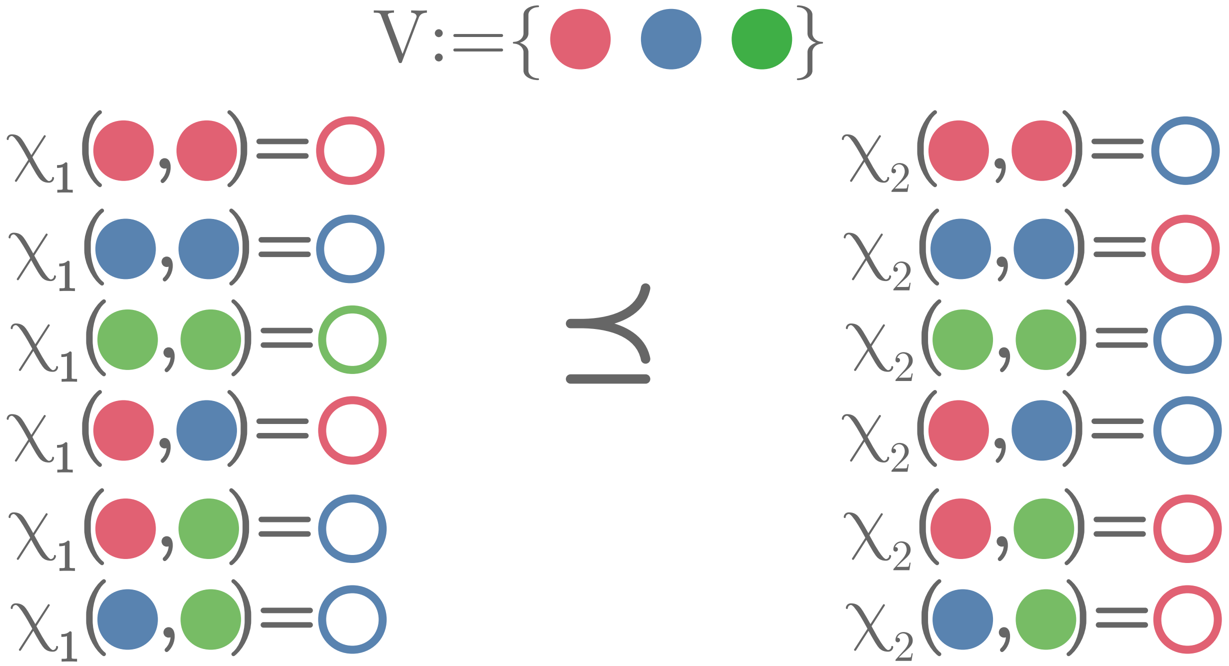
- $\chi_1 \preceq \chi_2$
- $\chi_1(\overline{v}) = \chi_1(\overline{w}) \Rightarrow \chi_2(\overline{v}) = \chi_2(\overline{w})$

Stable Colouring

- $\chi_\infty \text{ stable} \iff \forall \chi_2 \ \chi_2 \preceq \chi_\infty$

Foundations

Colourings: Example



$$\chi_1(v_1, v_2) = \chi_1(w_1, w_2) \Rightarrow \chi_2(v_1, v_2) = \chi_2(w_1, w_2)$$

Foundations

Canonisation

Graph Canonisation for a class $\mathcal{C}$

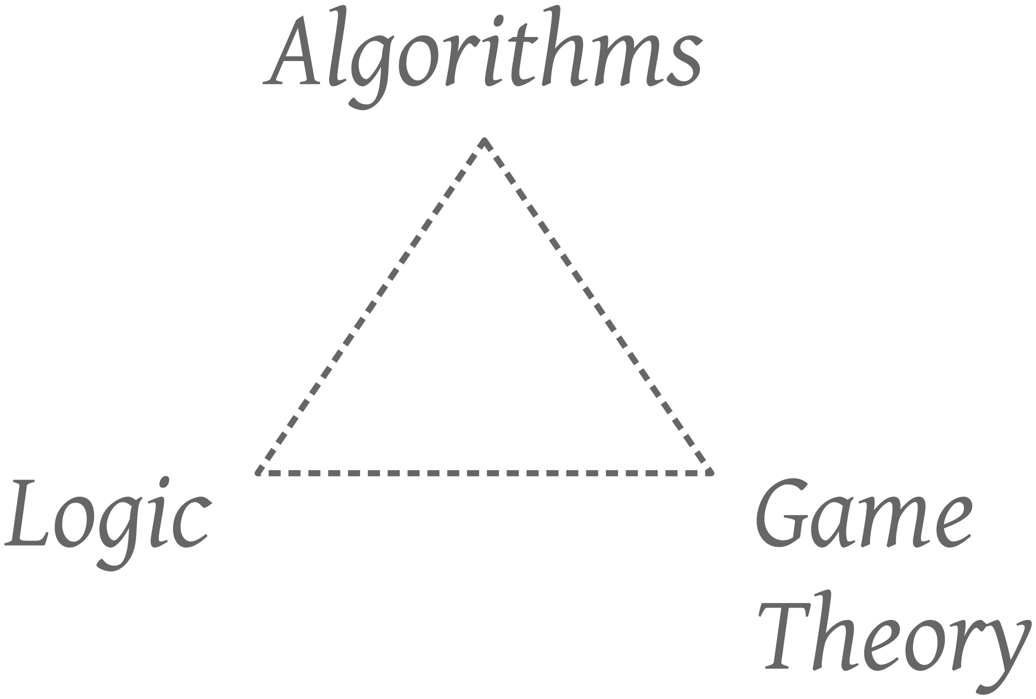
- $\kappa \colon \mathcal{C} \to \mathcal{C}$
- $\forall \mathrm{G} \in \mathcal{C},\ \kappa(\mathrm{G}) \cong \mathrm{G}$
- $\forall \mathrm{G}, \mathrm{H} \in \mathcal{C},\ \mathrm{G} \cong \mathrm{H} \implies \kappa(\mathrm{G}) = \kappa(\mathrm{H})$

Note that

- *Graph isomorphism for* $\mathcal{C} \leq_{\mathrm{P}}$ *graph canonisation for* $\mathcal{C}$

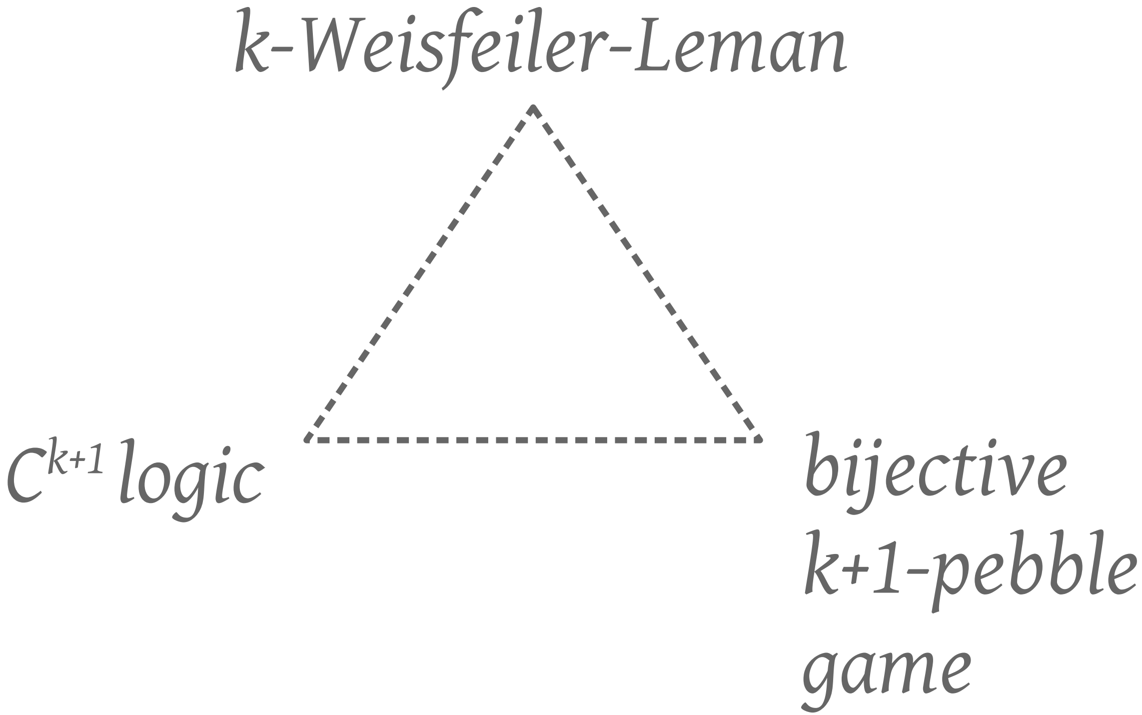
Weisfeiler-Leman

Correspondence



Correspondence

Specific



Algorithms

k-Weisfeiler-Leman

k-dimensional Weisfeiler Leman

- *Uses graph colouring*
- *Can establish non-isomorphism*
- *Not complete*

Idea:

- *Colour k-vertices*
- *Refine Colouring*

Algorithms

k-Weisfeiler-Leman

Notation:

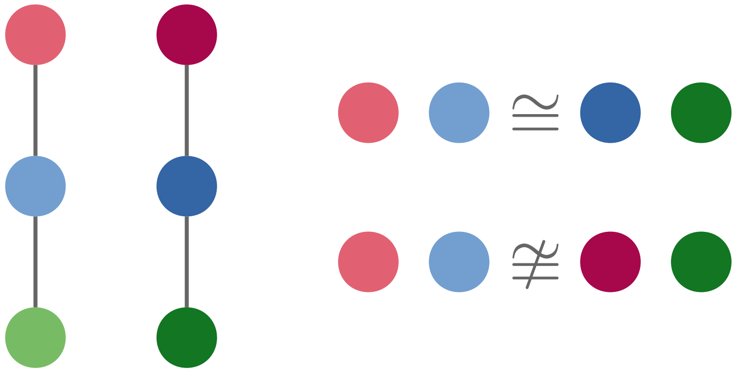
- $\overline{\mathrm{v}} \cong \overline{\mathrm{w}}$, *if* $\varphi : \mathrm{G}[\ \overline{\mathrm{v}}\ ] \to \mathrm{H}[\ \overline{\mathrm{w}}\ ]$, $\mathrm{v}_i \mapsto \mathrm{w}_i$ *isomorphism*

