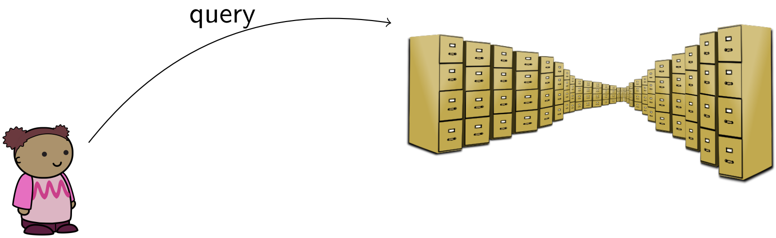
# Definability of Cai-Fürer-Immerman Problems in Choiceless Polynomial Time

Wied Pakusa, *Svenja Schalthöfer*, Erkal Selman

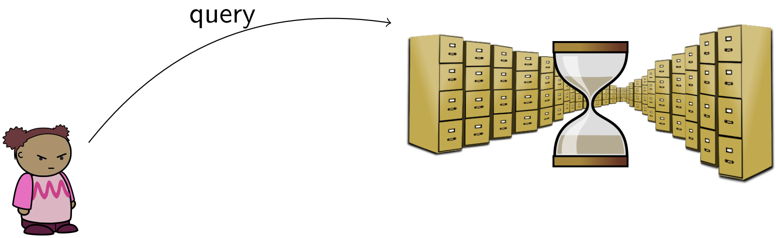
CSL 2016

# Origin: Database theory

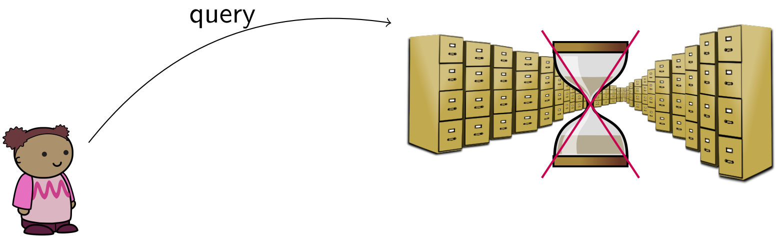
# Origin: Database theory



# Origin: Database theory



# Origin: Database theory



# The most important problem in finite model theory

## Question (Chandra, Harel 1982)

Is there a database query language expressing exactly the efficiently computable queries?

# The most important problem in finite model theory

### Question (Chandra, Harel 1982)

Is there a database query language expressing exactly the efficiently computable queries?

### Question (Gurevich 1988)

Is there a logic capturing PTIME?

# What is a logic capturing PTIME?

Informal definition: A logic $L$ captures $\textsc{Ptime}$ if it defines precisely those properties of finite structures that are decidable in polynomial time:

1. For every sentence $\psi \in L$, the set of finite models of $\psi$ is decidable in polynomial time.
2. For every $\textsc{Ptime}$-property $S$ of finite structures, there is a sentence $\psi \in L$ such that $S = \{\mathfrak{A} \in \mathsf{Fin} : \mathfrak{A} \models \psi\}$.

# The most important problem in finite model theory

**Question (Chandra, Harel 1982)**

Is there a database query language expressing exactly the efficiently computable queries?

**Question (Gurevich 1988)**

Is there a logic capturing $\textsc{Ptime}$?

**Theorem (Fagin)**

$\exists\mathrm{SO}$ *captures NP.*

# Candidates for a logic capturing PTIME

Ptime

$\subsetneq$

FO cannot define transitive closure

# Candidates for a logic capturing PTIME

PTIME

$\subsetneq$

FP captures PTIME on ordered structures

$\subsetneq$

FO cannot define transitive closure

# Candidates for a logic capturing PTIME

PTIME

$\subsetneq$

FPC — captures PTIME on many interesting classes of structures.

$\subsetneq$

FP — captures PTIME on ordered structures

$\subsetneq$

FO — cannot define transitive closure

# Candidates for a logic capturing PTIME

$\textsc{Ptime}$

$\subseteq$ $\subseteq$

$\mathrm{FP} + \mathsf{rk}$ $\mathrm{CPT} + \mathrm{C}$

$\subsetneq$ $\subsetneq$

$\mathrm{FPC}$ captures $\textsc{Ptime}$ on many interesting classes of structures.

$\subsetneq$

$\mathrm{FP}$ captures $\textsc{Ptime}$ on ordered structures

$\subsetneq$

$\mathrm{FO}$ cannot define transitive closure

# Candidates for a logic capturing PTIME

$\textsc{Ptime}$

$\subsetneq$ $\quad$ $\subseteq$

$\mathrm{FP} + \mathsf{rk}$ $\quad$ $\mathrm{CPT} + \mathrm{C}$

captures $\textsc{Ptime}$ on even more classes, and

**Theorem (Dawar,Richerby,Rossman)**

*The Cai-Fürer-Immerman query over ordered graphs is* $\mathrm{CPT}$*-definable.*

$\subsetneq$ $\quad$ $\subsetneq$

$\mathrm{FPC}$ captures $\textsc{Ptime}$ on many interesting classes of structures.

$\subsetneq$

$\mathrm{FP}$ captures $\textsc{Ptime}$ on ordered structures

$\subsetneq$

$\mathrm{FO}$ cannot define transitive closure

# Candidates for a logic capturing PTIME

$\textsc{Ptime}$

$=?$

$\mathrm{FP} + \mathsf{rk}$ $\quad$ $\mathrm{CPT} + \mathrm{C}$

$\not\supseteq$ $\quad$ $\subsetneq$

captures $\textsc{Ptime}$ on even more classes, and

**Theorem (Dawar,Richerby,Rossman)**

*The Cai-Fürer-Immerman query over ordered graphs is $\mathrm{CPT}$-definable.*

$\mathrm{FPC}$ captures $\textsc{Ptime}$ on many interesting classes of structures.

$\subsetneq$

$\mathrm{FP}$ captures $\textsc{Ptime}$ on ordered structures

$\subsetneq$

$\mathrm{FO}$ cannot define transitive closure

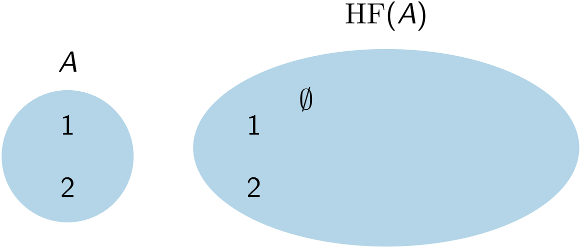
# The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

# The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

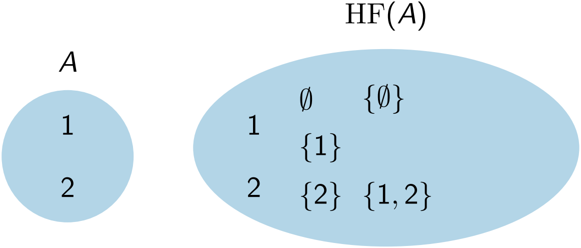
## Hereditarily finite sets



# The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

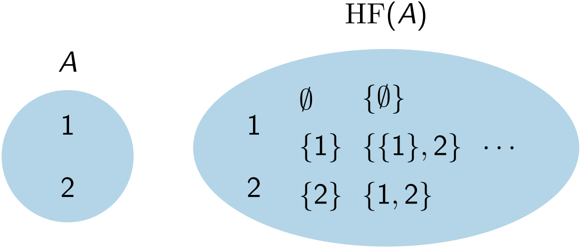
## Hereditarily finite sets



# The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets, polynomial resource bounds

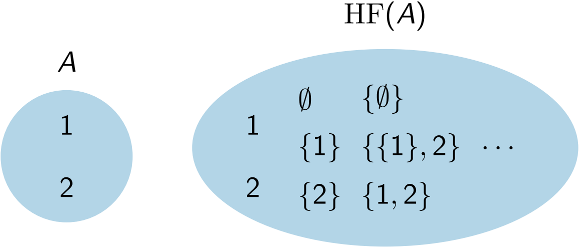
## Hereditarily finite sets



# The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

## Hereditarily finite sets



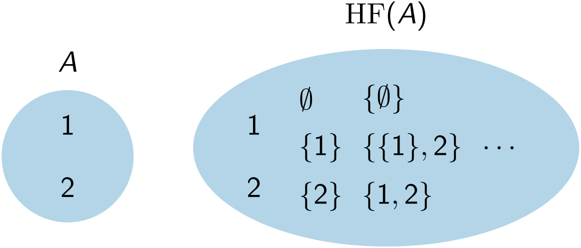

## Operations

- Set-theoretic operations (∅, ∈, ∪,...), boolean connectives
- Comprehension terms: $\{s(x) : x \in t : \varphi(x)\}$

