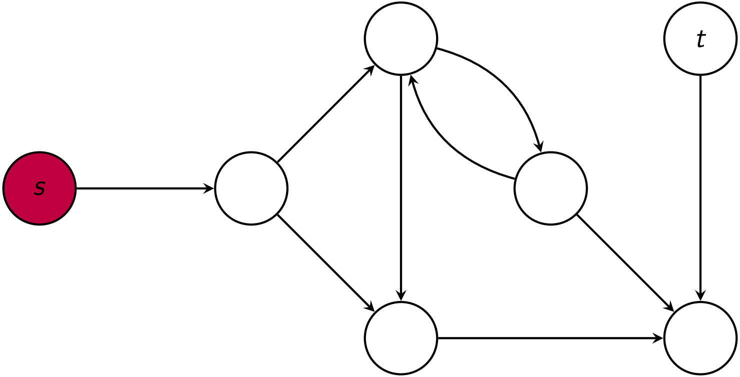
An Introduction to invariant logics

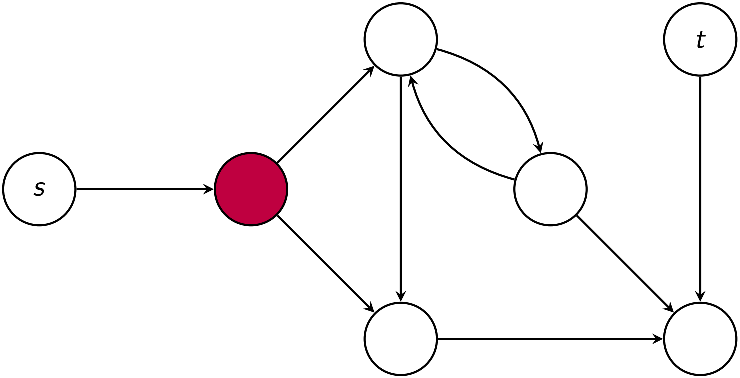
Julien Grange
LACL, Université Paris-Est Créteil

LMW 2023, Warsaw

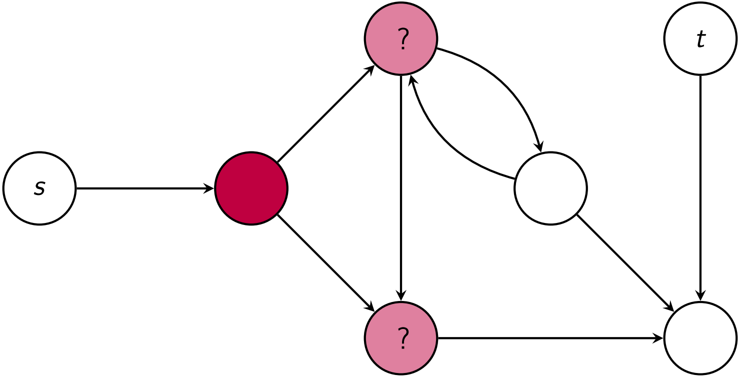
Reach(s,t) via BFS



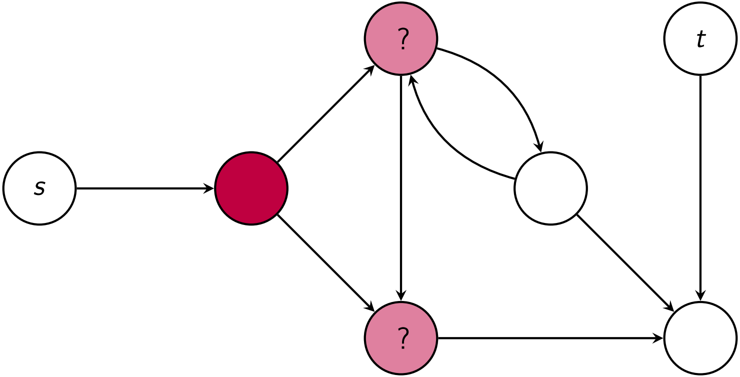
Reach(s,t) via BFS



Reach(s,t) via BFS



Reach(s,t) via BFS



- An order on vertices lets us make choices
- The end result does not depend on the order

Definition of $<$-inv $\mathcal{L}$

$\varphi \in \mathcal{L}(\Sigma, <)$ is **order-invariant** over a finite Σ-structure $\mathcal{A}$ if

$$\forall <_1, <_2, \qquad (\mathcal{A}, <_1) \models \varphi \quad \leftrightarrow \quad (\mathcal{A}, <_2) \models \varphi$$

