Homomorphism Indistinguishability for Comonadists

Comonads Online Meetup

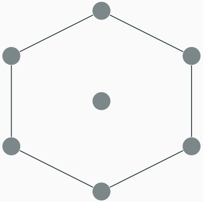
26 April 2023

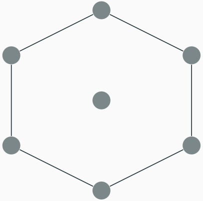
Tim Seppelt

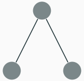
UnRAVeL Research Training Group – Uncertainty and Randomness in Algorithms, Verification, and Logic

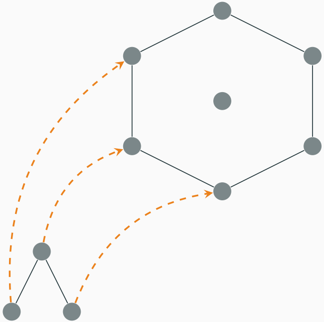
RWTH AACHEN UNIVERSITY

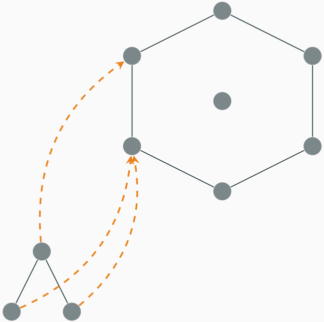
DFG Deutsche Forschungsgemeinschaft
German Research Foundation

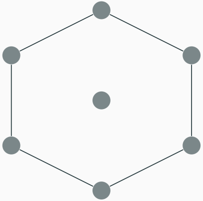


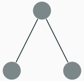




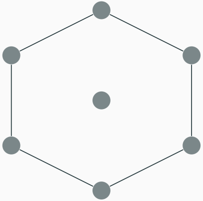


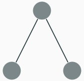




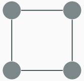


24

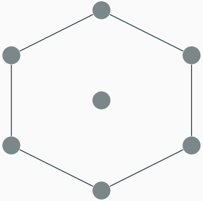
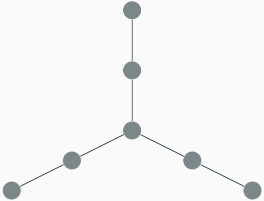
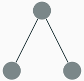
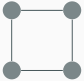


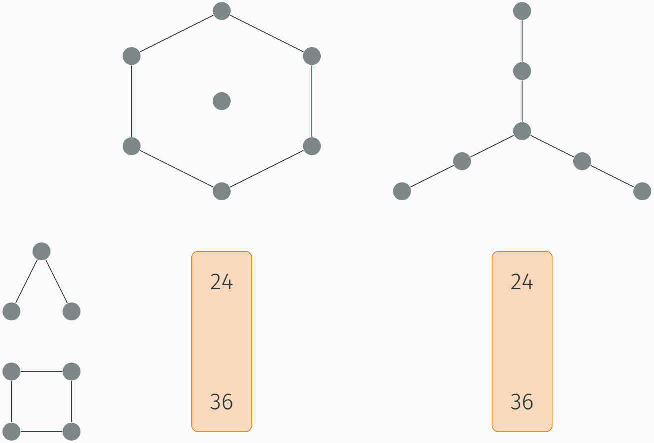


24



36

		
	24	24
	36	36

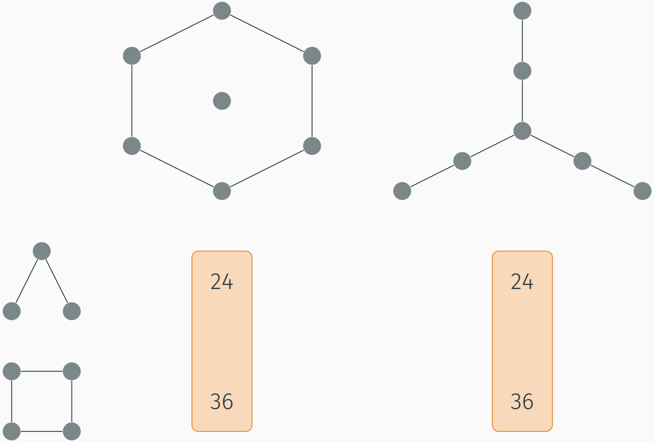


24

36

24

36

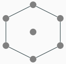
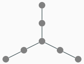


24

36

24

36

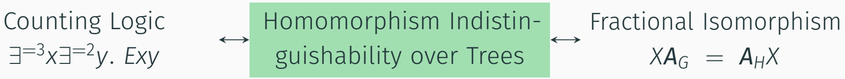
The graphs  and  are **homomorphism indistinguishable** over { , }.

Why Homomorphism Indistinguishability?

- Connections to graph properties in *finite model theory* and *algebraic graph theory*

Why Homomorphism Indistinguishability?

- Connections to graph properties in *finite model theory* and *algebraic graph theory*



Counting Logic
$\exists^{=3}x\exists^{=2}y.\ Exy$

↔

Homomorphism Indistinguishability over Trees

↔

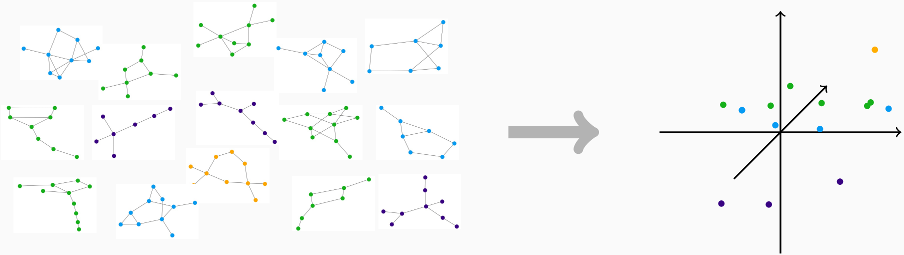
Fractional Isomorphism
$X\boldsymbol{A}_G = \boldsymbol{A}_H X$

Why Homomorphism Indistinguishability?

- Connections to graph properties in *finite model theory* and *algebraic graph theory*

Counting Logic
$\exists^{=3}x\exists^{=2}y.\ Exy$
↔
Homomorphism Indistinguishability over Trees
↔
Fractional Isomorphism
$X\boldsymbol{A}_G = \boldsymbol{A}_H X$

- Expressive *numerical graph invariants* for applications Illustration from Grohe (2020).



Outline

Matrix Equations for Homomorphism Indistinguishability

Outline

Matrix Equations for Homomorphism Indistinguishability

Towards a Theory of Homomorphism Indistinguishability

Outline

Matrix Equations for Homomorphism Indistinguishability

Towards a Theory of Homomorphism Indistinguishability

Open Questions

Matrix Equations for Homomorphism Indistinguishability

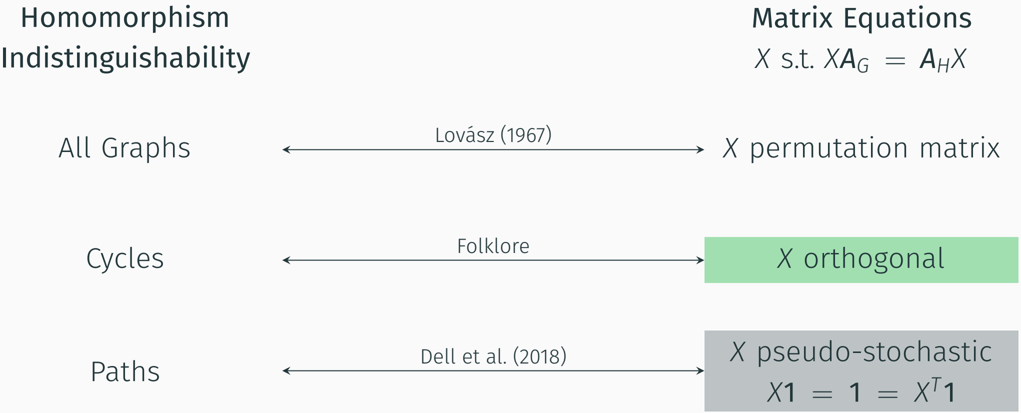
Matrix Equations for Homomorphism Indistinguishability

Homomorphism Indistinguishability		Matrix Equations X s.t. $X\boldsymbol{A}_G = \boldsymbol{A}_H X$
All Graphs	$\xleftrightarrow{\text{Lovász (1967)}}$	X permutation matrix

Matrix Equations for Homomorphism Indistinguishability

Homomorphism Indistinguishability		Matrix Equations X s.t. $XA_G = A_HX$
All Graphs	$\xleftrightarrow{\text{Lovász (1967)}}$	X permutation matrix
Cycles	$\xleftrightarrow{\text{Folklore}}$	X orthogonal

Matrix Equations for Homomorphism Indistinguishability



Homomorphism Indistinguishability

Matrix Equations

X s.t. $XA_G = A_H X$

All Graphs

Lovász (1967)

X permutation matrix

Cycles

Folklore

X orthogonal

Paths

Dell et al. (2018)

X pseudo-stochastic

$X\mathbf{1} = \mathbf{1} = X^T\mathbf{1}$

Matrix Equations for Homomorphism Indistinguishability

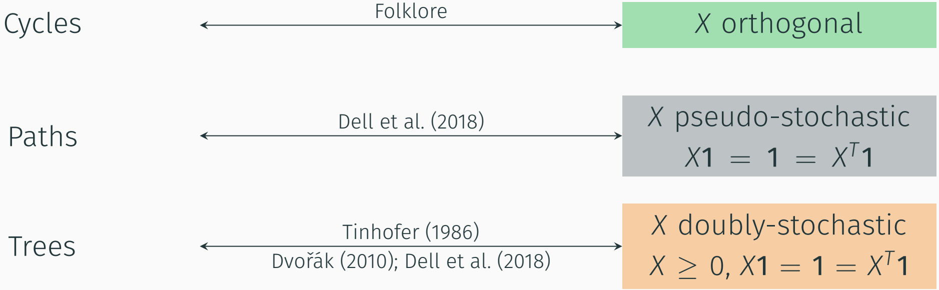
Homomorphism Indistinguishability		Matrix Equations X s.t. $XA_G = A_HX$
All Graphs	Lovász (1967) ⟷	X permutation matrix
Cycles	Folklore ⟷	X orthogonal
Paths	Dell et al. (2018) ⟷	X pseudo-stochastic $X\mathbf{1} = \mathbf{1} = X^T\mathbf{1}$
Trees	Tinhofer (1986) ⟷ Dvořák (2010); Dell et al. (2018)	X doubly-stochastic $X \geq 0$, $X\mathbf{1} = \mathbf{1} = X^T\mathbf{1}$

Matrix Equations for Homomorphism Indistinguishability

Homomorphism Indistinguishability

Matrix Equations

X s.t. $XA_G = A_H X$

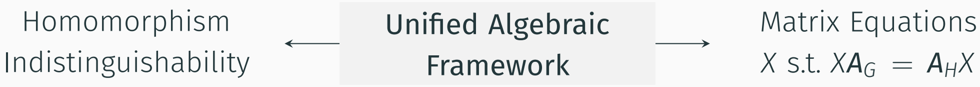


Cycles ⟷ (Folklore) ⟷ X orthogonal

Paths ⟷ (Dell et al. (2018)) ⟷ X pseudo-stochastic $X\mathbf{1} = \mathbf{1} = X^T\mathbf{1}$

