Modal Fixpoint Logics: When Logic Meets Games, Automata and Topology

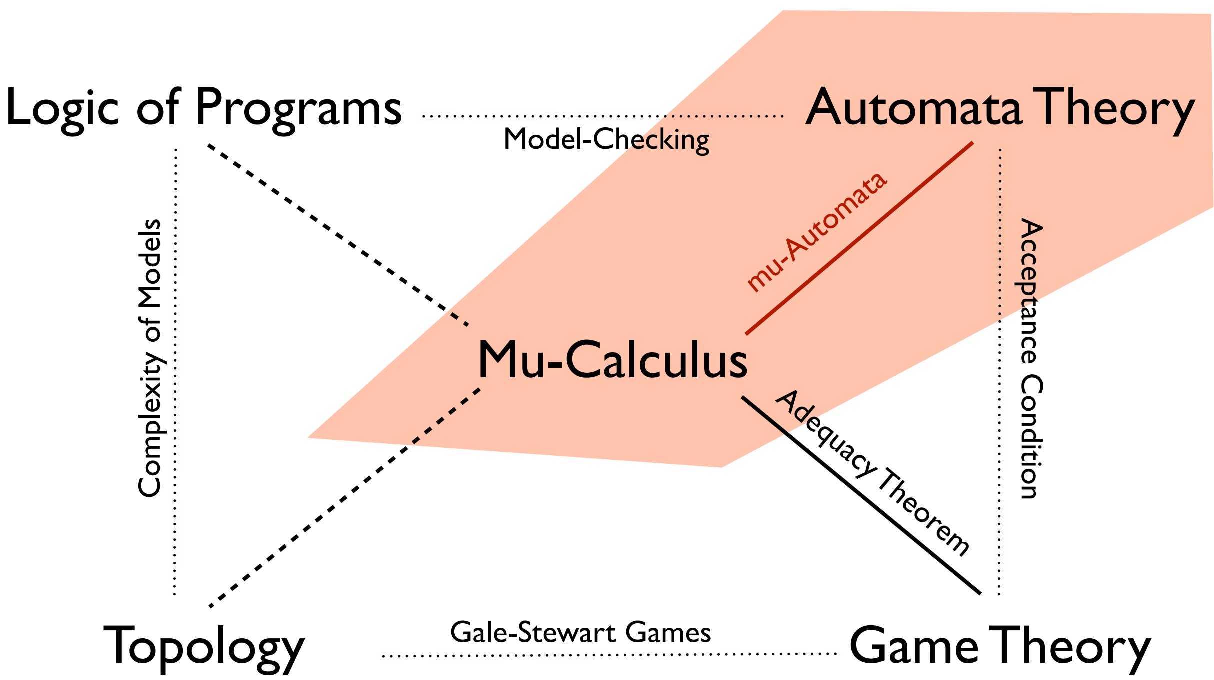
A. Facchini & D. Niwinski (U. Warsaw)

Lecture III

MSO vs Mu-Calculus

ESSLLI 2014, Tübingen, 11-22 August 2014

What we have seen yesterday...



Two automata-theoretic characterizations:

$$\varphi = \nu x.\mu y.(\Diamond x \vee p) \wedge (\Diamond y \vee \neg p)$$

1. modal automata

$$\mathbb{A} = (\{a, b\}, a, \Delta, \mathrm{rank})$$
$$\Delta(a) = \Delta(b) = (\Diamond a \vee p) \wedge (\Diamond b \vee \neg p)$$
$$\mathrm{rank}(a) = 2$$
$$\mathrm{rank}(b) = 1$$

Two automata-theoretic characterizations:

$$\varphi = \nu x.\mu y.(\Diamond x \vee p) \wedge (\Diamond y \vee \neg p)$$

2. mu-automata $\mathrm{Aut}(\mathrm{FO}^+)$

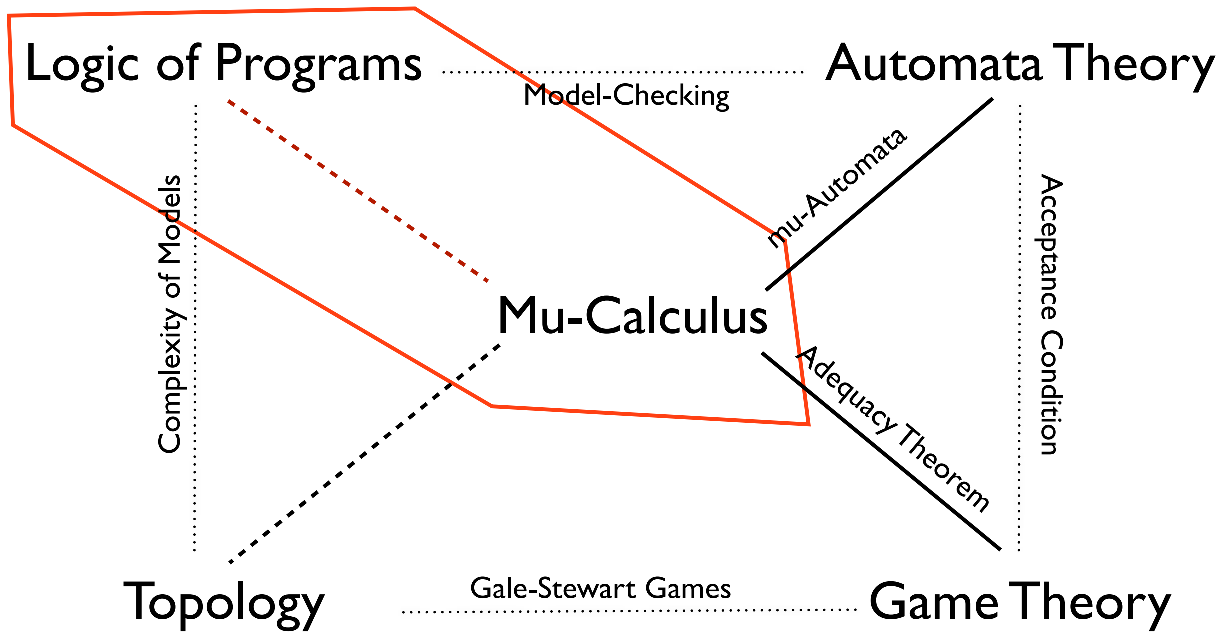
$$\mathbb{A} = (\{a, b\}, \wp P, a, \Delta, \mathrm{rank})$$
$$\Delta(a, Q) = \Delta(b, Q) = \begin{cases} \exists x.a(x) & \text{if } p \notin Q \\ \exists x.b(x) & \text{if } p \in Q \end{cases}$$
$$\mathrm{rank}(a) = 2$$
$$\mathrm{rank}(b) = 1$$

What we have seen yesterday...

Nice thing about mu-automata:

Simulation theorem
=
Normal form theorem

What we are going to see today...



The nice behavior of the mu-calculus:

(i) translatable into (fragment of) MSO
(ii) tree model property
(iii) small model property
(iv) Janin-Walukiewicz characterization theorem:

$$MSO/\underline{\leftrightarrow} = \mu ML \text{ (over all models)}$$

bisimulation invariance

Bisimulation invariance of the mu-Calculus

Theorem: Assume $\mathcal{K}, s_I \underline{\leftrightarrow} \mathcal{K}', s'_I$. Then for every $\phi \in \mu\text{ML}$:

$$\mathcal{K}, s_I \models \phi \text{ iff } \mathcal{K}', s'_I \models \phi$$

Theorem (Bounded Tree Model Property): Let $\phi \in \mu$ML. If ϕ is satisfiable, then it is satisfiable at the root of a tree whose branching degree is bounded by the size of ϕ.

Proof: Consider the tree unraveling of the model, then prune it by using the positional winning strategy for $\exists$ in the accepting game of $\mathbb{A}_\phi$ (non-det.) considering only the existential part of the transition.

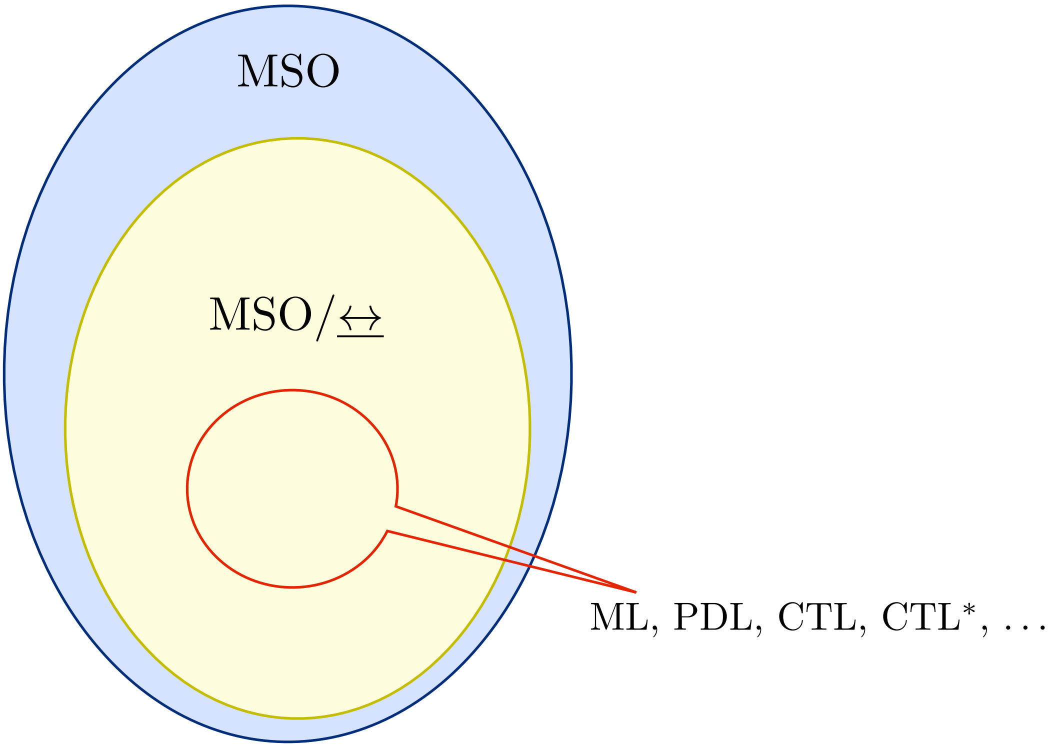
■

The case of the mu-calculus:

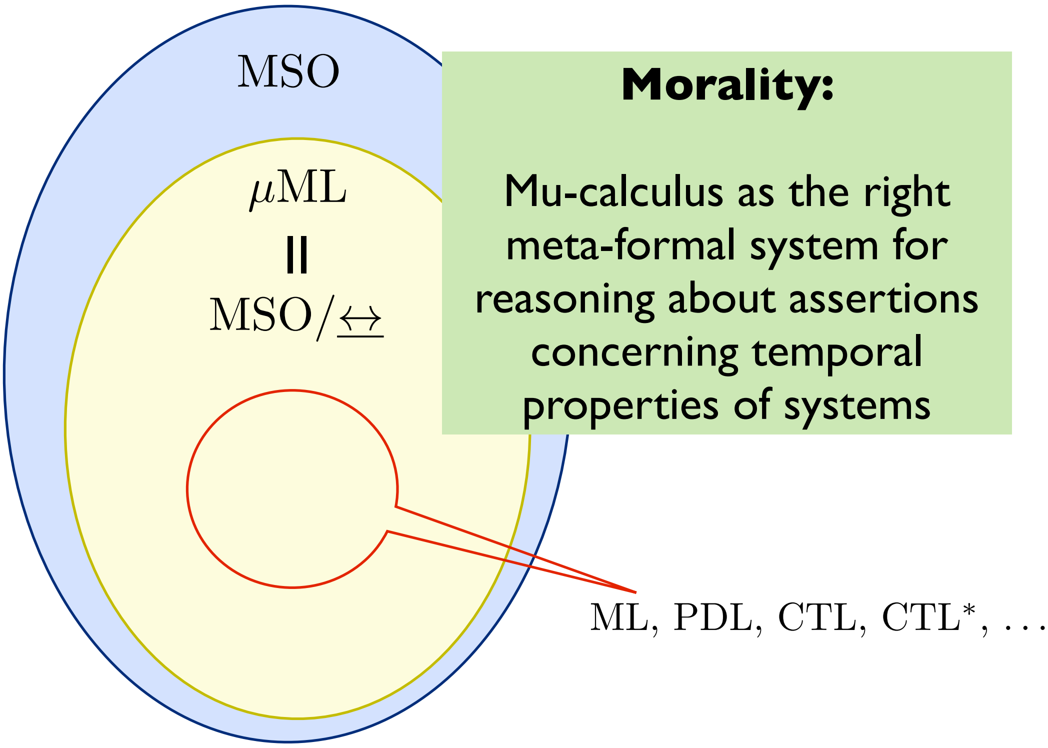
(i) translatable into (fragment of) MSO
(ii) tree model property
(iii) small model property
(iv) Janin-Walukiewicz characterization theorem:

$$MSO/\underline{\leftrightarrow} = \mu ML \text{ (over all models)}$$

General view



General view



Morality:

Mu-calculus as the right meta-formal system for reasoning about assertions concerning temporal properties of systems

Once more: why to bother about the Janin-Walukiewicz Theorem?