Definition of $<$-inv $\mathcal{L}$

$\varphi \in \mathcal{L}(\Sigma, <)$ is **order-invariant** over a finite Σ-structure $\mathcal{A}$ if

$$\forall <_1, <_2, \qquad (\mathcal{A}, <_1) \models \varphi \quad \leftrightarrow \quad (\mathcal{A}, <_2) \models \varphi$$

Definition (Order-Invariant Logic)

$<$-inv $\mathcal{L} := \{\varphi : \varphi$ is order-invariant over every finite $\mathcal{A}\}$

Definition of $<$-inv $\mathcal{L}$

$\varphi \in \mathcal{L}(\Sigma, <)$ is **order-invariant** over a finite Σ-structure $\mathcal{A}$ if

$$\forall <_1, <_2, \qquad (\mathcal{A}, <_1) \models \varphi \quad \leftrightarrow \quad (\mathcal{A}, <_2) \models \varphi$$

Definition (Order-Invariant Logic)

$<$-inv $\mathcal{L} := \{\varphi : \varphi$ is order-invariant over every finite $\mathcal{A}\}$

Definition of $<$-inv $\mathcal{L}$

$\varphi \in \mathcal{L}(\Sigma, <)$ is **order-invariant** over a finite Σ-structure $\mathcal{A}$ if

$$\forall <_1, <_2, \qquad (\mathcal{A}, <_1) \models \varphi \quad \leftrightarrow \quad (\mathcal{A}, <_2) \models \varphi$$

Definition (Order-Invariant Logic)

$<$-inv $\mathcal{L} := \{\varphi : \varphi$ is order-invariant over every finite $\mathcal{A}\}$

Proposition

If SAT($\mathcal{L}$) is undecidable, then $<$-inv $\mathcal{L}$'s syntax is not recursive.

Definition of $<$-inv $\mathcal{L}$

$\varphi \in \mathcal{L}(\Sigma, <)$ is **order-invariant** over a finite Σ-structure $\mathcal{A}$ if

$$\forall <_1, <_2, \qquad (\mathcal{A}, <_1) \models \varphi \quad \leftrightarrow \quad (\mathcal{A}, <_2) \models \varphi$$

Definition (Order-Invariant Logic)

$<$-inv $\mathcal{L} := \{\varphi : \varphi$ is order-invariant over every finite $\mathcal{A}\}$

Proposition

If SAT($\mathcal{L}$) is undecidable, then $<$-inv $\mathcal{L}$'s syntax is not recursive.

Proof. Add $U^{(1)}$ to Σ, and reduce sentence $\varphi \in \mathcal{L}$ to

$$\widehat{\varphi} := U(\texttt{first}) \wedge \varphi\,.$$

Then $\widehat{\varphi} \in <$-inv $\mathcal{L}$ iff φ has no model of size at least two. □

Expressive power

- If $\mathcal{L}$ (e.g. $\exists$SO) can define an order, $<$-inv $\mathcal{L} = \mathcal{L}$.
- For $\mathcal{L} \in \{\mathsf{MSO}, \mathsf{LFP}\}$,

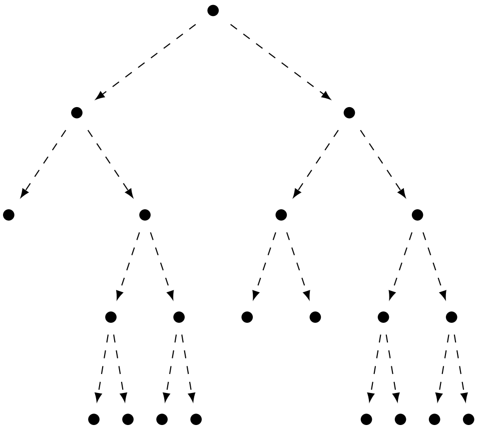
$$\begin{cases} \texttt{even\_size} \in <\text{-inv } \mathcal{L} \\ \texttt{even\_size} \notin \mathcal{L} \end{cases}$$

- What about $<$-inv FO ?

same_parity : All the branches have the same length parity

Complete unordered binary tree,
with the descendant relation.