Algorithms

k-Weisfeiler-Leman

Notation:

- $\overline{v} \cong \overline{w}$, *if* $\varphi : G[\,\overline{v}\,] \to H[\,\overline{w}\,],\ v_i \mapsto w_i$ *isomorphism*

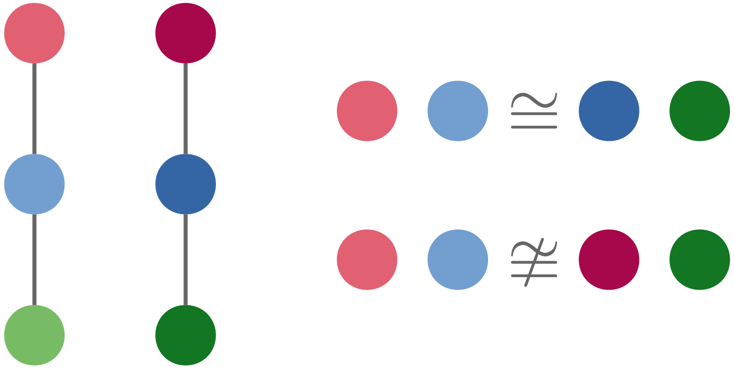


Algorithms

k-Weisfeiler-Leman

Notation:

- $\bar{v} \cong \bar{w}$, *if* $\varphi : G[\,\bar{v}\,] \to H[\,\bar{w}\,],\ v_i \mapsto w_i$ *isomorphism*
- $\bar{v}$ *and* $\bar{w}$ *i-neighbours,* *if* $v_j = w_j$ *for all* $j \neq i$

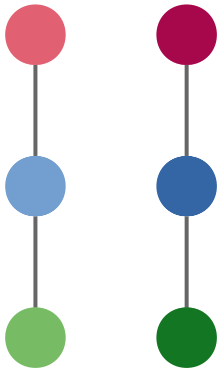


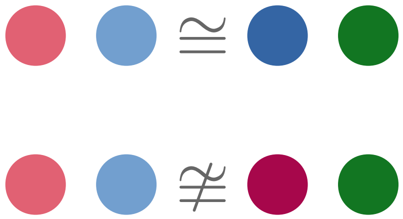
Algorithms

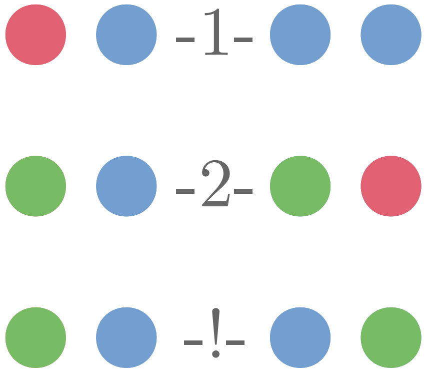
k-Weisfeiler-Leman

Notation:

- $\overline{v} \cong \overline{w}$, *if* $\varphi : G[\,\overline{v}\,] \rightarrow H[\,\overline{w}\,]$, $v_i \mapsto w_i$ *isomorphism*
- $\overline{v}$ *and* $\overline{w}$ *i-neighbours, if* $v_j = w_j$ *for all* $j \neq i$







Algorithms

k-Weisfeiler-Leman

Initialisation:

- $\overline{v}, \overline{w}$ *different colours if* $\overline{v} \ncong \overline{w}$

Step:

- $\overline{v}, \overline{w}$ *different colours if*
- $\exists\ i \leq k\ \exists\ c \in \mathcal{C}$ *different number of i-neighbours of colour* c

$\Rightarrow (G, \chi_\infty)$ stable, in $O(n^{k+1} \log n)$

Algorithms

k-Weisfeiler-Leman

$G \simeq_k H$ if and only if $\forall c \in \mathcal{C}\ |G_c| = |H_c|$

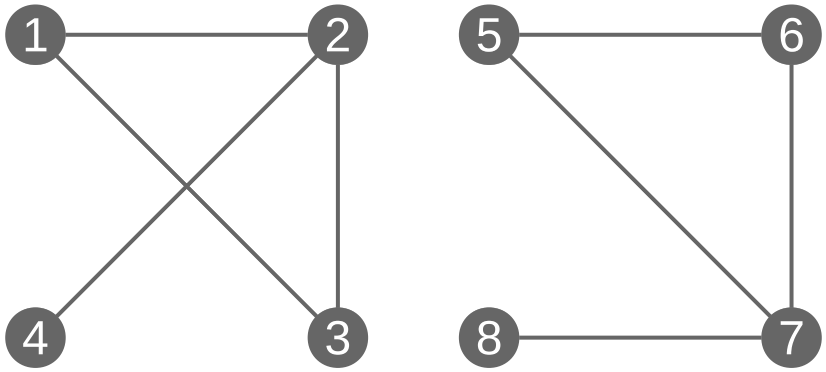
- $G_c := \{\ \overline{v} \mid \overline{v} \in V^k(G),\ \chi_\infty(\overline{v}) = c\ \}$
- $H_c := \{\ \overline{w} \mid \overline{w} \in V^k(H),\ \chi_\infty(\overline{w}) = c\ \}$

$G \not\simeq_k H$ if and only if $\exists c \in \mathcal{C}\ |G_c| \neq |H_c|$

Identifies $G :\Longleftrightarrow \forall\ H\ ,\ G \not\cong H \Rightarrow G \not\simeq_k H$

