Queries, Modalities, Relations, Trees, XPath
Lecture IV and V
Correspondence Languages
Definability, Expressivity and Equivalence

Tadeusz Litak

Department of Computer Science
University of Leicester

July 2010: draft

FO syntax

Definition

- Fix $k \in \mathrm{N}_0 \cup \{\infty\}$, Π a collection of labels and Σ a signature. The set of $FOL_{k,\Sigma,\Pi}$-formulas is defined as follows:

$$\phi ::= x_i = x_j \mid S(x_i, x_j) \mid P(x_i) \mid \neg\phi \mid \phi \vee \phi \mid \forall x_i.\phi$$

$$(S \in \Sigma, P \in \Pi, i, j < k)$$

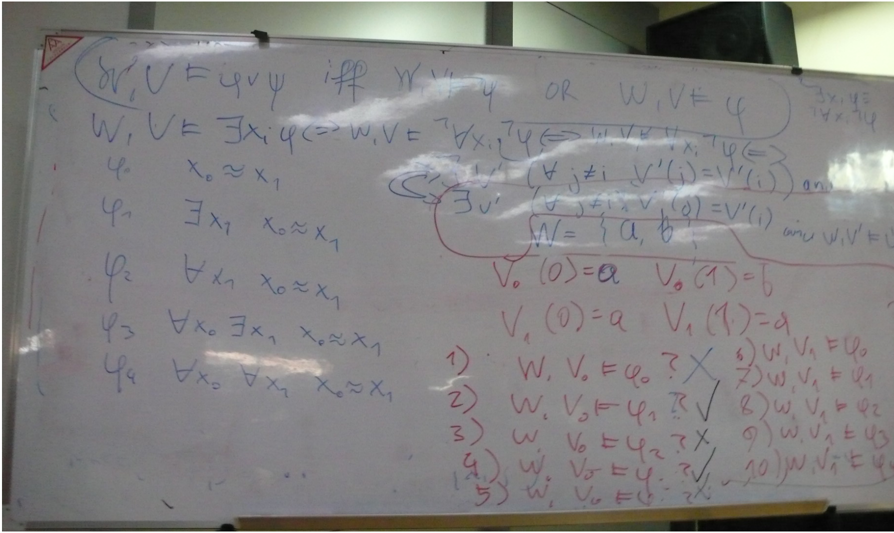
- A variable is free in a formula iff it is not inside the scope of any quantifier
- By $FOL^i_{k,\Sigma,\Pi}$, we denote the set of those $\phi \in FOL_{k,\Sigma,\Pi}$ s.t. x_j is free in ϕ iff $j < i$
- A sentence is a formula with no free variables, i.e., an element of $FOL^0_{k,\Sigma,\Pi}$

Exercise

1. Some exercises concerning free variables and sentences—on the whiteboard

If $k = \infty$, it is often dropped from the notation.
Σ can be dropped when it is clear from the context.
Π is dropped when either

- it is clear from the context or
- we are considering unlabelled structures


OR
φ0 x0 ≈ x1
φ1 ∃x1 x0 ≈ x1
φ2 ∀x1 x0 ≈ x1
φ3 ∀x0 ∃x1 x0 ≈ x1
φ4 ∀x0 ∀x1 x0 ≈ x1
W = {a, b}
V0 (0) = a V0 (1) = b
V1 (0) = a V1 (1) = a
1) W, V0 ⊨ φ0 ?
2) W, V0 ⊨ φ1 ?
3) W, V0 ⊨ φ2 ?