Once more: why to bother about the Janin-Walukiewicz Theorem?

instance of a more general problem

$$\mathcal{L}/\underline{\leftrightarrow} = \mathcal{M} \ \ (\textbf{over}\ \mathcal{C})$$

$\mathcal{L}/\underline{\leftrightarrow} = \mathcal{M}$ (over $\mathcal{C}$)

	$\mathcal{L}$	
	FO	
	MSO	
	WMSO	

$\mathcal{L}/\underline{\leftrightarrow} = \mathcal{M}$ (over $\mathcal{C}$)

	$\mathcal{M}$	
	ML	
	μ**ML**	
	AFMC	
	PDL	
	CTL	

$\mathcal{L}/\underline{\leftrightarrow} = \mathcal{M}$ **(over** $\mathcal{C}$**)**

Structures ($\mathcal{C}$)	
K	
K4	
$\mathcal{T}_2$	
$\mathbf{K}^f$	
GL	

$\mathcal{L}/\underline{\leftrightarrow} = \mathcal{M}$ **(over** $\mathcal{C}$**)**

Structures ($\mathcal{C}$)	$\mathcal{L}$	$\mathcal{M}$	**Reference**
K	**FO**	**ML**	van Benthem (1977)
	MSO	μ**ML**	Janin, Walukiewicz (1996)
	WMSO	μ_c**ML**	Carreiro, F., Venema, Zanasi (2014)
	WFMSO	**AFMC**	F., Venema, Zanasi (2013)
$\mathcal{T}_2$	**WMSO**	**AFMC**	Arnold, Niwinski (1992)
K4	**WMSO**	**ML**	ten Cate, F. (2011)
	MSO	**AFMC**	Alberucci, F. / Dawar, Otto (2008)
$\mathbf{K}^f$	**FO**	**ML**	Rosen (1997)
	MSO	???	-
$(\mathbb{N}, <)$	**FO**	**LTL**	Kamp (1968)
GL	**MSO**	**ML**	van Benthem (2006) / Alberucci, F. (2008)

A purely second-order variant of MSO

$$\phi ::= x = y \mid p(x) \mid R(x,y) \mid \phi \vee \phi \mid \neg\phi \mid \exists x.\phi \mid \exists p.\phi$$

with $p \in P$ and $x, y \in \mathcal{X}$.

A purely second-order variant of MSO

MSO'

$$\phi ::= x = y \mid p(x) \mid R(x,y) \mid \phi \vee \phi \mid \neg\phi \mid \exists x.\phi \mid \exists p.\phi$$

with $p \in P$ and $x, y \in \mathcal{X}$.

MSO

$$\phi ::= \downarrow p \mid p \subseteq q \mid R(p,q) \mid \phi \vee \phi \mid \neg\phi \mid \exists p.\phi$$

with $p \in P'$.

A purely second-order variant of MSO

Given a Kripke model $\mathcal{K}$, and $s \in S$,

- $\mathcal{K}, s \models \downarrow p$ iff $\rho(p) = \{s\}$,
- $\mathcal{K}, s \models p \subseteq q$ iff $\rho(p) \subseteq \rho(q)$,
- $\mathcal{K}, s \models R(p, q)$ iff $\forall s \in \rho(p), \exists t \in \rho(q)$ s.t. $(s, t) \in R$,
- $\ldots$
- $\mathcal{K}, s \models \exists p.\phi$ iff $\exists X \subseteq Q.\mathcal{K}[p \mapsto X], s \models \phi$.

p-variant

A purely second-order variant of MSO

Proposition:

- for every $\phi(x) \in MSO'$ there is $(\phi)^t \in MSO$ such that $\mathcal{K} \models \phi(s)$ iff $\mathcal{K}, s \models (\phi)^t$
- for every $\phi \in MSO$ there is $(\phi)_t(x) \in MSO$ such that $\mathcal{K}, s \models$ iff $\mathcal{K} \models (\phi)_t(s)$

Proof (sketch): For the first item, use the fact that

- $\text{Empty}(p) = \forall q.p \subseteq q$
- $\text{Sing}(p) = \neg\text{Empty}(p) \wedge \forall q(q \subseteq p \to (\text{Empty}(q) \vee p \subseteq q))$.

For the second item, just write the semantics of MSO in MSO′.

■

Theorem: There are effective translations $(\cdot)^{\bullet}$:MSO $\to \mu$ML and $(\cdot)_{\bullet} : \mu\text{ML} \to$ MSO, such that

1. $\phi \in$ MSO is bisimulation invariant iff $\phi \equiv \phi^{\bullet}$,
2. $\psi \equiv \psi_{\bullet}$ for every formula $\psi \in \mu$ML.

Proof idea

$\mu\mathbf{ML}$ = μ-automata
(over K)

Proof idea

(over trees)
$\mathbf{MSO}$ = $\mathbf{MSO}$-automata

$\mu\mathbf{ML}$ = μ-automata
(over K)

Proof idea

(over trees)

MSO = **MSO**-automata

translation ↑

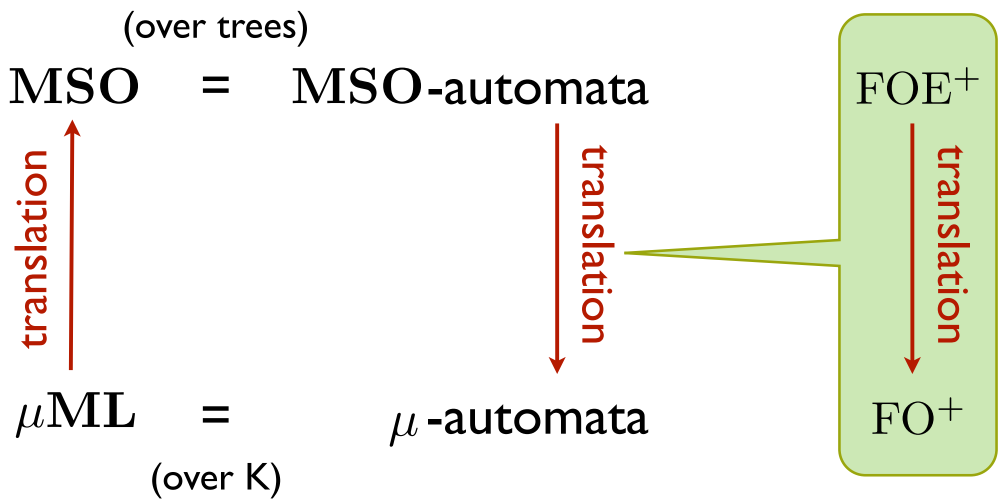
μ**ML** = μ-automata

(over K)

Proof idea

MSO-automata	$\text{Aut}(\text{FOE}^+)$ $\Delta : (a, c) \mapsto \varphi \in \text{FOE}^+(A)$
μ-automata	$\text{Aut}(\text{FO}^+)$ $\Delta : (a, c) \mapsto \varphi \in \text{FO}^+(A)$

Proof idea



Translating one-step logics

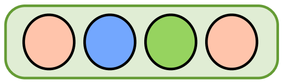
we want to find a translation satisfying:

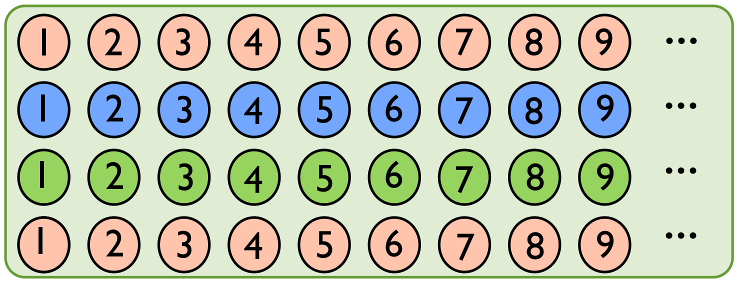
$$(\cdot)^{\bullet} : \mathrm{FOE}^{+}(A) \to \mathrm{FO}^{+}(A)$$

$$\mathbf{D}_{\omega} := (D_{\omega}, V_{\omega}) \models \varphi$$

iff

$$\mathbf{D} := (D, V) \models \varphi^{\bullet}$$

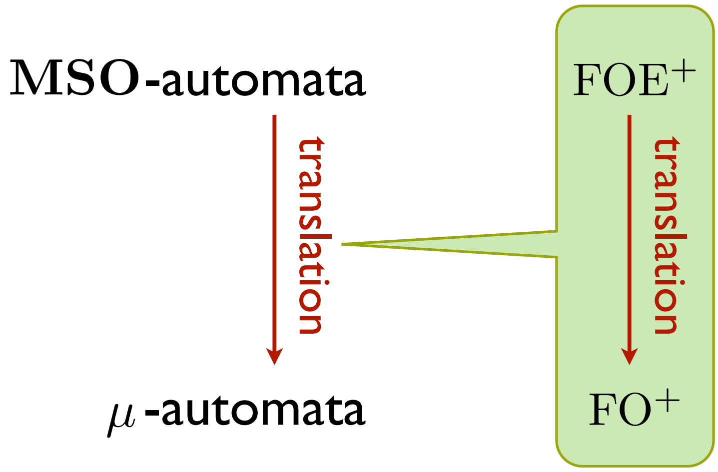
$(D, V) =$ 

$(D_\omega, V_\omega) =$ 

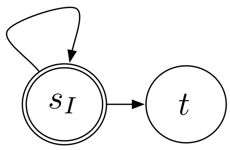
$(D_\omega, V_\omega) = (D \times \omega, V_\omega)$
where

$V_\omega((d, i)) = V(d)$

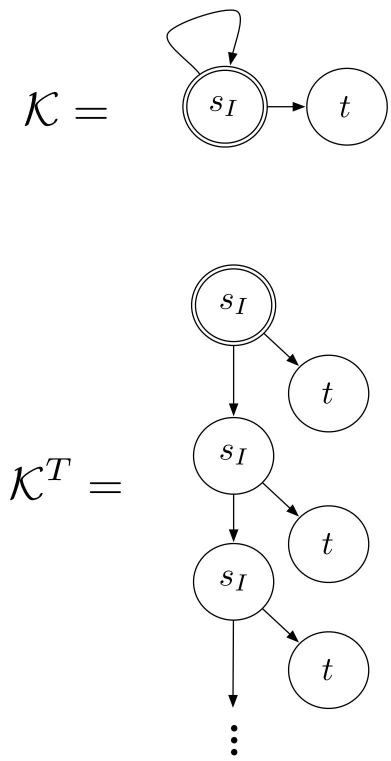
Translating one-step logics



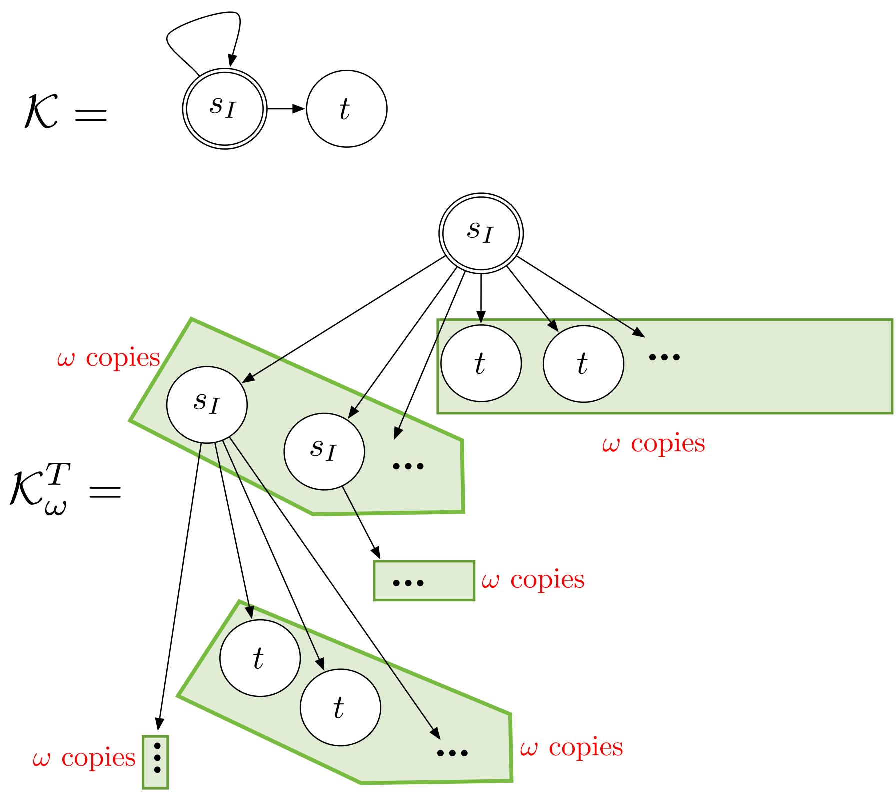
omega-tree unraveling

$\mathcal{K} =$ 

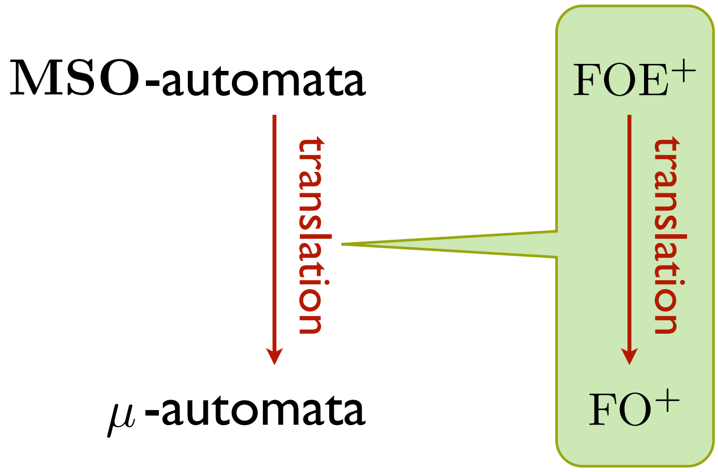
omega-tree unraveling



omega-tree unraveling



Translating one-step logics



Translating one-step logics

$$(\cdot)^\bullet : \mathrm{Aut}(\mathrm{FOE}^+) \to \mathrm{Aut}(\mathrm{FO}^+)$$
$$\Delta^\bullet(a, c) := (\Delta(a, c))^\bullet$$

