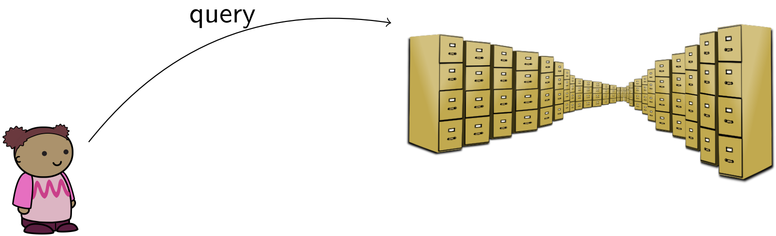
Definability of Cai-Fürer-Immerman Problems in Choiceless Polynomial Time

Wied Pakusa, *Svenja Schalthöfer*, Erkal Selman

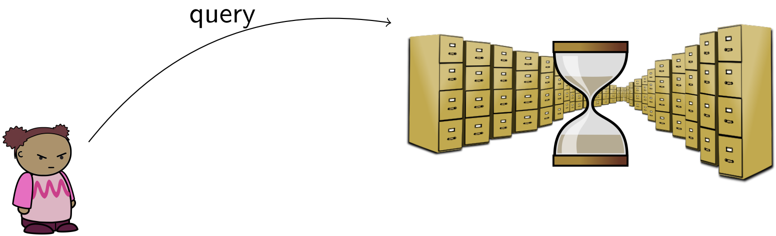
CSL 2016

Origin: Database theory

Origin: Database theory



Origin: Database theory



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 ψ 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

$\rotatebox{90}{$\subsetneq$}$

FPC — captures PTIME on many interesting classes of structures.

$\rotatebox{90}{$\subsetneq$}$

FP — captures PTIME on ordered structures

$\rotatebox{90}{$\subsetneq$}$

FO — cannot define transitive closure

Candidates for a logic capturing PTIME

$\textsc{Ptime}$

$\subseteq \quad \subseteq$

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

$\subsetneq \quad \subsetneq$

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

$\subsetneq$

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

$\subsetneq$

FO — cannot define transitive closure

Candidates for a logic capturing PTIME

$\textsc{Ptime}$

$\subseteq$ $\supseteq$

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

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

Theorem (Dawar,Richerby,Rossman)

The Cai-Fürer-Immerman query over ordered graphs is CPT*-definable.*

$\supsetneq$ $\subsetneq$

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

$\subsetneq$

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

$\subsetneq$

FO cannot define transitive closure

Candidates for a logic capturing PTIME

$\textsc{Ptime}$

$=?$

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

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

Theorem (Dawar,Richerby,Rossman)

The Cai-Fürer-Immerman query over ordered graphs is CPT*-definable.*

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

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

$\subsetneq$

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

$\subsetneq$

FO cannot define transitive closure

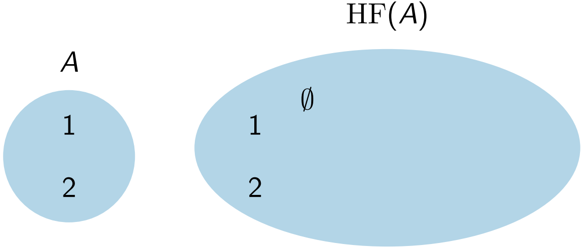
The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

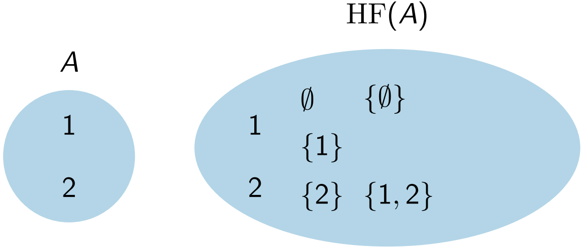
Hereditarily finite sets



The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

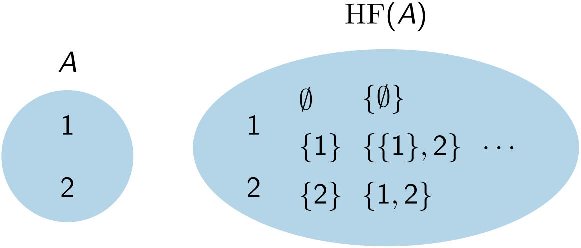
Hereditarily finite sets



The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

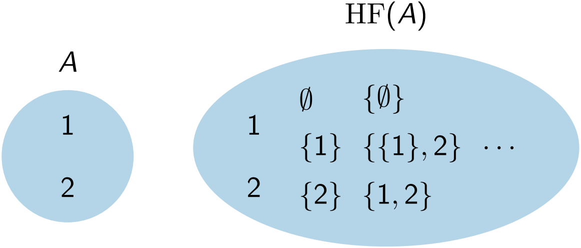
Hereditarily finite sets



The logic: Choiceless Polynomial Time

Iterated creation of hereditarily finite sets,
polynomial resource bounds

Hereditarily finite sets

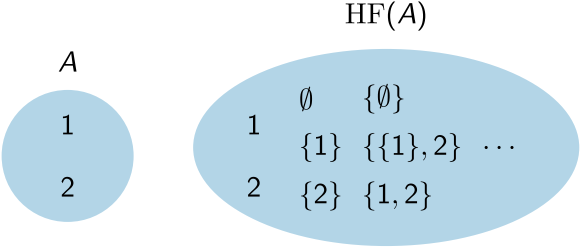


Operations

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

Iterated creation of hereditarily finite sets,
polynomial resource bounds

Hereditarily finite sets



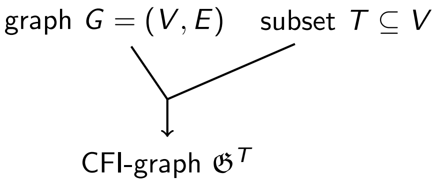
Operations

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

The benchmark: The Cai-Fürer-Immerman query

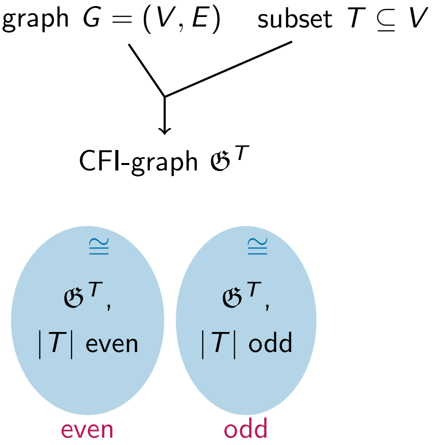
- $\mathrm{P\small{TIME}}$-computable
- not FPC-definable
- benchmark

The benchmark: The Cai-Fürer-Immerman query



- $\mathrm{P{\scriptsize TIME}}$-computable
- not FPC-definable
- benchmark

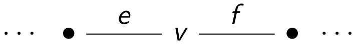
The benchmark: The Cai-Fürer-Immerman query



- PTIME-computable
- not FPC-definable
- benchmark

Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$



vertex v

$\rightsquigarrow$ gadget v^*

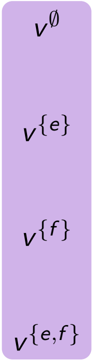
Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex v