Definition (FOL Satisfaction)

Let $\mathfrak{W} \in \mathsf{Str}(\Sigma)$ ($\mathfrak{W} := \langle \mathfrak{F}, \Lambda \rangle \in \mathsf{Str}(\Sigma, \Pi)$)

- A $FOL_{k,\Sigma,\Pi}$-valuation in $\mathfrak{W}$ is any mapping $k \mapsto \underline{\mathfrak{W}}$.
- Fix a valuation V in $\mathfrak{W}$. The FOL-satisfaction relation is defined as follows:

$$
\begin{array}{lll}
\mathfrak{W}, V \models x_i = x_j & \text{iff} & V(i) = V(j) \\
\mathfrak{W}, V \models S(x_i, x_j) & \text{iff} & V(i) S^{\mathfrak{W}} V(j) \\
\mathfrak{W}, V \models P(x_i) & \text{iff} & V(i) \in \Lambda(P) \\
\mathfrak{W}, V \models \neg\phi & \text{iff} & \mathfrak{W}, V \not\models \phi \\
\mathfrak{W}, V \models \phi \vee \psi & \text{iff} & \mathfrak{W}, V \models \phi \text{ or } \mathfrak{W}, V \models \psi \\
\mathfrak{W}, V \models \forall x_i.\phi & \text{iff} & \forall V' (\forall j \neq i. V(j) = V'(j)) \text{ implies} \\
 & & \qquad \mathfrak{W}, V' \models \phi
\end{array}
$$

Exercise

1. Does satisfaction of a sentence by $\mathfrak{W}$ depend on valuation?
2. Some exercises on satisfaction—on the whiteboard

FOL Theories of Models

Let $\mathfrak{W} \in \mathsf{Str}(\Sigma, \Pi)$ and $w_0, \ldots, w_{n-1} \in \underline{\mathfrak{W}}$.

FOL Theories of Models

Let $\mathfrak{W} \in \mathsf{Str}(\Sigma, \Pi)$ and $w_0, \ldots, w_{n-1} \in \underline{\mathfrak{W}}$.

Definition

- For $\phi \in FOL_{k,\Sigma,\Pi}$, $i_0, \ldots, i_{n-1} < k$ we define [validity]

$$\mathfrak{W} \vDash \phi[i_0/w_0, \ldots i_{n-1}/w_{n-1}]$$
$$\text{iff}$$
$$\forall V. \quad V(i_0) = w_0 \ldots V(i_{n-1}) = w_{n-1} \text{ implies } \mathfrak{W}, V \vDash \phi$$

- The FOL_k-theory of $\mathfrak{W}$ is defined as

$$\mathsf{Th}_{FOL_{k,\Sigma,\Pi}}(\mathfrak{W}) := \{\phi \in FOL^0_{k,\Sigma,\Pi} \mid \mathfrak{W} \vDash \phi\}$$

Exercise

1. Does validity of a sentence depend on i and/or V?
2. If ϕ is a sentence and $\phi \notin \mathrm{Th}_{FOL_{k,\Sigma,\Pi}}(\mathfrak{W})$, is it true that $\neg\phi \in \mathrm{Th}_{FOL_{k,\Sigma,\Pi}}(\mathfrak{W})$?
3. Universal closure of a formula—on the whiteboard

FOL Models and FOL Definability

FOL Models and FOL Definability

Definition

- Conversely, with every formula ϕ, we associate

$$\mathrm{Mod}(\{\phi\}) := \{\mathfrak{W} \mid \mathfrak{W} \vDash \phi\}$$

- Let $\mathfrak{W}$ be a (Π-labelled) Σ-structure and $\underline{\mathfrak{W}}$ its carrier, $\phi \in FOL^{i}_{k,\Sigma,\Pi}$

$$\phi^{\mathfrak{W}} := \{\langle w_0, \ldots w_{i-1}\rangle \mid \mathfrak{W} \vDash \phi[0/w_0, \ldots, (n-1)/w_{n-1}]\}$$

- For the particular case of $\phi \in FOL^{0}_{k,\Sigma,\Pi}$,

$$\phi^{\mathfrak{W}} = \top \text{ iff } \mathfrak{W} \vDash \phi \text{ and } \phi^{\mathfrak{W}} = \bot \text{ otherwise}$$