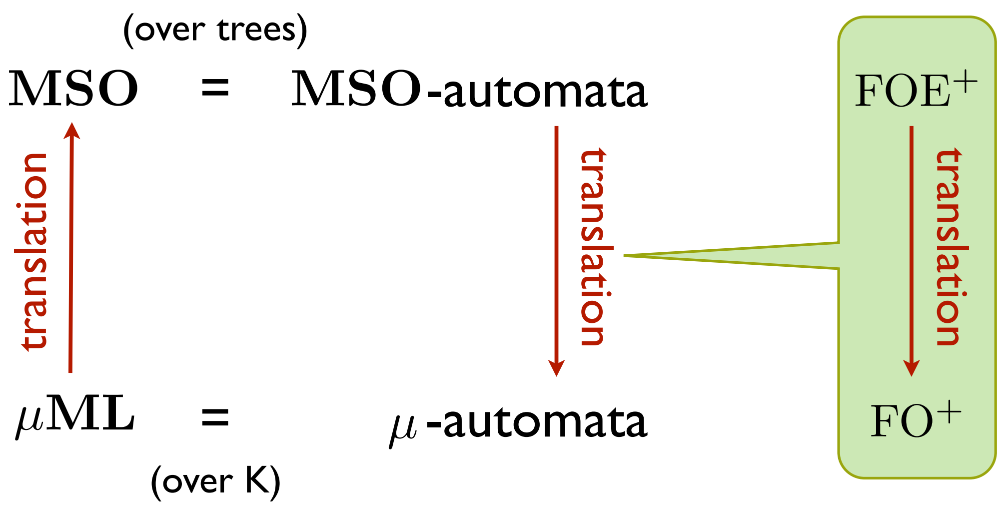
$$\mathcal{K}^T_\omega \in L(\mathbb{A})$$

ω-tree unraveling of $\mathcal{K}$

iff

$$\mathcal{K} \in L(\mathbb{A}^\bullet)$$

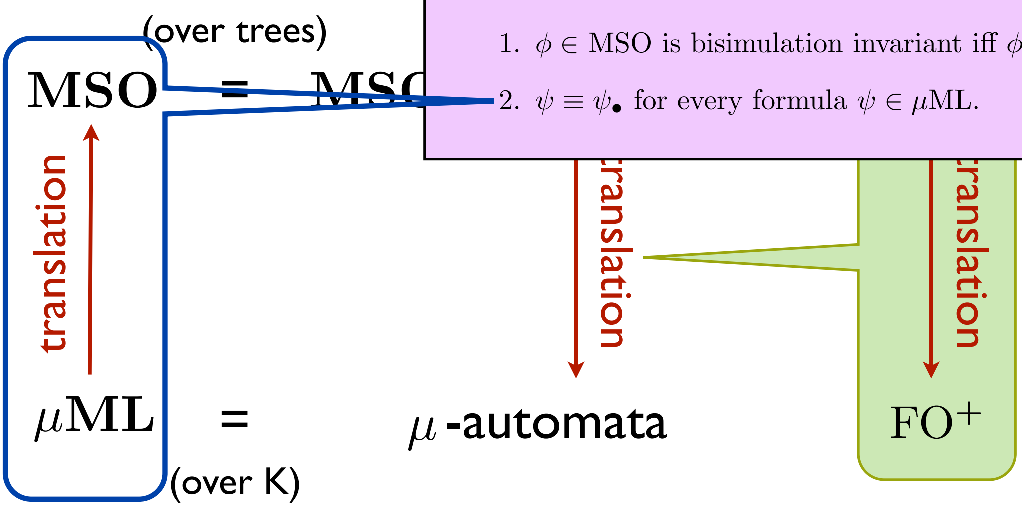
The Janin-Walukiewicz theorem as a corollary of this picture



The Janin-Walukiewicz theorem as a corollary of this picture

Theorem: There are effective translations $(\cdot)^{\bullet}$:MSO $\to$ μML and $(\cdot)_{\bullet}$: μML $\to$ MSO, such that

1. $\phi \in$ MSO is bisimulation invariant iff $\phi \equiv \phi^{\bullet}$,
2. $\psi \equiv \psi_{\bullet}$ for every formula $\psi \in \mu$ML.



For item I of the theorem we reason as follows:

Let $\phi \in$ MSO bisimulation invariant:

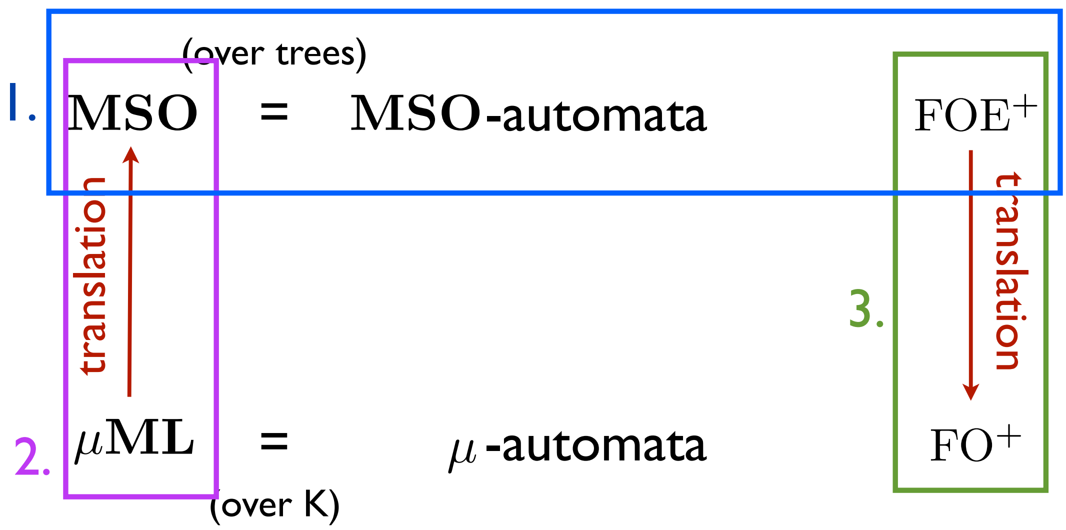
$\mathcal{K} \models \phi$ iff $\mathcal{K}^T_\omega \models \phi$
(bis. inv.)

iff $\mathcal{K}^T_\omega \in L(\mathbb{A}_\phi)$
(MSO=MSO-aut.)

iff $\mathcal{K} \in L((\mathbb{A}_\phi)^\bullet)$
(transl.)

iff $\mathcal{K} \models (\phi)^\bullet$
(mu-calculus=mu-automata)

What we want

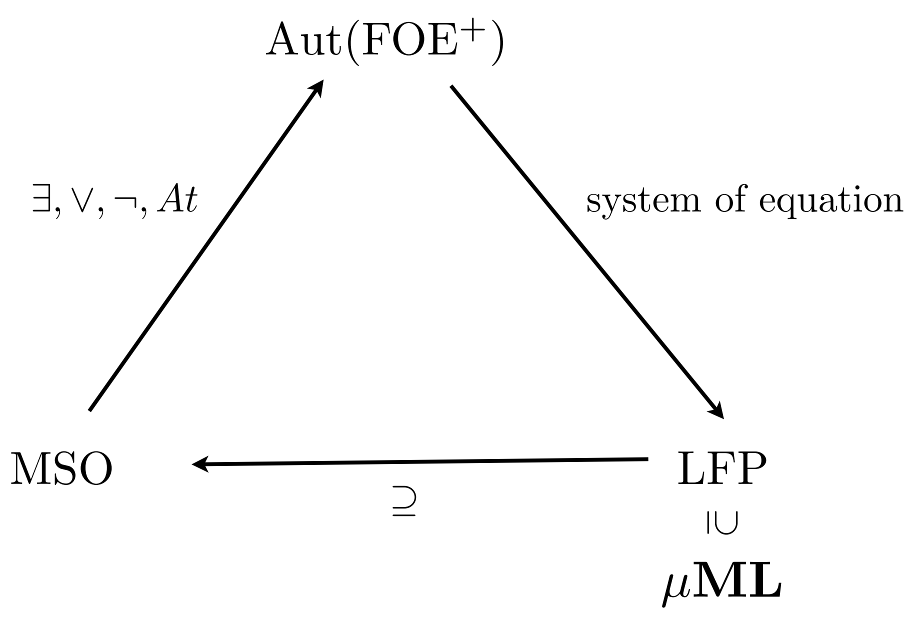


Let's start with

1. **MSO** = **MSO**-automata FOE^+ (over trees)

μ**ML** = μ-automata FO^+ (over K)

Automata for MSO



MSO-automata

Definition: A MSO-automaton (over Σ) is a tuple

$$\mathbb{A} = (A, \Sigma, a_I, \Delta, \Omega)$$

such that

- $a_I \in A$ (initial state)
- $\Delta : A \times \Sigma \to \mathrm{FOE}^+(A)$ (transition fct)
- $\mathrm{rank} : A \to \mathbb{N}$ (parity fct)

$\mathrm{Aut}(\mathrm{FOE}^+)$

Acceptance (parity) game $\mathcal{G}(\mathbb{A}, \mathcal{K})$

Let $\mathcal{K} = (S, R, \rho : S \to \Sigma)$ be a tree model over Σ.

Position	Player	Admissible moves	Parity
$(a, s) \in A \times S$	$\exists$	$\{V : A \to \wp(R[s]) \mid (R[s], V) \models \Delta(a, \rho(s))\}$	$\mathrm{rank}(a)$
$V : A \to \wp S$	$\forall$	$\{(b, t) \mid t \in V(b)\}$	$\max(\mathrm{rank}[A])$

Acceptance (parity) game $\mathcal{G}(\mathbb{A}, \mathcal{K})$

Definition: $\mathbb{A}$ accepts $(\mathcal{K}, s_I)$ iff $\exists$ has a winning strategy in $\mathcal{G}(\mathbb{A}, \mathcal{K})@(a_I, s_I)$

$$(\mathcal{K}, s_I) \in L(\mathbb{A})$$

where s_I is the root of $\mathcal{K}$.

Automata for MSO

Let $\mathbb{A} = (A, \wp P, a_I, \Delta, \mathrm{rank})$ be defined as follows.

$$\begin{array}{rcl} A & := & \{a_0\} \\ a_I & := & a_0 \\ \Delta(a_0, Q) & := & \begin{cases} \forall x\ a_0(x) & \text{If } q \in Q \text{ or } p \notin Q \\ \bot & \text{Otherwise} \end{cases} \\ \mathrm{rank}(a_0) & := & 0 \end{array}$$

Let $\mathbb{A} = (A, \wp P, a_I, \Delta, \text{rank})$ be defined as follows.

$$\begin{array}{rcl} A & := & \{a_0\} \\ a_I & := & a_0 \\ \Delta(a_0, Q) & := & \begin{cases} \forall x \; a_0(x) & \text{If } q \in Q \text{ or } p \notin Q \\ \bot & \text{Otherwise} \end{cases} \\ \text{rank}(a_0) & := & 0 \end{array}$$

$$L(\mathbb{A}) = \{\mathcal{K} \mid \mathcal{K}, s_I \models p \subseteq q\}$$

Let $\mathbb{A} = (A, \wp P, a_I, \Delta, \text{rank})$ be defined as follows.

$$\begin{array}{rcl} A & := & \{a_0, a_1\} \\ a_I & := & a_0 \\ \Delta(a_0, Q) & := & \begin{cases} \exists x \; (a_1(x) \wedge \forall y \; (y \neq x \to a_0(y))) & \text{If } p \in Q \\ \forall x \; (a_0(x)) & \text{Otherwise} \end{cases} \\ \Delta(a_1, Q) & := & \begin{cases} \bot & \text{If } q \notin Q \\ \exists x \; (a_1(x) \wedge \forall y \; (y \neq x \to a_0(y))) & \text{If } p \in Q \text{ and } q \in Q \\ \forall x \; (a_0(x)) & \text{Otherwise} \end{cases} \\ \text{rank}(a_0) & := & 0 \\ \text{rank}(a_1) & := & 0 \end{array}$$

Let $\mathbb{A} = (A, \wp P, a_I, \Delta, \text{rank})$ be defined as follows.

$$
\begin{array}{rcl}
A & := & \{a_0, a_1\} \\
a_I & := & a_0 \\
\Delta(a_0, Q) & := & \begin{cases} \exists x \ (a_1(x) \wedge \forall y \ (y \neq x \to a_0(y))) & \text{If } p \in Q \\ \forall x \ (a_0(x)) & \text{Otherwise} \end{cases} \\
\Delta(a_1, Q) & := & \begin{cases} \bot & \text{If } q \notin Q \\ \exists x \ (a_1(x) \wedge \forall y \ (y \neq x \to a_0(y))) & \text{If } p \in Q \text{ and } q \in Q \\ \forall x \ (a_0(x)) & \text{Otherwise} \end{cases} \\
\text{rank}(a_0) & := & 0 \\
\text{rank}(a_1) & := & 0
\end{array}
$$

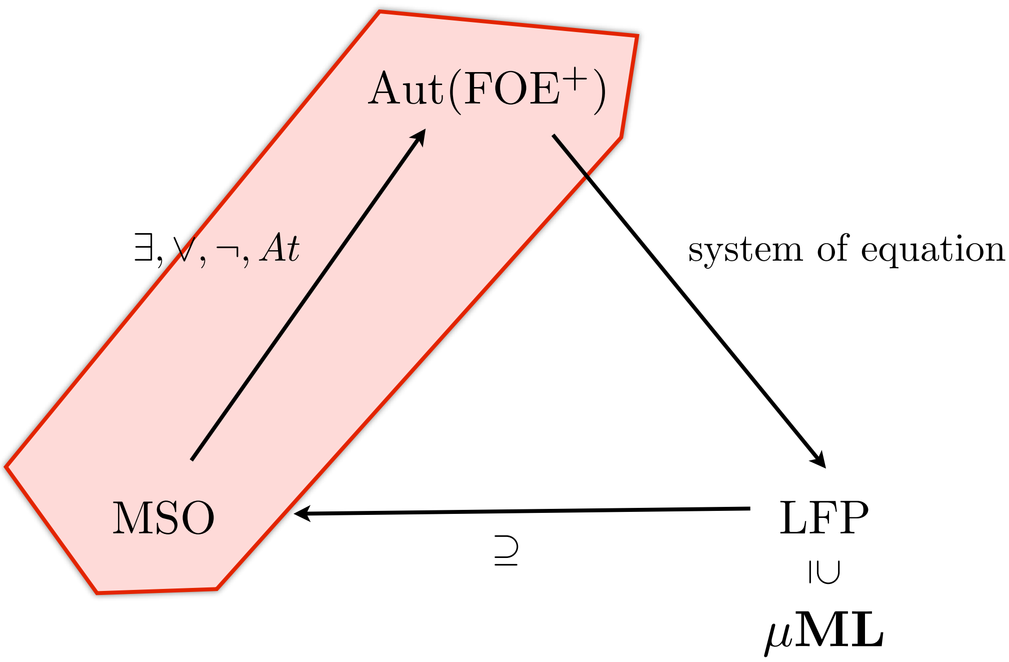
$$L(\mathbb{A}) = \{\mathcal{K} \mid \mathcal{K}, s_I \models R(p, q)\}$$

$$L(\mathbb{A}) = \{\mathcal{K} \mid \mathcal{K}, s_I \models \downarrow p\}$$

Let $\mathbb{A} = (A, \wp P, a_I, \Delta, \text{rank})$ be defined as follows.

$$
\begin{array}{rcl}
A & := & \{a_0, a_1\} \\
a_I & := & a_0 \\
\Delta(a_0, Q) & := & \begin{cases} \forall x\ a_1(x) & \text{If } p \in Q \\ \bot & \text{Otherwise} \end{cases} \\
\Delta(a_1, Q) & := & \forall x\ a_1(x) \\
\text{rank}(a_0) & := & 0 \\
\text{rank}(a_1) & := & 0
\end{array}
$$

$$L(\mathbb{A}) = \{\mathcal{K} \mid \mathcal{K}, s_I \models \downarrow p\}$$



Theorem: For every $\phi \in$ MSO there is an equivalent MSO-automaton $\mathbb{A}_\phi$.