# The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

## Hereditarily finite sets



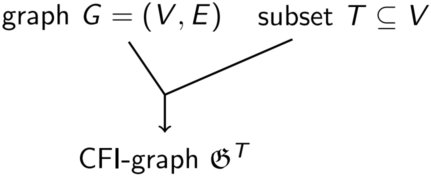

## Operations

- Set-theoretic operations ($\emptyset$, $\in$, $\cup$,...), boolean connectives
- Comprehension terms: $\{s(x) : x \in t : \varphi(x)\}$

# The benchmark: The Cai-Fürer-Immerman query

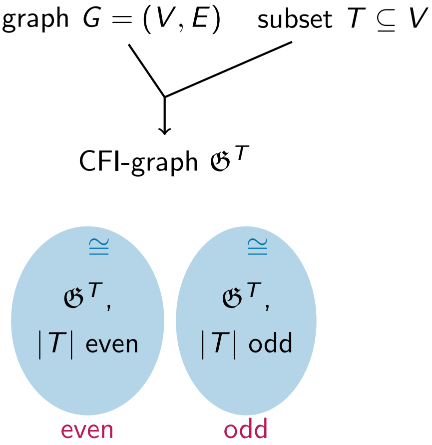
- PTIME-computable
- not FPC-definable
- benchmark

# The benchmark: The Cai-Fürer-Immerman query




- PTIME-computable
- not FPC-definable
- benchmark

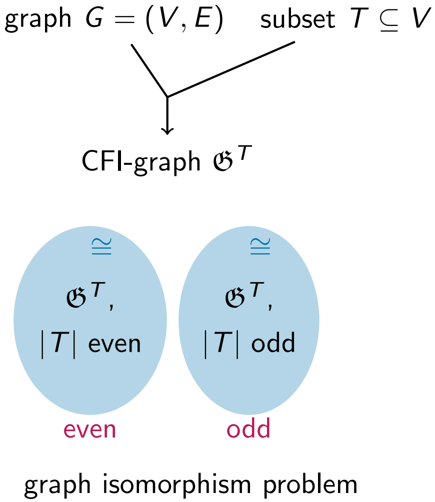
# The benchmark: The Cai-Fürer-Immerman query




- PTIME-computable
- not FPC-definable
- benchmark

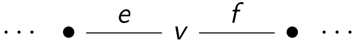
# The benchmark: The Cai-Fürer-Immerman query

- PTIME-computable
- not FPC-definable
- benchmark



# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

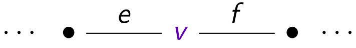


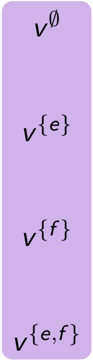

vertex $v$
$\rightsquigarrow$ gadget $v^*$

# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex $v$
$\rightsquigarrow$ gadget $v^*$

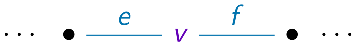


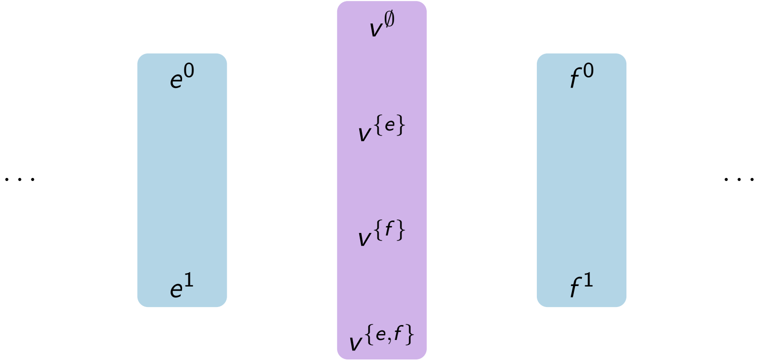



# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex $v$
$\rightsquigarrow$ gadget $v^*$

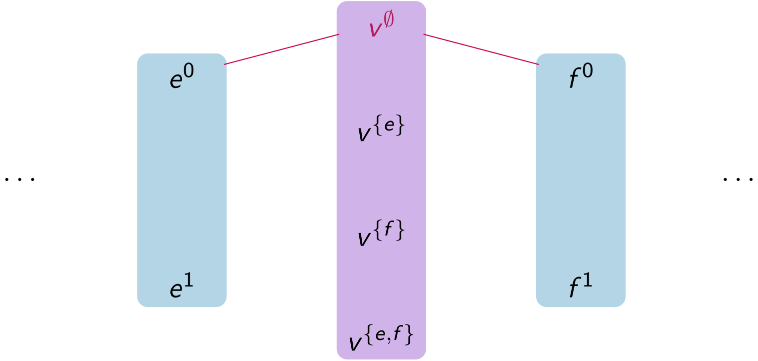


# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

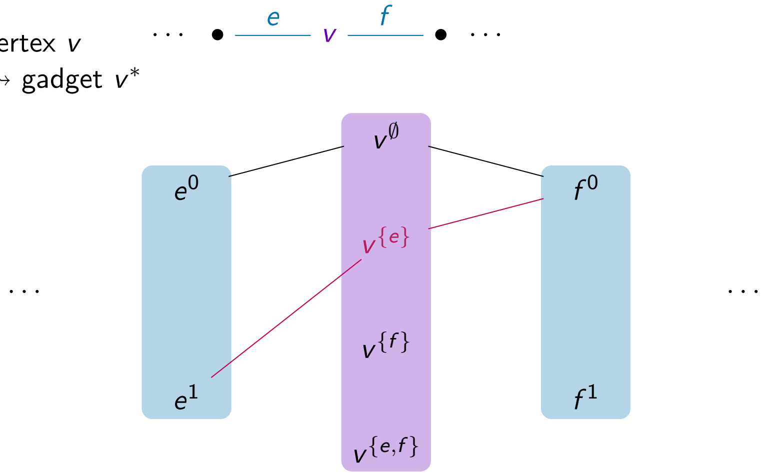

vertex $v$
$\rightsquigarrow$ gadget $v^*$



# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

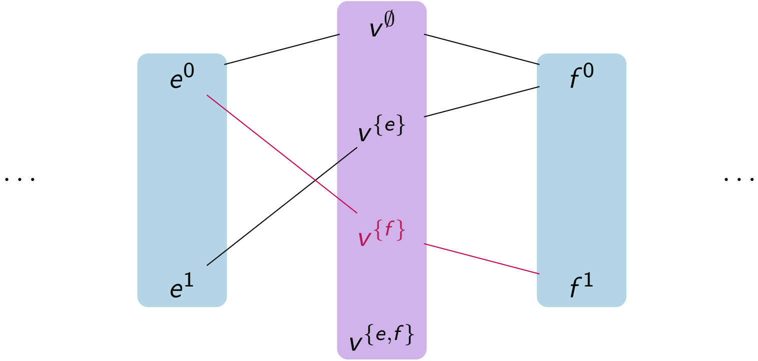
vertex $v$
$\rightsquigarrow$ gadget $v^*$



# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex $v$
$\rightsquigarrow$ gadget $v^*$

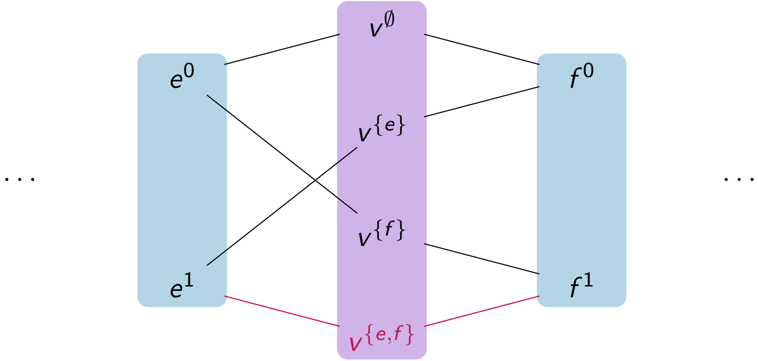



# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex $v$

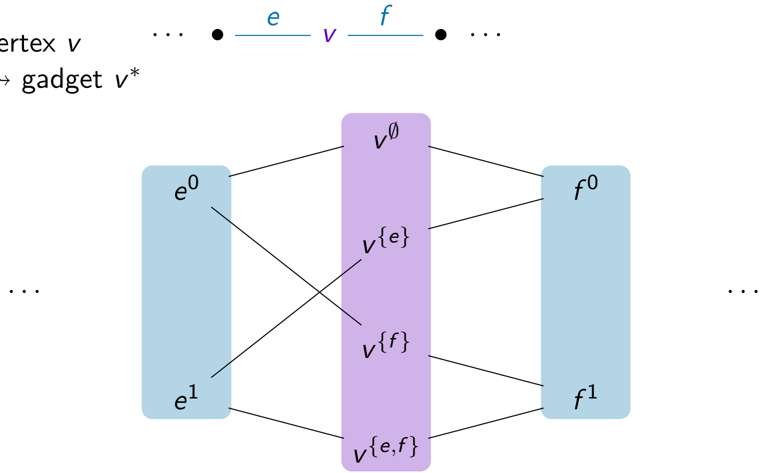
$\rightsquigarrow$ gadget $v^*$

# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}^T$

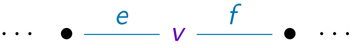
vertex $v$
$\rightsquigarrow$ gadget $v^*$