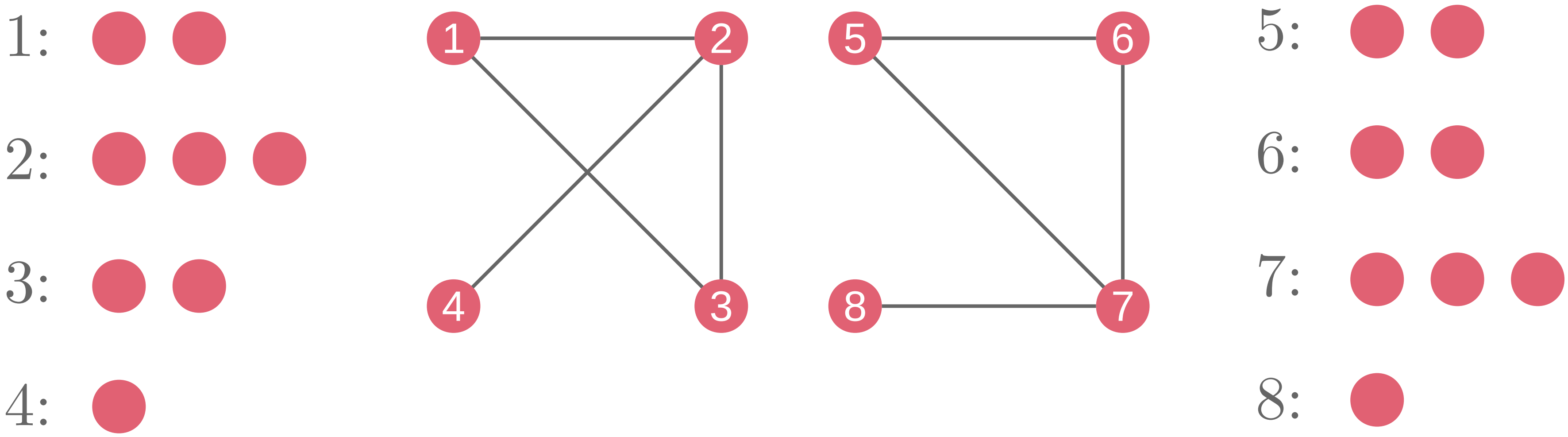
Algorithms

k-Weisfeiler-Leman: Example



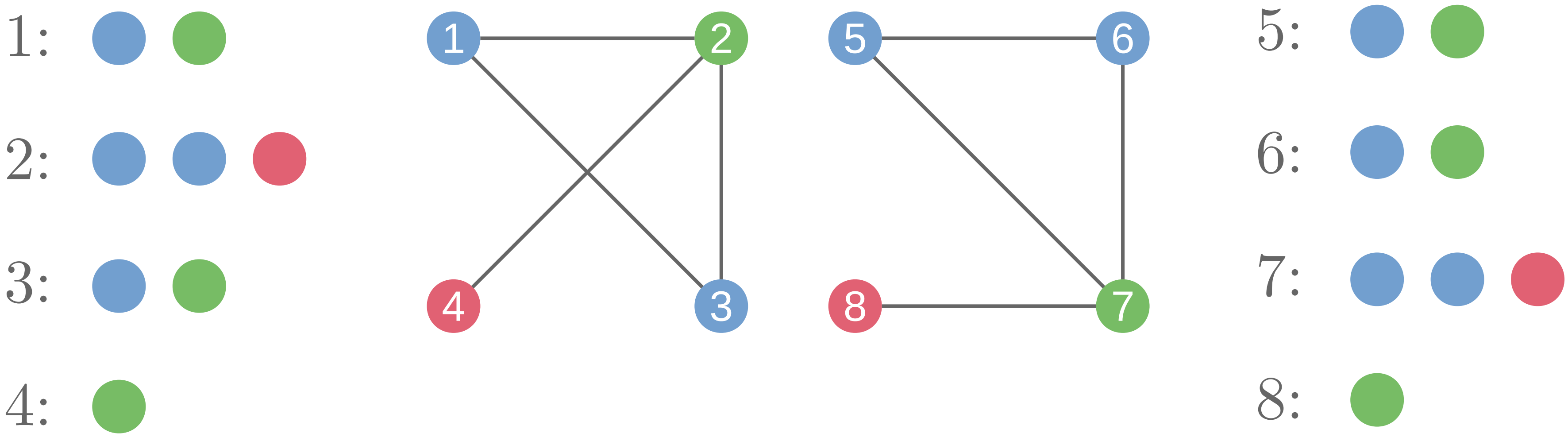
Algorithms

k-Weisfeiler-Leman: Example



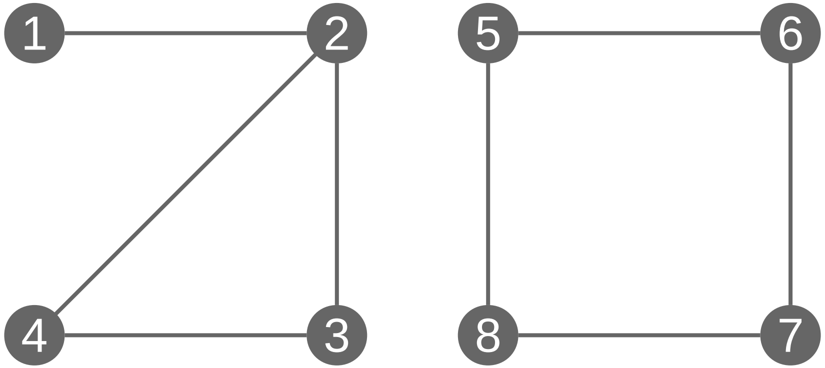
Algorithms

k-Weisfeiler-Leman: Example



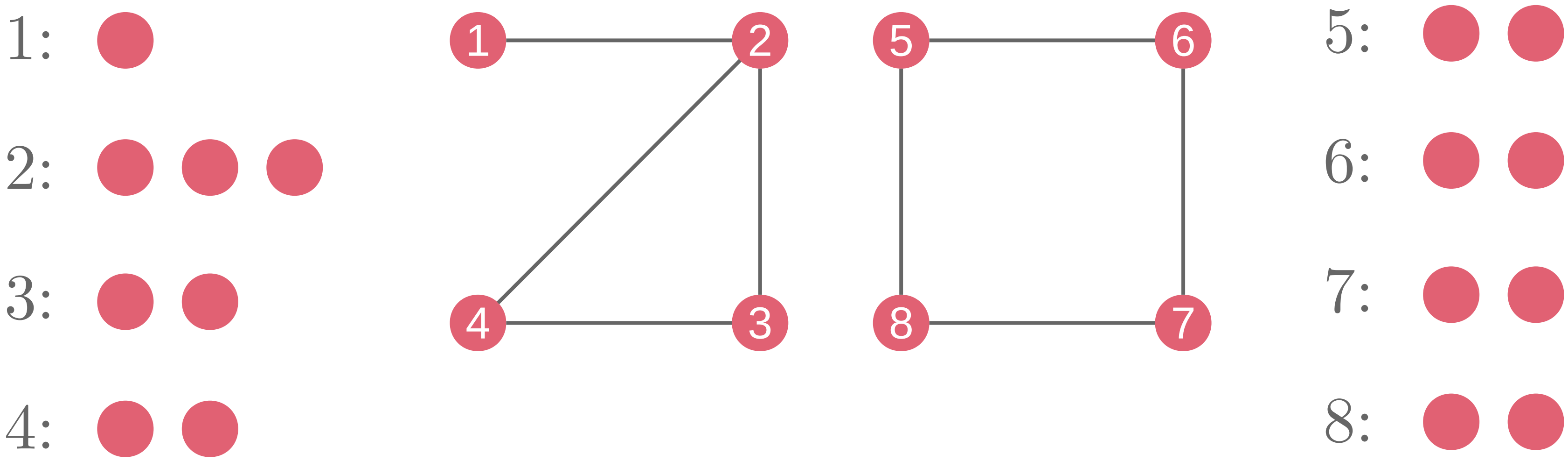
Algorithms

k-Weisfeiler-Leman: Example



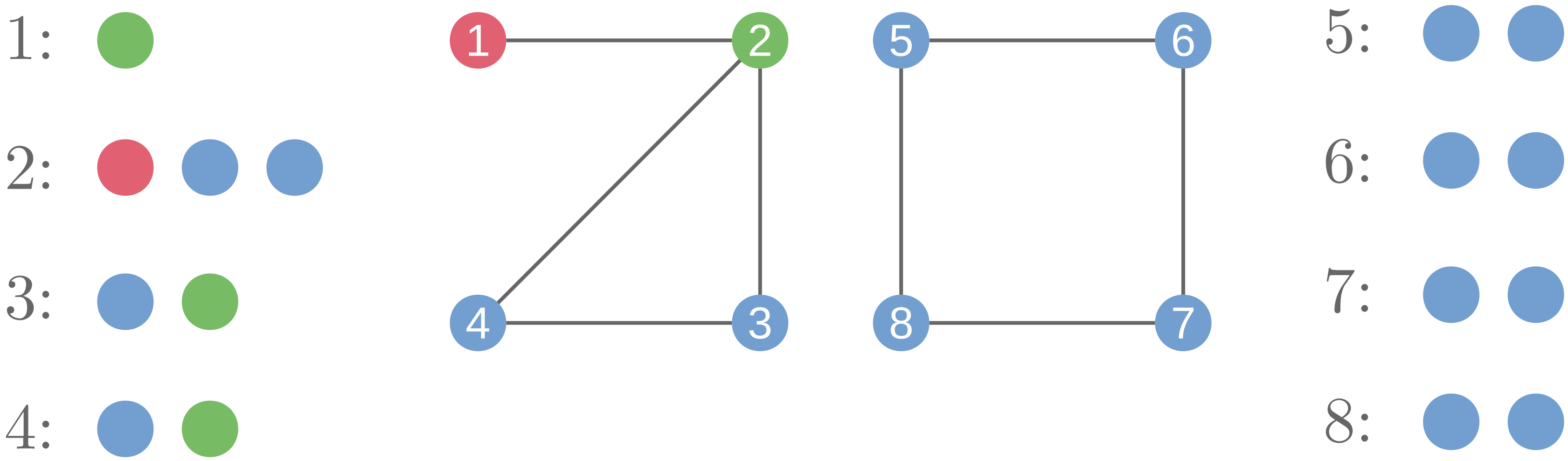
Algorithms

k-Weisfeiler-Leman: Example



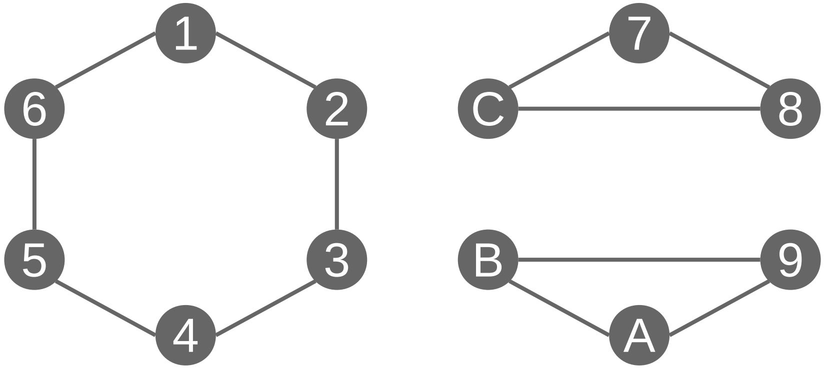
Algorithms

k-Weisfeiler-Leman: Example



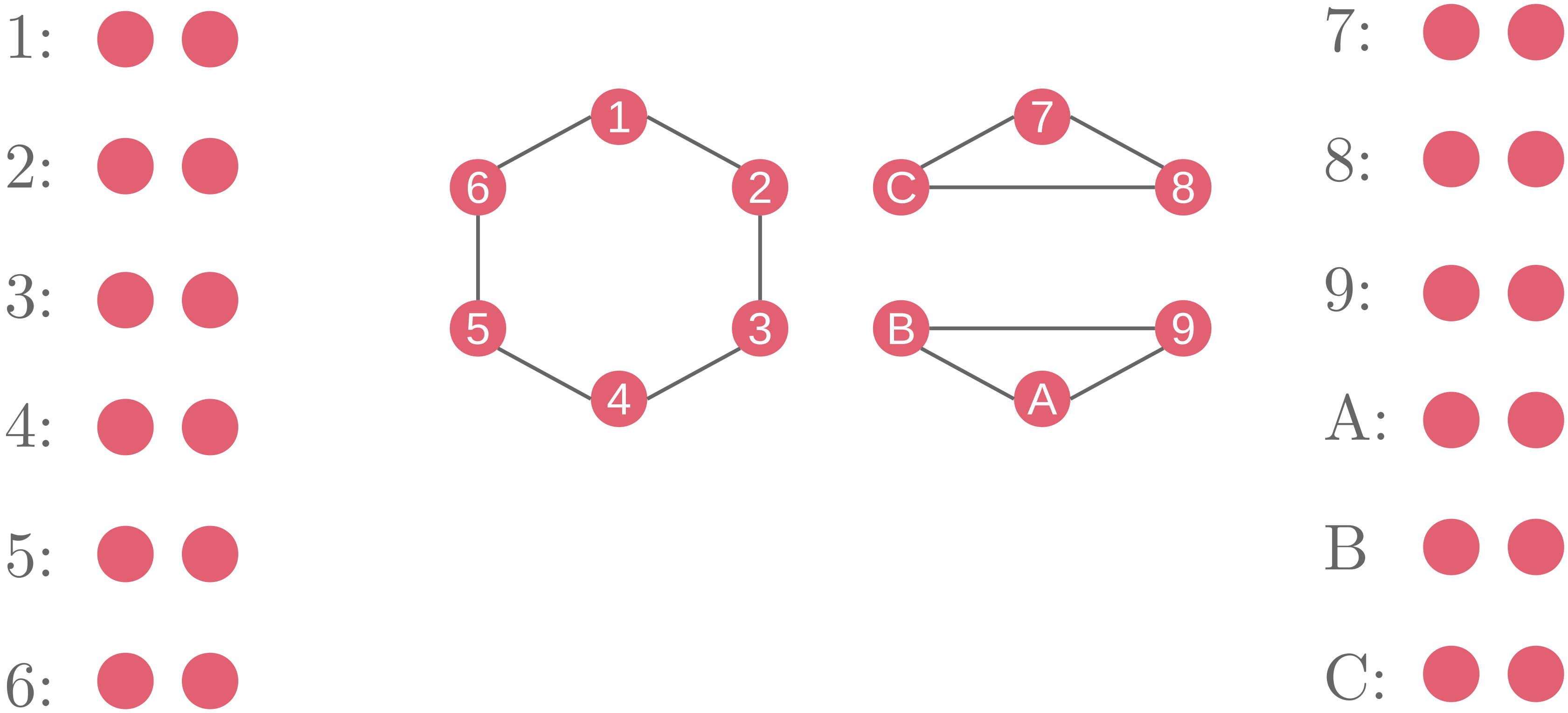
Algorithms

k-Weisfeiler-Leman: Example



Algorithms

k-Weisfeiler-Leman: Example



Algorithms

k-Weisfeiler-Leman: Example

1:

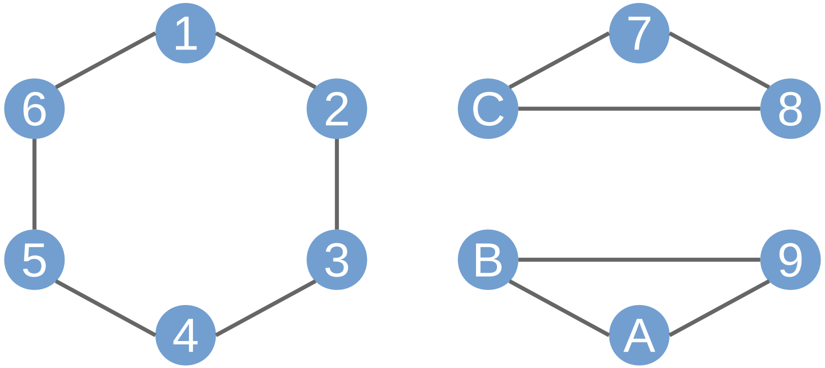
2:

3:

4:

5:

6:



7:

8:

9:

A:

B

C:

Game Theory

Bijective k-pebble game

Setup:

- *Spoiler and Duplicator*
- *Start Position:*
  - $((),())$
- *Rounds:*
  - $(\overline{v},\overline{w})$ *where* $\overline{v} \in E(G)^l$ *and* $\overline{w} \in E(H)^l$, $0 \leq l \leq k$

Game Theory

Bijective k-pebble game

Position $(\overline{v},\overline{w})=((v_1,\ .\ .\ .\ ,\ v_l),(w_1,\ .\ .\ .\ ,\ w_l))$

- *Spoiler chooses move*
  - *Remove pebble (if $l > 0$)*
  - *Add pebble (if $l < k$)*

Game Theory

Bijective k-pebble game

Remove move:

- *Spoiler picks* $\mathrm{i} \in [1,\ldots,\mathrm{l}]$
- *Next round starts with*