Proof: By induction on the structure of ϕ. Atomic cases and disjunction easy.

Proof (cont.): For the negation,

$$\overline{\cdot} : \begin{cases} a(x) \mapsto a(x) \\ \bot \mapsto \top \\ \top \mapsto \bot \\ x = y \mapsto x \neq y \\ x \neq y \mapsto x = y \\ \phi \vee \psi \mapsto \overline{\phi} \wedge \overline{\psi} \\ \phi \wedge \psi \mapsto \overline{\phi} \vee \overline{\psi} \\ \exists x.\phi \mapsto \forall x.\overline{\phi} \\ \forall x.\phi \mapsto \exists x.\overline{\phi} \end{cases}$$

Fact: Given $\phi, (D, V)$:
$(D, \overline{V}) \not\models \phi$ iff $(D, V) \models \overline{\phi}$.

Proof (cont.): For the negation,

$$\mathcal{A}_{\neg\phi} := (A_\phi, a_I, \overline{\Delta}, \overline{\text{rank}})$$

$$\begin{cases} \overline{\Delta}(a, Q) = \overline{\Delta(a, Q)} \\ \overline{\text{rank}}(a) = \text{rank}(a) + 1 \end{cases}$$

Proof (cont.): For quantification, we use the

Simulation Theorem: Every MSO-automaton is equivalent to a non-deterministic one.

$$\mathrm{diff}(x_1, \ldots, x_k) := \bigwedge_{i \neq j \text{ and } i,j \leq k} x_i \neq x_j$$

Formulation of the simulation theorem:

A **type** is a subset of P.

Let Q be a type.

- $\tau_Q(x) := \begin{cases} \top & \text{if } Q = \emptyset \\ \bigwedge_{p \in Q} p(x) \wedge \bigwedge_{p \notin Q} \neg p(x) & \text{else;} \end{cases}$
- $\tau_Q^+(x) := \begin{cases} \top & \text{if } Q = \emptyset \\ \bigwedge_{p \in Q} p(x) & \text{else.} \end{cases}$

Formulation of the simulation theorem:

Definition: A formula $\phi \in \mathrm{FOE}(A)$ is in basic normal form (BF(A)) if it is of the form

$$\nabla_{\mathrm{FOE}}(\overline{Q}, \Pi) := \exists \overline{x}.\mathrm{diff}(\overline{x}) \wedge \bigwedge_{i \leq k} \tau_{Q_i}(x_i) \wedge \forall y.\mathrm{diff}(\overline{x}, y) \to \bigvee_{T \in \Pi} \tau_T(y)$$

When each type in $\overline{Q} \cup \Pi$ is either empty or a singleton, we say that it is in special normal form (SBF(A)).

Formulation of the simulation theorem:

Definition: A formula $\phi \in \text{FOE}^+(A)$ is in basic normal form $(\text{BF}^+(\text{A}))$ if it is of the form

$$\nabla^+_{\text{FOE}}(\overline{Q}, \Pi) := \exists \overline{x}.\text{diff}(\overline{x}) \wedge \bigwedge_{i \leq k} \tau^+_{Q_i}(x_i) \wedge \forall y.\text{diff}(\overline{x}, y) \to \bigvee_{T \in \Pi} \tau^+_T(y)$$

When each type in $\overline{Q} \cup \Pi$ is either empty or a singleton, we say that it is in special normal form $(\text{SBF}^+(\text{A}))$.

Formulation of the simulation theorem:

Definition: A MSO-automaton $\mathbb{A}$ is non-deterministic if

$$\Delta : A \times \wp P \to \text{SLatt}(SBF^+(A))$$

Formulation of the simulation theorem:

Simulation Theorem: Every MSO-automaton is equivalent to MSO-automaton whose transition formulas are only in special normal form.

Formulation of the simulation theorem:

Simulation Theorem: Every MSO-automaton is equivalent to MSO-automaton whose transition formulas are only in special normal form.

How to use this theorem in order to prove that if $\|\phi(p)\|$ is recognizable then $\|\exists p.\phi(p)\|$ is also recognizable?

Functional winning strategies

Let $\mathcal{K} \in L(\mathbb{A})$ and $\mathbb{A}$ non deterministic

Consider the winning strategy σ for $\exists$ in the acceptance game

$$\sigma(a, s) = (D, V) \text{ s.t. } (D, V) \models \Delta(a, \rho(s))$$

Functional winning strategies

Let $\mathcal{K} \in L(\mathbb{A})$ and $\mathbb{A}$ non deterministic

Consider the winning strategy σ for $\exists$ in the acceptance game

$$\sigma(a, s) = (D, V) \text{ s.t. } (D, V) \models \Delta(a, \rho(s))$$

$$(D, V) \models \exists x_1 \exists x_2 . x_1 \neq x_2 \wedge a(x_1) \wedge a_2(x_2) \wedge \forall y. \text{diff}(y, x_1, x_2) \to (c_1(y) \vee c_2(y))$$

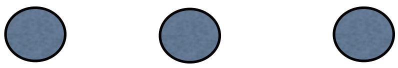
Functional winning strategies

Let $\mathcal{K} \in L(\mathbb{A})$ and $\mathbb{A}$ non deterministic

Consider the winning strategy σ for $\exists$ in the acceptance game

$$\sigma(a,s) = (D,V) \text{ s.t. } (D,V) \models \Delta(a,\rho(s))$$

$$(D,V) \models \exists x_1 \exists x_2 . x_1 \neq x_2 \wedge a(x_1) \wedge a_2(x_2) \wedge \forall y.\text{diff}(y,x_1,x_2) \rightarrow (c_1(y) \vee c_2(y))$$

$D =$ 

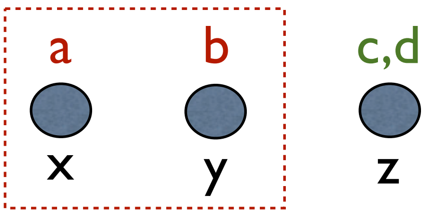
Functional winning strategies

Let $\mathcal{K} \in L(\mathbb{A})$ and $\mathbb{A}$ non deterministic

Consider the winning strategy σ for $\exists$ in the acceptance game

$$\sigma(a,s) = (D,V) \text{ s.t. } (D,V) \models \Delta(a,\rho(s))$$

$$(D,V) \models \exists x \exists y . x \neq y \wedge a(x) \wedge b(y) \wedge \forall z.\text{diff}(x,y,z) \rightarrow (c(z) \vee d(z))$$

$\sigma(a,s) = (D,V) =$ 

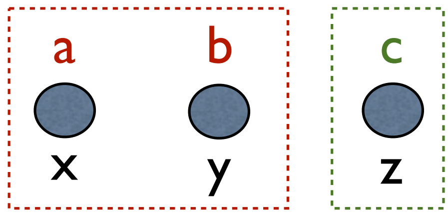
Functional winning strategies

Let $\mathcal{K} \in L(\mathbb{A})$ and $\mathbb{A}$ non deterministic

Consider the winning strategy σ for $\exists$ in the acceptance game

$$\sigma(a, s) = (D, V) \text{ s.t. } (D, V) \models \Delta(a, \rho(s))$$

$$(D, V) \models \exists x \exists y. x \neq y \wedge a(x) \wedge b(y) \wedge \forall z.\text{diff}(x, y, z) \rightarrow (c(z) \vee d(z))$$

$\sigma(a, s) = (D, V) =$ 

Functional winning strategies

Let $\mathcal{K} \in L(\mathbb{A})$ and $\mathbb{A}$ non deterministic

The positional winning strategy σ for $\exists$ in the acceptance game can be assumed to be **functional** i.e.

it induces a unique relabeling of $\mathcal{K}$ where:

- each node is labeled with an element from $A \cup \{\star\}$

$$\mathcal{K} = (S, R, \rho : S \rightarrow C)$$
$$\mapsto$$
$$\mathcal{K}_\sigma := (S, R, \rho_\sigma : S \rightarrow A \cup \{\star\})$$

From simulation to closure under existential quantification

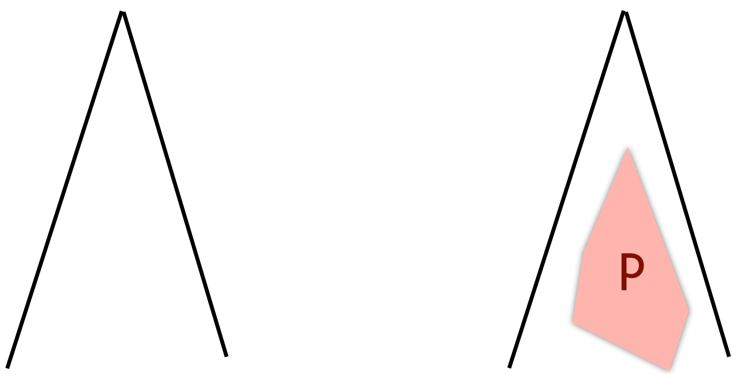
How to use this theorem in order to prove that if $\|\phi(p)\|$ is recognizable then $\|\exists p.\phi(p)\|$ is also recognizable?

we start by 're-formulating' this:

- from the point of view of a tree language
- from the point of view of automata

From simulation to closure under existential quantification

Let $\mathcal{K}' = (S, R, \rho)$ over P. A p-variant $\mathcal{K} = (S, R, \rho')$ is a tree over $P \cup \{p\}$ such that $\rho'|_P = \rho$.



p-variant

Given a tree language L over $P \cup \{p\}$:

$\exists p.L = \{\mathcal{K} \text{ over } P \mid \exists\, p\text{-variant } \mathcal{K}^p \text{ of } \mathcal{K} \text{ s.t. } \mathcal{K}^p \in L\}$

Given $\mathbb{A} = (A, a_I, \Delta, \text{rank})$ over $P \cup \{p\}$:

$\exists p.\mathbb{A} = (A, a_I, \Delta^{\exists}, \text{rank})$ is over P, with

$\Delta^{\exists}(a, c) := \Delta(a, c) \vee \Delta(a, c \cup \{p\})$

Note that if $\mathbb{A}$ non det., then $\mathbb{A}^{\exists}$ non-det. too.

Proposition: Given a letter p and a non-deterministic $\mathbb{A}$ on $P \cup \{p\}$,

$$L(\exists p.\mathbb{A}) = \exists p.L(\mathbb{A})$$

Proof: The direction from right to left is easy. Indeed, let $\mathcal{K}^p$ be a p-variant such that $\exists$ has a winning strategy σ in $\mathcal{G}(\mathbb{A}, \mathcal{K}^p)@(a_I, s_I)$. Then σ is also winning in $\mathcal{G}(\exists p.\mathbb{A}, \mathcal{K})@(a_I, s_I)$

From simulation to closure under existential quantification

$L(\exists p.\mathbb{A}) \subseteq \exists p.L(\mathbb{A})$

Proof (cont.): Let $\mathcal{K} \in L(\exists p.\mathbb{A})$ over P. Fix a functional winning strategy σ for $\exists$ in $\mathcal{G}(\exists p.\mathbb{A}, \mathcal{K})@(a_I, s_I)$. Define $\mathcal{K}^p$ by:

$$\rho^p(s) = \rho(s) \cup X$$

$$X = \begin{cases} \{p\} & \text{if } \rho_\sigma(s) = a \text{ and} \\ & \sigma(\Delta^\exists(a, \sigma(s)) \models \Delta(a, \sigma(s) \cup \{p\}) \\ \emptyset & \text{else.} \end{cases}$$

σ induces a w.s. for $\exists$ in $\mathcal{G}(\mathbb{A}, \mathcal{K}^p)@(a_I, s_I)$. ■

From simulation to closure under existential quantification

Theorem: For every $\phi \in$ MSO there is an equivalent MSO-automaton $\mathbb{A}_\phi$.

Finishing the proof: Base cases and booleans are ok. For quantification, by the Simulation Theorem we can assume that $\mathbb{A}$ is non-deterministic.

$\mathcal{K} \in L(\exists p.\mathbb{A}_\phi)$ iff

$\exists X \subseteq S$ and $\mathcal{K}[p \mapsto X] \in L(\mathbb{A}_\phi)$ iff

$\exists X \subseteq S$ and $\mathcal{K}[p \mapsto X], s_I \models \phi$ iff

$\mathcal{K}, s_I \models \exists p.\phi$

■

The Simulation Theorem

We have to prove the simulation theorem!