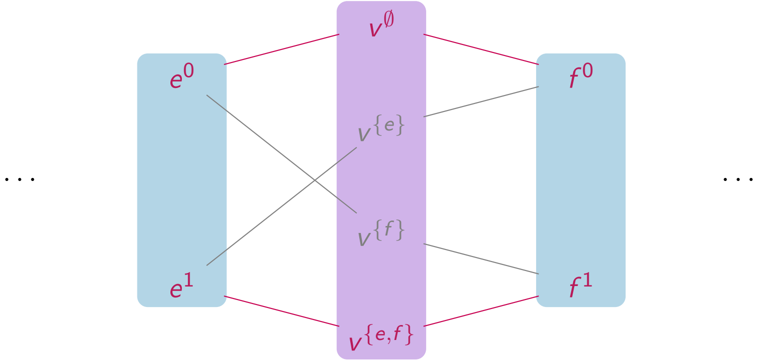
# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}^T$

vertex $v$




$\rightsquigarrow$ gadget $v^*$



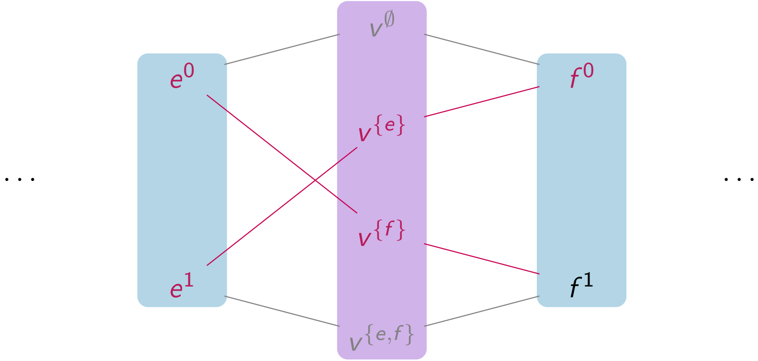

even gadget, $v \notin T$

# Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}^T$

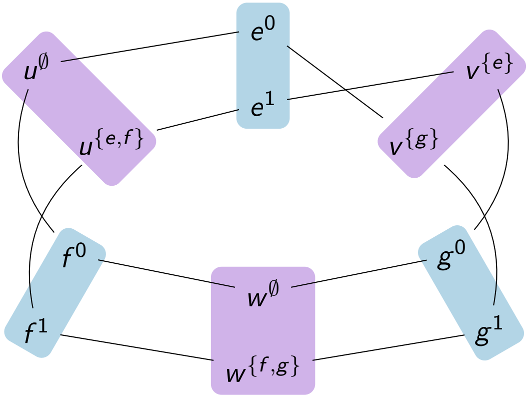
vertex $v$

$\rightsquigarrow$ gadget $v^*$


odd gadget, $v \in T$

# Cai-Fürer-Immerman graphs: example



# Results

**Theorem**
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.*

**Theorem**
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.*

**Theorem**
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

**Corollary**
$\approx$ *-free* $\mathrm{PIL} \not\equiv \mathrm{CPT}[\mathrm{rk} \leq k]$

Theorem
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.*

Theorem
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.*

Theorem
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

Corollary
$\approx$ *-free* $\mathrm{PIL} \neq \mathrm{CPT}[\mathrm{rk} \leq k]$

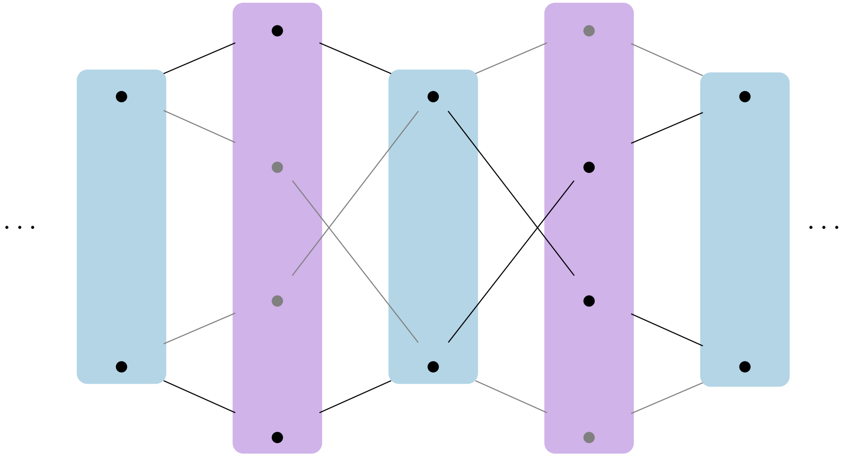
Theorem (Dawar,Richerby,Rossman)
*The CFI query over ordered graphs is* $\mathrm{CPT}$*-definable.*