  - $\overline{\mathrm{v}}=\left(\mathrm{v}_1, \ldots, \mathrm{v}_{\mathrm{i}-1}, \mathrm{v}_{\mathrm{i}+1}, \ldots, \mathrm{v}_{\mathrm{l}}\right)$
  - $\overline{\mathrm{w}}=\left(\mathrm{w}_1, \ldots, \mathrm{w}_{\mathrm{i}-1}, \mathrm{w}_{\mathrm{i}+1}, \ldots, \mathrm{w}_{\mathrm{l}}\right)$

Game Theory

Bijective k-pebble game

Add move:

- *Duplicator picks bijection* $\varphi : V(G) \to V(H)$
- *Spoiler chooses* $v \in V(G)$ *and sets* $w = \varphi(v)$
- *Next round starts with*
  - $\overline{v} = (v_1, \ldots, v_l, v)$
  - $\overline{w} = (w_1, \ldots, w_l, w)$

Game Theory

Bijective k-pebble game

Who wins $((v_1, \ldots, v_l),(w_1, \ldots, w_l))$:

- *Spoiler wins if :*
  - $\exists i \in [1,\ldots,l]\ v_i = \perp \not\Leftrightarrow w_i = \perp$
  - $\exists i \in [1,\ldots,l]\ \chi_G(v_i) \neq \chi_H(w_i)$
  - $\exists i, j \in [1,\ldots,l]\ v_i = v_j \not\Leftrightarrow w_i = w_j$
  - $\exists i, j \in [1,\ldots,l]\ v_i v_j \in E(G) \not\Leftrightarrow w_i w_j \in E(H)$
- *Duplicator wins if the game never ends*

Game Theory

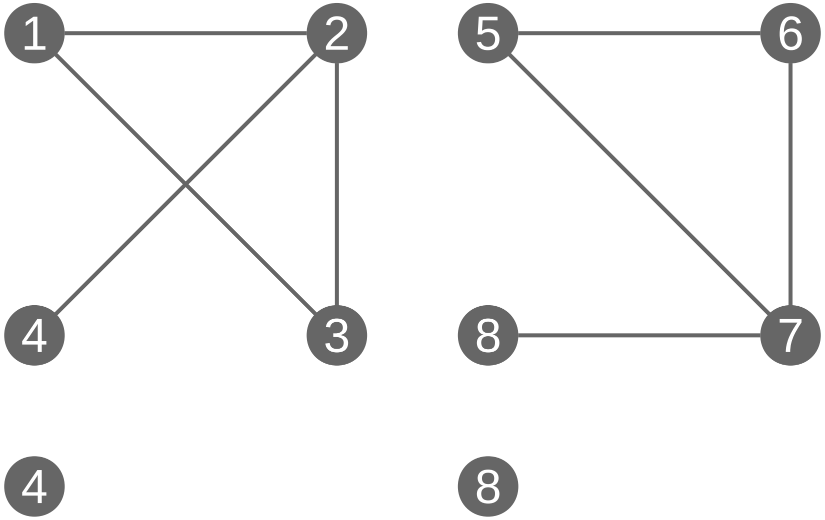
Bijective k-pebble game

1 $\mapsto$ 5

2 $\mapsto$ 6

3 $\mapsto$ 7

4 $\mapsto$ 8



Game Theory

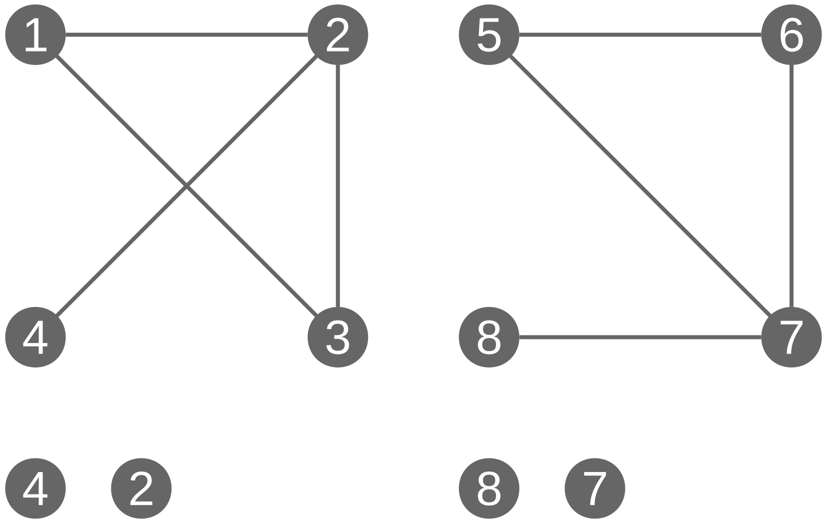
Bijective k-pebble game

$1 \mapsto 5$

$2 \mapsto 7$

$3 \mapsto 6$

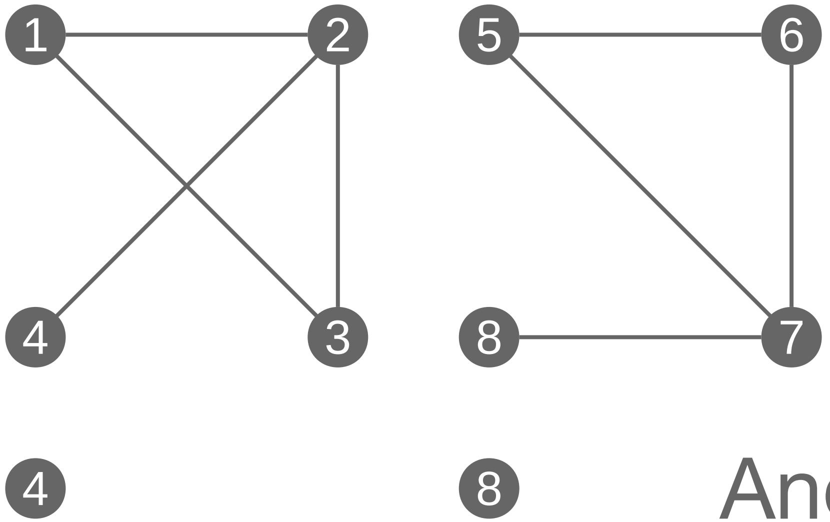
$4 \mapsto 8$



Game Theory

Bijective k-pebble game

1 ↦ 5
2 ↦ 7
3 ↦ 6
4 ↦ 8



And so on….