$\rightsquigarrow$ gadget v^*

$$\cdots \quad \bullet \overset{e}{\text{———}} v \overset{f}{\text{———}} \bullet \quad \cdots$$

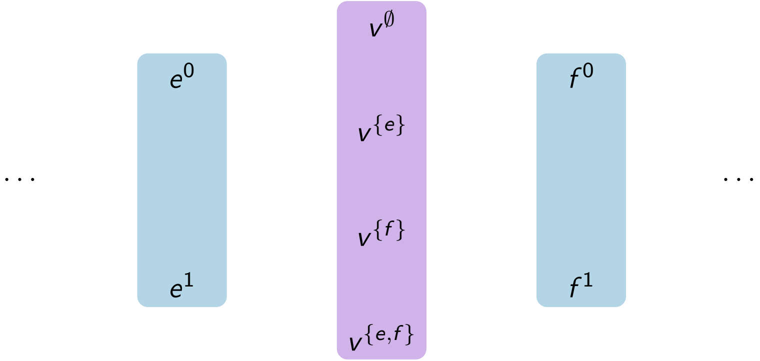


Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex v

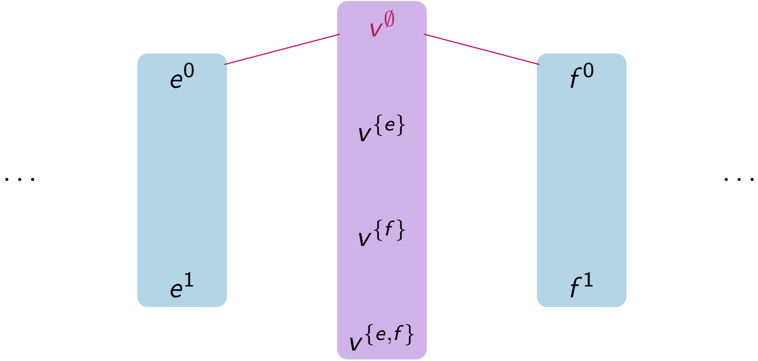
$\rightsquigarrow$ gadget v^*



Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

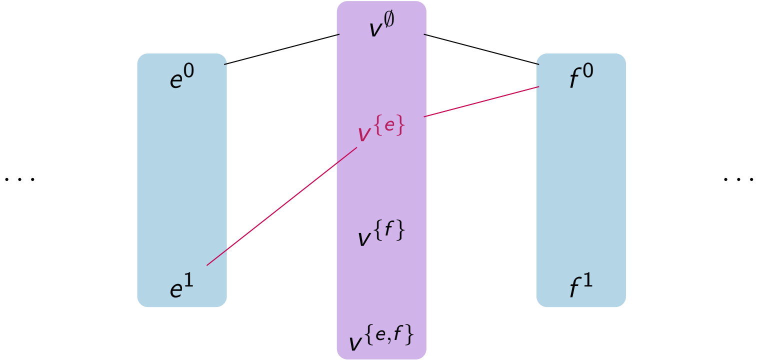
vertex v
$\rightsquigarrow$ gadget v^*



Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

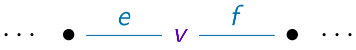
vertex v
$\rightsquigarrow$ gadget v^*

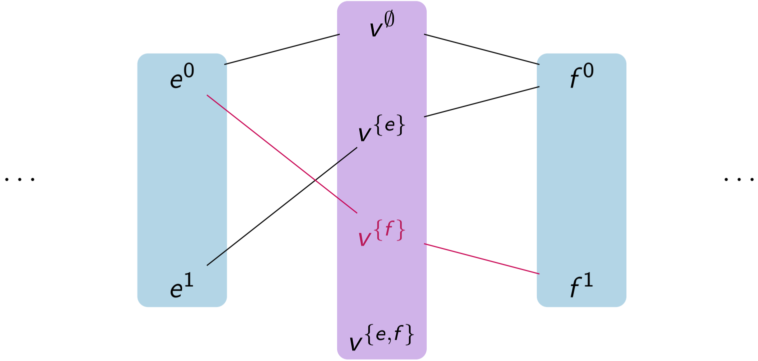


Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex v
$\rightsquigarrow$ gadget v^*