# Computing the parity of CFI graphs

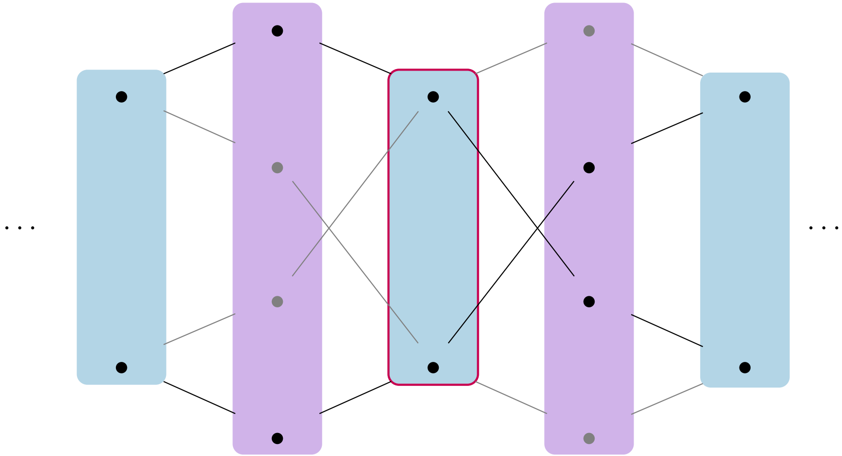
## Easy PTIME procedure

1. Label edge gadgets
2. Count odd vertex gadgets

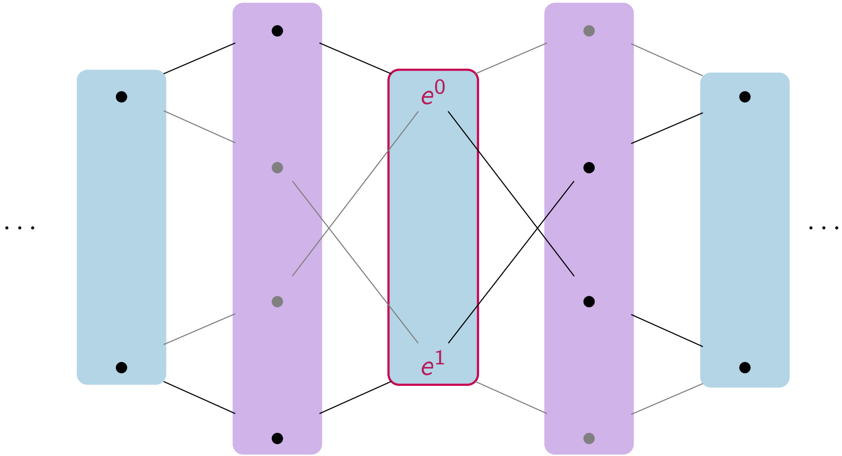
# Assigning edge labels



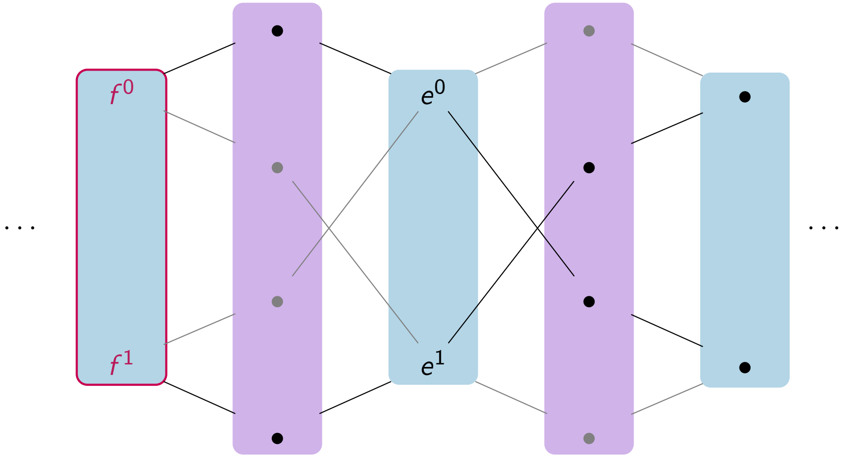
# Assigning edge labels



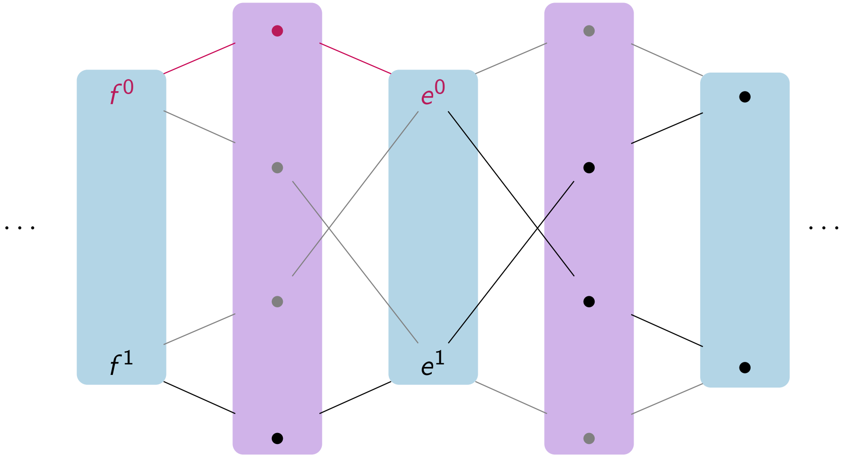
# Assigning edge labels



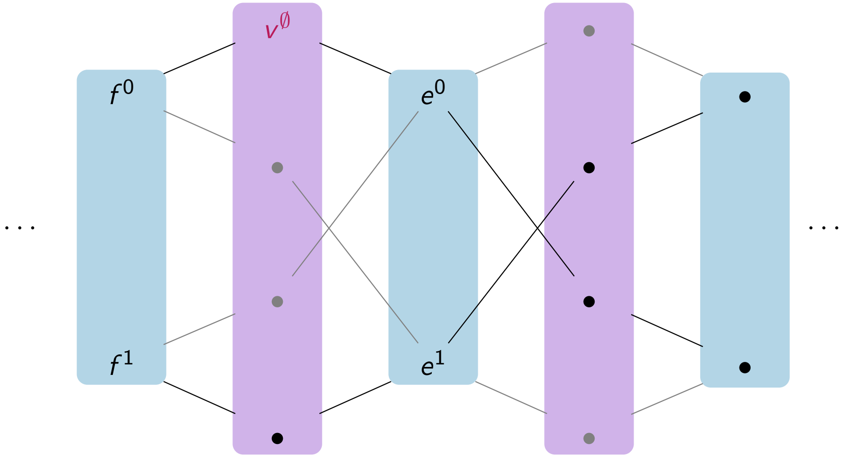
# Assigning edge labels



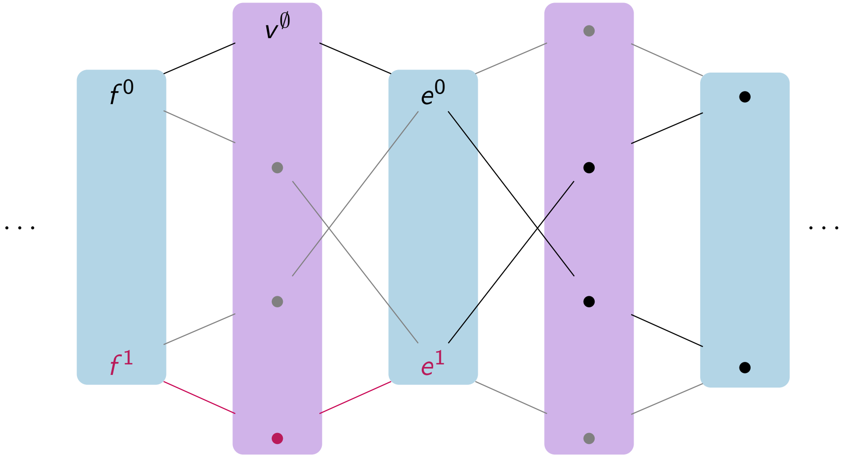
# Assigning edge labels



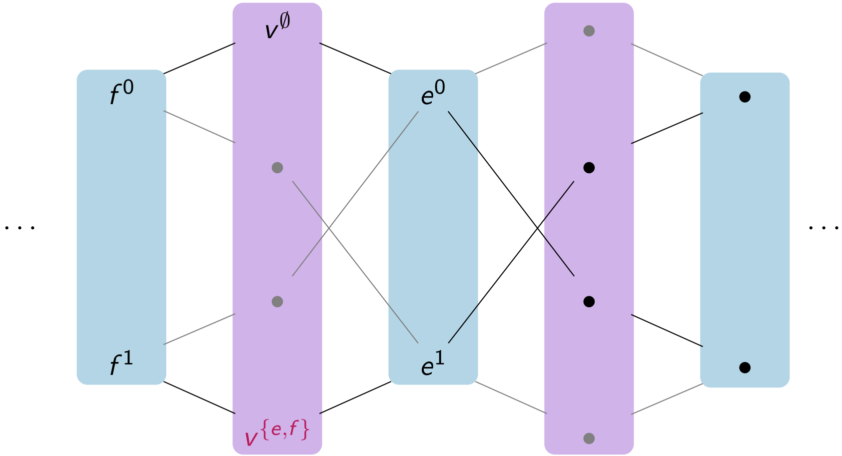