$(aa)^\star \notin$ FO :
same_parity $\notin$ FO

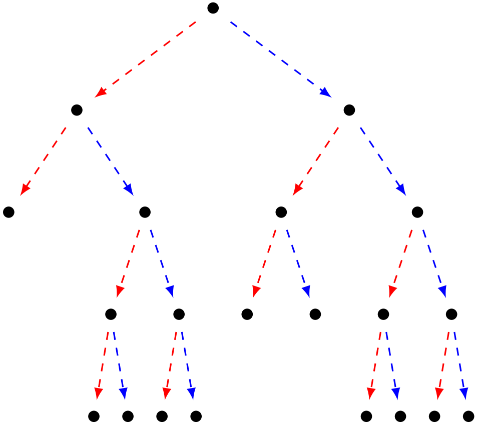


`same_parity` : All the branches have the same length parity

Complete unordered binary tree,
with the descendant relation.

$(aa)^\star \notin$ FO :
`same_parity` $\notin$ FO

$({\color{red}l}{\color{blue}r})^\star \in$ FO :

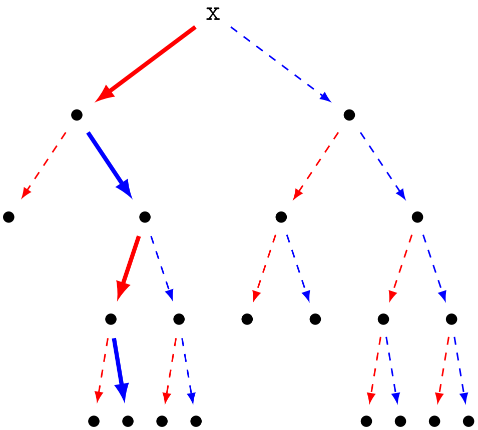


`same_parity` : All the branches have the same length parity

Complete unordered binary tree,
with the descendant relation.

$(aa)^\star \notin$ FO :
`same_parity` $\notin$ FO

$(lr)^\star \in$ FO :
`zigzag_even(x)` $\in$ FO



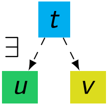
same_parity : All the branches have the same length parity

Complete unordered binary tree,
with the descendant relation.

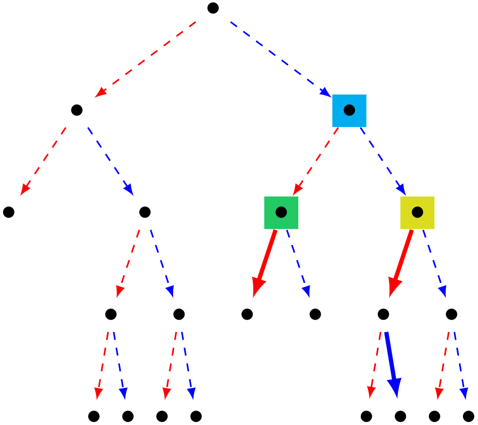
$(aa)^\star \notin$ FO :
same_parity $\notin$ FO

$(lr)^\star \in$ FO :
zigzag_even(x) $\in$ FO

$(\mathcal{T}, <) \models \neg$same_parity

iff $\exists$ 

, $\begin{cases} \neg\texttt{zigzag\_even}(u) \\ \texttt{zigzag\_even}(v) \end{cases}$



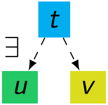
same_parity : All the branches have the same length parity

Complete unordered binary tree,
with the descendant relation.

$(aa)^\star \notin$ FO :
same_parity $\notin$ FO

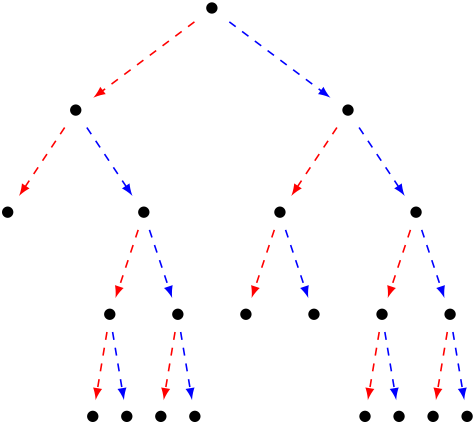
$(lr)^\star \in$ FO :
zigzag_even(x)$\in$ FO

$(\mathcal{T}, <) \models \neg$same_parity

iff $\exists$ 

, $\begin{cases} \neg\text{zigzag\_even}(u) \\ \text{zigzag\_even}(v) \end{cases}$

same_parity $\in$ $<$-inv FO



Always requiring invariance, we define

Succ-inv $\mathcal{L}$: adjacency relation in the order

Always requiring invariance, we define

Succ-inv $\mathcal{L}$: adjacency relation in the order

$+$-inv $\mathcal{L}$: addition on an initial segment of $\mathbb{N}$

Beyond order-invariance

Always requiring invariance, we define

Succ-inv $\mathcal{L}$: adjacency relation in the order

$+$-inv $\mathcal{L}$: addition on an initial segment of $\mathbb{N}$

Arb-inv $\mathcal{L}$: arbitrary numerical predicates

Descriptive complexity

Theorem (Fagin '74)