- $X \subseteq \underline{\mathfrak{W}}^{i}$ is $FOL^{i}_{k,\Sigma,\Pi}$-definable if $\exists \phi \in FOL^{i}_{k,\Sigma,\Pi}.\quad \phi^{\mathfrak{W}} = X$

Exercise

1. Some exercises on definability and expressivity—on the whiteboard

Modal Language

Definition

Fix Π a collection of labels and Σ a signature. The set of *$ML_{\Sigma,\Pi}$-formulas* is defined as follows:

$$\phi ::= P \mid \langle S \rangle \phi \mid \neg\phi \mid \phi \vee \phi \qquad (S \in \Sigma, P \in \Pi)$$

Note we have no quantifiers now!

(as in case of FOL, define other boolean connectives as abbreviations)

Semantics

Definition (ML-satisfaction)

Let $\mathfrak{W} := \langle \mathfrak{F}, \Lambda \rangle \in \mathsf{Str}(\Sigma, \Pi)$

$$\begin{array}{ll}
\mathfrak{W}, w \vDash P & \text{if } w \in \Lambda(P) \\
\mathfrak{W}, w \vDash \psi \vee \phi & \text{if } \mathfrak{W}, w \vDash \psi \text{ or } \mathfrak{W}, w \vDash \phi \\
\mathfrak{W}, w \vDash \neg\psi & \text{if not } \mathfrak{W}, w \vDash \psi \\
\mathfrak{W}, w \vDash \langle S \rangle \psi & \text{if } \exists y \in \underline{\mathfrak{W}}.(w S^{\mathfrak{W}} y \text{ and } \mathfrak{W}, y \vDash \psi)
\end{array}$$

Exercise

1. Exercises on satisfaction of standard modal formulas—on the whiteboard (introduce variable free as well?)

ML Theories of Models and Frames

Let $\mathfrak{W} \in \mathsf{Str}(\Sigma, \Pi)$ and $\mathfrak{F} = \mathsf{Str}(\Sigma)$.

ML Theories of Models and Frames

Let $\mathfrak{M} \in \mathsf{Str}(\Sigma, \Pi)$ and $\mathfrak{F} = \mathsf{Str}(\Sigma)$.

Definition

- For $\phi \in ML_{\Sigma,\Pi}$, we define [ML-global satisfaction]
  $\mathfrak{M} \models \phi$ iff for all $w \in \underline{\mathfrak{M}}$, $\mathfrak{M}, w \models \phi$
- For $\phi \in ML_{\Sigma,\Pi}$, we define [ML-validity]
  $\mathfrak{F} \models \phi$ iff for all $\mathfrak{M} = \langle \mathfrak{F}, \Lambda \rangle$, $\mathfrak{M} \models \phi$
- [ML-theories of structures]

$$\mathsf{Th}_{ML}(\mathfrak{M}) := \{\phi \in ML_{\Sigma,\Pi} \mid \mathfrak{M} \models \phi\}$$
$$\mathsf{Th}_{ML}(\mathfrak{F}) := \{\phi \in ML_{\Sigma,\Pi} \mid \mathfrak{F} \models \phi\}$$

Exercise

1. If ϕ is a sentence and $\phi \notin \mathrm{Th}_{ML}(\mathfrak{W})$, is it true that $\neg\phi \in \mathrm{Th}_{ML}(\mathfrak{W})$?
2. How about $\mathrm{Th}_{ML}(\mathfrak{F})$?

ML Models and ML Definability

ML Models and ML Definability

Definition

- Conversely, with every formula $\phi \in ML_{\Sigma,\Pi}$, we associate

$$\begin{aligned}\mathrm{Mod}_{\Sigma,\Pi}(\{\phi\}) &:= \{\mathfrak{W} = \langle \mathfrak{F}, \Lambda \rangle \in \mathrm{Str}(\Sigma, \Pi) \mid \mathfrak{W} \vDash \phi\} \\ \mathrm{Mod}_{\Sigma}(\{\phi\}) &:= \{\mathfrak{F} \in \mathrm{Str}(\Sigma) \mid \mathfrak{F} \vDash \phi\} \\ \phi^{\mathfrak{W}} &:= \{w \in \mathfrak{W} \mid \mathfrak{W}, w \vDash \phi\}\end{aligned}$$