Trees ⟷ (Tinhofer (1986); Dvořák (2010); Dell et al. (2018)) ⟷ X doubly-stochastic $X \geq 0$, $X\mathbf{1} = \mathbf{1} = X^T\mathbf{1}$

A Strategy for Matrix Equations Grohe, Rattan, S. (2022), Rattan & S. (2023)

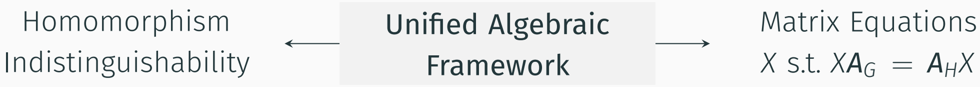


Homomorphism Indistinguishability

Unified Algebraic Framework

Matrix Equations X s.t. $X\boldsymbol{A}_G = \boldsymbol{A}_H X$

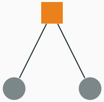
A Strategy for Matrix Equations Grohe, Rattan, S. (2022), Rattan & S. (2023)



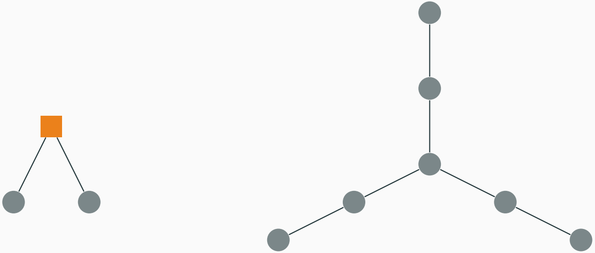
Homomorphism Indistinguishability ⟵ **Unified Algebraic Framework** ⟶ Matrix Equations X s.t. $X\boldsymbol{A}_G = \boldsymbol{A}_H X$

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
2. Define suitable **operations**
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations

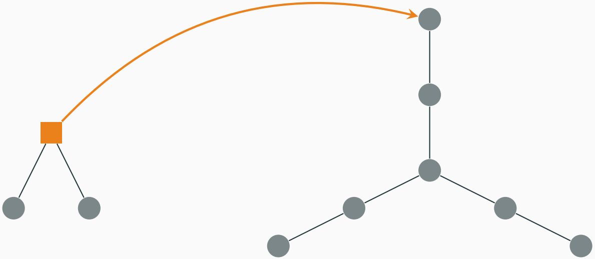
Labelled Graphs and Homomorphism Vectors



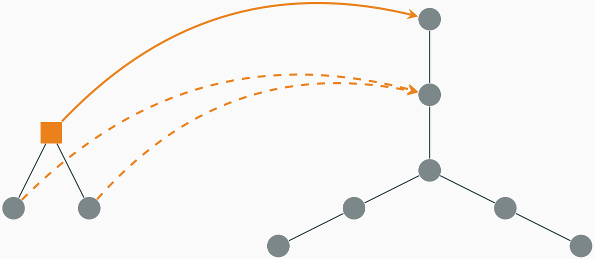
Labelled Graphs and Homomorphism Vectors



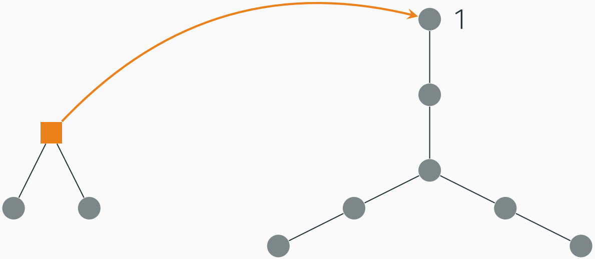
Labelled Graphs and Homomorphism Vectors



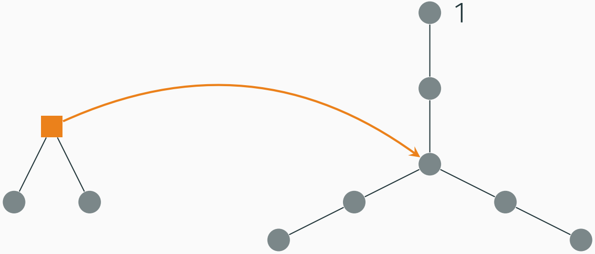
Labelled Graphs and Homomorphism Vectors



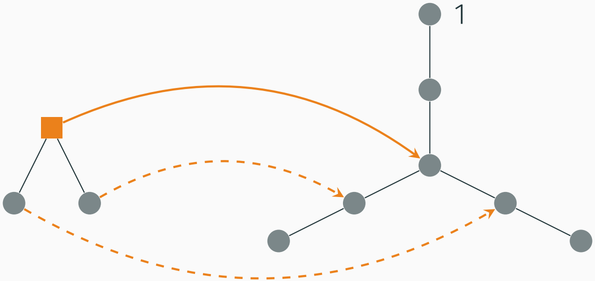
Labelled Graphs and Homomorphism Vectors



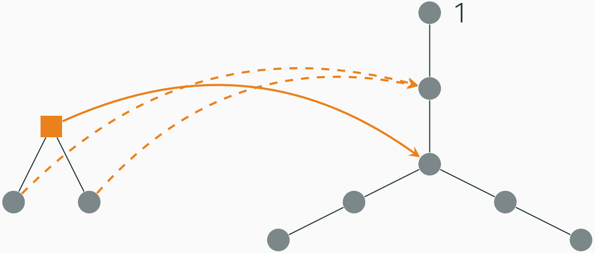
Labelled Graphs and Homomorphism Vectors



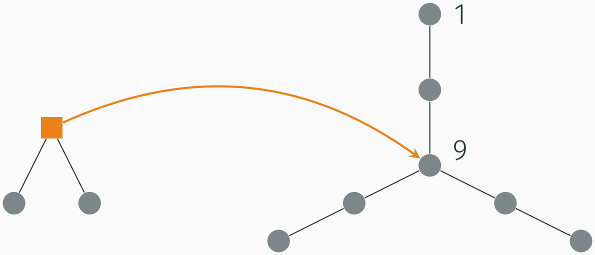
Labelled Graphs and Homomorphism Vectors



Labelled Graphs and Homomorphism Vectors

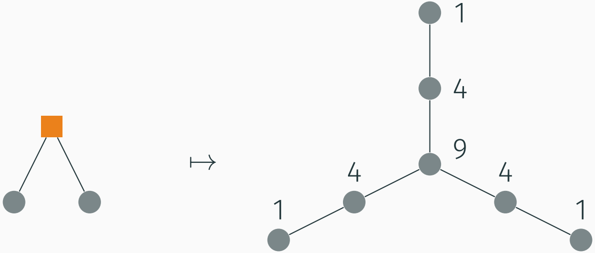


Labelled Graphs and Homomorphism Vectors

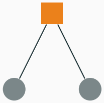


Labelled Graphs and Homomorphism Vectors

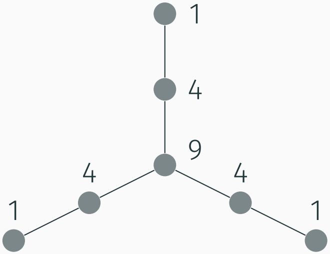
$$\mathcal{F} \longrightarrow \mathbb{C}^{V(G)}$$



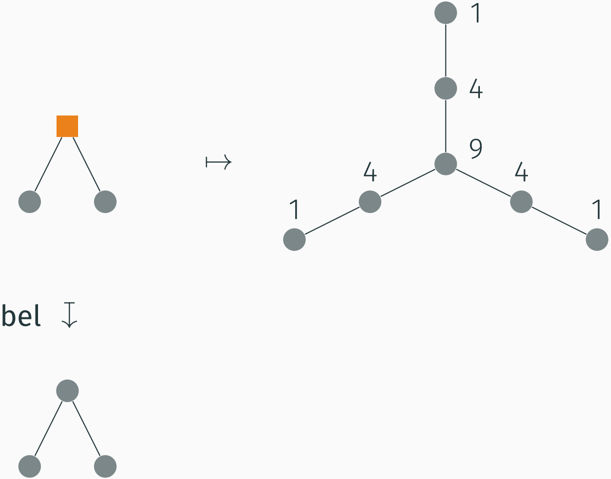
Combinatorial and Algebraic Operations: Unlabelling and Sum-of-Entries



$\mapsto$

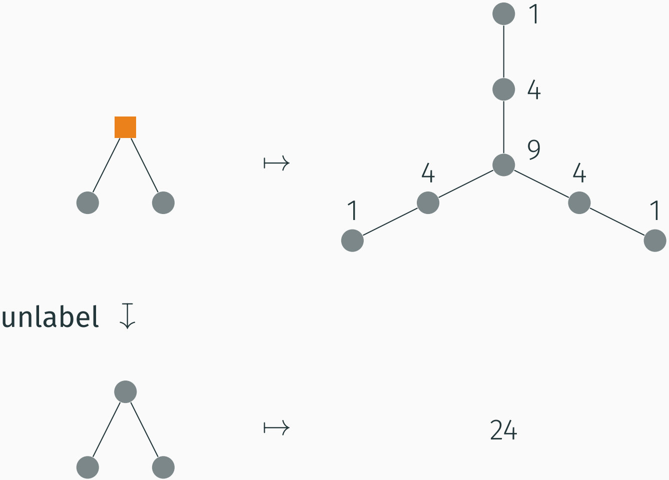


Combinatorial and Algebraic Operations: Unlabelling and Sum-of-Entries

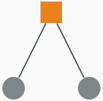


unlabel ↧

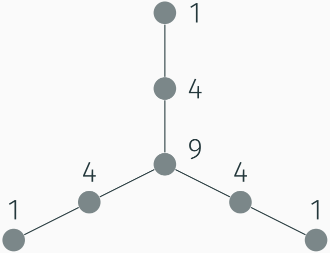
Combinatorial and Algebraic Operations: Unlabelling and Sum-of-Entries