Proof strategy:

1. We show that each one step FO formula is equivalent to a formula in normal form
2. same for the positive fragment
3. we use this normal form results to construct the equivalent non deterministic parity automaton

Normal forms for one-step logic

In the following we give

- Normal forms for arbitrary formulas of FOE and FOE^+,
- Strong forms of syntactic characterizations for the monotone fragments
- Normal forms for the monotone fragments.

Same can be done for FO and FO^+

Normal forms for one-step logic

Given a set A of (state) variables, the set of formula $\mathrm{FOE}(A)$ is defined as:

$$\phi ::= \top \mid \bot \mid x = y \mid x \neq y \mid a(x) \mid \neg a(x) \mid \phi \wedge \phi \mid \phi \vee \phi \mid \exists x.\phi \mid \forall x.\phi$$

with $a \in A$.

Normal forms for one-step logic

Given a set A of (state) variables, the set of formula $\mathrm{FOE}^{+}(A)$ is defined as:

$$\phi ::= \top \mid \bot \mid x = y \mid x \neq y \mid a(x) \mid \phi \wedge \phi \mid \phi \vee \phi \mid \exists x.\phi \mid \forall x.\phi$$

with $a \in A$.

A **type** is a subset of P.

Let Q be a type.

- $\tau_Q(x) := \begin{cases} \top & \text{if } Q = \emptyset \\ \bigwedge_{p \in Q} p(x) \wedge \bigwedge_{p \notin Q} \neg p(x) & \text{else;} \end{cases}$
- $\tau_Q^+(x) := \begin{cases} \top & \text{if } Q = \emptyset \\ \bigwedge_{p \in Q} p(x) & \text{else.} \end{cases}$

Definition: A formula $\phi \in \mathrm{FOE}(A)$ is in basic normal form (BF(A)) if it is of the form

$$\nabla_{\mathrm{FOE}}(\overline{Q}, \Pi) := \exists \overline{x}.\mathrm{diff}(\overline{x}) \wedge \bigwedge_{i \leq k} \tau_{Q_i}(x_i) \wedge \forall y.\mathrm{diff}(\overline{x}, y) \rightarrow \bigvee_{T \in \Pi} \tau_T(y)$$

Definition: A formula $\phi \in \mathrm{FOE}^{+}(A)$ is in basic normal form ($\mathrm{BF}^{+}(\mathrm{A})$) if it is of the form

$$\nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi) := \exists \overline{x}.\mathrm{diff}(\overline{x}) \wedge \bigwedge_{i \leq k} \tau^{+}_{Q_i}(x_i) \wedge \forall y.\mathrm{diff}(\overline{x}, y) \to \bigvee_{T \in \Pi} \tau^{+}_{T}(y)$$

Theorem: Every sentence of $\mathrm{FOE}(A)$ is equivalent to a disjunction of formulas in $\mathrm{BF}(\mathrm{A})$.

Proof: Given $\mathbf{D} = (D, V)$ and $\mathbf{D}' = (D', V')$, define

$$\mathbf{D} \sim_k^{=} \mathbf{D}' \iff \forall Q \subseteq A \; \big(|Q|_{\mathbf{D}} = |Q|_{\mathbf{D}'} < k \text{ or } |Q|_{\mathbf{D}}, |Q|_{\mathbf{D}'} \geq k\big)$$

$$|Q|_{\mathbf{D}} := \{d \in D \mid \mathbf{D} \models \tau_Q(d)\}$$

Proof (cont): It holds that

1. $\sim_k^{=}$ is an equivalence relation,
2. $\sim_k^{=}$ has finite index,
3. Every equivalence class E is characterized by a formula $\varphi_E^{=} \in \text{FOE}(A)$ with $qr(\varphi_E^{=}) = k$.

Proof (cont): It holds that

1. $\sim_k^{=}$ is an equivalence relation,
2. $\sim_k^{=}$ has finite index,
3. Every equivalence class E is characterized by a formula $\varphi_E^{=} \in \mathrm{FOE}(A)$ with $qr(\varphi_E^{=}) = k$.

By the fact that $\sim_k^{=}$ equals $\equiv_k$, every FOE sentence φ is equivalent to

$$\bigvee_{E:\|\varphi\|\cap E\neq\emptyset} \varphi_E^{=}$$

$D, V =$

(a) Normal forms for FOE

$D, V = \boxed{< k} \; \boxed{\geq k}$

Q_i

$$\exists x_1 \ldots \exists x_{n_i} \big(\mathrm{diff}(\overline{x}) \wedge \bigwedge_{1 \leq \ell \leq n_i} \tau_{Q_i}(x_\ell) \wedge \forall z.\mathrm{diff}(\overline{x}, z) \rightarrow \neg\tau_{Q_i}(z)\big),$$
$$n_i < k$$

(a) Normal forms for FOE

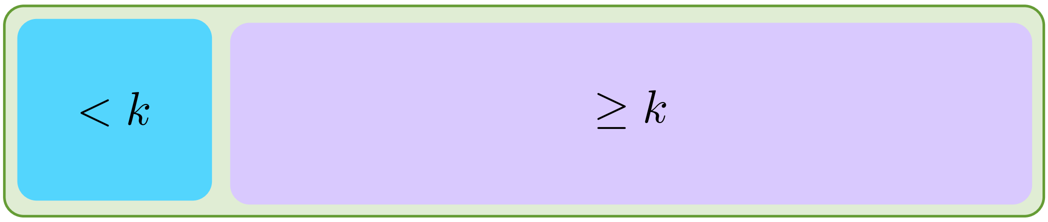
$D, V = \boxed{< k} \; \boxed{\geq k}$

$T \in \Pi$

$$\exists x_1 \ldots \exists x_k . \mathrm{diff}(\overline{x}) \wedge \bigwedge_{1 \leq \ell \leq k} \tau_T(x_\ell)$$

(a) Normal forms for FOE

$D, V =$



$$\varphi_E^{=} \equiv \nabla_{\mathrm{FOE}}(\overline{Q'}, \Pi)$$

The sequence $\overline{Q}'$ contains n_i occurrences of type Q_i and k occurrences of each type in Π.

Where we are in the proof of the Simulation Theorem

Proof strategy:

1. We show that each one step FO formula is equivalent to a formula in normal form
2. same for the positive fragment

3. we use this normal form results to construct the equivalent non deterministic parity automaton

Proof idea

1. we show that the positive fragment of FOE corresponds to its monotone fragment

2. we show that the previous normal form theorem for FOE provides us the expected normal form theorem for the positive fragment by using point 1

Definition: Given a one-step logic $\mathcal{L}(A)$ and $\varphi \in \mathcal{L}(A)$, We say that φ is monotone in $a \in A$ if for every (D, V) and assignment of first-order variables λ :

If $(D, V), \lambda \models \varphi$ and $V(a) \subseteq E$ then $(D, V[a \mapsto E]), \lambda \models \varphi$.

$\mathcal{L}\mathrm{C}_a(A)$

(b) Normal forms for positive FOE
- positive as monotone

Theorem: A sentence of $\mathrm{FOE}(A)$ is monotone in $a \in A$ iff it is equivalent to a sentence given by

$$\varphi ::= \psi \mid a(x) \mid \exists x.\varphi(x) \mid \forall x.\varphi(x) \mid \varphi \wedge \varphi \mid \varphi \vee \varphi$$

where $\psi \in \mathrm{FOE}(A \setminus \{a\})$

$\mathrm{FOEM}_a(A)$

Analgously for set of variables.

$\mathrm{FOEM}_{\mathrm{A}}(\mathrm{A}) = \mathrm{FOE}^{+}(\mathrm{A})$

(b) Normal forms for positive FOE
- positive as monotone

Proof: It follows by the following two lemmas.

Lemma 1: If $\varphi \in \mathrm{FOEM}_a(A)$ then φ is monotone in a;

Lemma 2: There exists an effective translation
$(-)^{\odot} : \mathrm{FOE}(A) \to \mathrm{FOEM}_a(A)$ such that
$\varphi \in \mathrm{FOE}(A)$ is monotone in a iff $\varphi \equiv \varphi^{\odot}$.

(b) Normal forms for positive FOE
- positive as monotone

Proof of Lemma 2: Define:

$$(\nabla_{FOE}(\overline{Q},\Pi))^{\odot} := \nabla^{a}_{FOE}(\overline{Q},\Pi)$$

$$\exists \overline{x}.\text{diff}(\overline{x}) \wedge \bigwedge_{i\leq k} \tau^{a}_{Q_i}(x_i) \wedge \forall y.\text{diff}(\overline{x},y) \to \bigvee_{T\in\Pi} \tau^{a}_{T}(x)$$

where $\tau^{a}_{Q}(x) := \bigwedge_{b\in Q} b(x) \wedge \bigwedge_{b\in A\setminus(Q\cup\{a\})} \neg b(x)$

(b) Normal forms for positive FOE
- positive as monotone

Proof of Lemma 2: Define:

$$(\nabla_{FOE}(\overline{Q},\Pi))^{\odot} := \nabla^{a}_{FOE}(\overline{Q},\Pi)$$

By Lemma 1, we have $\Leftarrow$.

Proof of Lemma 2: Define:

$$(\nabla_{FOE}(\overline{Q}, \Pi))^{\odot} := \nabla^{a}_{FOE}(\overline{Q}, \Pi)$$

For $\Rightarrow$ we check that:

$$(D, V) \models \phi \text{ iff } (D, V) \models \phi^{\odot}$$

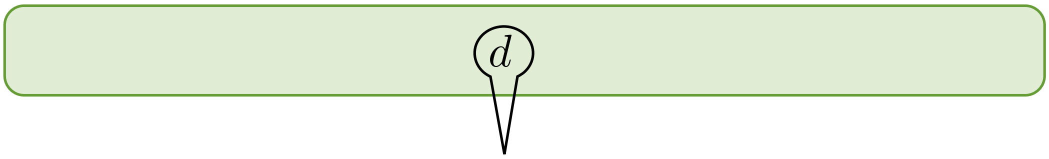
Proof of Lemma 2 (cont.): The direction $\Rightarrow$ is trivial.

For $\Leftarrow$ let $(D, V) \models \nabla^{a}_{FOE}(\overline{Q}, \Pi)$.

Proof of Lemma 2 (cont.): The direction $\Rightarrow$ is trivial.

For $\Leftarrow$ let $(D, V) \models \nabla^a_{FOE}(\overline{Q}, \Pi)$.

$D, V =$



witness of a a-positive type T in $\overline{Q} \cup \Pi$

$d \mapsto \tau^a_{T_d}$

Proof of Lemma 2 (cont.): Consider (D, V') with

$$V'(b) = \begin{cases} V(b) & a \neq b \\ V(b) \setminus \{d \in D \mid a \notin T_d\} & a = b \end{cases}$$

Proof of Lemma 2 (cont.): Consider (D, V') with

$$V'(b) = \begin{cases} V(b) & a \neq b \\ V(b) \setminus \{d \in D \mid a \notin T_d\} & a = b \end{cases}$$

It holds that $(D, V') \models \nabla_{FOE}(\overline{Q}, \Pi)$.

Thus $(D, V') \models \varphi$, and by monotonicity $(D, V) \models \varphi$.

$\blacksquare$

Proof idea

1. we show that the positive fragment of FOE corresponds to its monotone fragment

2. we show that the previous normal form theorem for FOE provides us the expected normal form theorem for the positive fragment by using point 1

(b) Normal forms for positive FOE - providing a normal form

Corollary:

1. φ is monotone in $a \in A$
   iff
   it is equivalent to a formula in $\bigvee \nabla^{a}_{\mathrm{FOE}}(\overline{Q}, \Pi)$.

2. φ is monotone in every $a \in A$
   (i.e., $\varphi \in \mathrm{FOE}^{+}(A)$) iff
   it is equivalent to a formula in the basic form
   $\bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$

Where we are in the proof of the Simulation Theorem

Proof strategy:

1. We show that each one step FO formula is equivalent to a formula in normal form
2. same for the positive fragment

3. we use this normal form results to construct the equivalent non deterministic parity automaton

Transition in normal form:

$$\Delta : A \times \wp P \to \mathrm{SLatt}(BF^+(A))$$

Transition for non-deterministic automata

$$\Delta : A \times \wp P \to \mathrm{SLatt}(SBF^+(A))$$

Definition (change of base): Let $\varphi := \nabla^+_{\mathrm{FOE}}(\overline{Q}, \Pi)$. For each type T in $\overline{Q} \cup \Pi$, we define the formula $\tau^\wp_T(x)$ as follows:

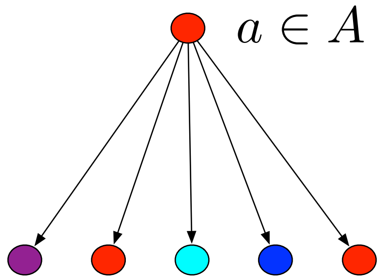
$$\tau^\wp_T(x) \quad := \quad \begin{cases} T(x) & \text{If } S \neq \emptyset \\ \top & \text{Otherwise} \end{cases}$$

We denote with $\varphi^\wp \in \mathrm{SBF}^+(A)$ the sentence

$$\exists x_1 \ldots x_k \; (\mathrm{diff}(\overline{x}) \wedge \bigwedge_{1 \leq i \leq k} \tau^\wp_{Q_i}(x_i) \wedge \forall z \; (\mathrm{diff}(\overline{x}, z) \to \bigvee_{T \in \Pi} \tau^\wp_T(z))).$$

In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

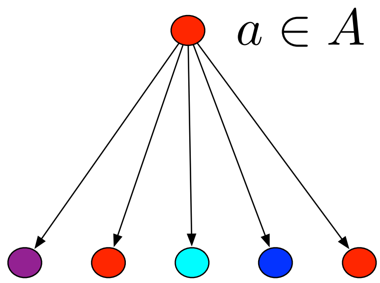


In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

$\exists$:

$D=$



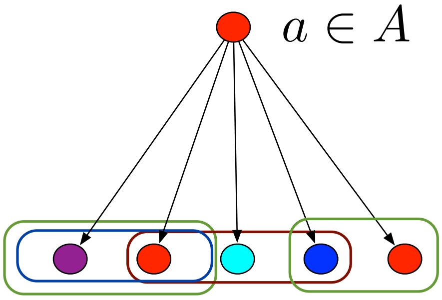
In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

$a \in A$

$\exists$:

$(D, V) =$



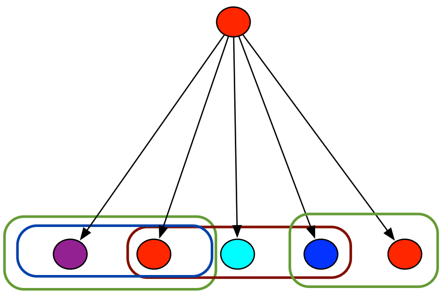
$$V : A \to \wp D$$

In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

$\exists$:

$(D, V) =$



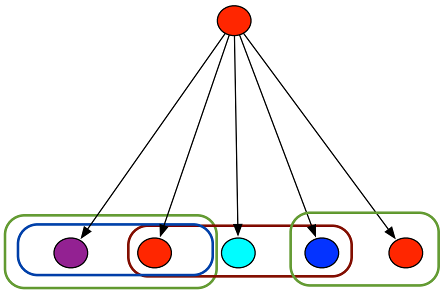
$\models \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$

$$V : A \to \wp D$$

In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

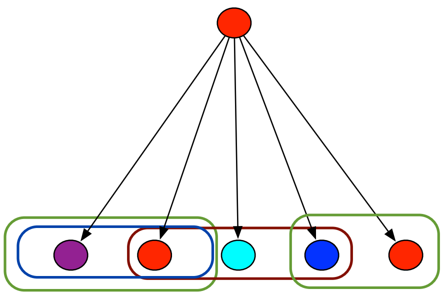
$\forall$:

$(D, V) =$  $\models \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$

In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

$\forall$:

$(D, V) =$  $\models \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$

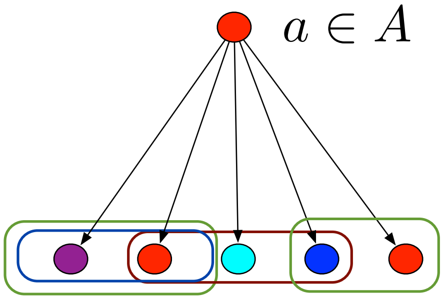
$a_i \in A$

In the search of non-determinism

$$\Delta : (a, Q) \mapsto \bigvee \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

$a \in A$

$\exists$:



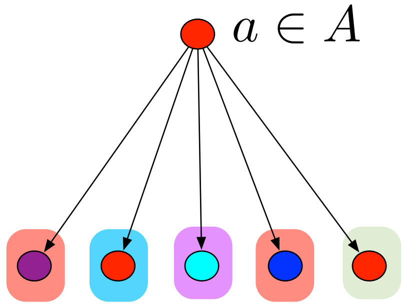
$$(D, V) = \quad \models \nabla^{+}_{\mathrm{FOE}}(\overline{Q}, \Pi)$$

$$V : A \to \wp D$$

In the search of non-determinism

$a \in A$

$\exists$:



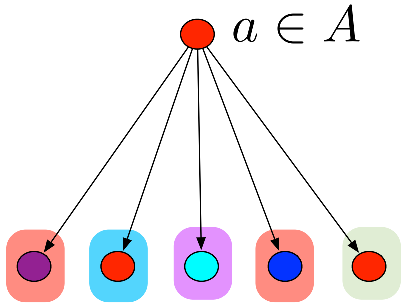
$$(D, V) =$$

$$V : \wp A \to \wp D$$

In the search of non-determinism

$a \in A$

$\exists$:



$(D, V) = \ \ldots \models \nabla^{\wp}_{\mathrm{FOE}}(\overline{Q}, \Pi)$

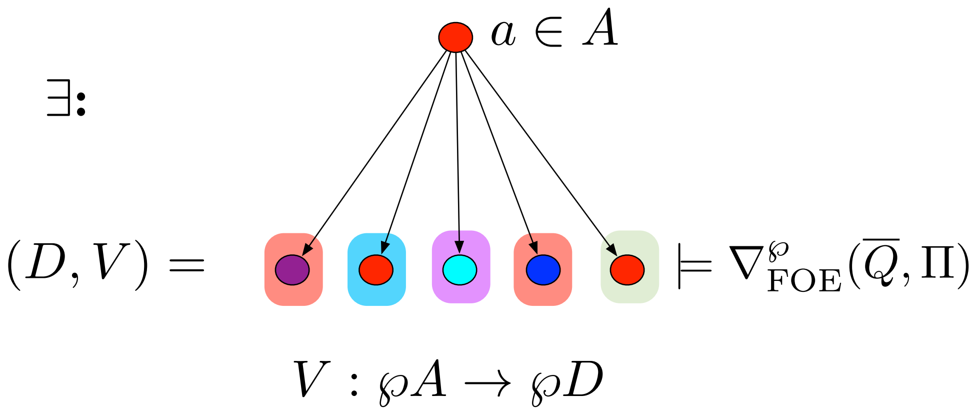
$$V : \wp A \to \wp D$$

In the search of non-determinism

Definition: Let $\mathbb{A} = (A, a_I, \Delta, \Omega)$ over C be an MSO-automaton.
Fix $a \in A$ and $c \in C$. The sentence $\Delta^{\star}(a, c)$ is defined as

$$\Delta^{\star}(a, c) := \Delta(a, c)[(a, b) \setminus b \mid b \in A],$$

where $\Delta(a, c)[(a, b) \setminus b \mid b \in A]$ denotes the sentence in $\mathrm{FOE}^{+}(A \times A)$ obtained by replacing each occurrence of an unary predicate $b \in A$ in $\Delta(a, c)$ with the unary predicate $(a, b) \in A \times A$.



Definition: Let $\mathbb{A} = (A, a_I, \Delta, \Omega)$ over C be an MSO-automaton.
Let $c \in C$ and $R \in \wp(A \times A)$.
There is a sentence $\Psi^{\#}_{R,c} \in \text{SLatt}(\text{BF}^+(A \times A))$ s.t.

$$\bigwedge_{a \in \text{Ran}(R)} \Delta^{\star}(a, c) \equiv \Psi^{\#}_{R,c}.$$

Let $\Psi_{R,c} \in \text{SLatt}(\text{SBF}^+(\wp(A \times A)))$ be $(\Psi^{\#}_{R,c})^{\wp}$.

In the search of non-determinism

Definition: Let $\mathbb{A} = (A, a_I, \Delta, \text{rank})$ over C be an MSO-automaton.
The automaton $\mathbb{A}^{\wp} = (A^{\wp}, a_I^{\wp}, \Delta^{\wp}, \text{NBT}_{\text{rank}})$ is given by

$$
\begin{aligned}
A^{\wp} &:= \wp(A \times A) \\
a_I^{\wp} &:= \{a_I, a_I\} \\
\Delta^{\wp}(R, c) &:= \Psi_{R,c} \\
\text{NBT}_{\text{rank}} &:= \{w \in (\wp(A \times A))^{\omega} \mid \\
&\quad \text{every trace in } w \text{ is good}\}.
\end{aligned}
$$

the max parity occurring infinitely often along $\text{rank}(w) \in \mathbb{N}$ is even

In the search of non-determinism

Proposition: $L(\mathbb{A}) = L(\mathbb{A}^{\wp})$.

Let $\mathbb{Z}$ be the deterministic parity automaton s.t. $L(\mathbb{Z}) = \mathrm{NBT}_{\mathrm{rank}}$.

Definition: The non-deterministic MSO-automaton $\mathbb{A}^N = (A^\wp \times Z, (a_I^\wp, z_I), \Delta^N, \mathrm{rank}^N)$ is given by:

$$\begin{aligned} \mathrm{rank}(q,z) &:= \mathrm{rank}_Z(z), \\ \Delta((q,z),c) &:= \bigvee\{\mathrm{Shift}_z(\varphi) \in \mathrm{SBF}^+(A^\wp \times Z) \mid \\ &\qquad \varphi \text{ is a disjunct of } \Delta^\wp(q,c)\}. \end{aligned}$$

$$\mathrm{Shift}_z(\varphi) := \varphi[(q, \Delta_Z(z,q))/q \mid q \in A^\wp]$$

Proposition: $L(\mathbb{A}^N) = L(\mathbb{A}^\wp)$.

Where we are in the proof of the Simulation Theorem

Proof strategy:

1. We show that each one step FO formula is equivalent to a formula in normal form
2. same for the positive fragment

3. we use this normal form results to construct the equivalent non deterministic parity automaton

■

Where we are

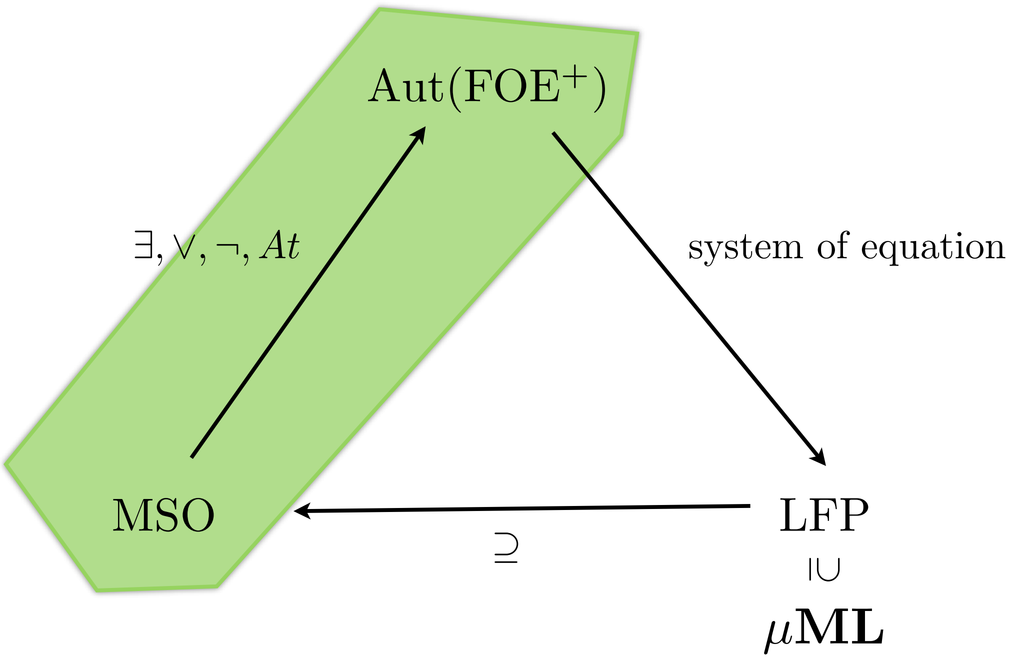
(over trees)

$\mathbf{MSO}$ $\subseteq$ $\mathbf{MSO}$-automata FOE^{+}

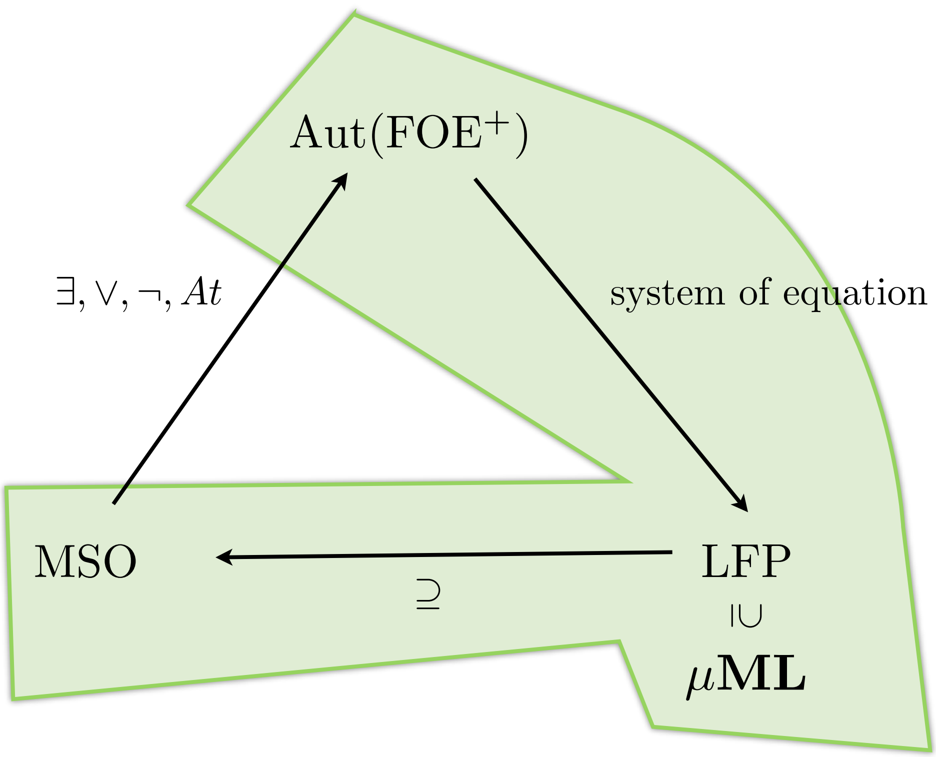
$\mu\mathbf{ML}$ = μ-automata FO^{+}

(over K)

Automata for MSO



Automata for MSO



Definition: The fixed point logic LFP is given by:

$$\varphi ::= q(x) \mid R(x,y) \mid x = y \mid \neg\varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \mu p.\varphi(p,x)$$

where

- $p, q \in P$, $x, y \in X$;
- moreover p occurs only positively in $\varphi(p,x)$ and
- x is the only free variable in $\varphi(p,x)$.