- $X \subseteq \underline{\mathfrak{W}}$ is *ML*-definable if $\exists \phi \in MO_{\Sigma,\Pi}.X = \phi^{\mathfrak{W}}$

Definition

- Fix $k \in \mathrm{N}_0 \cup \{\infty\}$, Π a collection of labels and Σ a signature. The set of $FOL_{k,\Sigma,\Pi}$-formulas is defined as follows:

$$\phi ::= x_i = x_j \mid S(x_i, x_j) \mid P(x_i) \mid \neg\phi \mid \phi \wedge \phi \mid \forall x_i.\phi$$

$$(S \in \Sigma, P \in \Pi, i, j < k)$$

- A variable is free in a formula iff it is not inside the scope of any quantifier
- By $FOL^i_{k,\Sigma,\Pi}$, we denote the set of those $\phi \in FOL_{k,\Sigma,\Pi}$ s.t. x_j is free in ϕ iff $j < i$
- A sentence is a formula with no free variables, i.e., an element of $FOL^0_{k,\Sigma,\Pi}$

Exercise

1. Some exercises on definability and expressivity—on the whiteboard (reflexivity? transitivity?)

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1
P	$P(x_0)$	$P(x_1)$

Definition (Standard Translation for ML)

	ST_0	ST_1
P	$P(x_0)$	$P(x_1)$
$\psi \wedge \phi$	$ST_0(\psi) \wedge ST_0(\phi)$	$ST_1(\psi) \wedge ST_1(\phi)$

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1
P	$P(x_0)$	$P(x_1)$
$\psi \wedge \phi$	$ST_0(\psi) \wedge ST_0(\phi)$	$ST_1(\psi) \wedge ST_1(\phi)$
$\neg\psi$	$\neg ST_0(\psi)$	$\neg ST_1(\psi)$

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1
P	$P(x_0)$	$P(x_1)$
$\psi \wedge \phi$	$ST_0(\psi) \wedge ST_0(\phi)$	$ST_1(\psi) \wedge ST_1(\phi)$
$\neg\psi$	$\neg ST_0(\psi)$	$\neg ST_1(\psi)$
$\langle S \rangle \phi$	$\exists x_1.(x_0 S x_1 \wedge ST_1(\phi))$	$\exists x_0.(x_1 S x_0 \wedge ST_0(\phi))$

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1
P	$P(x_0)$	$P(x_1)$
$\psi \wedge \phi$	$ST_0(\psi) \wedge ST_0(\phi)$	$ST_1(\psi) \wedge ST_1(\phi)$
$\neg\psi$	$\neg ST_0(\psi)$	$\neg ST_1(\psi)$
$\langle S \rangle \phi$	$\exists x_1.(x_0 S x_1 \wedge ST_1(\phi))$	$\exists x_0.(x_1 S x_0 \wedge ST_0(\phi))$

[the standard translation]

$$ST(\phi) := ST_0(\phi)$$

Standard translation

Definition (Standard Translation for ML)

	ST_0	ST_1
P	$P(x_0)$	$P(x_1)$
$\psi \wedge \phi$	$ST_0(\psi) \wedge ST_0(\phi)$	$ST_1(\psi) \wedge ST_1(\phi)$
$\neg\psi$	$\neg ST_0(\psi)$	$\neg ST_1(\psi)$
$\langle S \rangle \phi$	$\exists x_1.(x_0 S x_1 \wedge ST_1(\phi))$	$\exists x_0.(x_1 S x_0 \wedge ST_0(\phi))$