Combinatorial and Algebraic Operations: Unlabelling and Sum-of-Entries

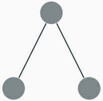


↦



unlabel ↧

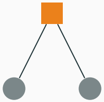
↧ sum of entries



↦

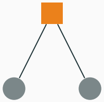
24

Combinatorial and Algebraic Operations: Gluing and Schur Product

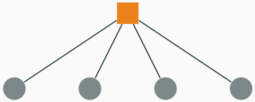


gluing

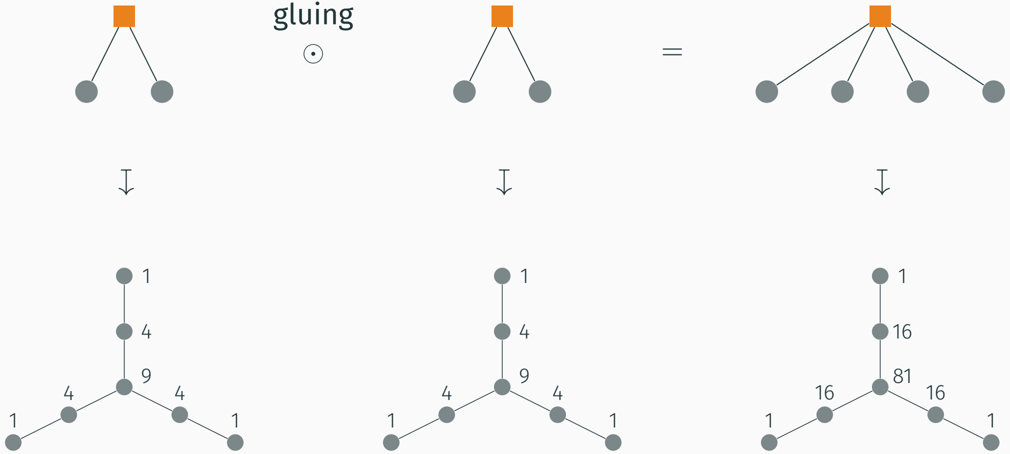
$\odot$



$=$



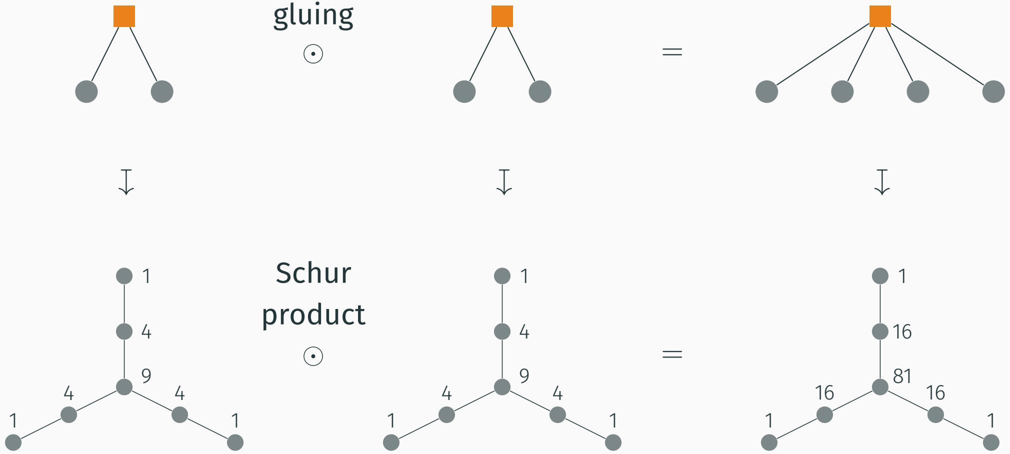
Combinatorial and Algebraic Operations: Gluing and Schur Product



gluing

$\odot$ $=$

Combinatorial and Algebraic Operations: Gluing and Schur Product

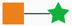


gluing $\odot$ $=$

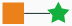
$\downarrow$

Schur product $\odot$ $=$

Bilabelled Graphs and Homomorphism Matrices

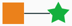


Bilabelled Graphs and Homomorphism Matrices

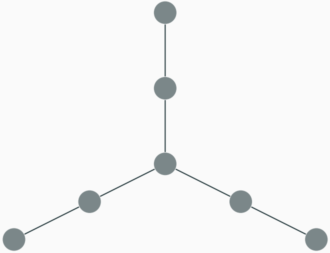


$$\mapsto \quad \left\{ \overbrace{\begin{matrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{matrix}} \right.$$

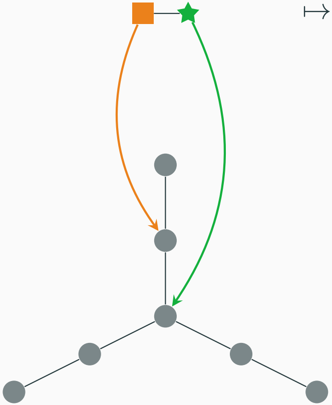
Bilabelled Graphs and Homomorphism Matrices



$$\mapsto \left\{ \overbrace{\begin{matrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{matrix}} \right.$$



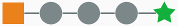
Bilabelled Graphs and Homomorphism Matrices



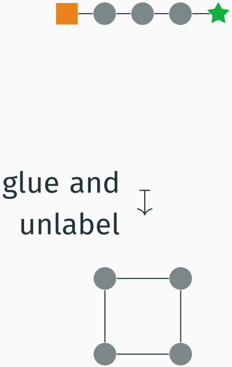
$\mapsto$

$$\left\{\overbrace{\begin{matrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{matrix}}\right.$$

Combinatorial and Algebraic Operations: Gluing+Unlabelling and Traces

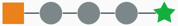


Combinatorial and Algebraic Operations: Gluing+Unlabelling and Traces



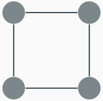
glue and unlabel

Combinatorial and Algebraic Operations: Gluing+Unlabelling and Traces

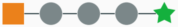


$$\mapsto \left\{ \overbrace{\begin{matrix} 12 & 0 & 4 & 0 & 4 & 0 & 4 \\ 0 & 6 & 0 & 5 & 0 & 5 & 0 \\ 4 & 0 & 2 & 0 & 1 & 0 & 1 \\ 0 & 5 & 0 & 6 & 0 & 5 & 0 \\ 4 & 0 & 1 & 0 & 2 & 0 & 1 \\ 0 & 5 & 0 & 5 & 0 & 6 & 0 \\ 4 & 0 & 1 & 0 & 1 & 0 & 2 \end{matrix}} \right.$$

glue and unlabel $\downarrow$



Combinatorial and Algebraic Operations: Gluing+Unlabelling and Traces

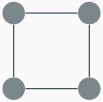


↦

$$\left\{\overbrace{\begin{matrix} 12 & 0 & 4 & 0 & 4 & 0 & 4 \\ 0 & 6 & 0 & 5 & 0 & 5 & 0 \\ 4 & 0 & 2 & 0 & 1 & 0 & 1 \\ 0 & 5 & 0 & 6 & 0 & 5 & 0 \\ 4 & 0 & 1 & 0 & 2 & 0 & 1 \\ 0 & 5 & 0 & 5 & 0 & 6 & 0 \\ 4 & 0 & 1 & 0 & 1 & 0 & 2 \end{matrix}}\right.$$

glue and unlabel ↧

↧ trace



↦

36

Examples: Trees, Paths, Cycles

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**

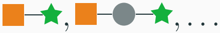
2. Define suitable **operations**

3. Prove that $\mathcal{F}$ is **finitely generated** under operations

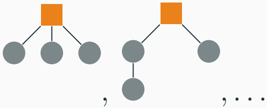
4. Define **representation** and recover system of equations

Examples: Trees, Paths, Cycles

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - paths, trees, cycles
2. Define suitable **operations**



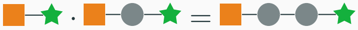
3. Prove that $\mathcal{F}$ is **finitely generated** under operations

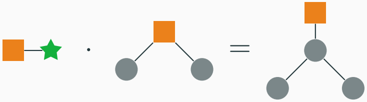


4. Define **representation** and recover system of equations

Examples: Trees, Paths, Cycles

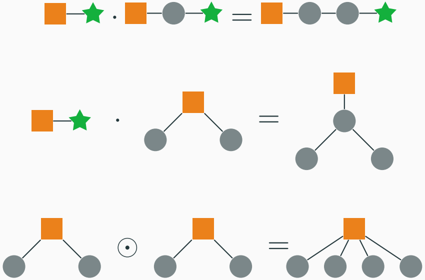
1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - paths, trees, cycles
2. Define suitable **operations**
   - gluing, series composition
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations





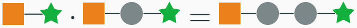
Examples: Trees, Paths, Cycles

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - paths, trees, cycles
2. Define suitable **operations**
   - gluing, series composition
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations

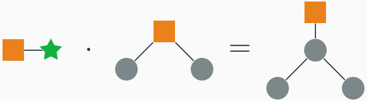


Examples: Trees, Paths, Cycles

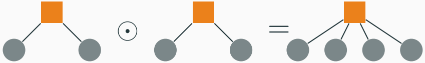
1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - paths, trees, cycles



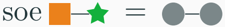
2. Define suitable **operations**
   - gluing, series composition
   - unlabelling, taking traces

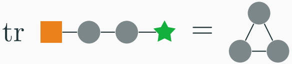


3. Prove that $\mathcal{F}$ is **finitely generated** under operations

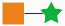


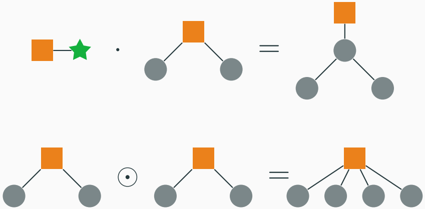
4. Define **representation** and recover system of equations





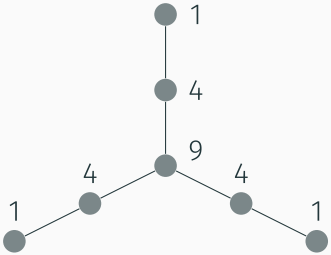
Examples: Trees, Paths, Cycles

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - paths, trees, cycles
2. Define suitable **operations**
   - gluing, series composition
   - unlabelling, taking traces
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
   - generator is 
4. Define **representation** and recover system of equations



Examples: Trees, Paths, Cycles

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - paths, trees, cycles
2. Define suitable **operations**
   - gluing, series composition
   - unlabelling, taking traces
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
   - generator is 🟧—★
4. Define **representation** and recover system of equations
   - homomorphism vectors and matrices
   - missing ingredient: *variants of theorem by Specht and Wiegmann*



Specht–Wiegmann: Unitary, Pseudo-Stochastic, Doubly-Stochastic

When are complex square matrices $A_1, \ldots, A_n$ and $B_1, \ldots, B_n$ simultaneously similar?

Specht–Wiegmann: Unitary, Pseudo-Stochastic, Doubly-Stochastic

When are complex square matrices $A_1, \ldots, A_n$ and $B_1, \ldots, B_n$ simultaneously similar?

Theorem

X unitary

$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

X pseudo-stochastic

$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

X doubly-stochastic

$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

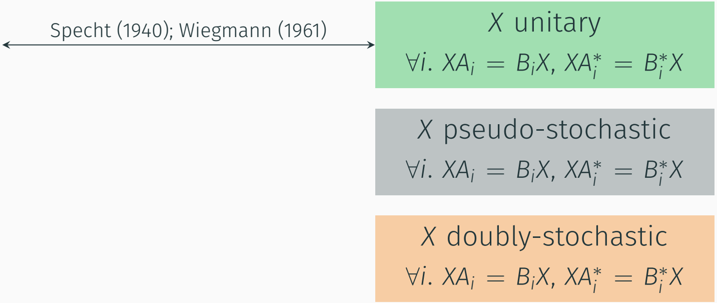
Specht–Wiegmann: Unitary, Pseudo-Stochastic, Doubly-Stochastic

When are complex square matrices $A_1, \ldots, A_n$ and $B_1, \ldots, B_n$ simultaneously similar?