# Assigning edge labels



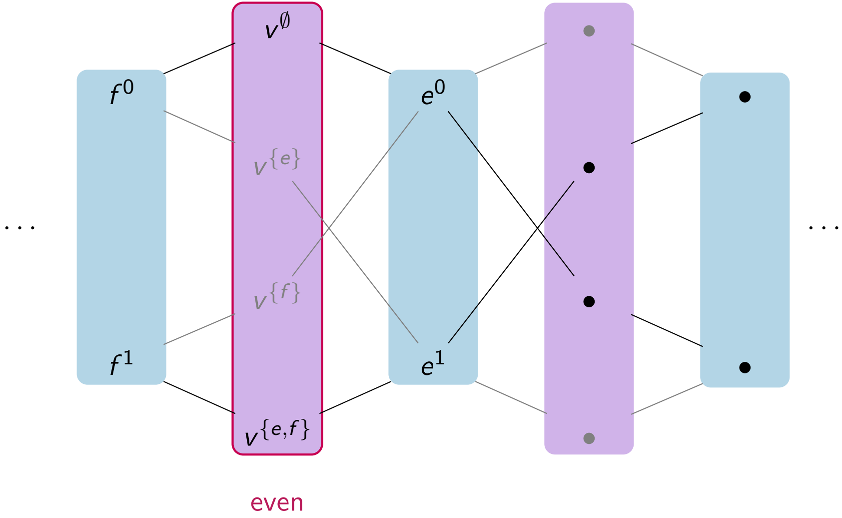
# Assigning edge labels



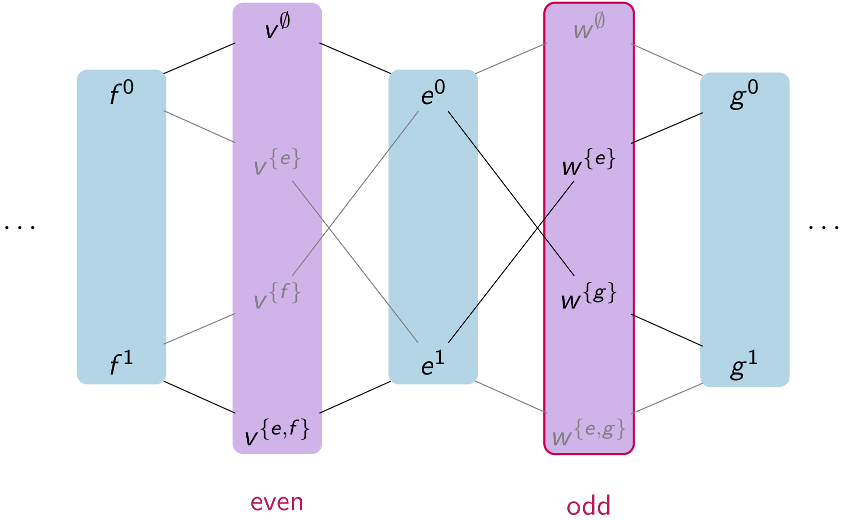
# Assigning edge labels



# Assigning edge labels



# Assigning edge labels



# Computing the parity of CFI graphs

## Easy PTIME procedure

1. Label edge gadgets
2. Count odd vertex gadgets

# Computing the parity of CFI graphs

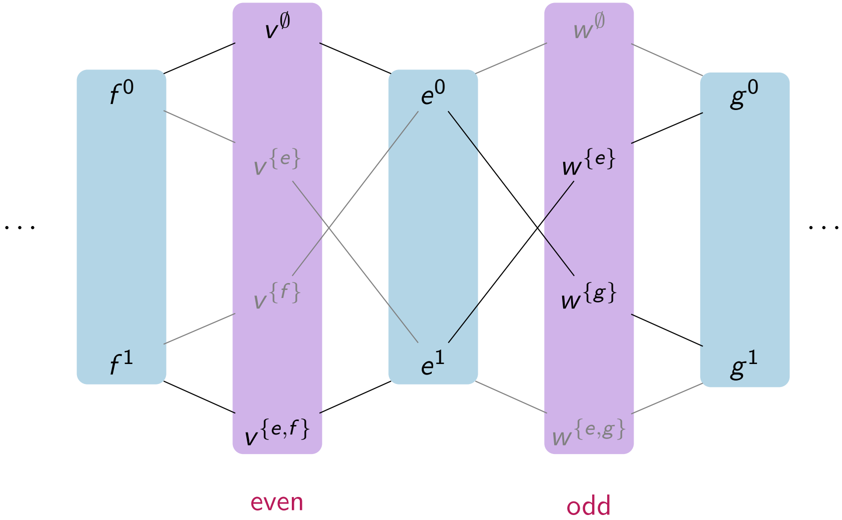
## Easy PTIME procedure

1. Label edge gadgets
2. Count odd vertex gadgets

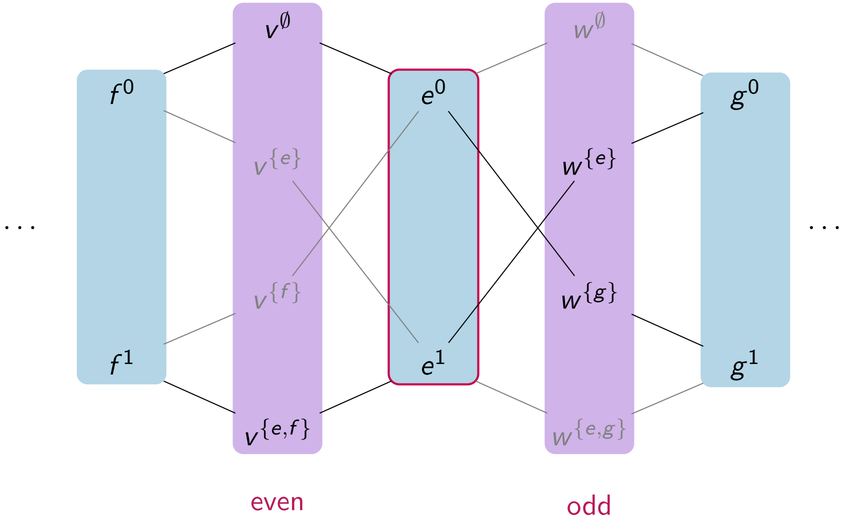
## CPT procedure (Dawar, Richerby, Rossman)