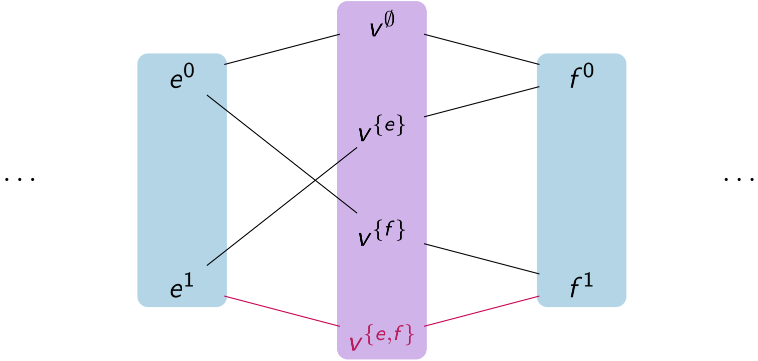


Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}$

vertex v

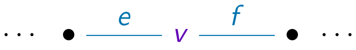
$\rightsquigarrow$ gadget v^*

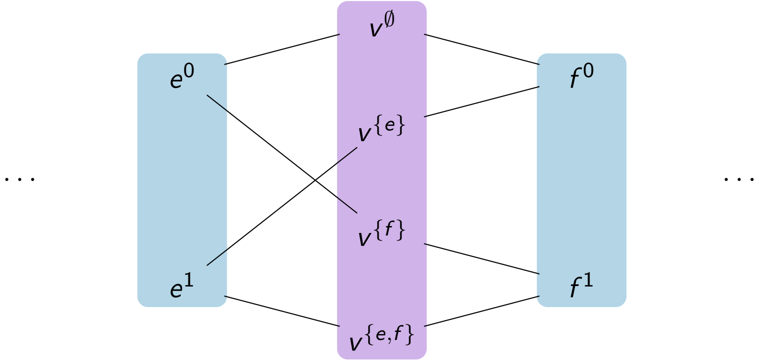


Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}^T$

vertex v
$\rightsquigarrow$ gadget v^*



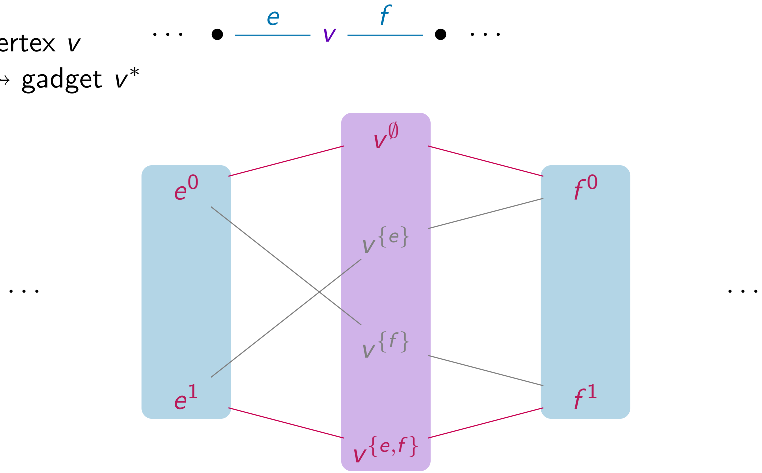


Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}^T$

vertex v

$\rightsquigarrow$ gadget v^*



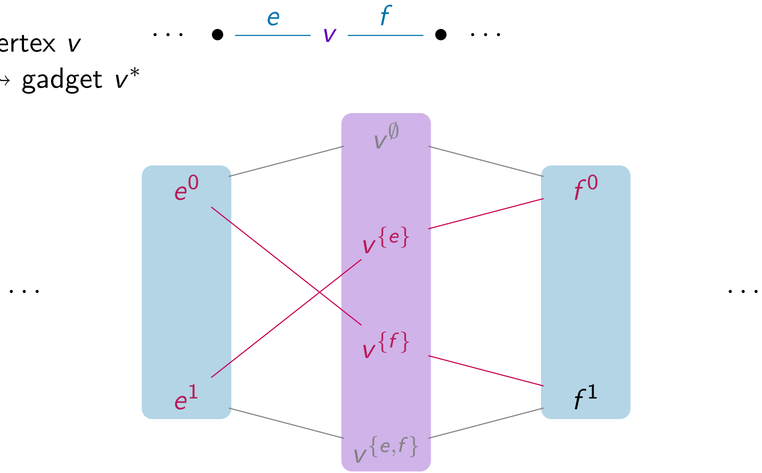
even gadget, $v \notin T$

Cai-Fürer-Immerman graphs: construction

Graph $\mathfrak{G}^T$

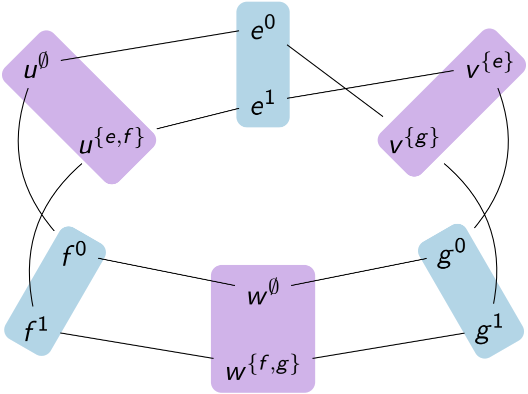
vertex v

$\rightsquigarrow$ gadget v^*



odd gadget, $v \in T$

Cai-Fürer-Immerman graphs: example



Theorem

The CFI query over graphs with logarithmic colour classes is CPT*-definable.*

Theorem

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

Theorem

The CFI query over complete graphs is not 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 CPT*-definable.*

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

Theorem
The CFI query over complete graphs is not CPT*-definable without using set-like objects.*

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

Use:

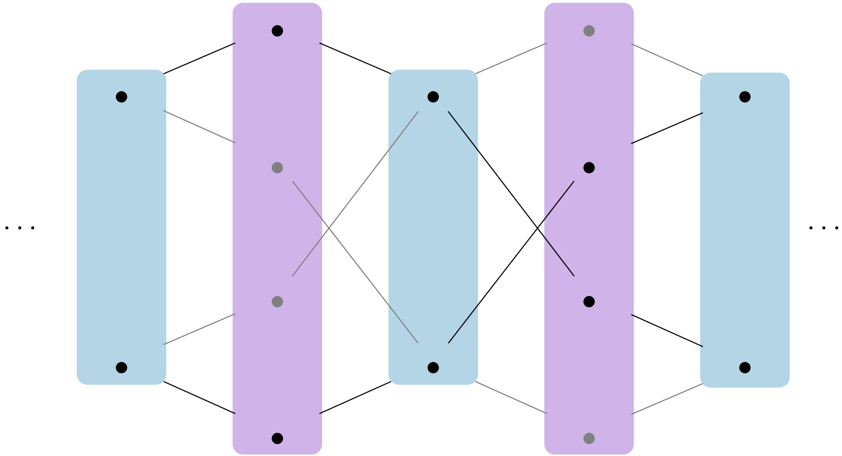
Theorem (Dawar,Richerby,Rossman)
The CFI query over ordered graphs is CPT*-definable.*