Game Theory

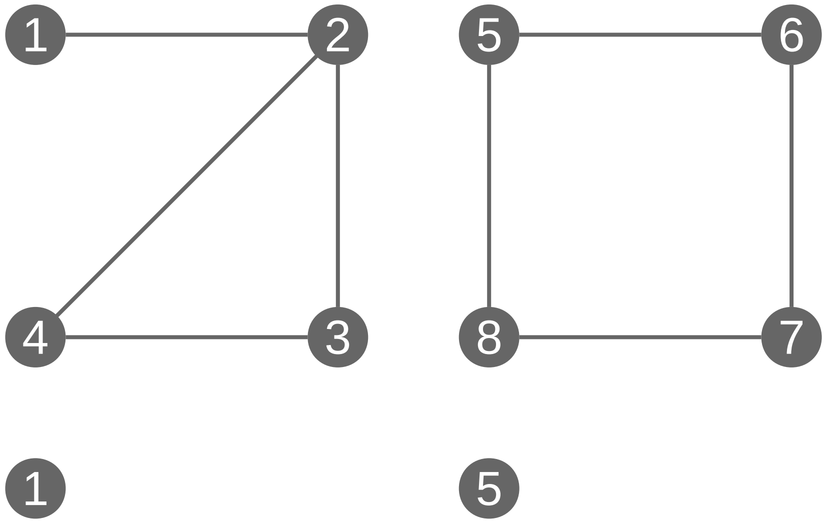
Bijective k-pebble game

$1 \mapsto 5$

$2 \mapsto 7$

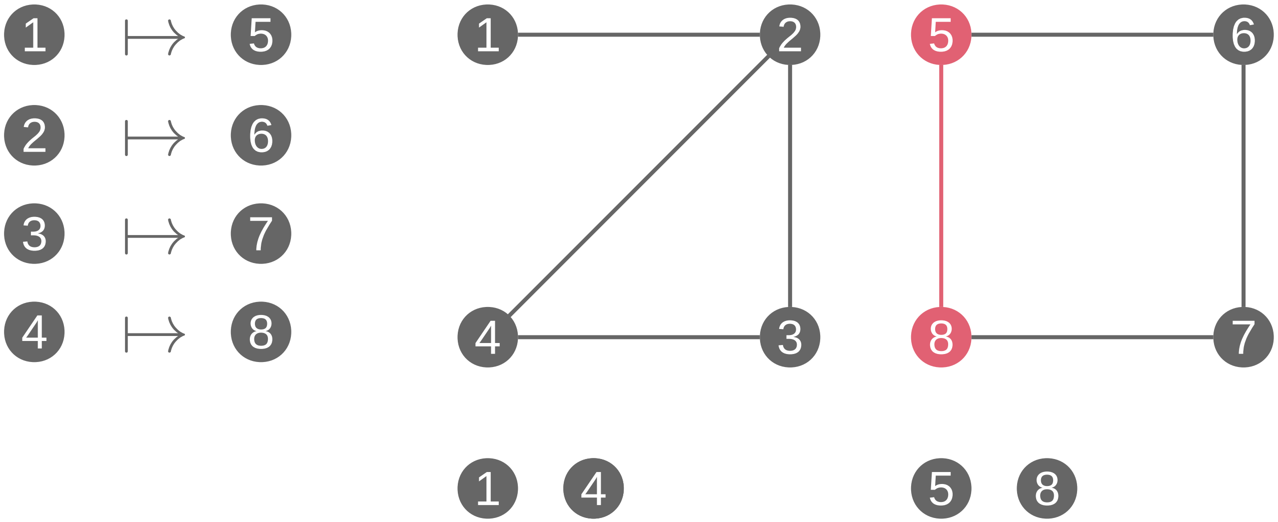
$3 \mapsto 6$

$4 \mapsto 8$



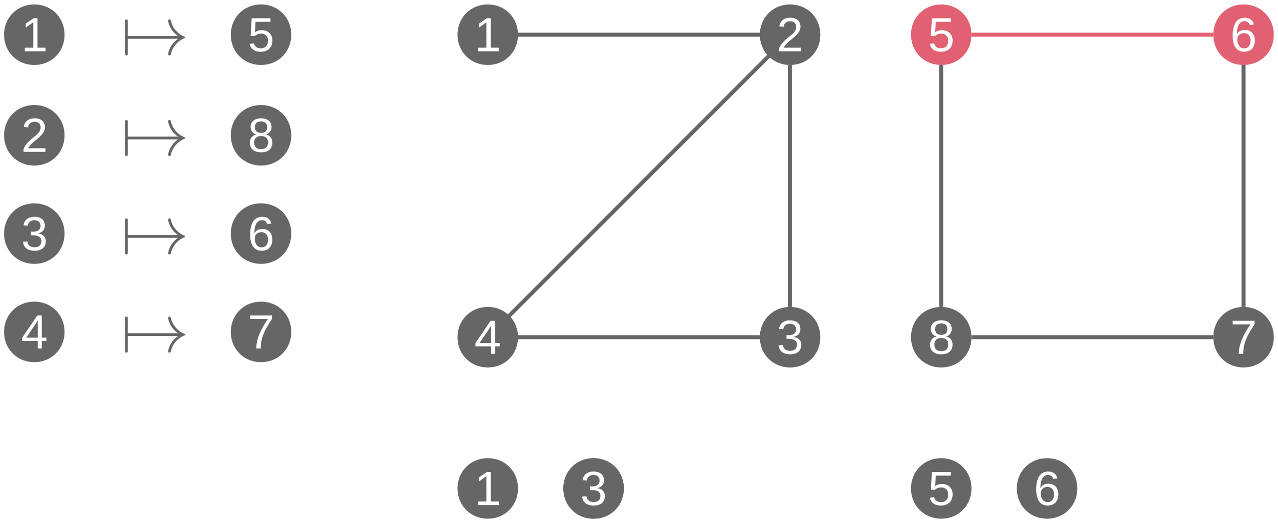
Game Theory

Bijective k-pebble game



Game Theory

Bijective k-pebble game



Game Theory

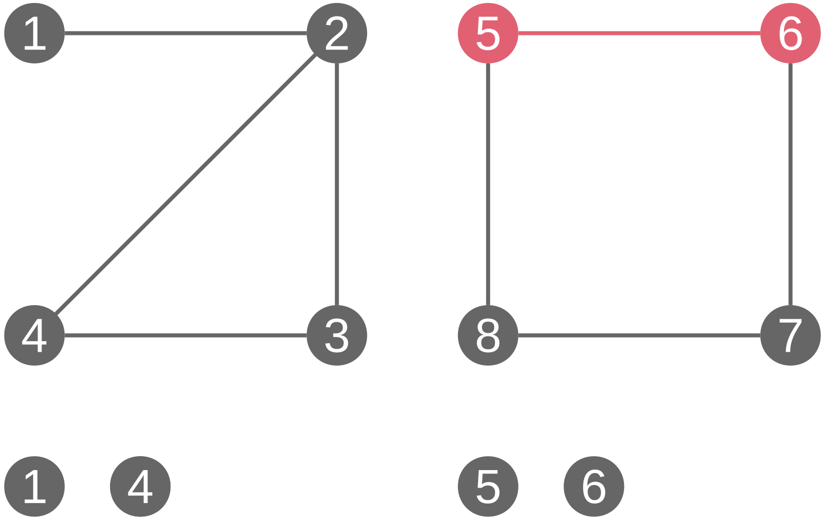
Bijective k-pebble game

1 ↦ 5

2 ↦ 7

3 ↦ 8

4 ↦ 6



Game Theory

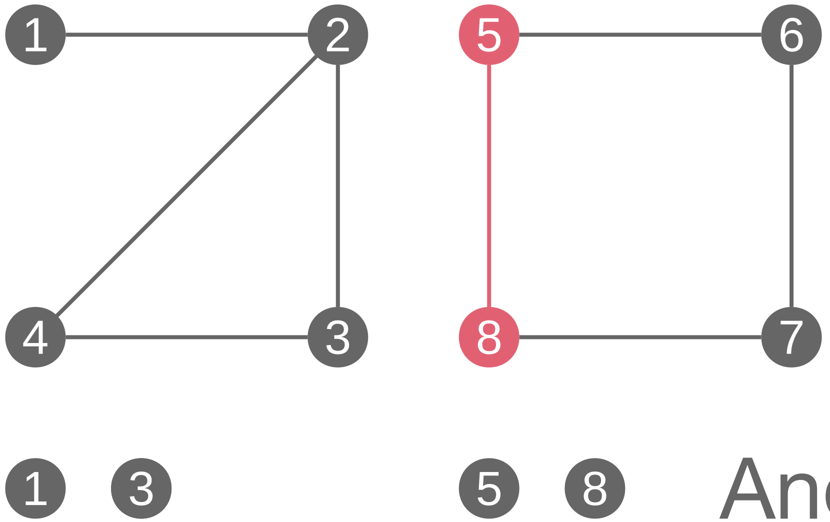
Bijective k-pebble game

$1 \mapsto 5$

$2 \mapsto 6$

$3 \mapsto 8$

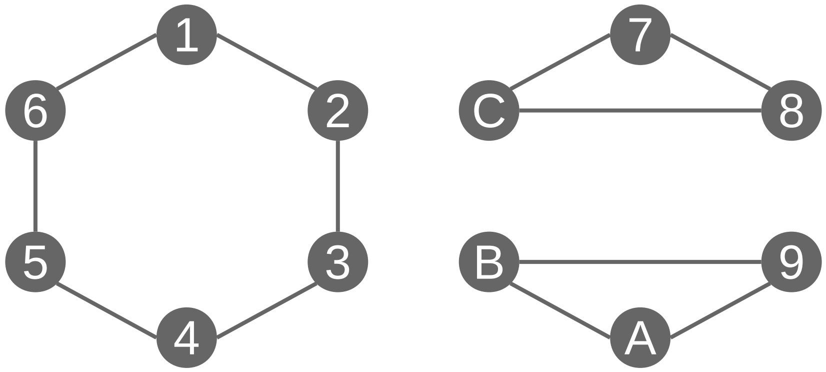
$4 \mapsto 7$



And so on….