$\exists$SO *describes* NP

Descriptive complexity

Theorem (Fagin '74)

($<$*-inv*) $\exists$SO *describes* NP

Descriptive complexity

Theorem (Fagin '74)

($<$-*inv*) $\exists$SO *describes* NP

Theorem (Immerman,Vardi '82)

$<$-inv LFP *describes* PTIME

Descriptive complexity

Theorem (Fagin '74)

($<$*-inv*) $\exists$SO *describes* NP

Theorem (Immerman,Vardi '82)

$<$-inv LFP *describes* PTIME

Theorem (Immerman '88)

$<$-inv TC *describes* NLOGSPACE

Descriptive complexity

Theorem (Fagin '74)

($<$-*inv*) ∃SO *describes* NP

Theorem (Immerman,Vardi '82)

$<$-inv LFP *describes* PTIME

Theorem (Immerman '88)

$<$-inv TC *describes* NLOGSPACE

Theorem (Makowsky '97)

Arb-inv FO *describes* AC^0

Descriptive complexity

Theorem (Fagin '74)

($<$-*inv*) $\exists$SO *describes* NP

Theorem (Immerman,Vardi '82)

$<$-inv LFP *describes* PTIME

Theorem (Immerman '88)

$<$-inv TC *describes* NLOGSPACE

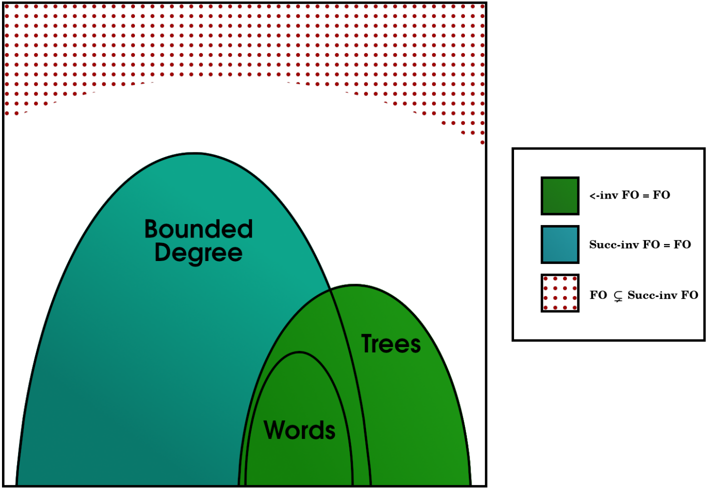
Theorem (Makowsky '97)

Arb-inv FO *describes* AC^0

Main lines of research :

- find decidable syntax for invariant logics
- understand their expressive power

Zoom on FO (expressive power)



Gurevich, [Rossman '07], [Benedikt, Segoufin '09], [G. '20]

Zoom on FO (locality)

Definition (Locality)

A formula is **local with radius** r if it cannot distinguish between elements with isomorphic r-neighborhood.

Theorem (Gaifman '88)

FO *is local with constant radius.*

Zoom on FO (locality)

Definition (Locality)

A formula is **local with radius** r if it cannot distinguish between elements with isomorphic r-neighborhood.

Theorem (Gaifman '88)

FO *is local with constant radius.*

Theorem (Grohe, Schwentick '00)

$<$-inv FO *is local with constant radius.*

Zoom on FO (locality)

Definition (Locality)

A formula is **local with radius** r if it cannot distinguish between elements with isomorphic r-neighborhood.

Theorem (Gaifman '88)

FO *is local with constant radius.*

Theorem (Grohe, Schwentick '00)

$<$-inv FO *is local with constant radius.*

Theorem (Anderson et al. '11)

Arb-inv FO *is local with a polylogarithmic radius.*

Brains needed !

Some interesting open questions :

- Which density is needed for $<$-inv FO/Succ-inv FO to go beyond FO ?
- Is $+$-inv FO local with constant radius ?

Brains needed !

Some interesting open questions :

- Which density is needed for $<$-inv FO/Succ-inv FO to go beyond FO ?
- Is $+$-inv FO local with constant radius ?

Thanks !