Computing the parity of CFI graphs

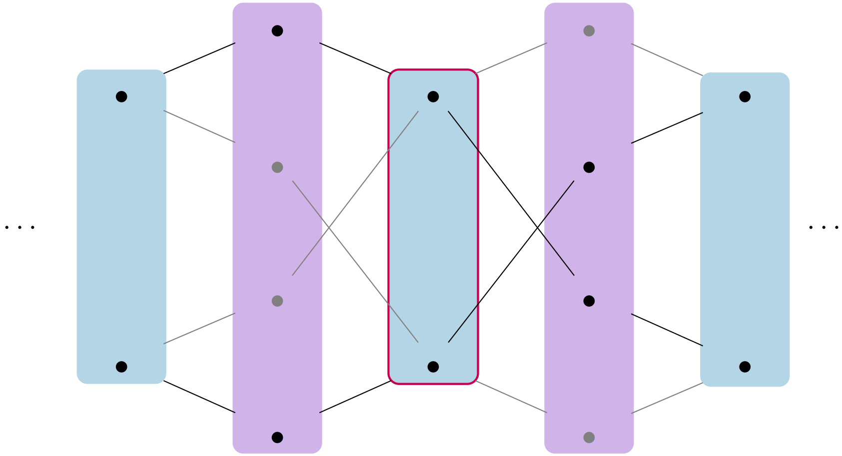
Easy PTIME procedure

1. Label edge gadgets
2. Count odd vertex gadgets

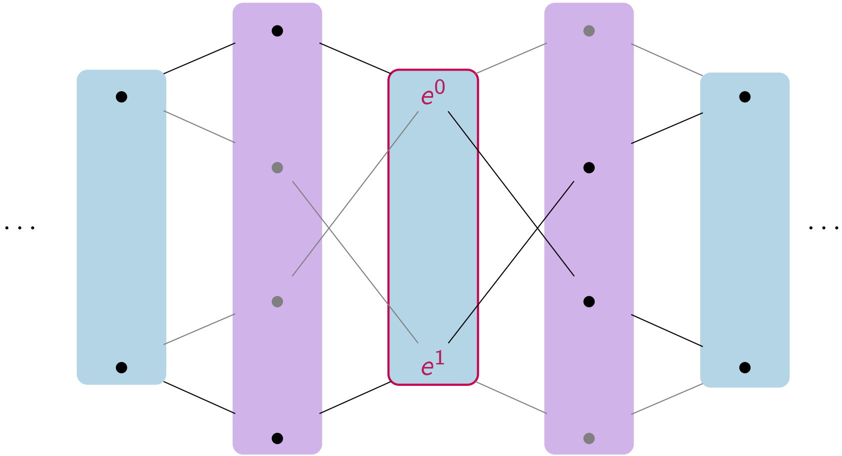
Assigning edge labels



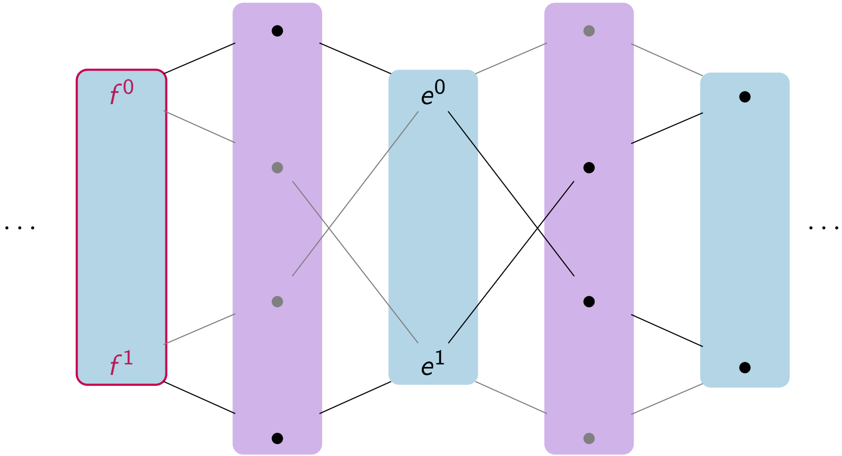
Assigning edge labels



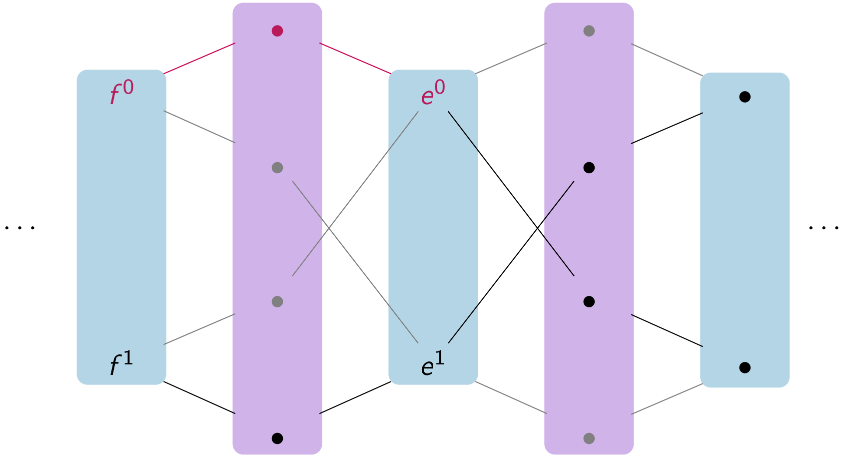
Assigning edge labels



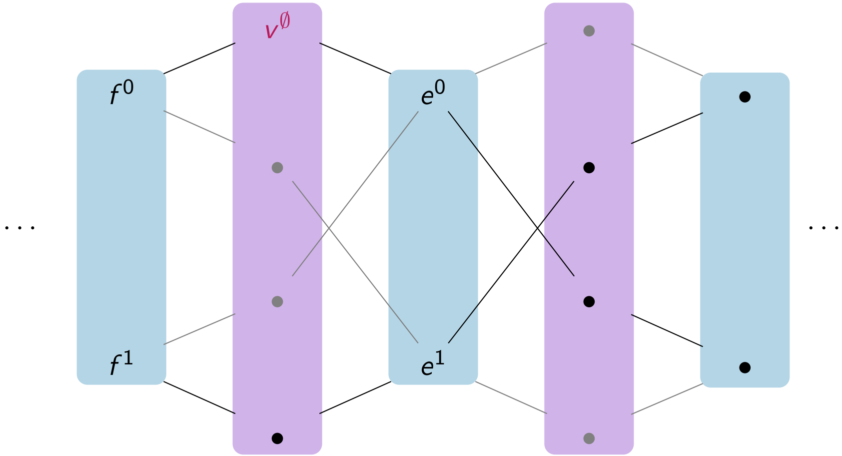
Assigning edge labels



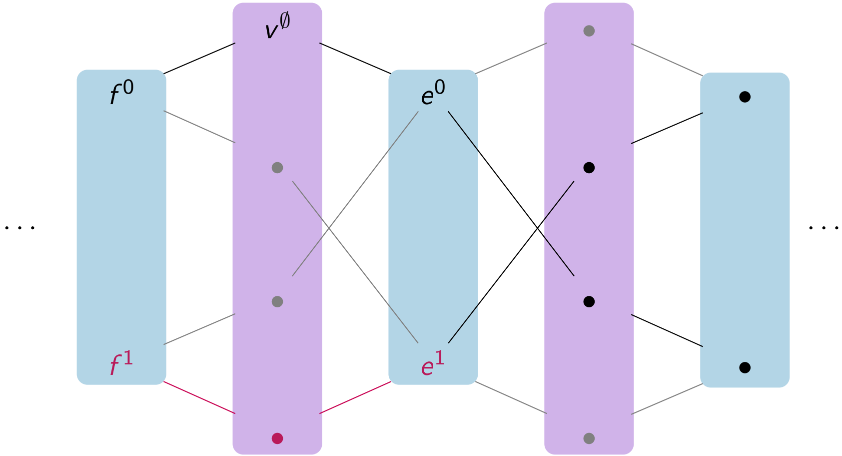