1. Construct super-symmetric objects
2. Label edge gadgets
3. Count odd vertex gadgets

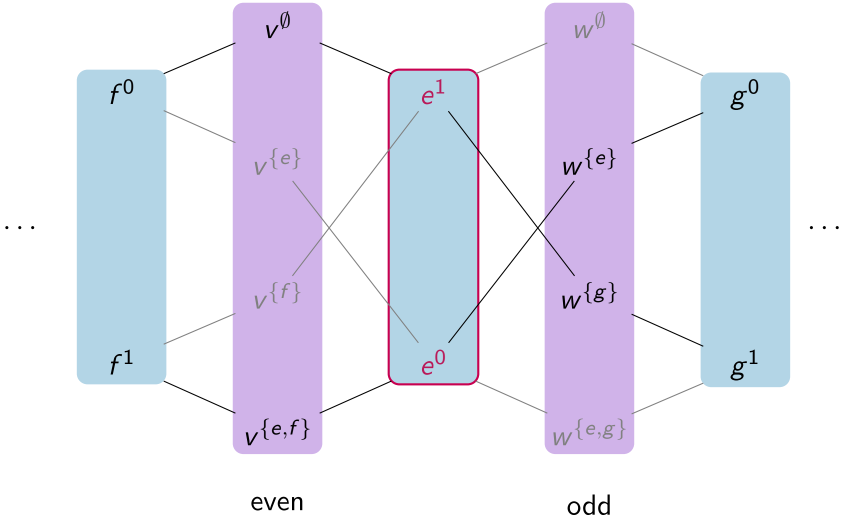
# Why super-symmetry is useful



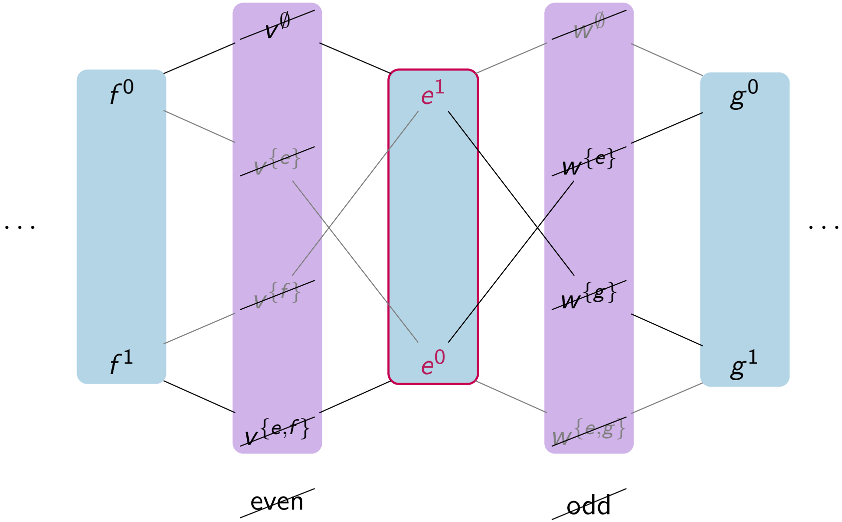
# Why super-symmetry is useful



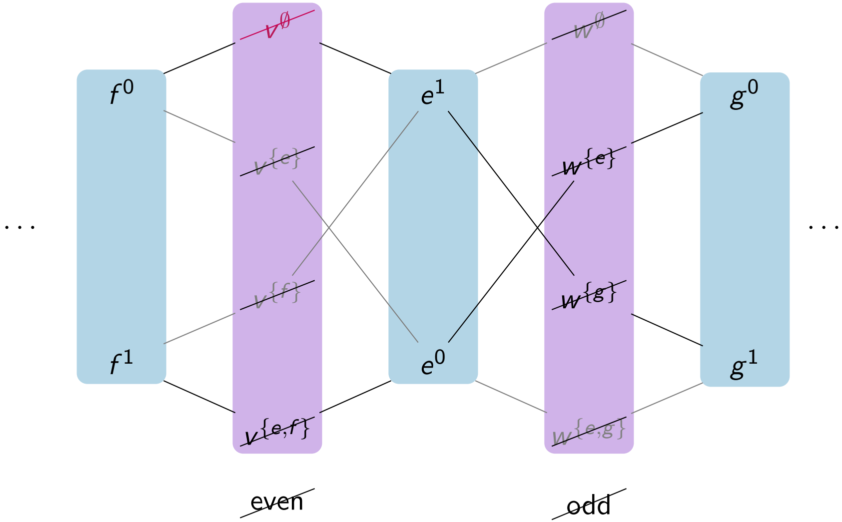
# Why super-symmetry is useful



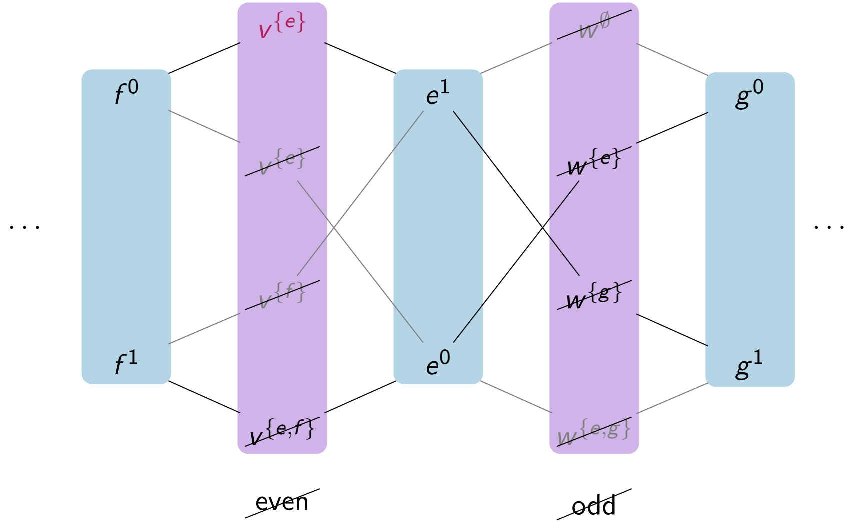
# Why super-symmetry is useful



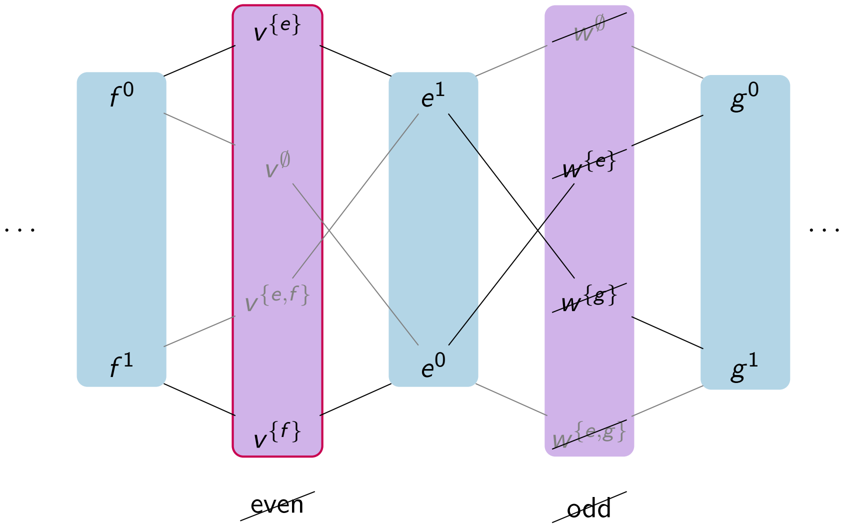
# Why super-symmetry is useful



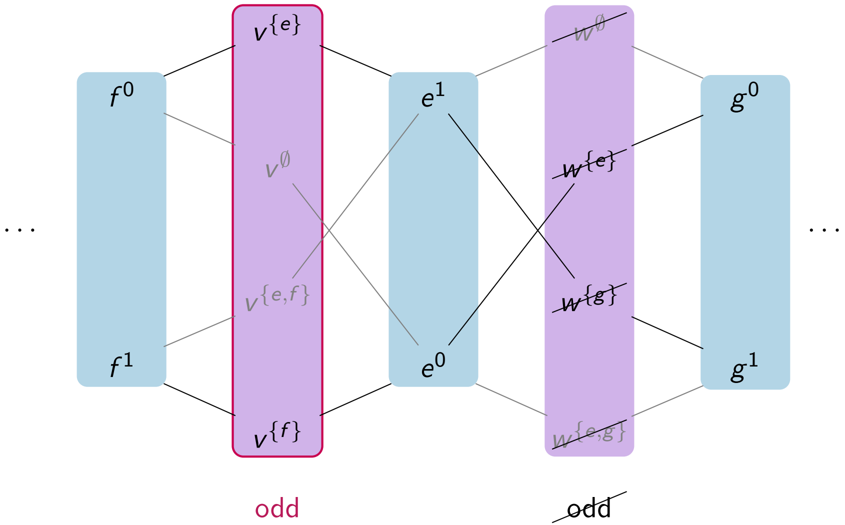
# Why super-symmetry is useful



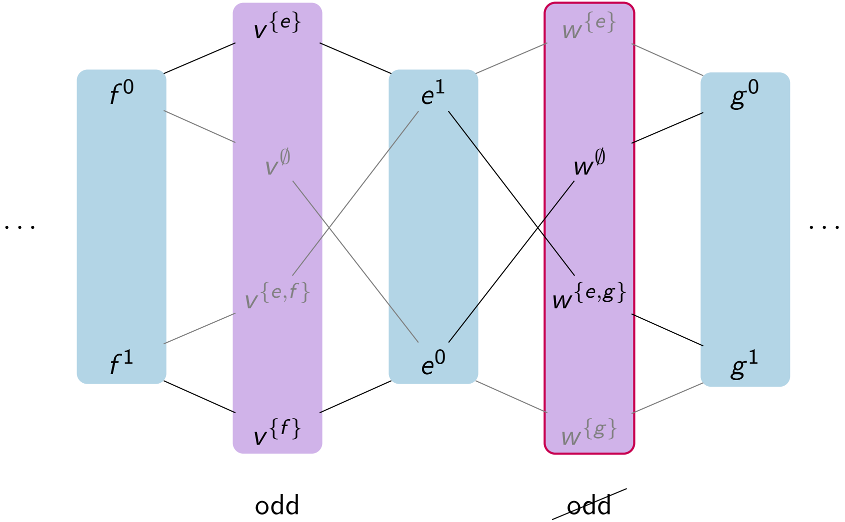
# Why super-symmetry is useful



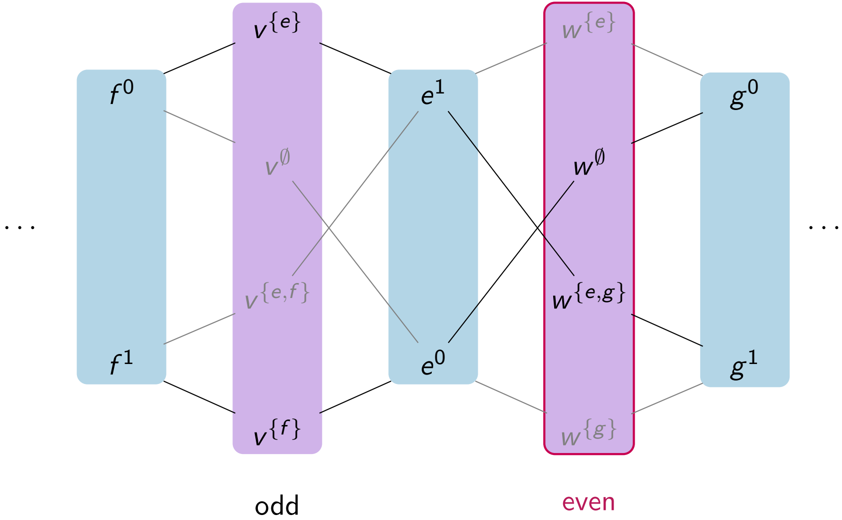
# Why super-symmetry is useful



# Why super-symmetry is useful



# Why super-symmetry is useful



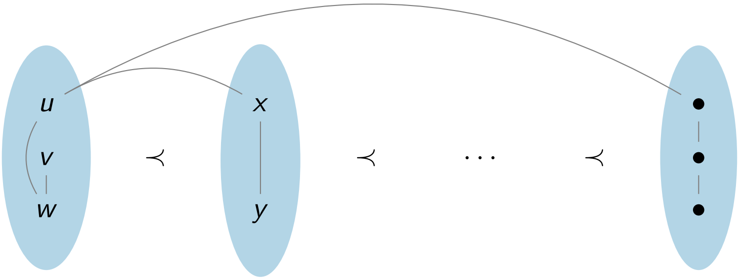
**Theorem**
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.*

**Theorem**
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.*

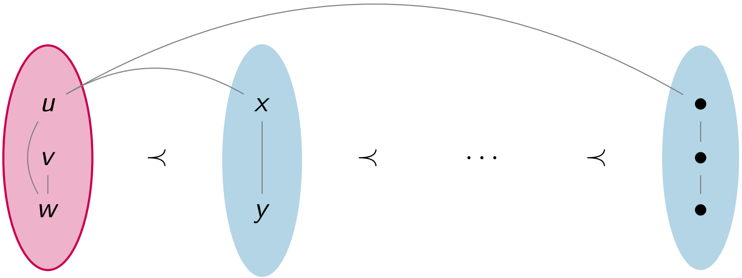
**Theorem**
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

**Corollary**
$\approx$ *-free* $\mathrm{PIL} \not\equiv \mathrm{CPT}[\mathrm{rk} \leq k]$

# Graphs with colour classes of logarithmic size

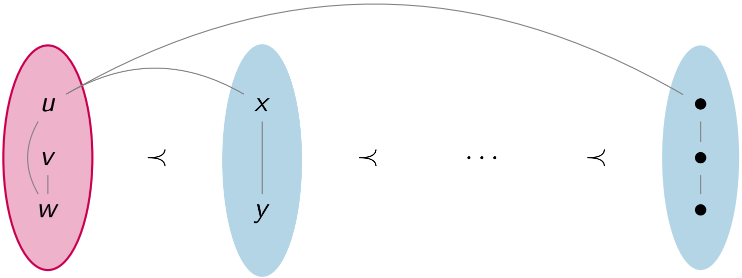


# Graphs with colour classes of logarithmic size




$\mu_{\{u\}}$ $\mu_{\{v\}}$ $\mu_{\{w\}}$

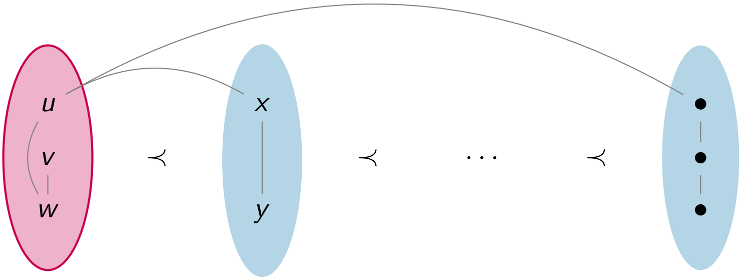
# Graphs with colour classes of logarithmic size