Game Theory

Bijective k-pebble game : k= 2



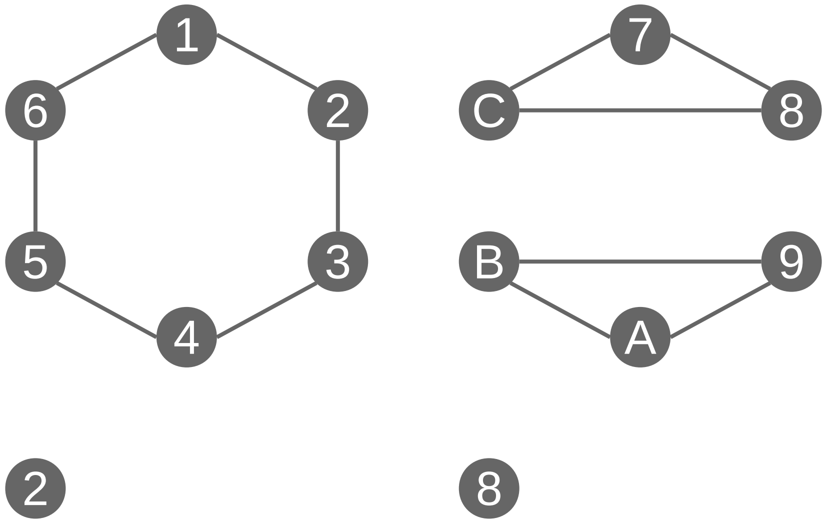
Game Theory

Bijective k-pebble game : k= 2

1 $\mapsto$ 7

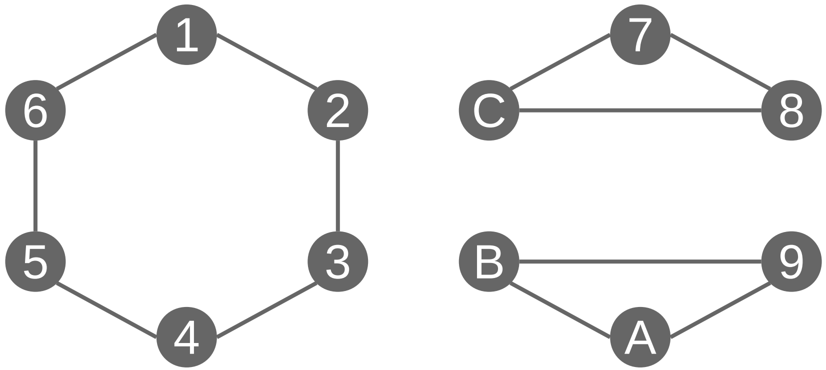
2 $\mapsto$ 8

3 $\mapsto$ C



Game Theory

Bijective k-pebble game : k= 3



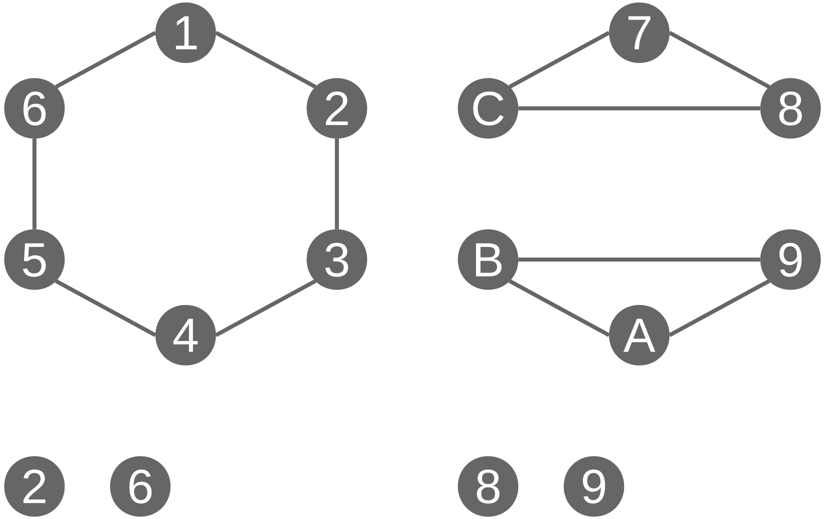
Game Theory

Bijective k-pebble game : k= 3

1 $\mapsto$ 8

6 $\mapsto$ 9

1 $\mapsto$?



Logic

C^{k+1} logic

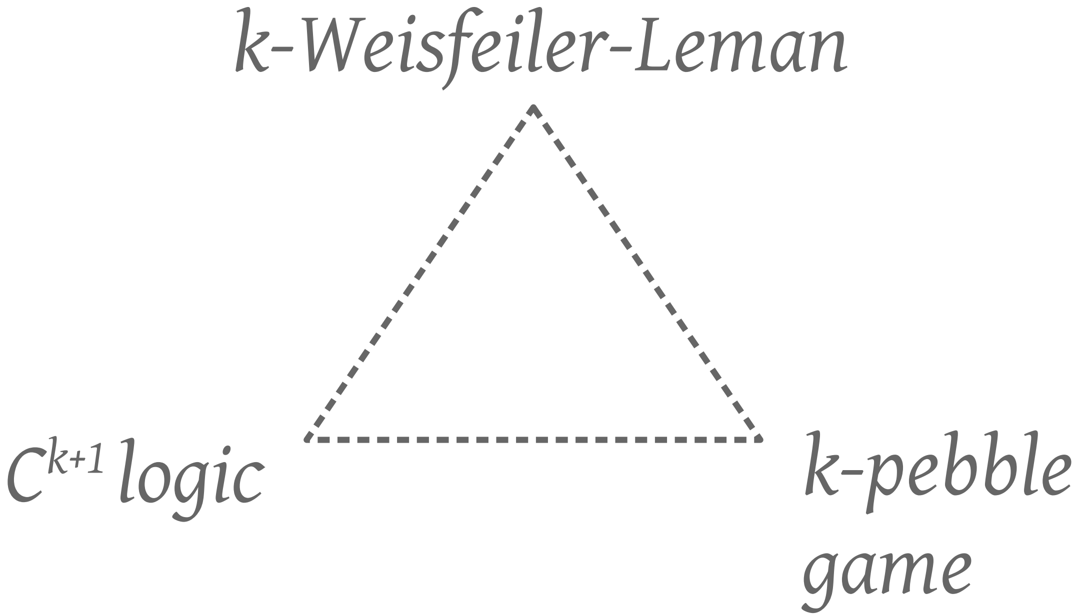
C^{k+1} logic

- *first-order Logic*
- *Counting quantifier, e.g.* $\exists^{\geq i} x \ \varphi(x)$
- *k-variable fragment*

$$G \equiv_{C^{k+1}} H :\Longleftrightarrow \forall \varphi \in C^{k+1} \ G \models \varphi \Leftrightarrow H \models \varphi$$

Correspondence

Specific



Correspondence

Theorems

For the graphs G and H

- $G \simeq_k H \iff$ *Duplicator wins* $BP_{k+1}(G,H)$.
- $G \simeq_k H \iff G \equiv_{C^{k+1}} H$
- G *(k)-identified* $\Longrightarrow \varphi_G \in C^{k+1}$ *characterises* G

Split and Flip

Why?

General Idea

For $l \in O(k)$, l-WLA identifies G if $rw(G) \leq k$

- *For* $X \subseteq V(G)$, *s.t.* $\rho_G(X) \leq k$
- *Pebbling splits* G *in* C
  - $C \subseteq X$ *or* $C \subseteq \overline{X}$
  - *independent*

Split Pair

Definition

(A,B) (ordered) split pair for $X \subseteq V(G)$

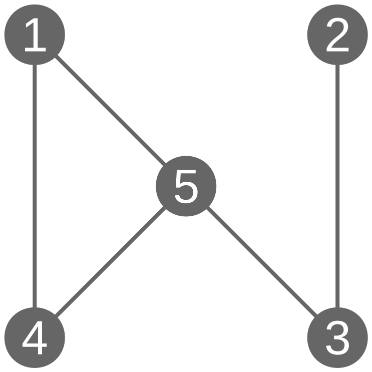
- $A \subseteq X$ *and* $B \subseteq \overline{X}$ $(A,B$ *tuple*$)$
- $\mathrm{vec}_X(A)$ *basis of* $\langle \mathrm{vec}_X(X) \rangle$
- $\mathrm{vec}_{\overline{X}}(B)$ *basis of* $\langle \mathrm{vec}_{\overline{X}}(\overline{X}) \rangle$

And

- $v \in X, \quad \mathrm{vec}_X(v) = (a_{vw})_{w \in \overline{X}}$ with $a_{vw} = 1 \Leftrightarrow vw \in E(G)$