Assigning edge labels



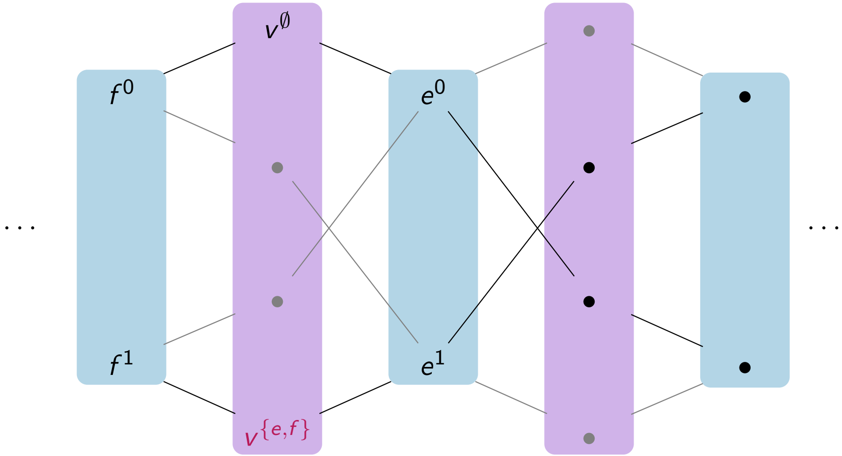
Assigning edge labels



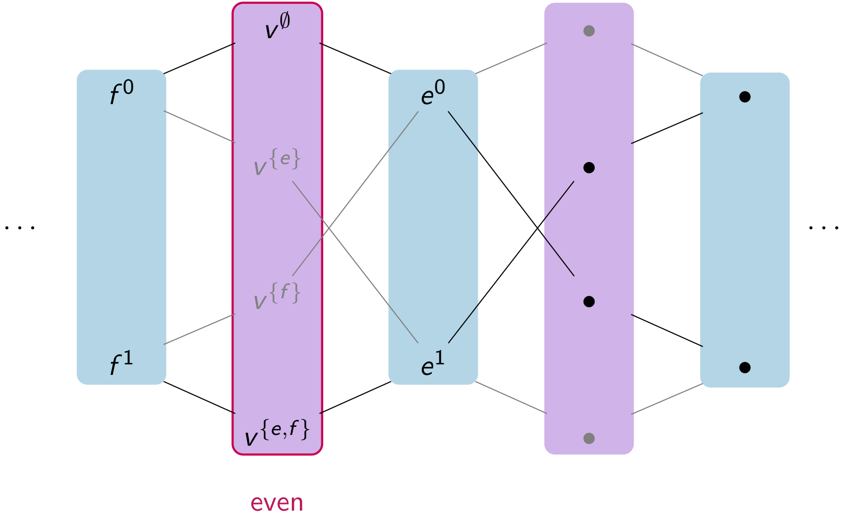
Assigning edge labels



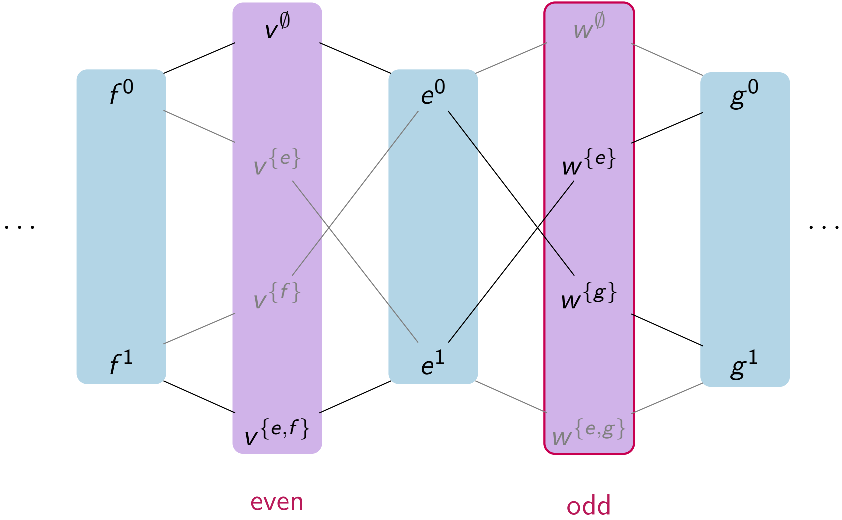
Assigning edge labels



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

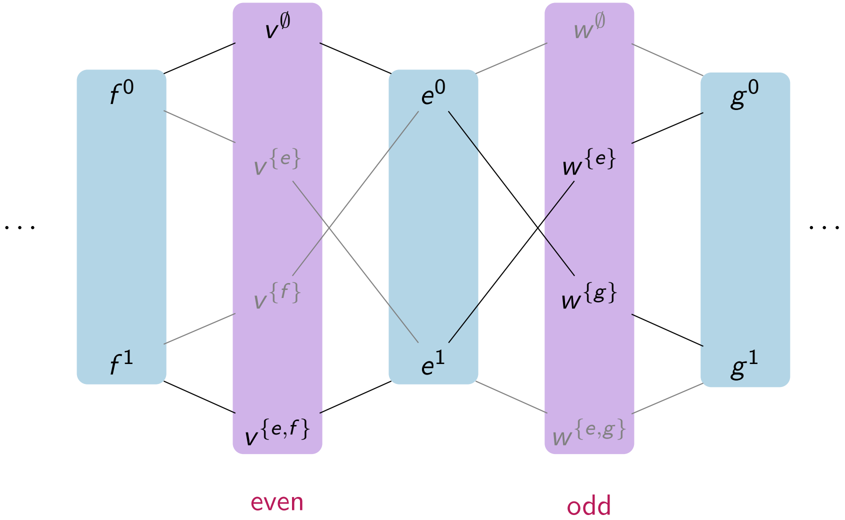
Easy PTIME procedure

1. Label edge gadgets
2. Count odd vertex gadgets

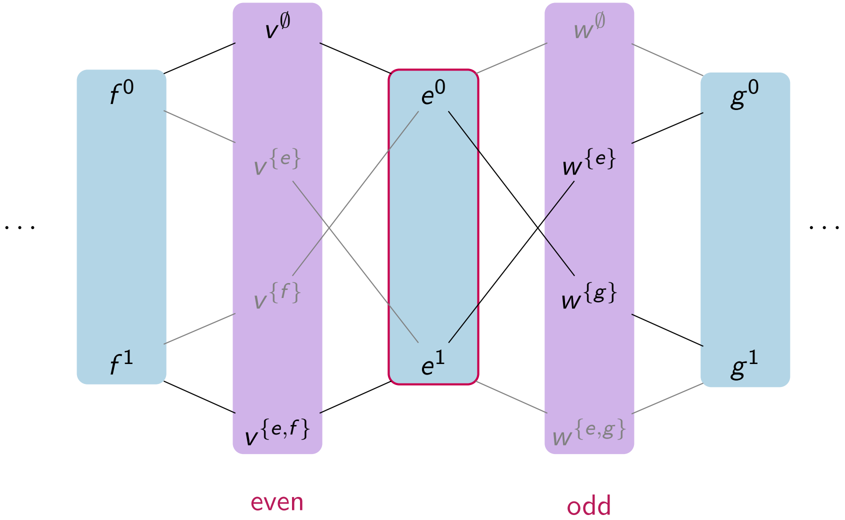
CPT procedure (Dawar, Richerby, Rossman)