[the standard translation]

$$ST(\phi) := ST_0(\phi)$$

[universal closure of ST]

$$ST^u(\phi) := \forall x_0.ST_0(\phi)$$

Exercise

1. Compute the Standard Translation of several example formulas — on the whiteboard

Fact

For any $\phi \in ML_{\Sigma,\Pi}$, $ST(\phi) \in FOL^1_{2,\Sigma,\Pi}$

Lemma

Let $\phi \in ML_{\Sigma,\Pi}$, $\mathfrak{W} \in \text{Str}(\Sigma, \Pi)$. *Then*

- *For any* $w \in W$, $\quad \mathfrak{W}, w \models \phi$ *iff* $w \in [ST(\phi)]^{\mathfrak{W}}$
- $\mathfrak{W} \models \phi$ *iff* $\mathfrak{W} \models ST^u(\phi) = \forall x_0.ST_0(\phi)$

Lemma

Let $\phi \in ML_{\Sigma,\Pi}$, $\mathfrak{W} \in \mathrm{Str}(\Sigma,\Pi)$. *Then*

- *For any* $w \in W$, $\quad \mathfrak{W}, w \vDash \phi$ *iff* $w \in [ST(\phi)]^{\mathfrak{W}}$
- $\mathfrak{W} \vDash \phi$ *iff* $\mathfrak{W} \vDash ST^{u}(\phi) = \forall x_0.ST_0(\phi)$

Corollary

- *Let* $\mathfrak{W} \in \mathrm{Str}(\Sigma,\Pi)$. *Then all MO-definable subsets of* $\underline{\mathfrak{W}}$ *are* $FOL^{1}_{2,\Sigma,\Pi}$*-definable*
- *Modally definable classes of labelled structures are* $FOL^{0}_{2,\Sigma,\Pi}$*-definable*

Definition

Let $\mathfrak{W}, \mathfrak{V} \in \mathsf{Str}(\Sigma, \Pi)$. A relation $Z \subseteq \underline{\mathfrak{W}} \times \underline{\mathfrak{V}}$ is a bisimulation if for any $S \in \Sigma$, $w \in \mathfrak{W}$, $v \in \mathfrak{V}$ if wZv then

1. $wS^{\mathfrak{W}}w'$ implies there is $v' \in \underline{\mathfrak{V}}$ s.t. $vS^{\mathfrak{V}}v'$ and $w'Zv'$ [forth]
2. $vS^{\mathfrak{W}}v'$ implies there is $w' \in \underline{\mathfrak{W}}$ s.t. $wS^{\mathfrak{W}}w'$ and $w'Zv'$ [back]

Exercise

1. Prove that modal formulas are preserved by bisimulations
2. How about FO formulas?

Contrast with the notion of isomorphism

Bisimulations are central

- to automata-based view of modal formulas
- hence, to coalgebraic approach to logic
- to other computer science formalisms such as process algebras . . .

Exercise

1. Prove that finite sibling-ordered trees are bisimiliar iff they are ismorphic
2. Does this result hold for arbitrary trees?

Theorem (Van Benthem Characterization Theorem)

A formula $\phi \in FOL^1_{\Sigma,\Pi}$ is equivalent to $ST(\psi)$ for some $\psi \in ML_{\Sigma,\Pi}$ if it is invariant for bisimulations

- (explain what invariant means!)
- (explain what equivalent means!)

Standard Translation for Unlabelled Structures?

Recall again definition of

Validity for Unlabelled Structures

For $\phi \in ML_{\Sigma,\Pi}$, we define [ML-validity]
$\mathfrak{F} \models \phi$ iff for all $\mathfrak{W} = \langle \mathfrak{F}, \Lambda \rangle$, $\mathfrak{W} \models \phi$

Standard Translation for Unlabelled Structures?

Recall again definition of

Validity for Unlabelled Structures

For $\phi \in ML_{\Sigma,\Pi}$, we define [ML-validity]
$\mathfrak{F} \vDash \phi$ iff for all $\mathfrak{W} = \langle \mathfrak{F}, \Lambda \rangle$, $\mathfrak{W} \vDash \phi$

That would require universal closure of ST to look like this:

$$ST^u(\phi) = \forall P_0 \ldots P_{n-1} \forall x_0 ST_0(\phi)$$

where $P_0, \ldots, P_{n-1}$ are all the Π-labels occurring in ϕ
This is not FO-quantification!