Theorem

For every word w,

$$\operatorname{tr} w_A = \operatorname{tr} w_B.$$



Specht (1940); Wiegmann (1961)

X unitary

$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

X pseudo-stochastic

$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

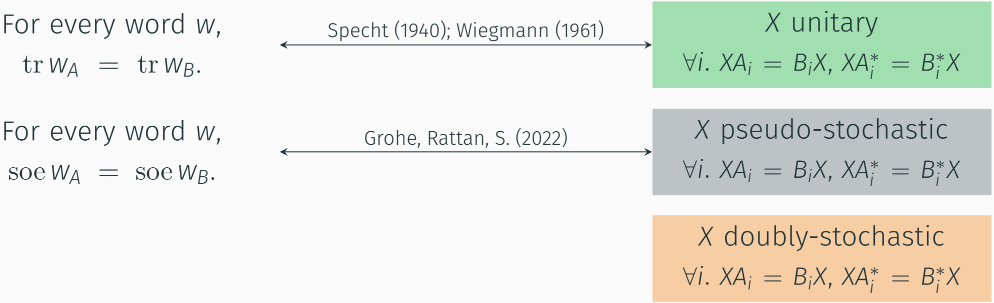
X doubly-stochastic

$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

Specht–Wiegmann: Unitary, Pseudo-Stochastic, Doubly-Stochastic

When are complex square matrices $A_1, \dots, A_n$ and $B_1, \dots, B_n$ simultaneously similar?

Theorem



For every word w,

$$\operatorname{tr} w_A = \operatorname{tr} w_B.$$

$\xleftrightarrow{\text{Specht (1940); Wiegmann (1961)}}$

X unitary

$$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$$

For every word w,

$$\operatorname{soe} w_A = \operatorname{soe} w_B.$$

$\xleftrightarrow{\text{Grohe, Rattan, S. (2022)}}$

X pseudo-stochastic

$$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$$

X doubly-stochastic

$$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$$

Specht–Wiegmann: Unitary, Pseudo-Stochastic, Doubly-Stochastic

When are complex square matrices $A_1, \dots, A_n$ and $B_1, \dots, B_n$ simultaneously similar?

Theorem

For every word w,
$$\operatorname{tr} w_A = \operatorname{tr} w_B.$$
$$\xleftrightarrow{\text{Specht (1940); Wiegmann (1961)}}$$
X unitary
$$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$$

For every word w,
$$\operatorname{soe} w_A = \operatorname{soe} w_B.$$
$$\xleftrightarrow{\text{Grohe, Rattan, S. (2022)}}$$
X pseudo-stochastic
$$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$$

For every tree t,
$$\operatorname{soe} t_A = \operatorname{soe} t_B.$$
$$\xleftrightarrow{\text{Grohe, Rattan, S. (2022)}}$$
X doubly-stochastic
$$\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$$

Specht–Wiegmann: Words

Let Γ be the set of finite words over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

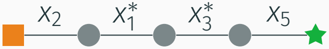
Γ forms an *involution monoid.*

Specht–Wiegmann: Words

Let Γ be the set of finite words over $\{x_1, \dots, x_n, x_1^*, \dots, x_n^*\}$.

Γ forms an *involution monoid*.

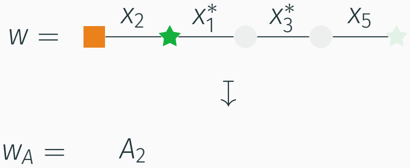
$w =$



Specht–Wiegmann: Words

Let Γ be the set of finite words over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

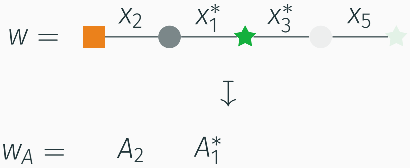
Γ forms an *involution monoid*.



Specht–Wiegmann: Words

Let Γ be the set of finite words over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

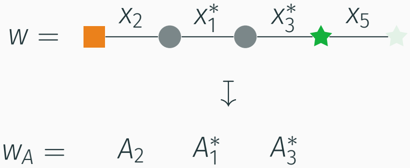
Γ forms an *involution monoid.*



Specht–Wiegmann: Words

Let Γ be the set of finite words over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

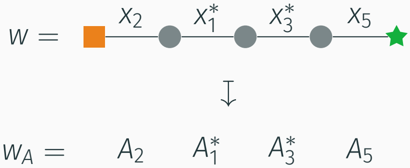
Γ forms an *involution monoid*.



Specht–Wiegmann: Words

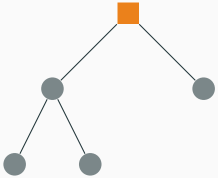
Let Γ be the set of finite words over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

Γ forms an *involution monoid.*



Specht–Wiegmann: Trees

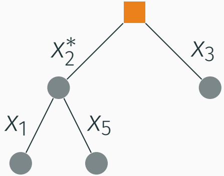
Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.



Specht–Wiegmann: Trees

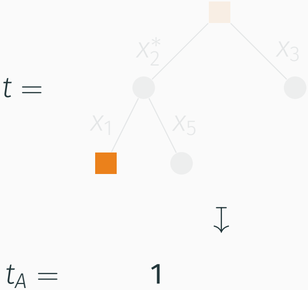
Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

$t =$



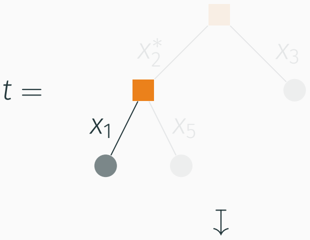
Specht–Wiegmann: Trees

Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.



Specht–Wiegmann: Trees

Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

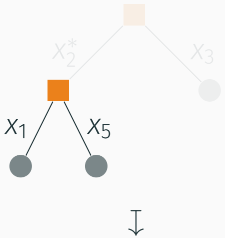


$$t_A = (A_1 \mathbf{1})$$

Specht–Wiegmann: Trees

Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

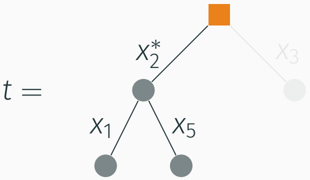
$t =$



$$t_A = \quad ((A_1\mathbf{1}) \odot (A_5\mathbf{1}))$$

Specht–Wiegmann: Trees

Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.



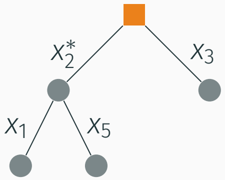
$$\downarrow$$

$$t_A = A_2^* \left((A_1 \mathbf{1}) \odot (A_5 \mathbf{1}) \right)$$

Specht–Wiegmann: Trees

Consider *trees* over $\{x_1, \ldots, x_n, x_1^*, \ldots, x_n^*\}$.

$t =$



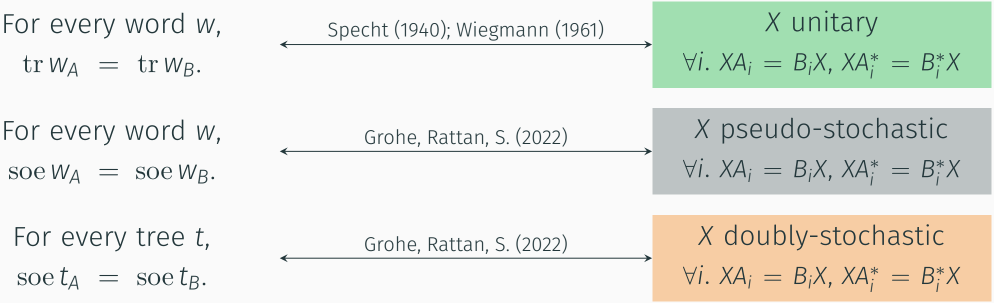
$$\downarrow$$

$$t_A = A_2^* \left((A_1 \mathbf{1}) \odot (A_5 \mathbf{1}) \right) \odot (A_3 \mathbf{1})$$

Specht–Wiegmann: Unitary, Pseudo-Stochastic, Doubly-Stochastic

Let $A_1, \dots, A_n$ and $B_1, \dots, B_n$ be square matrices.

Theorem

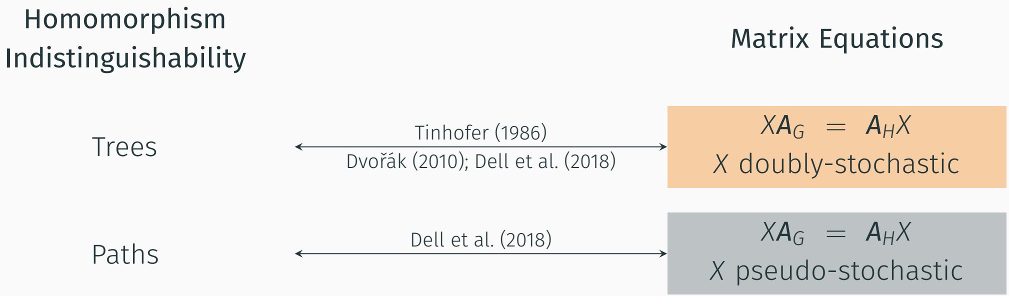


For every word w, $\operatorname{tr} w_A = \operatorname{tr} w_B$. $\xleftrightarrow{\text{Specht (1940); Wiegmann (1961)}}$ X unitary $\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

For every word w, $\operatorname{soe} w_A = \operatorname{soe} w_B$. $\xleftrightarrow{\text{Grohe, Rattan, S. (2022)}}$ X pseudo-stochastic $\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

For every tree t, $\operatorname{soe} t_A = \operatorname{soe} t_B$. $\xleftrightarrow{\text{Grohe, Rattan, S. (2022)}}$ X doubly-stochastic $\forall i.\ XA_i = B_iX,\ XA_i^* = B_i^*X$

Graphs of Bounded Pathwidth and Sherali–Adams Relaxation



Homomorphism Indistinguishability

Matrix Equations

Trees

Tinhofer (1986)

Dvořák (2010); Dell et al. (2018)

$$XA_G = A_HX$$

X doubly-stochastic

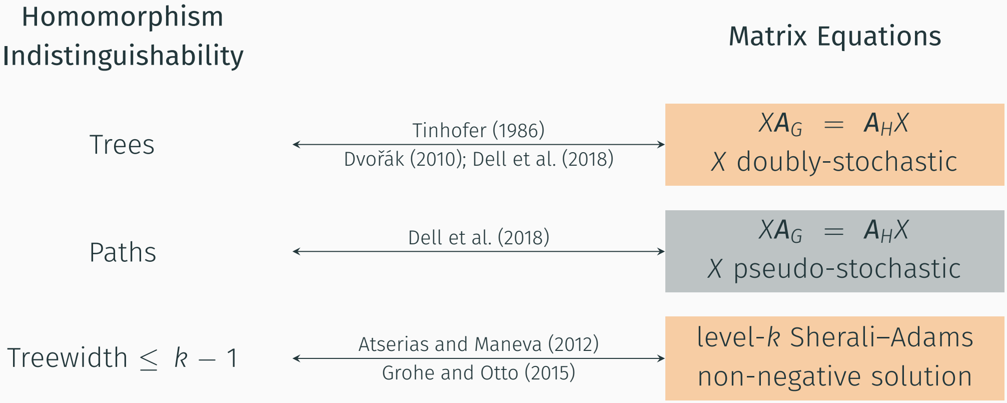
Paths

Dell et al. (2018)

$$XA_G = A_HX$$

X pseudo-stochastic

Graphs of Bounded Pathwidth and Sherali–Adams Relaxation



Homomorphism Indistinguishability

Matrix Equations

Trees

Tinhofer (1986)