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

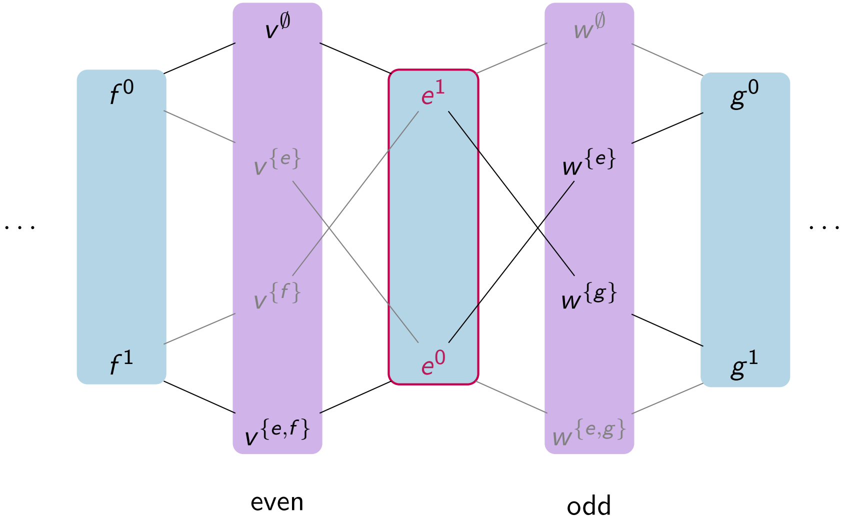
Why super-symmetry is useful



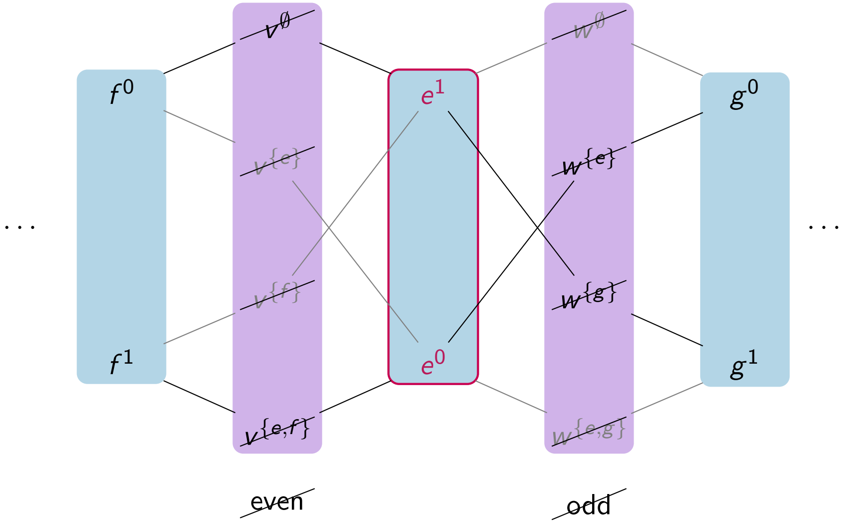
Why super-symmetry is useful



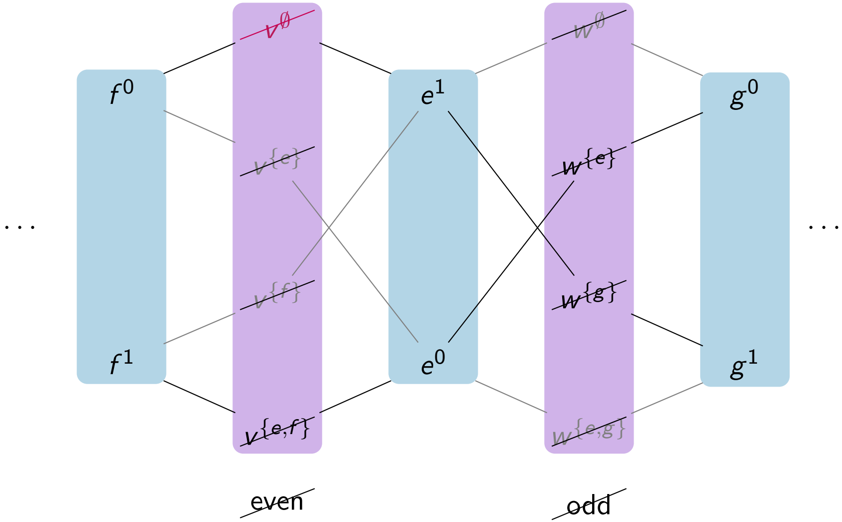
Why super-symmetry is useful



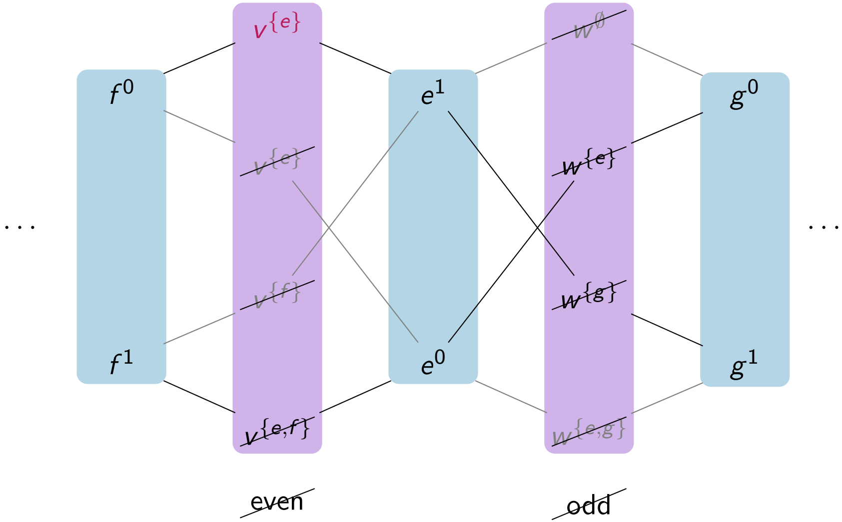
Why super-symmetry is useful



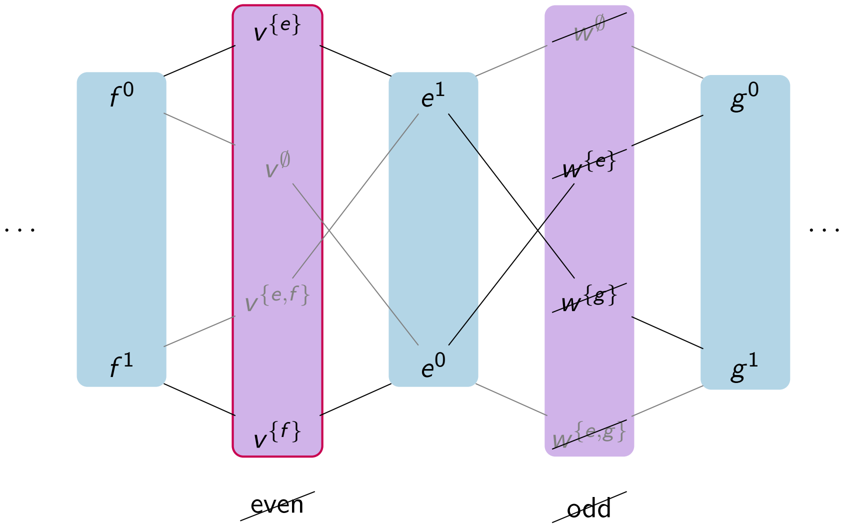
Why super-symmetry is useful



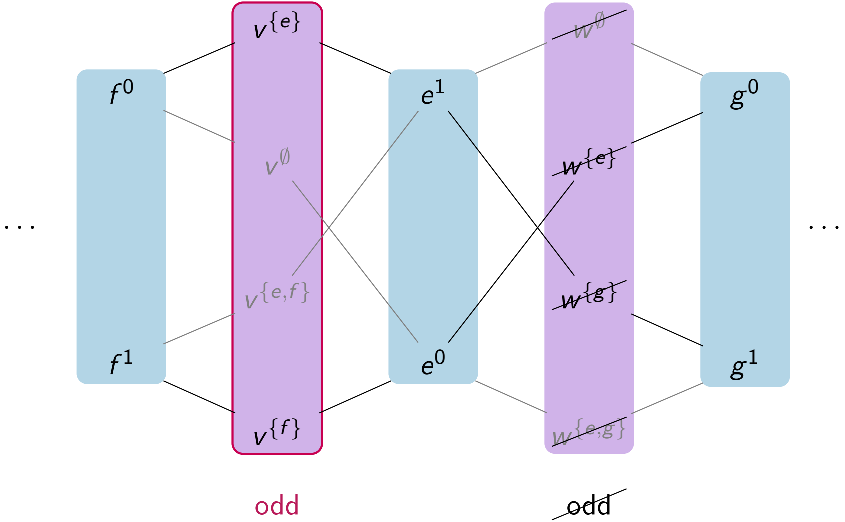
Why super-symmetry is useful



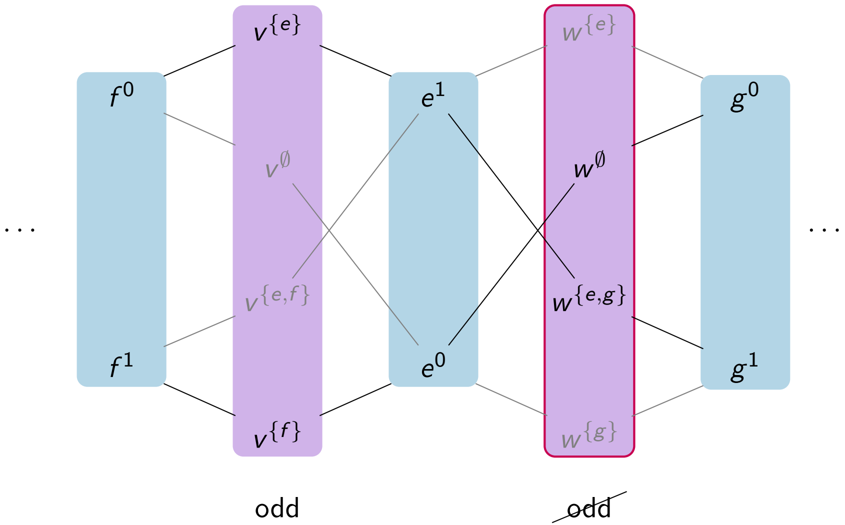
Why super-symmetry is useful



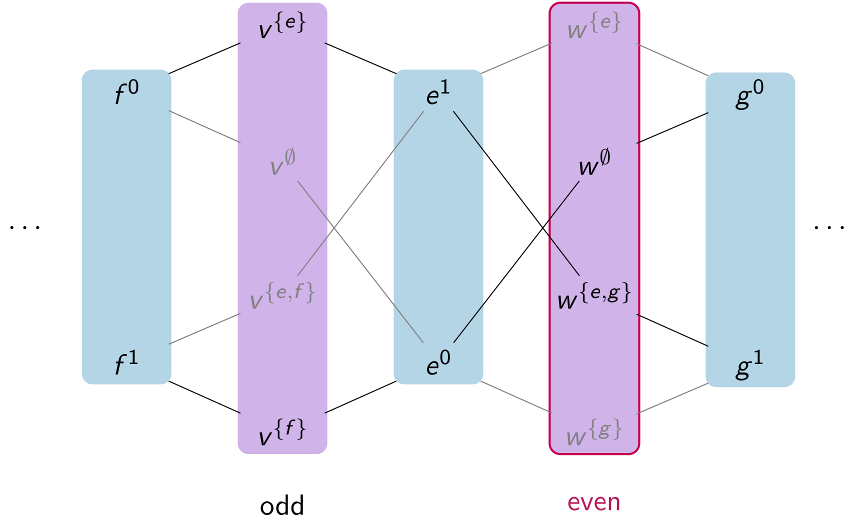
Why super-symmetry is useful



Why super-symmetry is useful



Why super-symmetry is useful



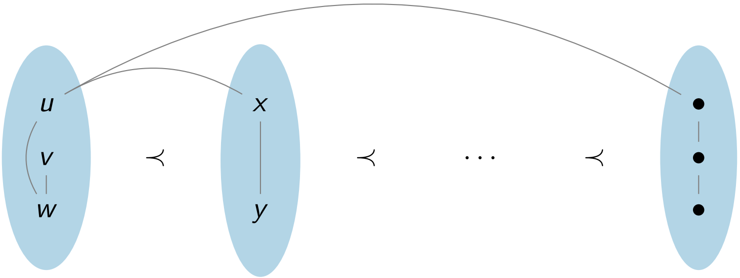
Theorem
The CFI query over graphs with logarithmic colour classes is CPT*-definable.*

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

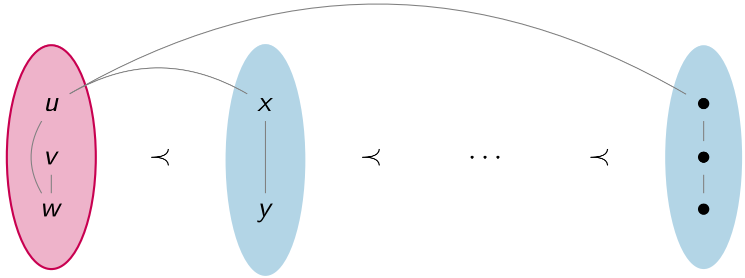
Theorem
The CFI query over complete graphs is not CPT*-definable without using set-like objects.*

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

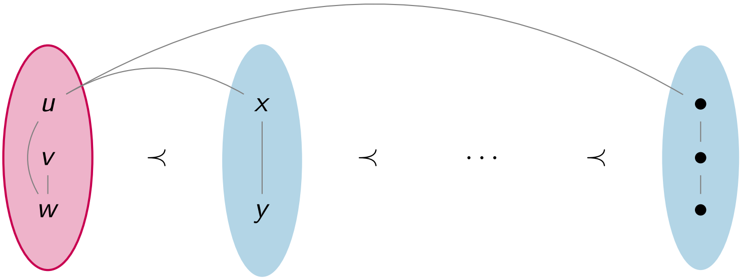