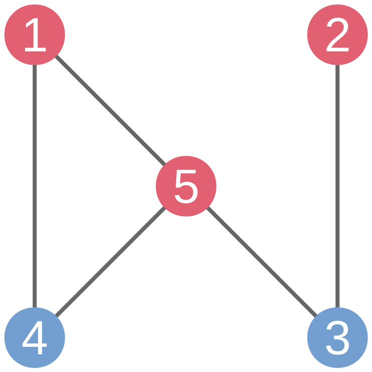
Split Pair

Example



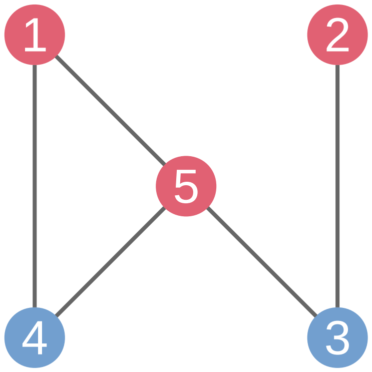
Split Pair

Example



Split Pair

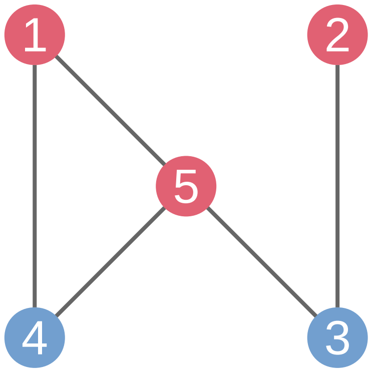
Example



	3	4
1	0	1
2	1	0
5	1	1

Split Pair

Example

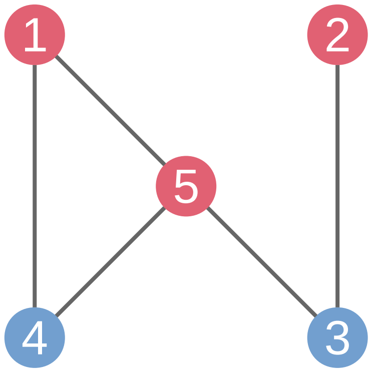


	3	4
1	0	1
2	1	0
5	1	1

	1	2	5
3	0	1	1
4	1	0	1

Split Pair

Example



	3	4
1	0	1
2	1	0
5	1	1

	1	2	5
3	0	1	1
4	1	0	1

Flip Function

Definition

For $\chi : V \to \mathcal{C}$, f: $\mathcal{C} \times \mathcal{C} \to \{0,1\}$ is a flip function

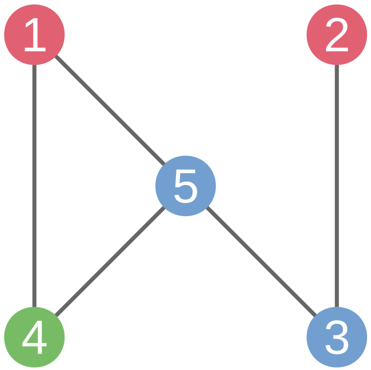
- *If* $f(c_1, c_2) = f(c_2, c_1)$ *for all* $c_1, c_2 \in \mathcal{C}$

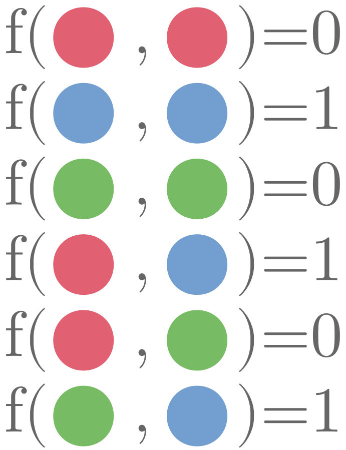
$G^f = (V, E^f, \chi)$ is a flipped graph of G where

- $E^f := \{vw \mid vw \in E(G) \wedge f(\chi(v), \chi(w)) = 0\} \cup$
  $\{vw \mid v \neq w \wedge vw \notin E(G) \wedge f(\chi(v), \chi(w)) = 1\}$

Flip Function

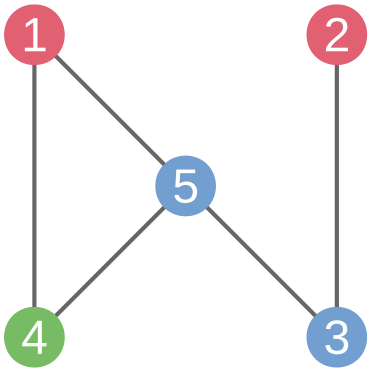
Example

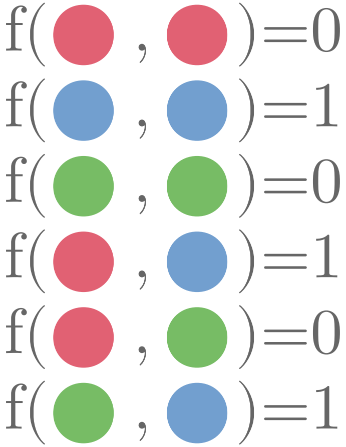


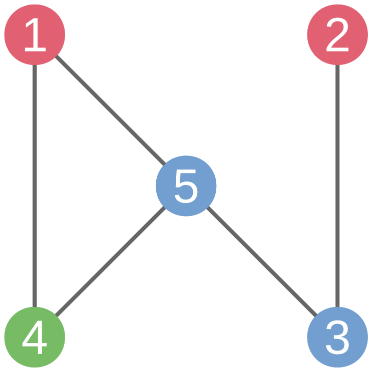


Flip Function

Example





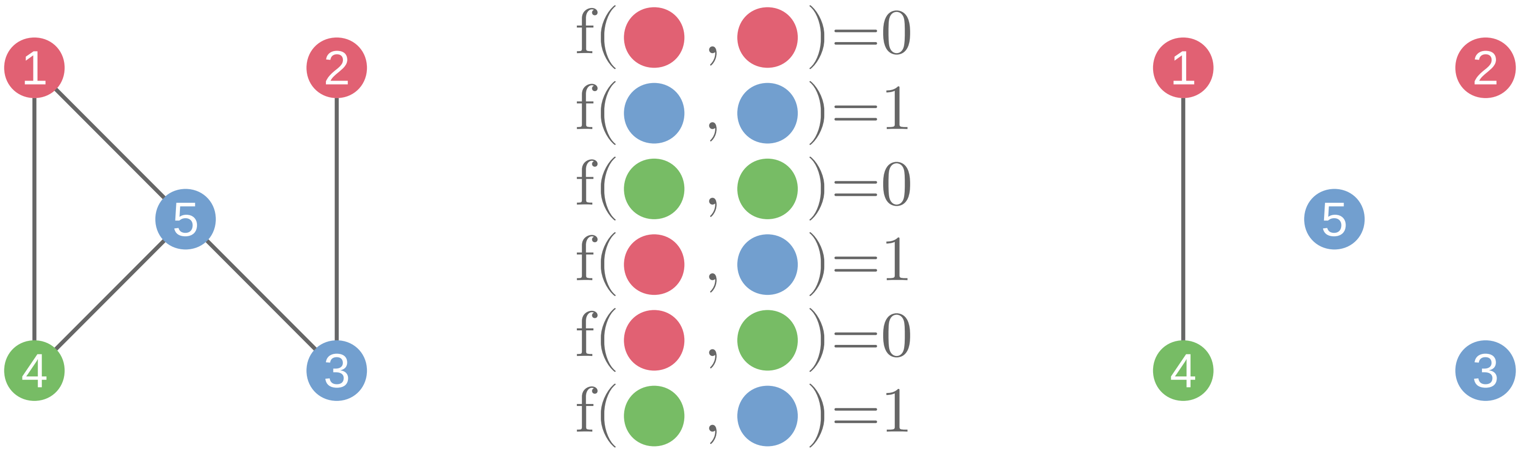


$$\{vw \mid vw \in E(G) \wedge f(\chi(v),\chi(w))=0\}$$

$$\{vw \mid v \neq w \wedge vw \notin E(G) \wedge f(\chi(v),\chi(w))=1\}$$

Flip Function

Example

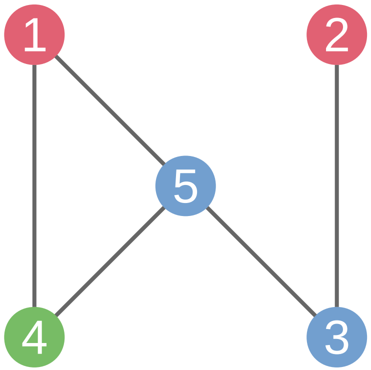


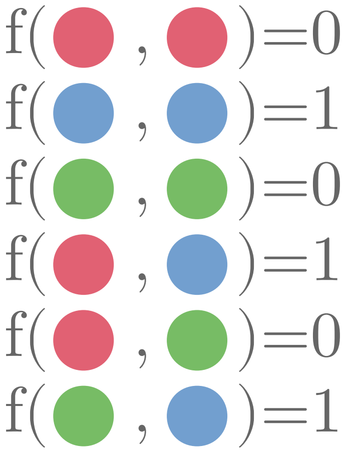
$$\{vw \mid vw \in E(G) \wedge f(\chi(v),\chi(w))=0\}$$

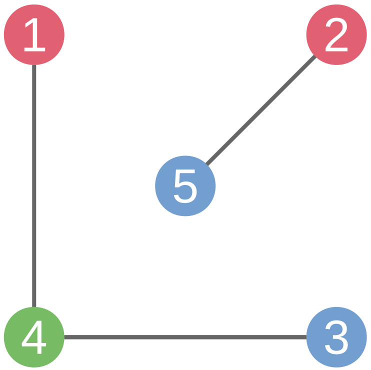
$$\{vw \mid v \neq w \wedge vw \notin E(G) \wedge f(\chi(v),\chi(w))=1\}$$