Dvořák (2010); Dell et al. (2018)

$XA_G = A_HX$

X doubly-stochastic

Paths

Dell et al. (2018)

$XA_G = A_HX$

X pseudo-stochastic

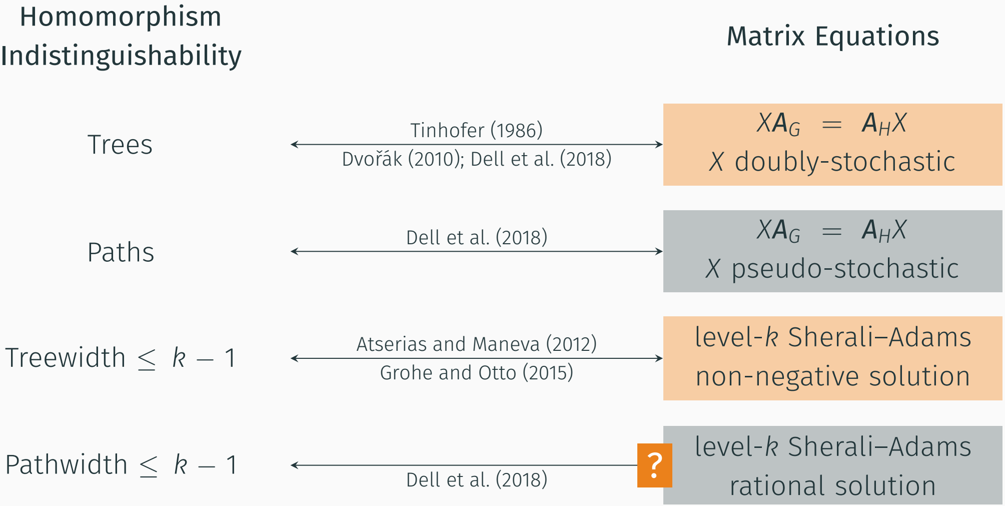
Treewidth $\leq k - 1$

Atserias and Maneva (2012)

Grohe and Otto (2015)

level-k Sherali–Adams non-negative solution

Graphs of Bounded Pathwidth and Sherali–Adams Relaxation



Homomorphism Indistinguishability

Matrix Equations

Trees

Tinhofer (1986)

Dvořák (2010); Dell et al. (2018)

$XA_G = A_HX$

X doubly-stochastic

Paths

Dell et al. (2018)

$XA_G = A_HX$

X pseudo-stochastic

Treewidth $\leq k - 1$

Atserias and Maneva (2012)

Grohe and Otto (2015)

level-k Sherali–Adams non-negative solution

Pathwidth $\leq k - 1$

Dell et al. (2018)

?

level-k Sherali–Adams rational solution

Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

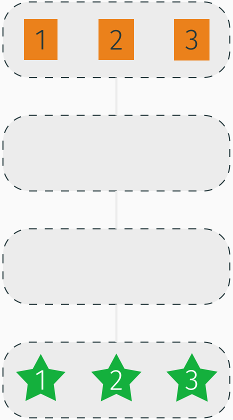
Homomorphism Indistinguishability		Matrix Equations
Trees	Tinhofer (1986) ↔ Dvořák (2010); Dell et al. (2018)	$XA_G = A_HX$ X doubly-stochastic
Paths	Dell et al. (2018) ↔	$XA_G = A_HX$ X pseudo-stochastic
Treewidth $\leq k-1$	Atserias and Maneva (2012) ↔ Grohe and Otto (2015)	level-k Sherali–Adams non-negative solution
Pathwidth $\leq k-1$	**Grohe, Rattan, S. (2022)** ↔ Dell et al. (2018)	level-k Sherali–Adams rational solution

Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
2. Define suitable **operations**
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations

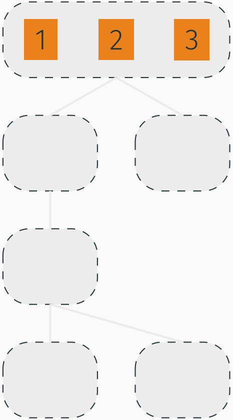
Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labels in a single bag of the tree or path decomposition.
2. Define suitable **operations**
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations



Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

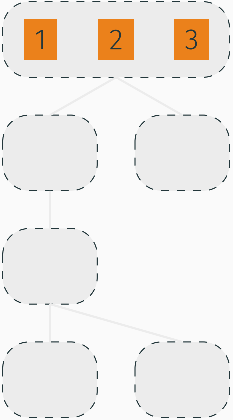
1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labels in a single bag of the tree or path decomposition.
2. Define suitable **operations**
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations



Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

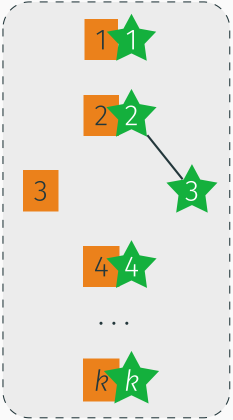
1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labels in a single bag of the tree or path decomposition.
2. Define suitable **operations**
   - gluing, series composition
3. Prove that $\mathcal{F}$ is **finitely generated** under operations

4. Define **representation** and recover system of equations



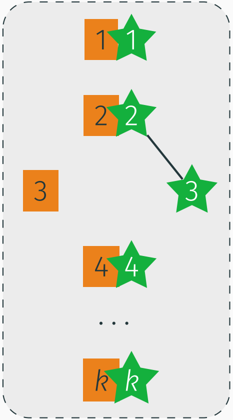
Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labels in a single bag of the tree or path decomposition.
2. Define suitable **operations**
   - gluing, series composition
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
   - generator is not 🟧—★ but **basal graphs**, i.e. bilabelled single bag.
4. Define **representation** and recover system of equations



Graphs of Bounded Pathwidth and Sherali–Adams Relaxation

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labels in a single bag of the tree or path decomposion.
2. Define suitable **operations**
   - gluing, series composition
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
   - generator is not 🟧—★ but **basal graphs**, i.e. bilabelled single bag.
4. Define **representation** and recover system of equations
   - homomorphism tensors and Specht–Wiegmann

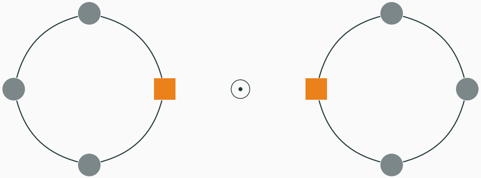


Limitations: Warped Wheel

The pieces **labelling**, **operations**, **finite generation**, and **representation** have to fit together.

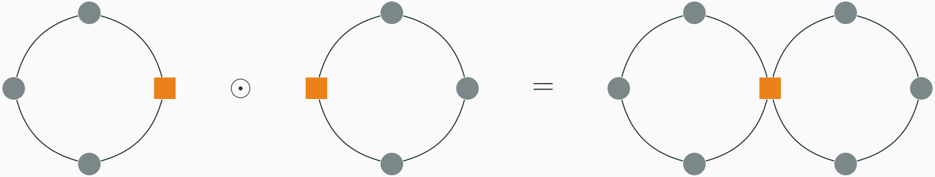
Limitations: Warped Wheel

The pieces **labelling**, **operations**, **finite generation**, and **representation** have to fit together.



Limitations: Warped Wheel

The pieces **labelling**, **operations**, **finite generation**, and **representation** have to fit together.



A Comonadic Strategy

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**

2. Define suitable **operations**

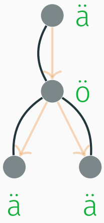
3. Prove that $\mathcal{F}$ is **finitely generated** under operations

4. Define **representation** and recover system of equations

A Comonadic Strategy

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labelled coalgebras of pebbling comonad $\mathbb{P}_{k,d}$ from Dawar et al. (2021).
2. Define suitable **operations**
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations

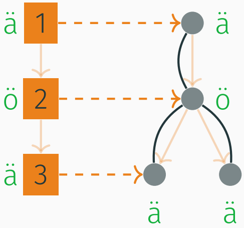
(A, α)



A Comonadic Strategy

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labelled coalgebras of pebbling comonad $\mathbb{P}_{k,d}$ from Dawar et al. (2021).
2. Define suitable **operations**
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations

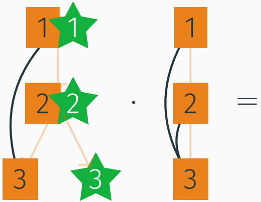
$$(L, \lambda) \quad \to \quad (A, \alpha)$$



A Comonadic Strategy

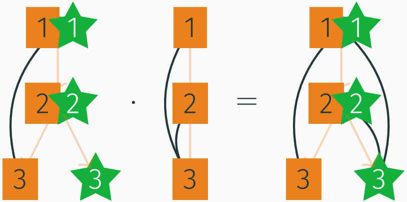
1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labelled coalgebras of pebbling comonad $\mathbb{P}_{k,d}$ from Dawar et al. (2021).
2. Define suitable **operations**
   - pushouts in $\mathrm{EM}(\mathbb{P}_{k,d})$
3. Prove that $\mathcal{F}$ is **finitely generated** under operations

4. Define **representation** and recover system of equations



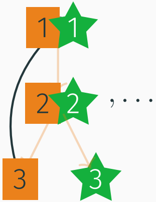
A Comonadic Strategy

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labelled coalgebras of pebbling comonad $\mathbb{P}_{k,d}$ from Dawar et al. (2021).
2. Define suitable **operations**
   - pushouts in $\mathrm{EM}(\mathbb{P}_{k,d})$
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
4. Define **representation** and recover system of equations



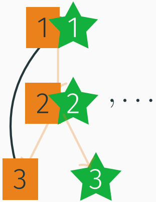
A Comonadic Strategy

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labelled coalgebras of pebbling comonad $\mathbb{P}_{k,d}$ from Dawar et al. (2021).
2. Define suitable **operations**
   - pushouts in $\mathrm{EM}(\mathbb{P}_{k,d})$
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
   - requires $\mathbb{P}_{k,d}$-specific argument
4. Define **representation** and recover system of equations



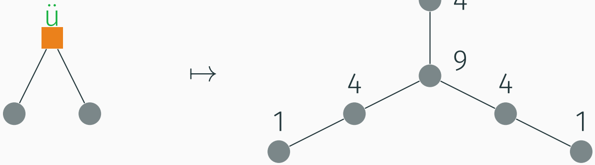
A Comonadic Strategy

1. Construct family $\mathcal{F}$ of **(bi)labelled graphs**
   - labelled coalgebras of pebbling comonad $\mathbb{P}_{k,d}$ from Dawar et al. (2021).
2. Define suitable **operations**
   - pushouts in $\mathrm{EM}(\mathbb{P}_{k,d})$
3. Prove that $\mathcal{F}$ is **finitely generated** under operations
   - requires $\mathbb{P}_{k,d}$-specific argument
4. Define **representation** and recover system of equations
   - **augmented homomorphism tensors** and Specht–Wiegmann