$\mu_{\{u\}}$ $\mu_{\{v\}}$ $\mu_{\{w\}}$

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

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

Graphs with colour classes of logarithmic size



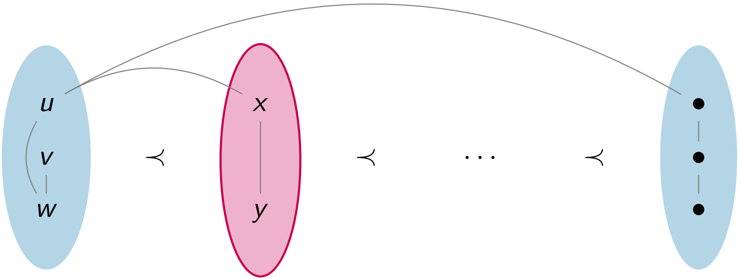
$\mu_{\{u\}}$ $\mu_{\{v\}}$ $\mu_{\{w\}}$

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

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

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

Graphs with colour classes of logarithmic size

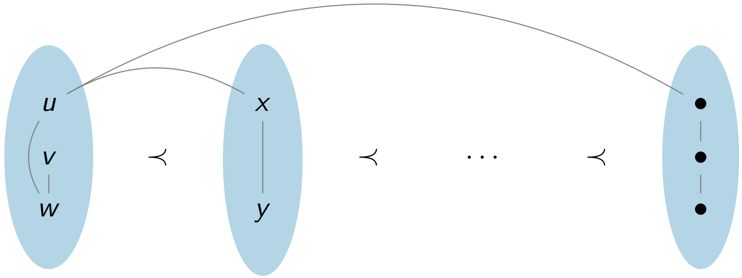


$\mu_{\{u,v,w\}}$

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

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

Graphs with colour classes of logarithmic size



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

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

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

μ_V

Theorem
The CFI query over graphs with logarithmic colour classes is CPT-definable. ✓

Theorem
The CFI query over graphs with large degree is CPT-definable using only sets of bounded rank.