Flip Function

Example







$$\{vw \mid vw \in E(G) \wedge f(\chi(v),\chi(w))=0\}$$

$$\{vw \mid v \neq w \wedge vw \notin E(G) \wedge f(\chi(v),\chi(w))=1\}$$

Flip Function

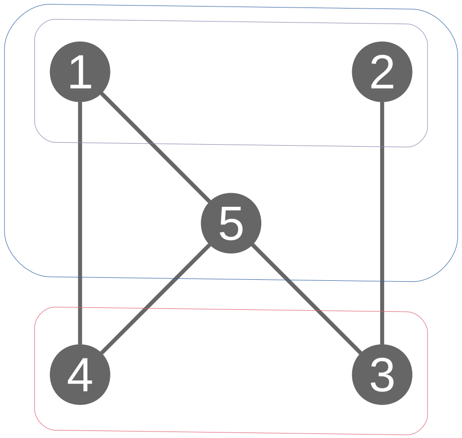
Theorem

There exists a flip function f for $G' := (G,\ \chi_\infty^{(\overline{a},\overline{b})})$

- $C \in \mathrm{Comp}(G', f)$ s.t. $C \subseteq X$ or $C \subseteq \overline{X}$

Flip Function

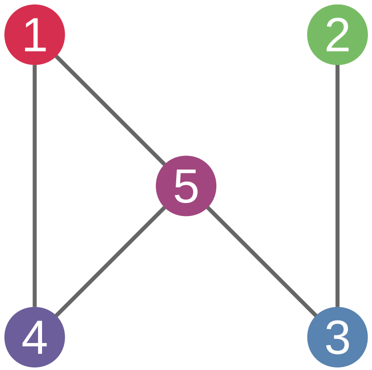
Example



$$X^{((1,2),(3,4)\ )} \Rightarrow X_{\infty}^{((1,2),(3,4)\ )}$$

Flip Function

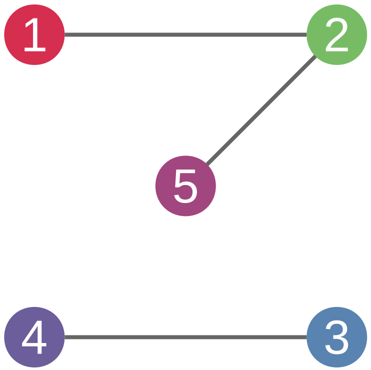
Example



$f(x,y) :=$

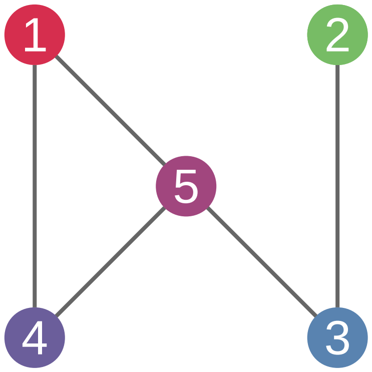
$\bullet\,\bullet \mapsto 0$

$\text{else} \mapsto 1$



Flip Function

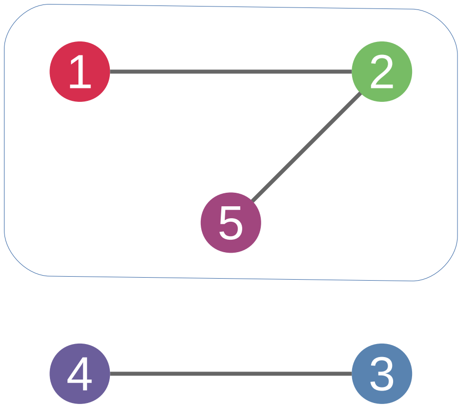
Example



$$f(x,y) :=$$

$$\mapsto 0$$

$$\text{else} \mapsto 1$$



Flip Function

Theorem

φ: $\mathrm{V(G)} \mapsto \mathrm{V(H)}$, bijection

- φ: $\mathrm{G} \cong \mathrm{H} \Leftrightarrow \mathrm{G^f} \cong \mathrm{H^f}$

Similarly,

- *Stable colouring*
- *Game*
  - *Wins from the same position*

Proof (Idea)

Proof

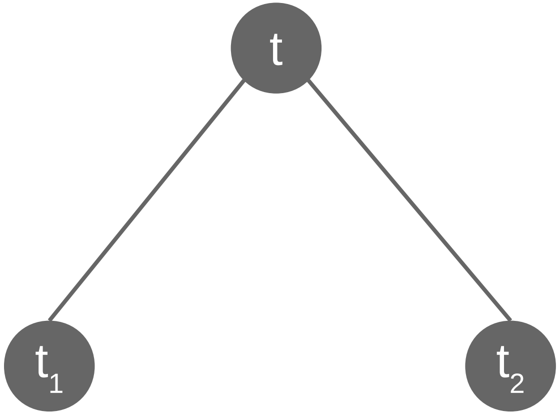
Idea

Rank decomposition

- *Play along*
- *For* X *(small rank) find* C*'s*
  - $C \subseteq X$ *or* $C \subseteq \overline{X}$
- *Can be treated independently*

Can remove pebbles from parent!

$$X = X_1 \uplus X_2$$



$X_1 := \gamma(t_1)$ $X_2 := \gamma(t_2)$

Proof

Induction Hypothesis

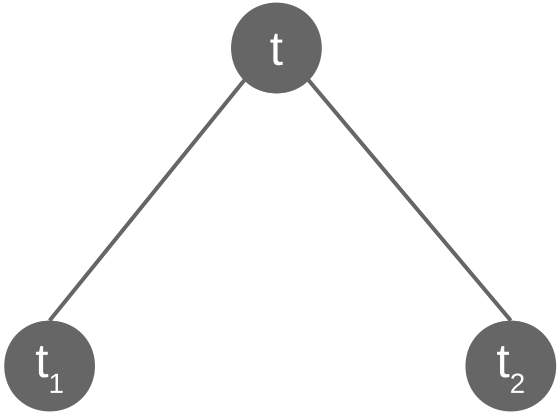
Position $((\overline{a},\overline{b},v),(\overline{a}',\overline{b}',v')) \Rightarrow$ Spoiler wins

- $(\overline{a},\overline{b})$ *ordered split pair* $t \in V(T)$ (i.e. $\gamma(t)$)
- $v \in \gamma(t)$
- f *flip function wrt.* $X = \gamma(t)$
  - $C \in \mathrm{Comp}((G, \chi_\infty{}^{(\overline{a},\overline{b})}),f)$ s.t. $v \in C$
  - $C' \in \mathrm{Comp}((H, \chi_\infty{}^{(\overline{a}',\overline{b}')}),f)$ s.t. $v' \in C'$
- $(G[C], \chi_\infty{}^{(\overline{a},\overline{b},v)}) \ncong (H[C'], \chi_\infty{}^{(\overline{a}',\overline{b}',v')})$

Proof

Induction Step: Idea

$$X = X_1 \uplus X_2$$



$$X_1 := \gamma(t_1) \qquad X_2 := \gamma(t_2)$$

$$|\gamma(t)| > 1 \Rightarrow \textit{children}$$

Spoiler moves to:

$$(\alpha,\alpha') := ((\bar{a},\bar{b},\bar{a}_1,\bar{b}_1,\bar{a}_2,\bar{b}_2,v),(\bar{a}',\bar{b}',\bar{a}_1',\bar{b}_1',\bar{a}_2',\bar{b}_2',v'))$$

Proof

Induction Step: Idea

From:

$$(\alpha,\alpha'):=((\overline{a},\overline{b},\overline{a}_1,\overline{b}_1,\overline{a}_2,\overline{b}_2,v),(\overline{a}',\overline{b}',\overline{a}_1',\overline{b}_1',\overline{a}_2',\overline{b}_2',v'))$$

Goal:

$$((\overline{a}_1,\overline{b}_1,z),(\overline{a}_1',\overline{b}_1',z')) \text{ or } ((\overline{a}_2,\overline{b}_2,z),(\overline{a}_2',\overline{b}_2',z'))$$

$\Rightarrow$ IH

Proof

Induction Step: Idea Case 1

Can find

- $C_1 \in \mathrm{Comp}((G, \chi_\infty^{(\bar{a}1,\bar{b}1)}), f_1)$,
- $C_1' \in \mathrm{Comp}((H, \chi_\infty^{(\bar{a}1',\bar{b}1')}), f_1)$
- $(G[C_1], \chi_\infty^{(\alpha)}) \not\cong (H[C_1'], \chi_\infty^{(\alpha')})$

Can find z, z'

- $z = v,\ z' = v'$
- $z = w,\ z' = \sigma(w)$

Proof

Induction Step: Idea Case 1

Obtain $(G[C_1], \chi_\infty^{(\alpha,z)}) \not\cong (H[C_1'], \chi_\infty^{(\alpha',z')})$

- C_1 *connected component in* G^{f1}
- C_1' *connected component in* H^{f1}
- $\Rightarrow$ *can remove pebbles outside* C_1 *without worry*
  - *without effecting the colouring restricted to* C_1

$\Rightarrow (G[C_1], \chi_\infty^{(\overline{a1},\overline{b1},z)}) \not\cong (H[C_1'], \chi_\infty^{(\overline{a1}',\overline{b1}',z')}) \Rightarrow$ IH

Proof

What about the rank width?

$(\overline{a},\overline{b})$, $(\overline{a}_1,\overline{b}_1)$ and $(\overline{a}_2,\overline{b}_2)$

- *Linear basis of* $\gamma(t)$, $\gamma(t)$ *and* $\gamma(t)$

$\mathrm{rw}(G) = k \Rightarrow$ size smaller $k \Rightarrow 6k$

- *actually* $6k + 5$

Can be improved to $3k + 5$

Contributions

Detail

$(3k+4)$-dim Weisfeiler-Leman algorithm identifies

- *graphs rank width at most k*

Isomorphism in $O(n^{3k+5} \log n)$, $rw(G) \leq k$

Sentence from C^{3k+5} characterises, $rw(G) \leq k$

Canonisation algorithm in $O(n^{3k+7} \log n)$, $rw(G) \leq k$

Thank you.