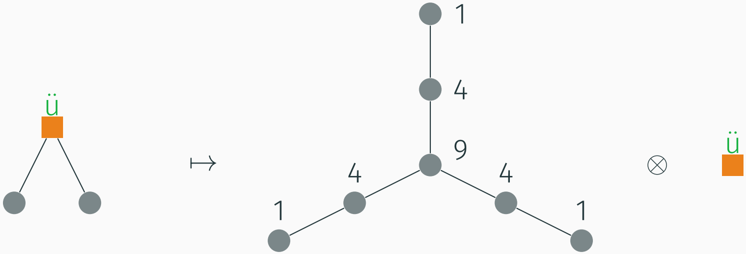
Augmented Homomorphism Representation

$$\mathcal{F} \longrightarrow \mathbb{C}^{V(G)}$$

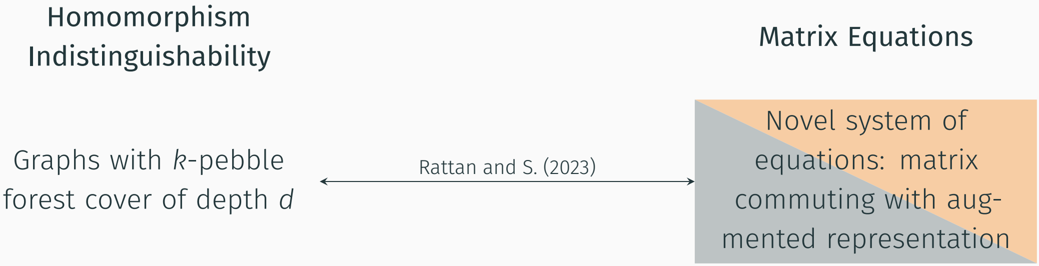


Augmented Homomorphism Representation

$$\mathcal{F} \longrightarrow \mathbb{C}^{V(G)} \otimes \bigoplus_{(L,\lambda)} \mathbb{C}$$



Graphs admitting k-pebble forest covers of depth d



Homomorphism Indistinguishability

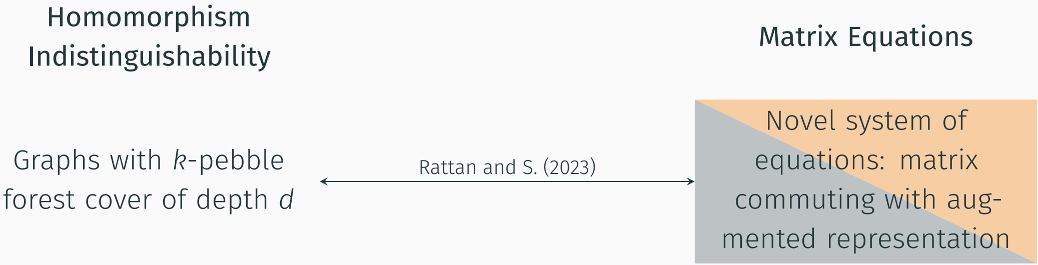
Matrix Equations

Graphs with k-pebble forest cover of depth d

Rattan and S. (2023)

Novel system of equations: matrix commuting with augmented representation

Graphs admitting k-pebble forest covers of depth d



Homomorphism Indistinguishability

Matrix Equations

Graphs with k-pebble forest cover of depth d

Rattan and S. (2023)

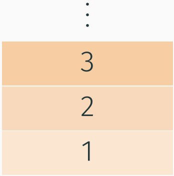
Novel system of equations: matrix commuting with augmented representation

This characterises *logical equivalence over* $\mathsf{C}_k \cap \mathsf{C}^d$, and with some modifications indistinguishability after *d rounds of the k-dimensional Weisfeiler–Leman algorithm*.

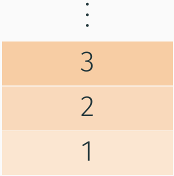
Towards a Theory of Homomorphism Indistinguishability

G and H are isomorphic iff
integer program $\mathrm{ISO}(G,H)$ is feasible

G and H are isomorphic iff
integer program $\mathrm{ISO}(G, H)$ is feasible

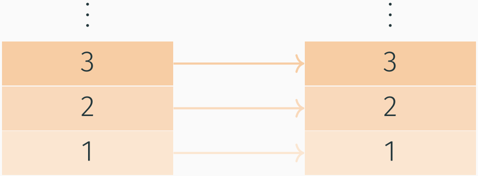

⋮
3
2
1

Lasserre
semidefinite prog.


⋮
3
2
1

Sherali–Adams
linear prog.

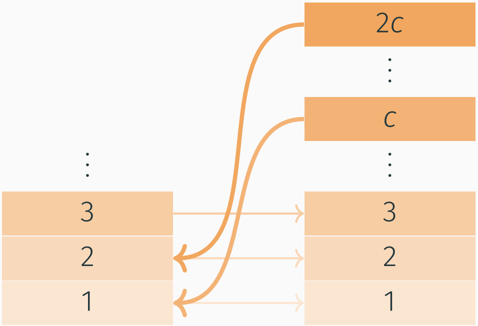
G and H are isomorphic iff
integer program $\mathrm{ISO}(G, H)$ is feasible



Lasserre
semidefinite prog.

Sherali–Adams
linear prog.

G and H are isomorphic iff
integer program $\mathrm{ISO}(G, H)$ is feasible

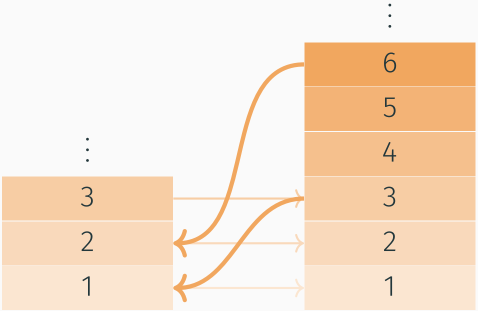


Lasserre
semidefinite prog.

Sherali–Adams
linear prog.

Atserias and Ochremiak (2018)

G and H are isomorphic iff
integer program $\mathrm{ISO}(G, H)$ is feasible



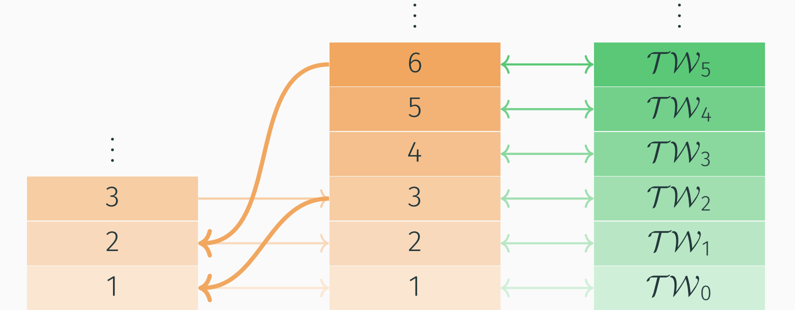
Lasserre
semidefinite prog.

Sherali–Adams
linear prog.

Atserias and Ochremiak (2018), **Roberson & S. (2023)**

G and H are isomorphic iff

integer program $\mathrm{ISO}(G, H)$ is feasible



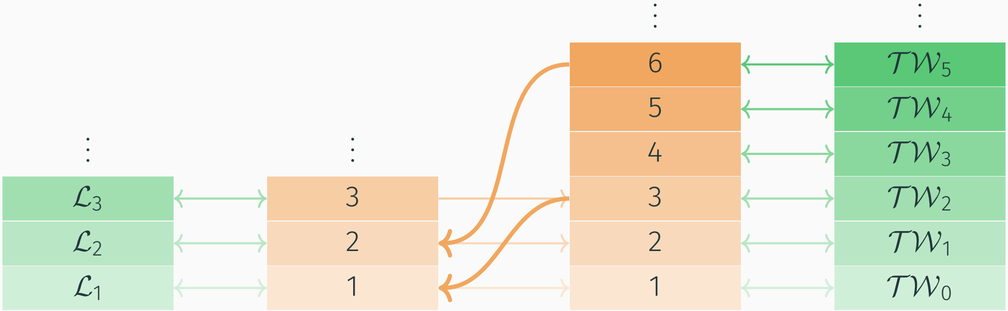
Lasserre
semidefinite prog.

Sherali–Adams
linear prog.

Homomorphism
Indistinguishability

Atserias and Ochremiak (2018), **Roberson & S. (2023)**, Grohe and Otto (2015), Atserias and Maneva (2012), Dvořák (2010)

G and H are isomorphic iff
integer program $\mathrm{ISO}(G, H)$ is feasible



Homomorphism Indistinguishability

Lasserre semidefinite prog.

Sherali–Adams linear prog.

Homomorphism Indistinguishability

Atserias and Ochremiak (2018), **Roberson & S. (2023)**, Grohe and Otto (2015), Atserias and Maneva (2012), Dvořák (2010)

From Equations to Graphs

Equations

homomorphism tensors,

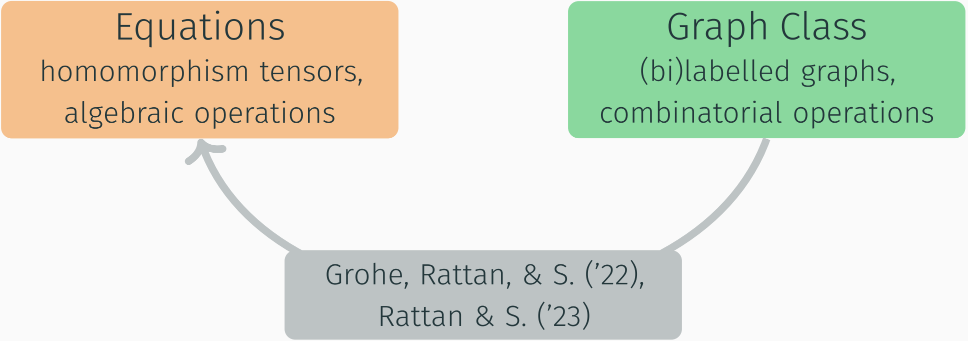
algebraic operations

Graph Class

(bi)labelled graphs,

combinatorial operations

From Equations to Graphs



Equations

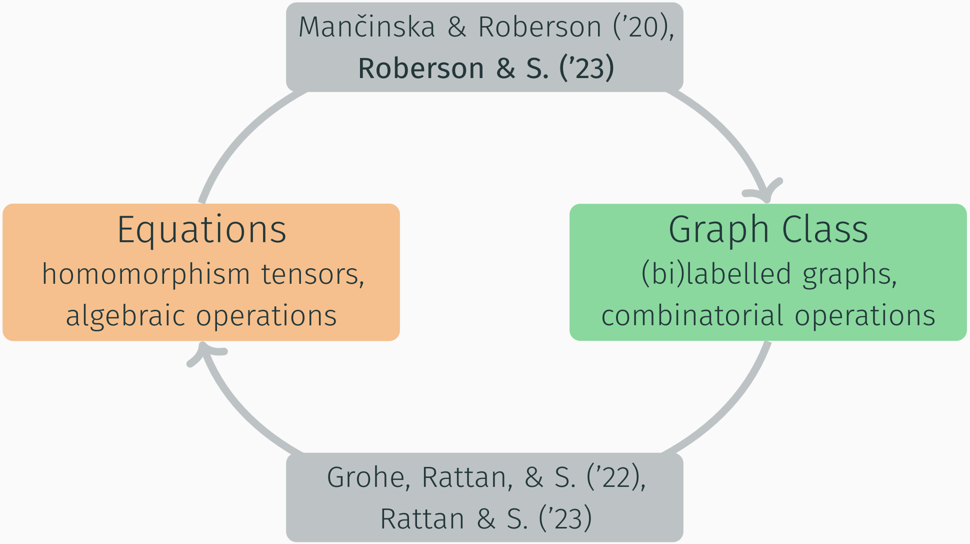
homomorphism tensors,
algebraic operations

Graph Class

(bi)labelled graphs,
combinatorial operations

Grohe, Rattan, & S. ('22),
Rattan & S. ('23)

From Equations to Graphs



Mančinska & Roberson ('20),

Roberson & S. ('23)

Equations

homomorphism tensors,

algebraic operations

Graph Class

(bi)labelled graphs,

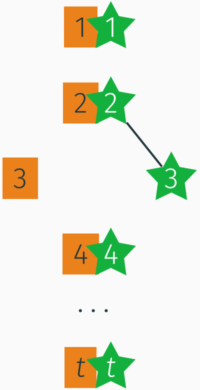
combinatorial operations

Grohe, Rattan, & S. ('22),

Rattan & S. ('23)

The Graph Class $\mathcal{L}_t$

A (t, t)-bilabelled graph is *atomic* if all its vertices are labelled.