$\mu_{\{u\}}$ $\mu_{\{v\}}$ $\mu_{\{w\}}$

$\mu_{\{u,v\}}$ $\mu_{\{u,w\}}$ $\mu_{\{v,w\}}$

# Graphs with colour classes of logarithmic size

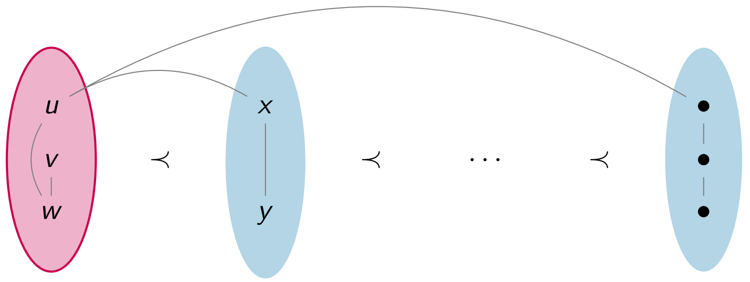



$\mu_{\{u\}}$ $\mu_{\{v\}}$ $\mu_{\{w\}}$

$\mu_{\{u,v\}}$ $\mu_{\{u,w\}}$ $\mu_{\{v,w\}}$

$\mu_{\{u,v,w\}}$

# Graphs with colour classes of logarithmic size



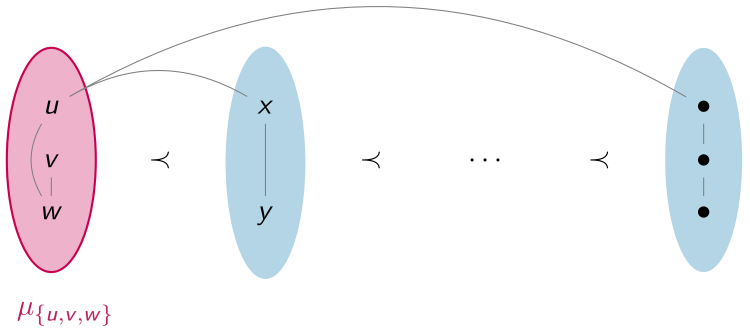

$\mu_{\{u\}}$ $\mu_{\{v\}}$ $\mu_{\{w\}}$

$\mu_{\{u,v\}}$ $\mu_{\{u,w\}}$ $\mu_{\{v,w\}}$

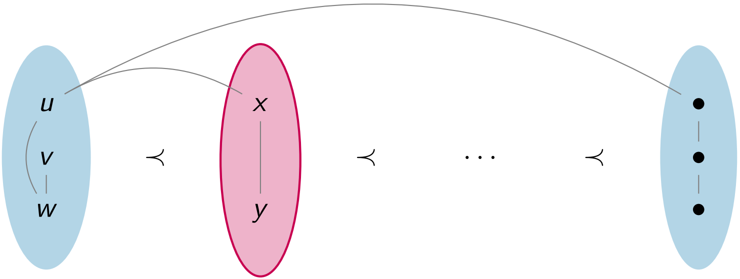
$\mu_{\{u,v,w\}}$

Construct $\mathcal{O}(2^{|C|})$ many sets

# Graphs with colour classes of logarithmic size



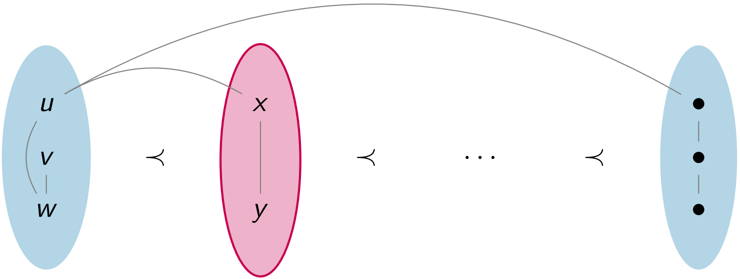
# Graphs with colour classes of logarithmic size




$\mu_{\{u,v,w\}}$

$\mu_{\{u,v,w,x\}}$ $\mu_{\{u,v,w,y\}}$

# Graphs with colour classes of logarithmic size

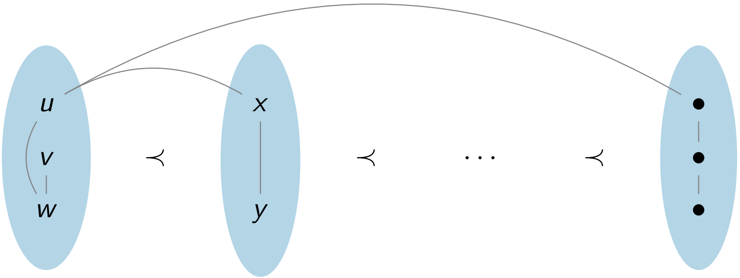



$\mu_{\{u,v,w\}}$

$\mu_{\{u,v,w,x\}}$ $\mu_{\{u,v,w,y\}}$

$\mu_{\{u,v,w,x,y\}}$

# Graphs with colour classes of logarithmic size




$\mu_{\{u,v,w\}}$

$\mu_{\{u,v,w,x\}}$ $\mu_{\{u,v,w,y\}}$

$\mu_{\{u,v,w,x,y\}}$ $\ddots$

$\mu_V$

# Results

**Theorem**
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.* ✓

**Theorem**
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.*

**Theorem**
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

**Corollary**
$\approx$ *-free* $\mathrm{PIL} \neq \mathrm{CPT}[\mathrm{rk} \leq k]$

Theorem

*The CFI query over graphs with logarithmic colour classes is $\mathrm{CPT}$-definable.* ✓

Theorem

*The CFI query over graphs with large degree is $\mathrm{CPT}$-definable using only sets of bounded rank.*

Theorem

*The CFI query over complete graphs is not $\mathrm{CPT}$-definable without using set-like objects.*

Corollary

$\approx$ *-free* $\mathrm{PIL} \not\equiv \mathrm{CPT}[\mathrm{rk} \leq k]$

**Theorem**
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.* ✓

**Theorem**
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.*

**Theorem**
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

**Corollary**
$\approx$ *-free* $\mathrm{PIL} \neq \mathrm{CPT}[\mathrm{rk} \leq k]$

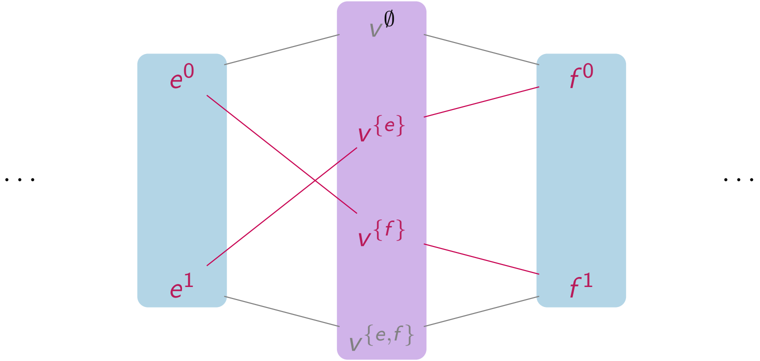
**Theorem (Dawar, Richerby, Rossman)**
*The CFI-query over ordered graphs is not definable in* $\mathrm{CPT}$ *using only sets of bounded rank.*

# Graphs with large degree: Keeping the rank small

- Access to all subsets of $V$

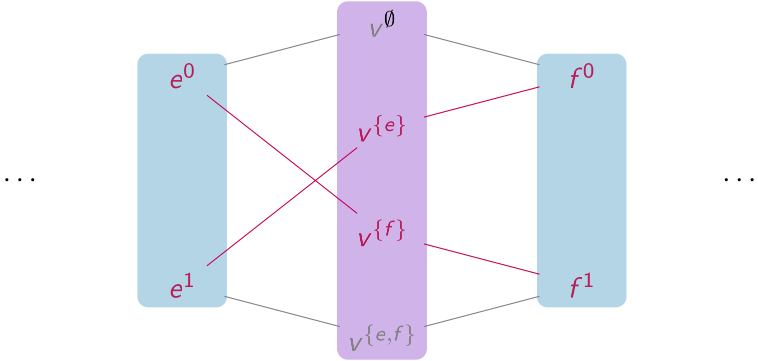
# Graphs with large degree: Keeping the rank small