The semantics of the fixpoint formula $\mu p.\phi(p,x)$ is the expected one: given $\mathcal{K}$ and $s \in S$,

$$\mathcal{K} \models \mu p.\phi(p,s)$$

iff

$$s \in \text{lfp}.F_\phi = \bigcap \{X \subseteq S \mid F_\phi(X) \subseteq X\}, \text{ where}$$

$$F_\phi(X) := \{t \in T \mid \mathcal{K}[p \mapsto X] \models \phi(p,t)\}.$$

Proposition: There is an effective translation $(-)^{\circledast} : \mu ML \to \mathrm{LFP}$ s.t. for every $\mathcal{K}, s \in S$ the following are equivalent:

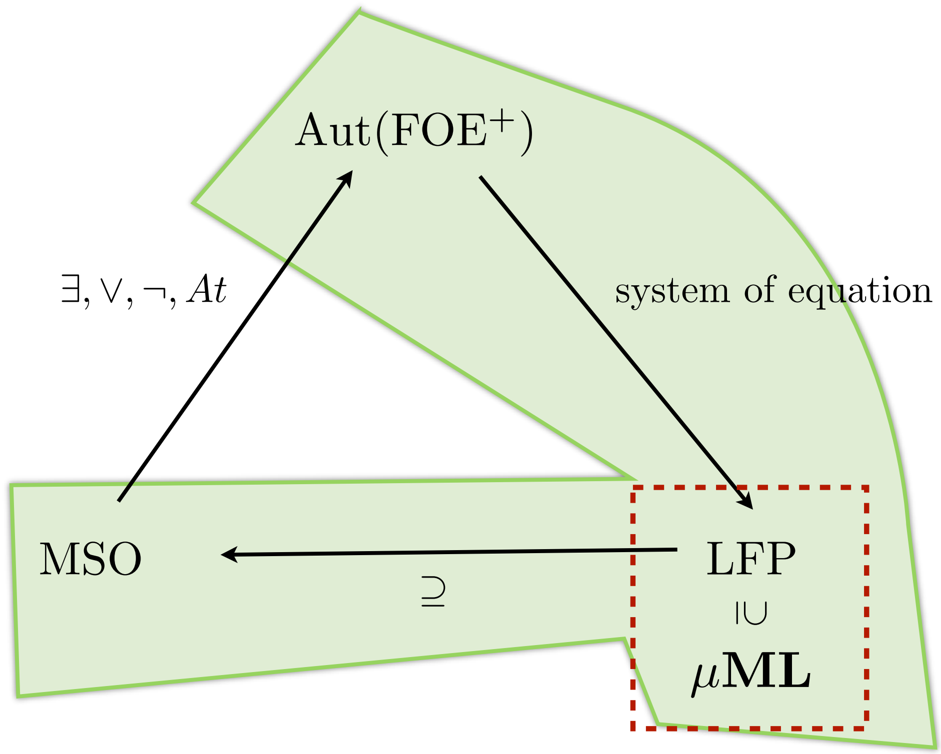
- $(\mathcal{K}, s) \models \varphi$,
- $\mathcal{K} \models (\varphi)^{\circledast}(s)$.

Proof: Consider translation $(-)^{\circledast}_x : \mu ML \to \mathrm{LFP}$ for $x \in X$ given by:

- $(p)^{\circledast}_x = p(x)$,
- $(\Diamond\phi)^{\circledast}_x = \exists y. R(x, y) \wedge (\phi)^{\circledast}_y$,
- $(\neg\phi)^{\circledast}_x = \neg(\phi)^{\circledast}_x$,
- $(\psi \wedge \phi)^{\circledast}_x = (\psi)^{\circledast}_x \wedge (\phi)^{\circledast}_x$,
- $(\mu p.\phi)^{\circledast}_x = \mu p.(\phi)^{\circledast}_x$,

$\blacksquare$

Automata for MSO



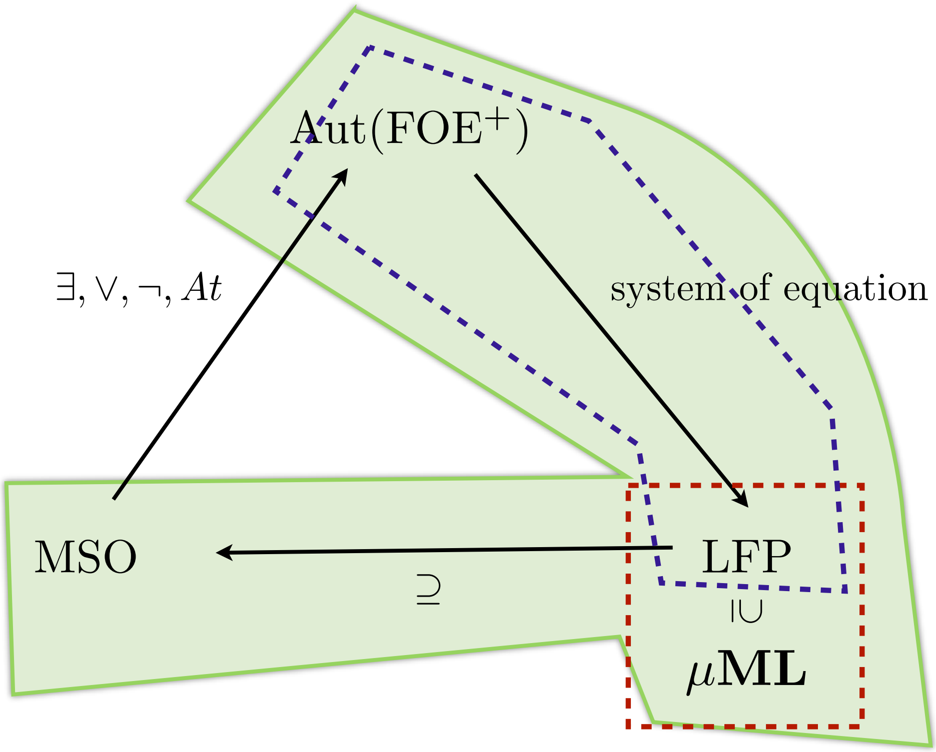
From the MSO-automata to LFP

Proposition: For every MSO-automaton there is an equivalent formula in LFP.

Proof: Proceed like for modal automata and μ-formulas.

$\blacksquare$

Automata for MSO



From the LFP to MSO

Proposition: There is a translation $(-)^{\ominus}$: LFP $\to$ MSO s.t. for every $\mathcal{K}$, and valuation V the following are equivalent:

- $\mathcal{K}, V \models \varphi$,
- $\mathcal{K}, V \models (\varphi)^{\ominus}$.

Proof:

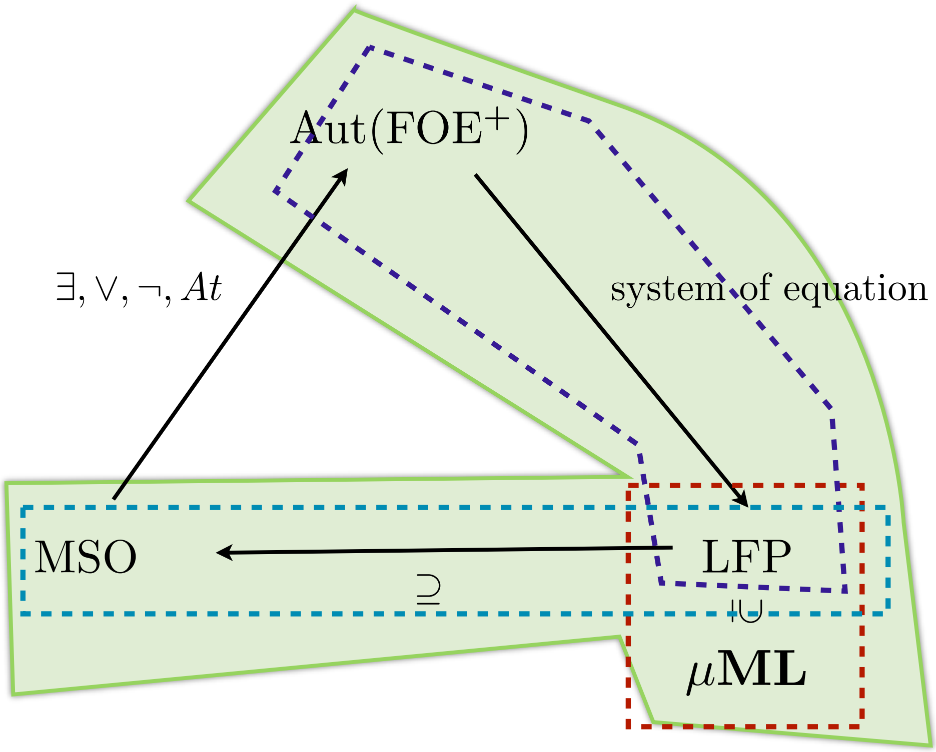
$$(\mu p.\phi(p,x))^{\ominus}$$

$$=$$

$$\exists X.\big(Xx \wedge \forall Y.(\forall y.(\phi(p,y)^{\ominus} \to Yy) \to \forall z.(Xz \to Yz))\big)$$

■

Automata for MSO



Where we are

(over trees)

MSO = **MSO**-automata $\quad$ FOE$^+$

translation ↑

μ**ML** = μ-automata $\quad$ FO$^+$

(over TS)

Finishing the proof

Proposition: Let $(-)^\bullet : \text{FOE}^+(A) \to \text{FO}^+(A)$ given by

$$(\nabla^+_{FOE}(\overline{Q}, \Pi))^\bullet = \nabla^+_{FO}(\overline{Q}, \Pi)$$

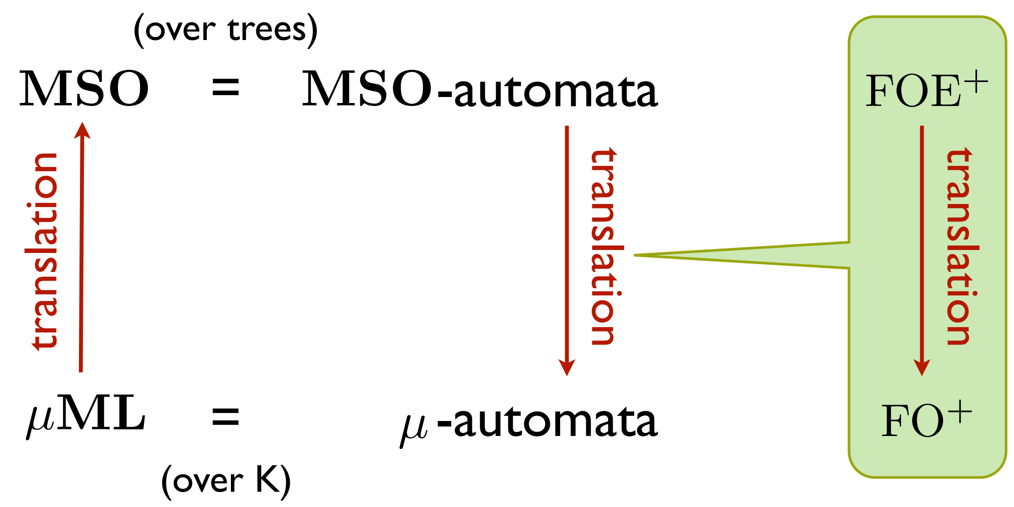
$$\nabla^+_{\text{FOE}}(\overline{Q}, \Pi) := \exists \overline{x}.\text{diff}(\overline{x}) \wedge \bigwedge_{i \leq k} \tau^+_{Q_i}(x_i) \wedge \forall z.\text{diff}(\overline{x}, z) \to \bigvee_{T \in \Pi} \tau^+_T(z)$$

$$\nabla^+_{\text{FO}}(\overline{Q}, \Pi) := \exists \overline{x}. \bigwedge_{i \leq k} \tau^+_{Q_i}(x_i) \wedge \forall z. \bigvee_{T \in \Pi} \tau^+_T(z)$$

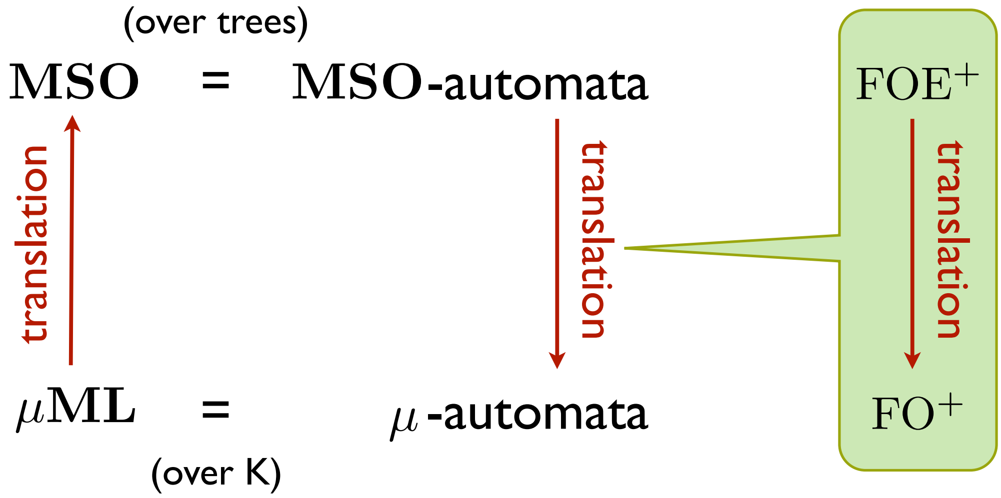
$$\mathbf{D} \models \phi^\bullet \text{ iff } \mathbf{D}_\omega \models \phi$$