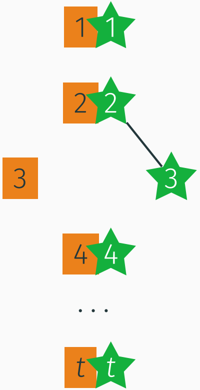


The Graph Class $\mathcal{L}_t$

A (t, t)-bilabelled graph is *atomic* if all its vertices are labelled.

The class $\mathcal{L}_t$ is generated by atomic graphs under

- series composition,
- parallel composition with atomic graphs,
- permutation of labels.



Syntactic Properties of the Graph Class $\mathcal{L}_t$

- $\mathcal{L}_t \subseteq \mathcal{TW}_{3t-1}$,

Syntactic Properties of the Graph Class $\mathcal{L}_t$

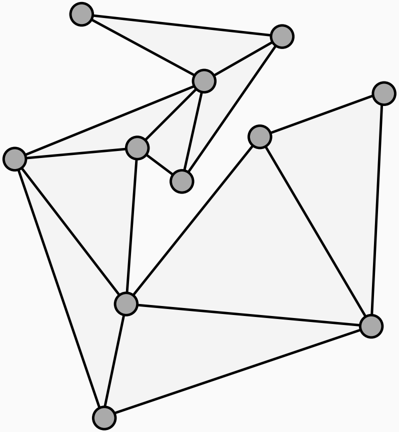
- $\mathcal{L}_t \subseteq \mathcal{TW}_{3t-1}$,
- $\mathcal{L}_t$ contains the clique K_{3t},

Syntactic Properties of the Graph Class $\mathcal{L}_t$

- $\mathcal{L}_t \subseteq \mathcal{TW}_{3t-1}$,
- $\mathcal{L}_t$ contains the clique K_{3t},
- $\mathcal{L}_t$ is minor-closed,

Syntactic Properties of the Graph Class $\mathcal{L}_t$

- $\mathcal{L}_t \subseteq \mathcal{TW}_{3t-1}$,
- $\mathcal{L}_t$ contains the clique K_{3t},
- $\mathcal{L}_t$ is minor-closed,
- $\mathcal{L}_1$ is the class of all outerplanar graphs.



Syntax and Semantics: Roberson's Conjecture

$\mathcal{L}_t$ is a class of graphs of treewidth $\leq 3t - 1$ containing K_{3t}.

Syntax and Semantics: Roberson's Conjecture

$\mathcal{L}_t$ is a class of graphs of treewidth $\leq 3t - 1$ containing K_{3t}.

Although $\mathcal{L}_t \not\subseteq \mathcal{TW}_{3t-2}$, it could well be that $G \equiv_{\mathcal{TW}_{3t-2}} H \implies G \equiv_{\mathcal{L}_t} H$.

Syntax and Semantics: Roberson's Conjecture

$\mathcal{L}_t$ is a class of graphs of treewidth $\leq 3t - 1$ containing K_{3t}.

Although $\mathcal{L}_t \not\subseteq \mathcal{TW}_{3t-2}$, it could well be that $G \equiv_{\mathcal{TW}_{3t-2}} H \implies G \equiv_{\mathcal{L}_t} H$.

The homomorphism distinguishing closure of a graph class $\mathcal{F}$ is

$$\mathrm{cl}(\mathcal{F}) = \{K \text{ graph} \mid G \equiv_{\mathcal{F}} H \implies \hom(K, G) = \hom(K, H)\}.$$

Syntax and Semantics: Roberson's Conjecture

$\mathcal{L}_t$ is a class of graphs of treewidth $\leq 3t-1$ containing K_{3t}.

Although $\mathcal{L}_t \not\subseteq \mathcal{TW}_{3t-2}$, it could well be that $G \equiv_{\mathcal{TW}_{3t-2}} H \implies G \equiv_{\mathcal{L}_t} H$.

The homomorphism distinguishing closure of a graph class $\mathcal{F}$ is

$$\mathrm{cl}(\mathcal{F}) = \{K \text{ graph} \mid G \equiv_{\mathcal{F}} H \implies \hom(K, G) = \hom(K, H)\}.$$

Conjecture (Roberson (2022))

Every minor-closed union-closed graph class is homomorphism distinguishing closed.

Roberson's Conjecture: State of Affairs

Conjecture (Roberson (2022))

Every minor-closed union-closed graph class is homomorphism distinguishing closed.

Roberson's Conjecture: State of Affairs

Conjecture (Roberson (2022))

Every minor-closed union-closed graph class is homomorphism distinguishing closed.

- treewidth $\leq k$, — Neuen (2023)
- planar graphs, — Roberson (2022)
- essentially finite graph classes. — S. (2023)

Roberson's Conjecture: State of Affairs

Conjecture (Roberson (2022))

Every minor-closed union-closed graph class is homomorphism distinguishing closed.

- treewidth $\leq k$, Neuen (2023)
- planar graphs, Roberson (2022)
- essentially finite graph classes. S. (2023)

Corollary (Roberson and S. (2023))

For every $t \geq 1$, there are graphs G and H such that $G \simeq^{\mathsf{SA}}_{3t-1} H$ and $G \not\simeq^{\mathsf{L}}_{t} H$.

Games for Roberson's Conjecture

Theorem (Neuen (2023))

For every $t \geq 0$, the class $\mathcal{TW}_t$ is homomorphism distinguishing closed.

Games for Roberson's Conjecture

Theorem (Neuen (2023))

For every $t \geq 0$*, the class* $\mathcal{TW}_t$ *is homomorphism distinguishing closed.*

Using a CFI-like construction of Roberson (2022), it suffices to show the following:

Games for Roberson's Conjecture

Theorem (Neuen (2023))

For every $t \geq 0$, the class $\mathcal{TW}_t$ is homomorphism distinguishing closed.

Using a CFI-like construction of Roberson (2022), it suffices to show the following:

Claim

If $G \notin \mathcal{TW}_k$ and G is connected then $G_0 \equiv_{\mathcal{TW}_k} G_1$.

Duplicator can play like robber evading $k + 1$ cops on G.

Games for Roberson's Conjecture

Theorem (Neuen (2023))

For every $t \geq 0$, the class $\mathcal{TW}_t$ is homomorphism distinguishing closed.

Using a CFI-like construction of Roberson (2022), it suffices to show the following:

Claim

If $G \notin \mathcal{TW}_k$ and G is connected then $G_0 \equiv_{\mathcal{TW}_k} G_1$.

Duplicator can play like robber evading $k + 1$ cops on G.

Question

Can game comonads yield more such results?

Properties of Homomorphism Indistinguishability Relations

Let’s forget about the graph class $\mathcal{F}$ and think of the equivalence relation $\equiv_{\mathcal{F}}$!

Properties of Homomorphism Indistinguishability Relations

Let's forget about the graph class $\mathcal{F}$ and think of the equivalence relation $\equiv_{\mathcal{F}}$!

Observation ($\equiv_{\mathcal{F}}$ is preserved under categorical products)

If $G_1 \equiv_{\mathcal{F}} H_1$ and $G_2 \equiv_{\mathcal{F}} H_2$ then $G_1 \times G_2 \equiv_{\mathcal{F}} H_1 \times H_2$.

Properties of Homomorphism Indistinguishability Relations

Let's forget about the graph class $\mathcal{F}$ and think of the equivalence relation $\equiv_{\mathcal{F}}$!

Observation ($\equiv_{\mathcal{F}}$ is preserved under categorical products)

If $G_1 \equiv_{\mathcal{F}} H_1$ and $G_2 \equiv_{\mathcal{F}} H_2$ then $G_1 \times G_2 \equiv_{\mathcal{F}} H_1 \times H_2$.

The $\hom(F, -)$-functor maps products to products.

In the language of Marsden, Jakl, Shah (2023): There is a Kleisli law for the product functor $(G, H) \mapsto G \times H$.

Properties of Homomorphism Indistinguishability Relations

Closure properties of $\mathcal{F}$ correspond to preservation properties of $\equiv_{\mathcal{F}}$.

Properties of Homomorphism Indistinguishability Relations

Closure properties of $\mathcal{F}$ correspond to preservation properties of $\equiv_{\mathcal{F}}$.

Theorem (S. (2023))

For every homomorphism distinguishing closed graph class $\mathcal{F}$, tfae:

$\mathcal{F}$ is closed under	***$\equiv_{\mathcal{F}}$ is preserved under***	
minors	*complements*	$G \mapsto \overline{G}$

Properties of Homomorphism Indistinguishability Relations

Closure properties of $\mathcal{F}$ correspond to preservation properties of $\equiv_{\mathcal{F}}$.

Theorem (S. (2023))

For every homomorphism distinguishing closed graph class $\mathcal{F}$, tfae:

$\mathcal{F}$ is closed under	***$\equiv_{\mathcal{F}}$ is preserved under***	
minors	*complements*	$G \mapsto \overline{G}$
summands	*disjoint unions*	$(G, H) \mapsto G + H$

Properties of Homomorphism Indistinguishability Relations

Closure properties of $\mathcal{F}$ correspond to preservation properties of $\equiv_{\mathcal{F}}$.

Theorem (S. (2023))

For every homomorphism distinguishing closed graph class $\mathcal{F}$, tfae:

$\mathcal{F}$ is closed under	***$\equiv_{\mathcal{F}}$ is preserved under***	
minors	*complements*	$G \mapsto \overline{G}$
summands	*disjoint unions*	$(G, H) \mapsto G + H$
subgraphs	*full complements*	$G \mapsto \widehat{G}$
induced subgraphs	*left lexicographic products*	$H \mapsto G[H]$ *for every* G
contracting edges	*right lexicographic products*	$G \mapsto G[H]$ *for every* H.