Theorem
The CFI query over complete graphs is not 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 CPT*-definable.* ✓

Theorem

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

Theorem

The CFI query over complete graphs is not 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 CPT-definable. ✓

Theorem

The CFI query over graphs with large degree is CPT-definable using only sets of bounded rank.

Theorem

The CFI query over complete graphs is not CPT-definable without using set-like objects.

Corollary

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

Theorem (Dawar, Richerby, Rossman)

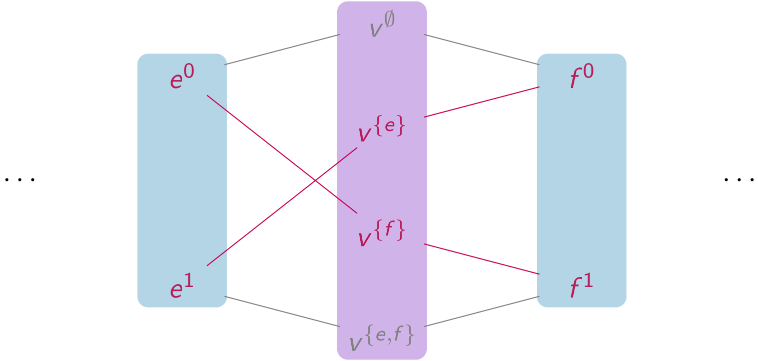
The CFI-query over ordered graphs is not definable in CPT using only sets of bounded rank.

Graphs with large degree: Keeping the rank small

- Access to all subsets of V

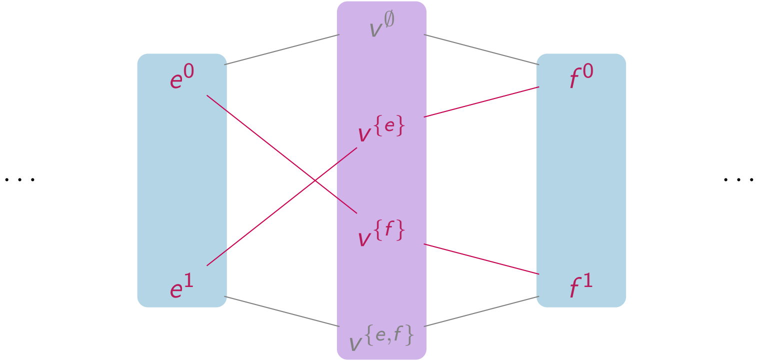
Graphs with large degree: Keeping the rank small

- 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 CPT*-definable.* ✓

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

Theorem
The CFI query over complete graphs is not 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 CPT*-definable.* ✓

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

Theorem
The CFI query over complete graphs is not CPT*-definable without using set-like objects.*

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

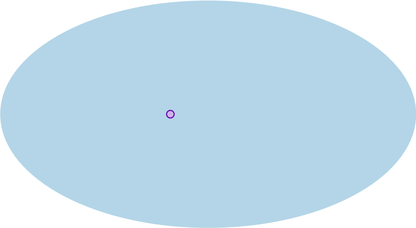
Proving lower bounds

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

Proving lower bounds

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

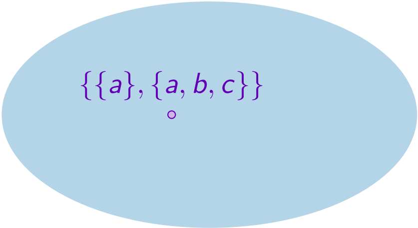
Move in $\mathrm{HF}(\mathfrak{A})$:



Proving lower bounds

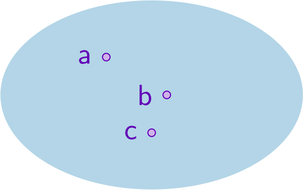
CPT-program $\rightsquigarrow \mathcal{C}^k$-formula over HF($\mathfrak{A}$)

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

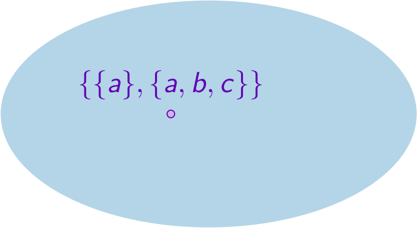


CPT-program $\rightsquigarrow \mathcal{C}^k$-formula over HF($\mathfrak{A}$)

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



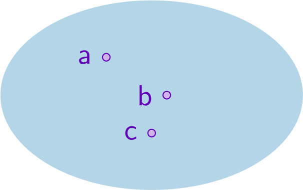
Move in HF($\mathfrak{A}$):



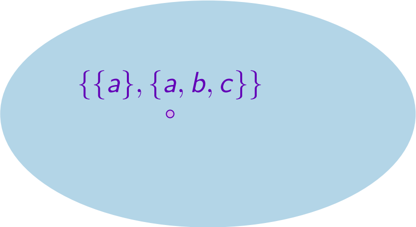
Proving lower bounds

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

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



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

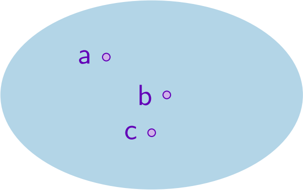


S supports x if $\mathrm{Stab}^\bullet(S) \leq \mathrm{Stab}(x)$

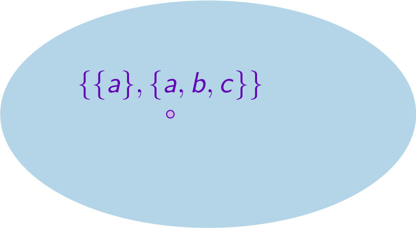
Proving lower bounds

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

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



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



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

Example

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

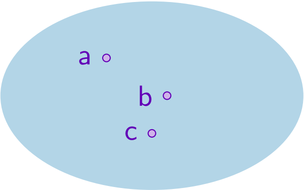
Supports of $\{a, b, c\}$: A, $\{a, b, c\}$, $\{d, e\}$

Proving lower bounds

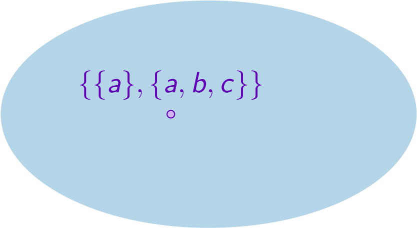
sets with small support

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

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



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



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, $\{a, b, c\}$, $\{d, e\}$

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 $\mathcal{K}_n$.

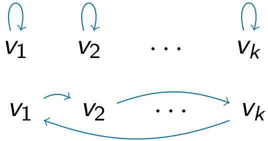
$x = \{v_1, \ldots, v_k\}$

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

Sequence-like objects: strong supports

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

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

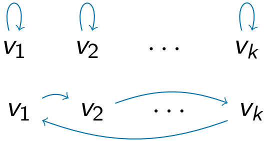


$y = (v_1, \ldots, v_k)$

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 CPT*-definable.* ✓

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

Theorem
The CFI query over complete graphs is not CPT*-definable without using set-like objects.* ✓

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