Mutual Incomparability

Properties ML-definable over unlabelled structures which are not FOL-definable:

- Löb
- optionally McKinsey, Grzegorczyk, Van Benthem's cyclic return ...

Mutual Incomparability

Properties ML-definable over unlabelled structures which are not FOL-definable:

- Löb
- optionally McKinsey, Grzegorczyk, Van Benthem's cyclic return . . .

Properties FOL-definable over unlabelled structures which are not ML-definable:

- irreflexivity, antisymmetry . . .

Mutual Incomparability

Properties ML-definable over unlabelled structures which are not FOL-definable:

- Löb
- optionally McKinsey, Grzegorczyk, Van Benthem's cyclic return . . .

Properties FOL-definable over unlabelled structures which are not ML-definable:

- irreflexivity, antisymmetry . . .

Properties which are both ML- and FOL-definable

- reflexivity, transitivity, symmetry
- a relation is the converse of another one . . .

Mutual Incomparability

Properties ML-definable over unlabelled structures which are not FOL-definable:

- Löb
- optionally McKinsey, Grzegorczyk, Van Benthem's cyclic return ...

Properties FOL-definable over unlabelled structures which are not ML-definable:

- irreflexivity, antisymmetry ...

Properties which are both ML- and FOL-definable

- reflexivity, transitivity, symmetry
- a relation is the converse of another one ...

Properties which are neither ML- nor FOL-definable

- finiteness
- transitive closure (but mention VB about combination of transitive closure and Löb!)

MSO: One Logic To Rule Them All!

Definition

- Fix Σ a signature. The set of *$MSO_{\Sigma,\Pi}$-formulas* is defined as follows:

$$\phi ::= x_i{=}x_j \mid S(x_i, x_j) \mid P(x_i) \mid \neg\phi \mid \phi \wedge \phi \mid \forall x_i.\phi \mid \forall P.\phi$$

$$(S \in \Sigma, i, j < \infty)$$

MSO: One Logic To Rule Them All!

Definition

- Fix Σ a signature. The set of *$MSO_{\Sigma,\Pi}$-formulas* is defined as follows:

$$\phi ::= x_i = x_j \mid S(x_i, x_j) \mid P(x_i) \mid \neg\phi \mid \phi \wedge \phi \mid \forall x_i.\phi \mid \forall P.\phi$$

$$(S \in \Sigma, i, j < \infty)$$

- An additional clause in the satisfaction definition

$$\mathfrak{F}, V \vDash \forall P.\phi \quad \text{iff} \quad \forall X \subseteq \underline{\mathfrak{W}}.\forall \mathfrak{W} = \langle \mathfrak{F}, \Lambda \rangle . \; \Lambda(P) = X \text{ implies } \mathfrak{W}, V \vDash \phi$$

Exercise

1. Define transitive closure in MSO

And Still Some More Languages: DRA and TRA

These are languages for pairs of points—i.e., for arcs

And Still Some More Languages: DRA and TRA

These are languages for pairs of points—i.e., for arcs

Definition

- Fix Σ a signature. The set of $DRA_{\Sigma,\Pi}$-formulas is defined as follows:

$$\phi ::= ?P \mid S \mid \cdot \mid \sim\phi \mid \phi \cup \phi \mid \phi/\phi$$

- Fix Σ a signature. The set of $TRA_{\Sigma,\Pi}$-formulas is defined as follows:

$$\phi ::= ?P \mid S \mid \cdot \mid \neg\phi \mid \phi \cup \phi \mid \phi/\phi \mid \phi^{\smile}$$