- Access to all subsets of $V$



# Graphs with large degree: Keeping the rank small

- Access to all subsets of $V$
- Intuition: “Ordered” objects need nesting



**Theorem**
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.* ✓

**Theorem**
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.* ✓

**Theorem**
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

**Corollary**
$\approx$ *-free* $\mathrm{PIL} \not\equiv \mathrm{CPT}[\mathrm{rk} \leq k]$

# Results

**Theorem**
*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.* ✓

**Theorem**
*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.* ✓

**Theorem**
*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.*

**Corollary**
$\approx$ *-free* $\mathrm{PIL} \not\equiv \mathrm{CPT}[\mathrm{rk} \leq k]$

# Lower bound: Set-like objects are necessary

**Theorem**

*The CFI query over complete graphs is not definable by any* $\mathrm{CPT}$*-program using only sequence-like objects.*

# Lower bound: Set-like objects are necessary

### Theorem

*The CFI query over complete graphs is not definable by any* $\mathrm{CPT}$*-program using only sequence-like objects.*

### Technique (Dawar, Richerby, Rossman)

- $\mathrm{CPT}$-program $\rightsquigarrow$ $\mathcal{C}^k$-formula over HF($\mathfrak{A}$)

# Lower bound: Set-like objects are necessary

**Theorem**

*The CFI query over complete graphs is not definable by any $\mathrm{CPT}$-program using only sequence-like objects.*

**Technique (Dawar, Richerby, Rossman)**

sets with small support

- $\mathrm{CPT}$-program $\rightsquigarrow$ $\mathcal{C}^k$-formula over $\mathrm{HF}(\mathfrak{A})_\ell$

# Lower bound: Set-like objects are necessary

**Theorem**

*The CFI query over complete graphs is not definable by any $\mathrm{CPT}$-program using only sequence-like objects.*

**Technique (Dawar, Richerby, Rossman)**

sets with small support

- $\mathrm{CPT}$-program $\rightsquigarrow$ $\mathcal{C}^k$-formula over $\mathsf{HF}(\mathfrak{A})_\ell$
- Show $\mathsf{HF}(\mathfrak{A})_\ell \equiv^{\mathcal{C}^k} \mathsf{HF}(\mathfrak{B})_\ell$

pebble game

# Lower bound: Set-like objects are necessary

### Theorem

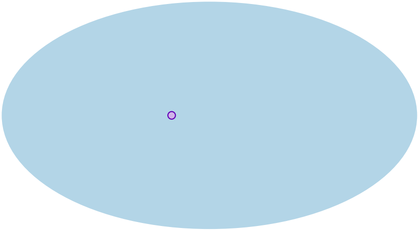
*The CFI query over complete graphs is not definable by any* $\mathrm{CPT}$*-program using only sequence-like objects.*

### Technique (Dawar, Richerby, Rossman)

sets with small support

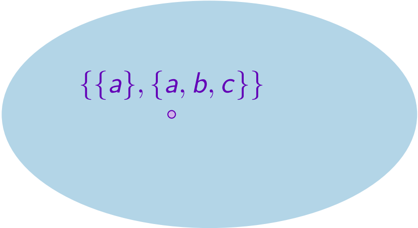
- $\mathrm{CPT}$-program $\rightsquigarrow$ $\mathcal{C}^k$-formula over $\mathsf{HF}(\mathfrak{A})_\ell$
- Show $\mathsf{HF}(\mathfrak{A})_\ell \equiv^{\mathcal{C}^k} \mathsf{HF}(\mathfrak{B})_\ell$ (pebble game)
- $\mathfrak{A} \equiv^{\mathcal{C}^{\ell \cdot k}} \mathfrak{B} \Rightarrow \mathsf{HF}(\mathfrak{A})_\ell \equiv^{\mathcal{C}^k} \mathsf{HF}(\mathfrak{B})_\ell$ if $\mathfrak{A}, \mathfrak{B}$ $\mathcal{C}^{\ell \cdot k}$-homogeneous

Move in HF($\mathfrak{A}$):



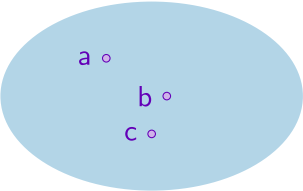
# Supports

Move in HF($\mathfrak{A}$):

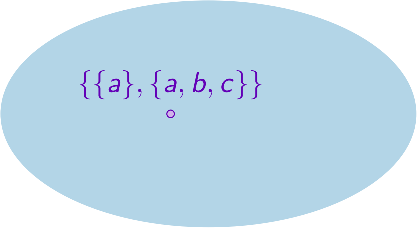


# Supports

"Corresponding" move in $\mathfrak{A}$:

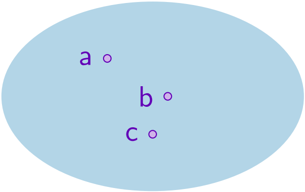



Move in HF($\mathfrak{A}$):

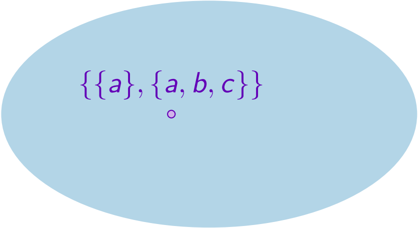


# Supports

"Corresponding" move in $\mathfrak{A}$:

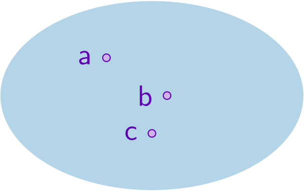



Move in HF($\mathfrak{A}$):

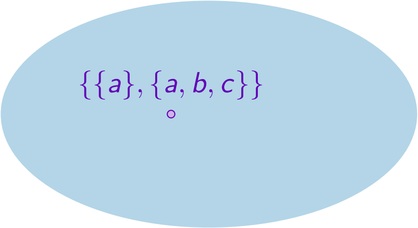



$S$ supports $x$ if $\mathrm{Stab}^{\bullet}(S) \leq \mathrm{Stab}(x)$

"Corresponding" move in $\mathfrak{A}$:




Move in HF($\mathfrak{A}$):




$S$ supports $x$ if $\mathrm{Stab}^{\bullet}(S) \leq \mathrm{Stab}(x)$

Example

$A = \{a, b, c, d, e\}$
Supports of $\{a, b, c\}$: $A$, $\{a, b, c\}$, $\{d, e\}$

# Lower bound: Set-like objects are necessary

**Theorem**

*The CFI query over complete graphs is not definable by any $\mathrm{CPT}$-program using only sequence-like objects.*

**Technique (Dawar, Richerby, Rossman)**

sets with small support

- $\mathrm{CPT}$-program $\rightsquigarrow$ $\mathcal{C}^k$-formula over $\mathsf{HF}(\mathfrak{A})_\ell$
- Show $\mathsf{HF}(\mathfrak{A})_\ell \equiv^{\mathcal{C}^k} \mathsf{HF}(\mathfrak{B})_\ell$

pebble game

- $\mathfrak{A} \equiv^{\mathcal{C}^{\ell \cdot k}} \mathfrak{B} \Rightarrow \mathsf{HF}(\mathfrak{A})_\ell \equiv^{\mathcal{C}^k} \mathsf{HF}(\mathfrak{B})_\ell$ if $\mathfrak{A}, \mathfrak{B}$ $\mathcal{C}^{\ell \cdot k}$-homogeneous

What are sequence-like objects?

# Sequence-like objects: strong supports

$v_1, \ldots, v_k$ vertices of $\mathcal{K}_n$.

$x = \{v_1, \ldots, v_k\}$

$y = (v_1, \ldots, v_k)$

# Sequence-like objects: strong supports

$v_1, \dots, v_k$ vertices of $\mathcal{K}_n$.

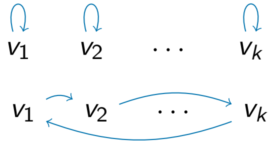
$x = \{v_1, \dots, v_k\}$


$y = (v_1, \dots, v_k)$

# Sequence-like objects: strong supports

$v_1, \ldots, v_k$ vertices of $\mathcal{K}_n$.

$x = \{v_1, \ldots, v_k\}$

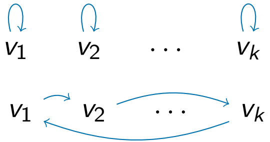



$y = (v_1, \ldots, v_k)$

# Sequence-like objects: strong supports

$v_1, \ldots, v_k$ vertices of $\mathcal{K}_n$.

$x = \{v_1, \ldots, v_k\}$




$y = (v_1, \ldots, v_k)$


$\rightsquigarrow$ strong support

**Theorem**

*The CFI query over graphs with logarithmic colour classes is* $\mathrm{CPT}$*-definable.* ✓

**Theorem**

*The CFI query over graphs with large degree is* $\mathrm{CPT}$*-definable using only sets of bounded rank.* ✓

**Theorem**

*The CFI query over complete graphs is not* $\mathrm{CPT}$*-definable without using set-like objects.* ✓

**Corollary**

$\approx$ *-free* $\mathrm{PIL} \not\equiv \mathrm{CPT}[\mathrm{rk} \leq k]$ ✓