$\blacksquare$

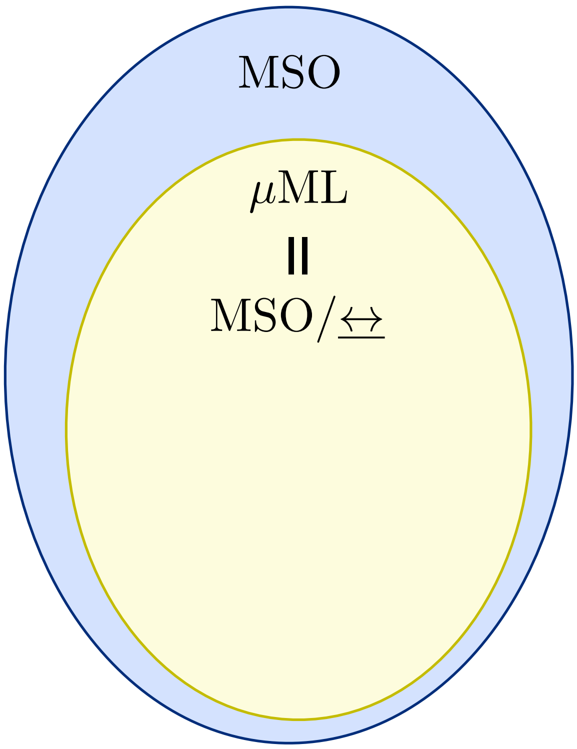
Closing the cycle



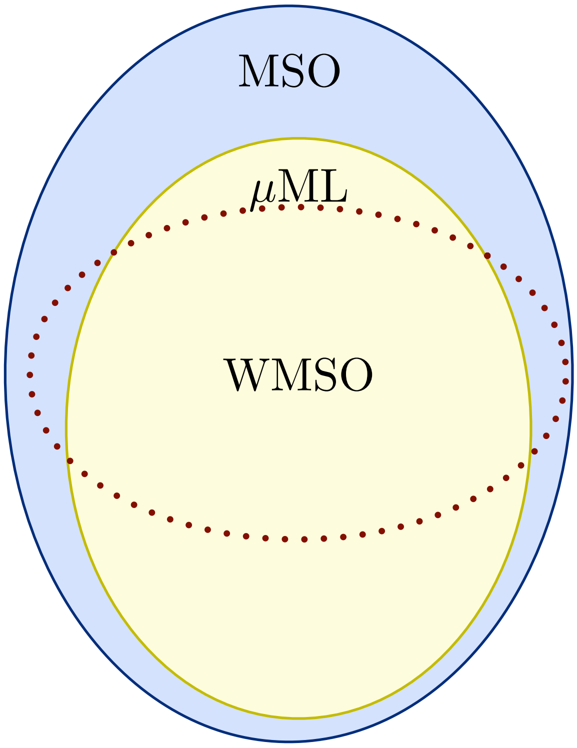
Question:
where this picture breaks on finite models ???



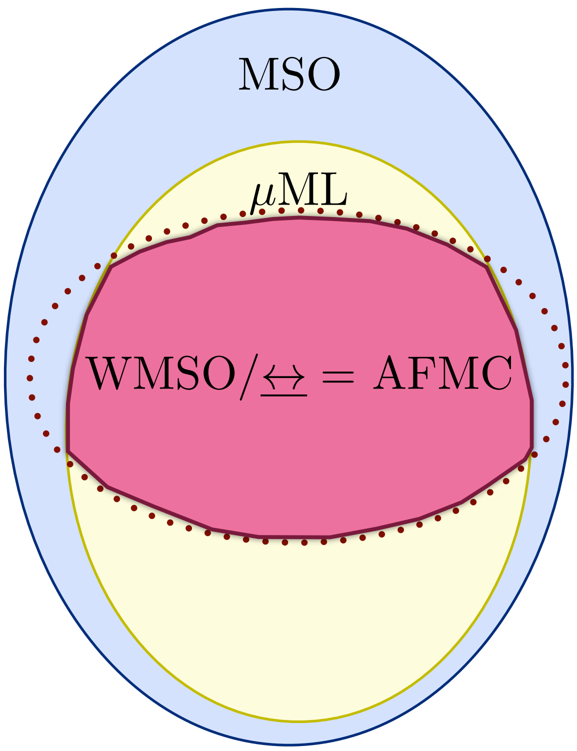
Over all models

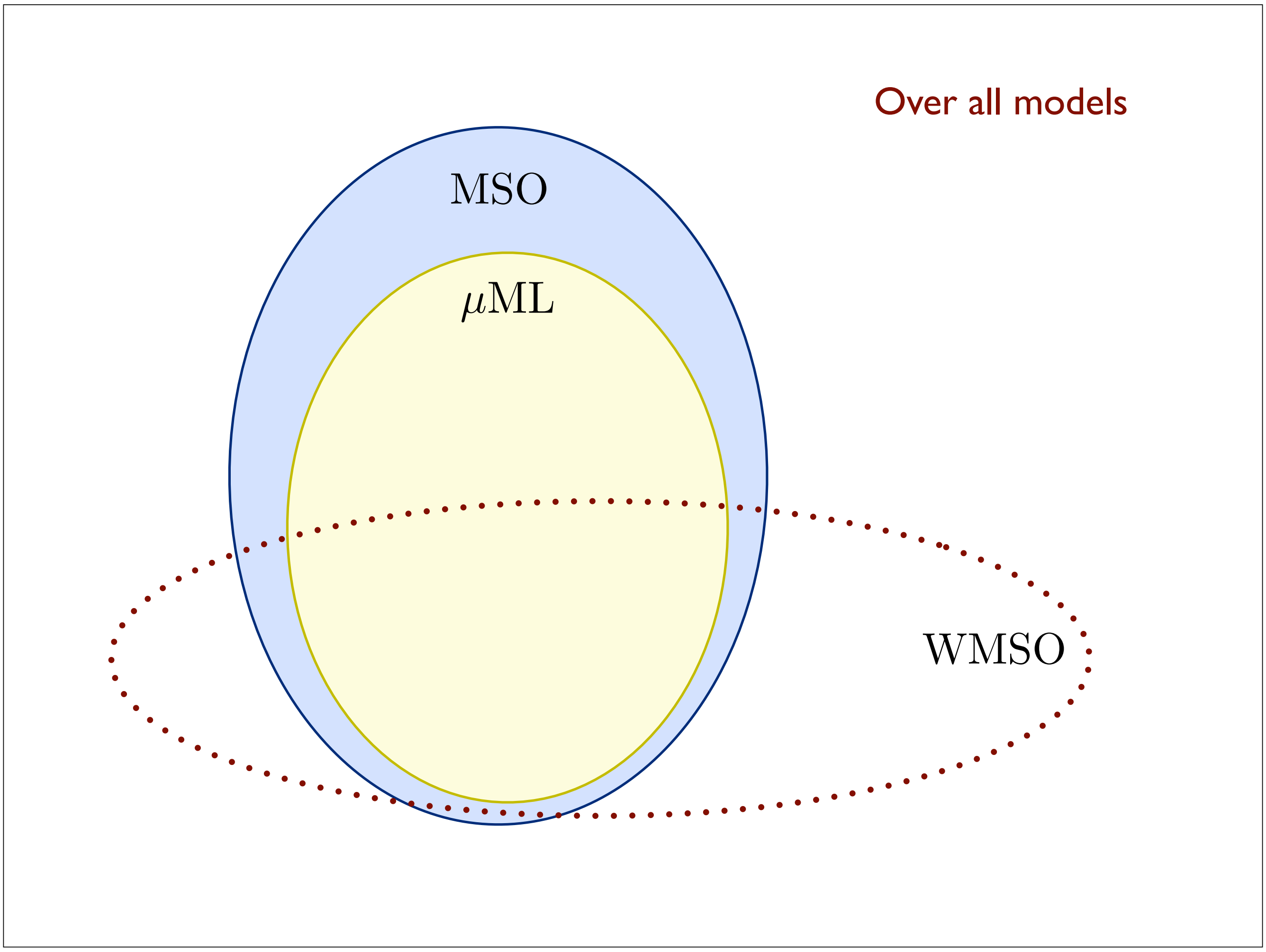


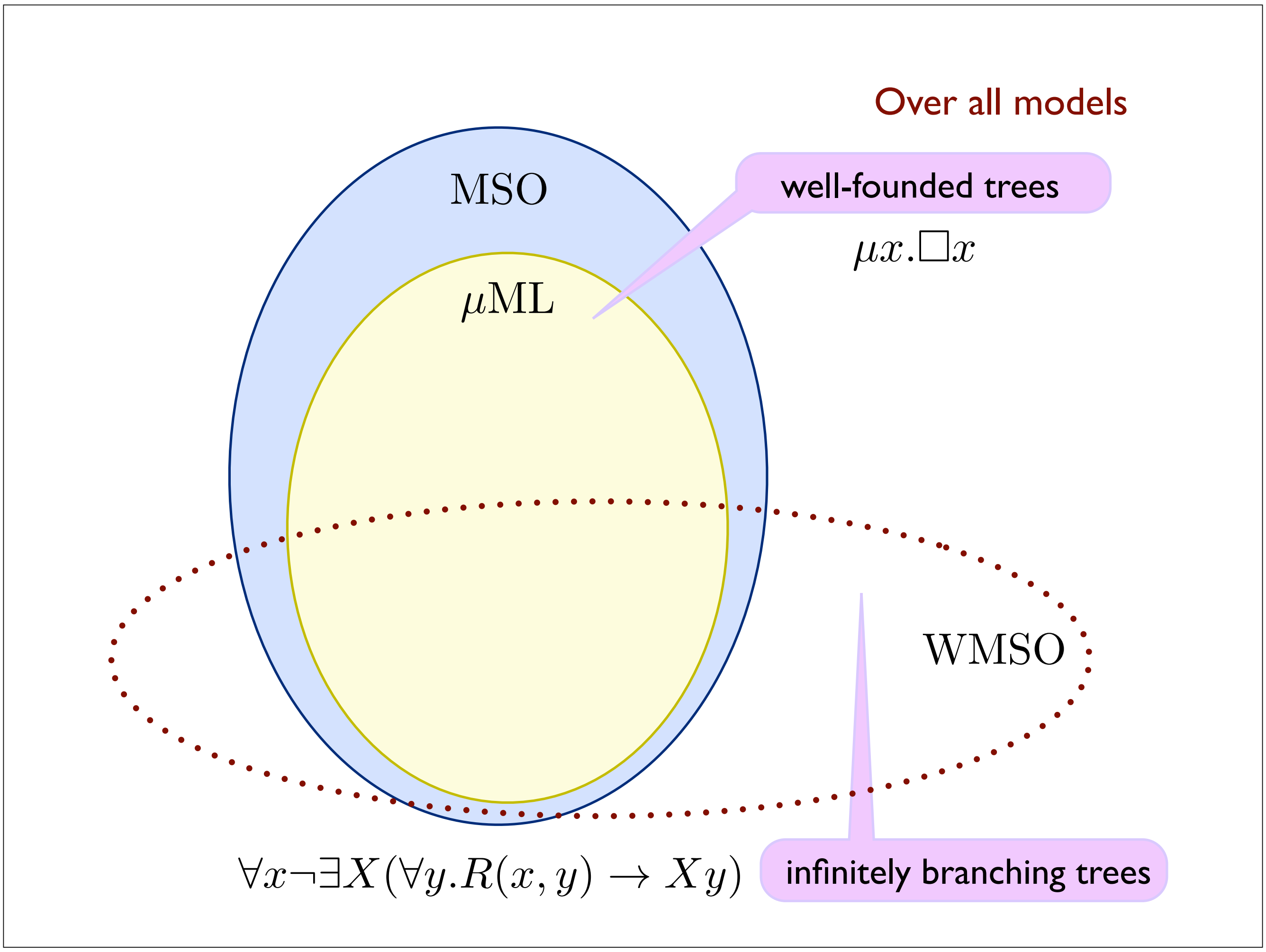
Over finitely branching trees



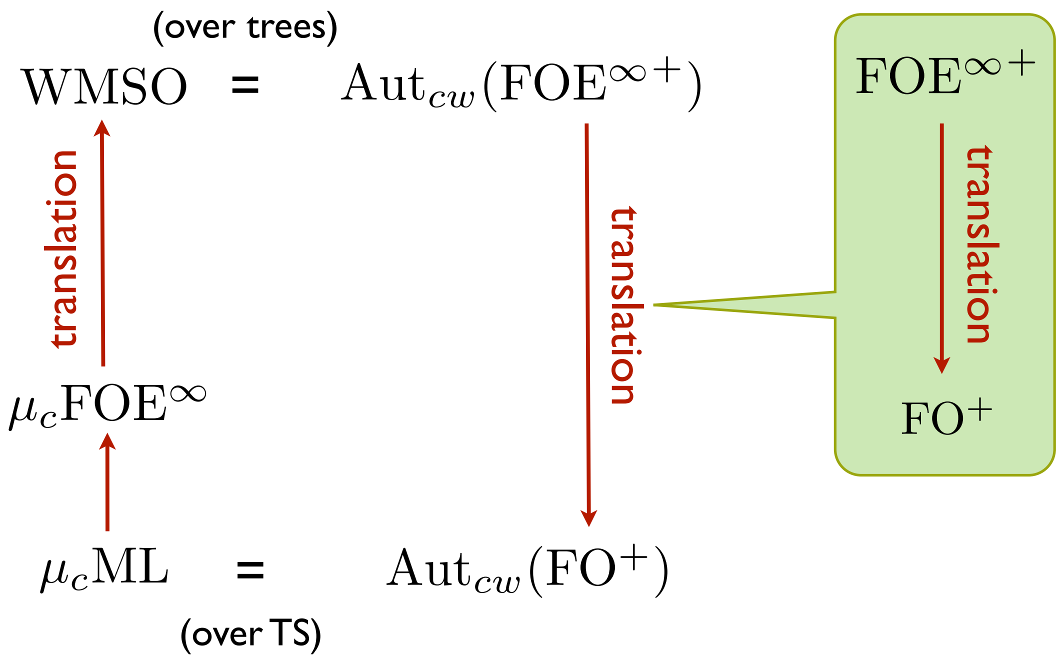
Over finitely branching trees




Over all models
MSO
μML
WMSO


Over all models
MSO
well-founded trees
μx.□x
μML
WMSO
∀x¬∃X(∀y.R(x, y) → Xy)
infinitely branching trees

The same strategy works for WMSO [Carreiro, F., Venema, Zanasi (2014)]



What we have seen today...