Logic meets Minors

Logic meets Minors

Corollary (S. (2023))

Let $\equiv$ be an equivalence relation between graphs.

Logic meets Minors

Corollary (S. (2023))

Let $\equiv$ be an equivalence relation between graphs. If

- $G \equiv H \iff \overline{G} \equiv \overline{H}$ *for all G and H,*

Logic meets Minors

Corollary (S. (2023))

Let $\equiv$ be an equivalence relation between graphs. If

- $G \equiv H \iff \overline{G} \equiv \overline{H}$ *for all G and H,*
- $\equiv$ *is a homomorphism indistinguishability relation, then*

Logic meets Minors

Corollary (S. (2023))

Let $\equiv$ be an equivalence relation between graphs. If

- $G \equiv H \iff \overline{G} \equiv \overline{H}$ *for all G and H,*
- $\equiv$ *is a homomorphism indistinguishability relation, then*

$\equiv$ *is a homomorphism indistinguishability over a minor-closed graph class.*

Logic meets Minors

Corollary (S. (2023))

Let $\equiv$ be an equivalence relation between graphs. If

- $G \equiv H \iff \overline{G} \equiv \overline{H}$ *for all G and H,*
- $\equiv$ *is a homomorphism indistinguishability relation, then*

$\equiv$ *is a homomorphism indistinguishability over a minor-closed graph class.*

Examples include logical equivalences and systems of equations.

Logic meets Minors

Corollary (S. (2023))

Let $\equiv$ be an equivalence relation between graphs. If

- $G \equiv H \iff \overline{G} \equiv \overline{H}$ *for all G and H,*
- $\equiv$ *is a homomorphism indistinguishability relation, then*

$\equiv$ *is a homomorphism indistinguishability over a minor-closed graph class.*

Examples include logical equivalences and systems of equations.

Corollary (Atserias et al. (2021))

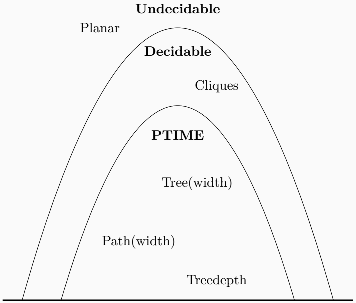
$\equiv_{\mathsf{FO}_k}$ *is not a homomorphism indistinguishability relation.*

Open Questions

Open Questions I

What is the complexity of deciding whether G and H are homomorphism indistinguishable over $\mathcal{F}$?

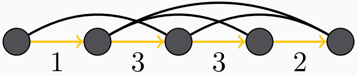
- Succinct matrix equations yield algorithms
- …may be used to prove undecidability



Open Questions II

Can matrix equations be cooked up for other graph classes?

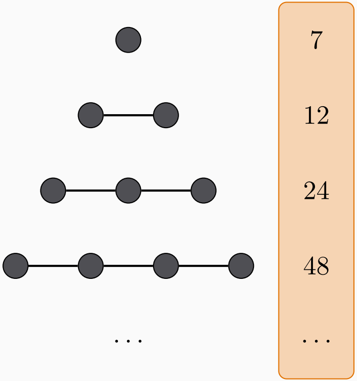
- path-like or tree-like graph classes, e.g. bounded cutwidth
- with comonadic strategy, only finite generation seems to be an issue



Open Questions III

When is a function $h\colon \mathcal{F} \to \mathbb{N}$ such that $h = \hom(-, H)$ for some graph H?

- Lovász and Schrijver (2009) answer this for $\mathcal{F} = \{\text{all graphs}\}$ using algebras of labelled graphs
- Applications in reconstruction



7

12

24

48

$\dots$

Lovász and Schrijver (2009)

Let $\mathcal{C}$ be a category such that

- $\mathcal{C}$ is locally finite,
- $\mathcal{C}$ has pushouts and an initial object 0,
- every morphism is the product of an epimorphism and a monomorphism,
- there is a *generator* $G \in \operatorname{obj} \mathcal{C}$, i.e. $\forall F \exists n \in \mathbb{N}.\ nG \twoheadrightarrow F$.

Lovász and Schrijver (2009)

Let $\mathcal{C}$ be a category such that

- $\mathcal{C}$ is locally finite,
- $\mathcal{C}$ has pushouts and an initial object 0,
- every morphism is the product of an epimorphism and a monomorphism,
- there is a *generator* $G \in \operatorname{obj} \mathcal{C}$, i.e. $\forall F \exists n \in \mathbb{N}.\ nG \twoheadrightarrow F$.

Then $h \colon \operatorname{obj} \mathcal{C} \to \mathbb{R}$ is of the form $h = \hom(-, H)$ if and only if

- $h(0) = 1$,
- h is multiplicative over coproducts,
- the matrix $N(h, L)$ is positive semidefinite for every L.

Lovász and Schrijver (2009)

Let $\mathcal{C}$ be a category such that

- $\mathcal{C}$ is locally finite,
- $\mathcal{C}$ has pushouts and an initial object 0,
- every morphism is the product of an epimorphism and a monomorphism,
- there is a *generator* $G \in \operatorname{obj} \mathcal{C}$, i.e. $\forall F \exists n \in \mathbb{N}.\ nG \twoheadrightarrow F$.

Then $h\colon \operatorname{obj} \mathcal{C} \to \mathbb{R}$ is of the form $h = \hom(-, H)$ if and only if

- $h(0) = 1$,
- h is multiplicative over coproducts,
- the matrix $N(h, L)$ is positive semidefinite for every L.

Question

Characterise $h\colon \operatorname{im} U^{\mathbb{C}} \to \mathbb{R}$ *of the form* $h = \hom_{\Sigma}(-, H) = \hom_{\mathrm{EM}(\mathbb{C})}(-, F^{\mathbb{C}} H)$.

Conclusion

- Matrix Equations for Homomorphism Indistinguishability

Conclusion

- Matrix Equations for Homomorphism Indistinguishability
    - (bi)labelled graphs, operations, finite generation, representation

Conclusion

- Matrix Equations for Homomorphism Indistinguishability
    - (bi)labelled graphs, operations, finite generation, representation
    - versions of Specht–Wiegmann Theorem

Conclusion

- Matrix Equations for Homomorphism Indistinguishability
    - (bi)labelled graphs, operations, finite generation, representation
    - versions of Specht–Wiegmann Theorem
- Towards a Theory of Homomorphism Indistinguishability

Conclusion

- Matrix Equations for Homomorphism Indistinguishability
    - (bi)labelled graphs, operations, finite generation, representation
    - versions of Specht–Wiegmann Theorem
- Towards a Theory of Homomorphism Indistinguishability
    - Roberson’s Conjecture

Conclusion

- Matrix Equations for Homomorphism Indistinguishability
  - (bi)labelled graphs, operations, finite generation, representation
  - versions of Specht–Wiegmann Theorem
- Towards a Theory of Homomorphism Indistinguishability
  - Roberson’s Conjecture
  - properties of homomorphism indistinguishability relations

Conclusion

- Matrix Equations for Homomorphism Indistinguishability
  - (bi)labelled graphs, operations, finite generation, representation
  - versions of Specht–Wiegmann Theorem
- Towards a Theory of Homomorphism Indistinguishability
  - Roberson's Conjecture
  - properties of homomorphism indistinguishability relations
- **Check out Grohe et al. (2022); Rattan and Seppelt (2023); Roberson and Seppelt (2023); Seppelt (2023)**!

References

Atserias, A., Kolaitis, P. G., and Wu, W.-L. (2021). On the Expressive Power of Homomorphism Counts. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pages 1–13.

Atserias, A. and Maneva, E. (2012). Sherali–Adams Relaxations and Indistinguishability in Counting Logics. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pages 367–379, New York, NY, USA. Association for Computing Machinery.

Bibliography ii

Atserias, A. and Ochremiak, J. (2018). Definable ellipsoid method, sums-of-squares proofs, and the isomorphism problem. In Dawar, A. and Grädel, E., editors, *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, pages 66–75. ACM.

Dawar, A., Jakl, T., and Reggio, L. (2021). Lovász-type theorems and game comonads. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pages 1–13. IEEE.

Dell, H., Grohe, M., and Rattan, G. (2018). Lovász Meets Weisfeiler and Leman. In Chatzigiannakis, I., Kaklamanis, C., Marx, D., and Sannella, D., editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 40:1–40:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Dvořák, Z. (2010). On recognizing graphs by numbers of homomorphisms. *Journal of Graph Theory*, 64(4):330–342.

Bibliography iii

Grohe, M. (2020). word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data. In Suciu, D., Tao, Y., and Wei, Z., editors, *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2020, Portland, OR, USA, June 14-19, 2020*, pages 1–16. ACM.

Grohe, M. and Otto, M. (2015). Pebble Games and Linear Equations. *J. Symb. Log.*, 80(3):797–844.

Grohe, M., Rattan, G., and Seppelt, T. (2022). Homomorphism Tensors and Linear Equations. In Bojańczyk, M., Merelli, E., and Woodruff, D. P., editors, *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*, volume 229 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 70:1–70:20, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISSN: 1868-8969.

Lovász, L. (1967). Operations with structures. *Acta Mathematica Academiae Scientiarum Hungarica*, 18(3):321–328.

Bibliography iv

Lovász, L. and Schrijver, A. (2009). Semidefinite Functions on Categories. *Electron. J. Comb.*, 16(2).

Mančinska, L. and Roberson, D. E. (2020). Quantum isomorphism is equivalent to equality of homomorphism counts from planar graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 661–672.

Neuen, D. (2023). Homomorphism-Distinguishing Closedness for Graphs of Bounded Tree-Width. arXiv:2304.07011 [cs, math].

Rattan, G. and Seppelt, T. (2023). Weisfeiler-Leman and Graph Spectra. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2268–2285. _eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611977554.ch87.

Roberson, D. E. (2022). Oddomorphisms and homomorphism indistinguishability over graphs of bounded degree. Number: arXiv:2206.10321.

Bibliography v

Roberson, D. E. and Seppelt, T. (2023). Lasserre Hierarchy for Graph Isomorphism and Homomorphism Indistinguishability. arXiv:2302.10538 [cs, math].

Seppelt, T. (2023). Logical Equivalences, Homomorphism Indistinguishability, and Forbidden Minors. arXiv:2302.11290 [cs, math].

Specht, W. (1940). Zur Theorie der Matrizen. II. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 50:19–23.

Tinhofer, G. (1986). Graph isomorphism and theorems of Birkhoff type. *Computing*, 36(4):285–300.

Wiegmann, N. A. (1961). Necessary and sufficient conditions for unitary similarity. *Journal of the Australian Mathematical Society*, 2(1):122–126. Edition: 2009/04/09 Publisher: Cambridge University Press.

Bibliography vi

Title Picture: “Bicycle race scene. A peloton of six cyclists crosses the finish line in front of a crowded grandstand, observed by a referee.” (1895) by Calvert Lithographic Co., Detroit, Michigan, Public Domain, via Wikimedia Commons. `https://commons.wikimedia.org/wiki/File:Bicycle_race_scene,_1895.jpg`