And Still Some More Languages: DRA and TRA

These are languages for pairs of points—i.e., for arcs

Definition

- Fix Σ a signature. The set of $DRA_{\Sigma,\Pi}$-formulas is defined as follows:

$$\phi ::= ?P \mid S \mid \cdot \mid {\sim}\phi \mid \phi \cup \phi \mid \phi / \phi$$

- Fix Σ a signature. The set of $TRA_{\Sigma,\Pi}$-formulas is defined as follows:

$$\phi ::= ?P \mid S \mid \cdot \mid \neg\phi \mid \phi \cup \phi \mid \phi / \phi \mid \phi^{\smile}$$

Note we have no quantifiers again—like in ML

(as in case of FOL and ML, define other boolean connectives for TRA as abbreviations)

Semantics

Definition (DRA/TRA Satisfaction)

Let $\mathfrak{W} := \langle \mathfrak{F}, \Lambda \rangle \in \mathsf{Str}(\Sigma, \Pi)$

$$
\begin{array}{ll}
\mathfrak{W}, w, v \models ?P & \text{if } w = v \text{ and } w \in \Lambda(P) \\
\mathfrak{W}, w, v \models S & \text{if } wS^{\mathfrak{W}}v \\
\mathfrak{W}, w, v \models \cdot & \text{if } w = v \\
\mathfrak{W}, w, v \models \sim\phi & \text{if } w = v \text{ and } \forall v' \in \underline{\mathfrak{W}}.\mathfrak{W}, w, v' \not\models \phi \\
\mathfrak{W}, w, v \models \neg\phi & \text{if } \mathfrak{W}, w, v \not\models \phi \\
\mathfrak{W}, w, v \models \phi \cup \psi & \text{if } \mathfrak{W}, w, v \models \phi \text{ or } \mathfrak{W}, w, v \models \psi \\
\mathfrak{W}, w, v \models \phi/\psi & \text{if } \exists v' \in \underline{\mathfrak{W}}.\mathfrak{W}, w, v' \models \phi \text{ and } \mathfrak{W}, v', v \models \psi \\
\mathfrak{W}, w, v \models \phi^{\smile} & \text{if } \mathfrak{W}, v, w \models \phi
\end{array}
$$

Exercise

1. Provide a definition of $\sim$ in TRA, deduce it is more expressive than DRA
2. Provide an analogue of Standard Translation for TRA/DRA (i.e., ST^{DRA} and ST^{TRA}) into $FOL^2_{3,\Sigma,\Pi}$
3. Deduce the same consequences as for ML over labelled structures

Theorem (Tarski-Givant)

Over labelled structures, $TRA_{\Sigma,\Pi}$ and $FOL^2_{3,\Sigma,\Pi}$ are equally expressive.

(An accessible proof given by Venema)

Comparing ML and DRA

A (slightly modified) diagram of Johan Van Benthem:

$\underline{\mathfrak{W}}$				$\underline{\mathfrak{W}}^2$
unary properties	$\longrightarrow$	*modes*	$\longrightarrow$	binary relations
of states	$\longleftarrow$	*projections*	$\longleftarrow$	between states
propositional operators				program operators
ML				DRA/TRA

Examples of modes:

$$?X := \{\langle x, x\rangle \mid x \in X\} \qquad \text{(testing)}$$

$$!X := \{\langle w, x\rangle \mid w \in \underline{\mathfrak{W}}, x \in X\} \qquad \text{(realizing)}$$

Examples of projections:

$$\langle R\rangle := \{w \in \underline{\mathfrak{W}} \mid \exists v \in \underline{\mathfrak{W}}. wR^{\mathfrak{W}}v\} \qquad \text{(domain)}$$

$$\pi^{-1}(R) := \{w \in \underline{\mathfrak{W}} \mid \exists v \in \underline{\mathfrak{W}}. vR^{\mathfrak{W}}w\} \qquad \text{(codomain)}$$

$$\sim R := \{w \in \underline{\mathfrak{W}} \mid \forall v \in \underline{\mathfrak{W}}. \neg(wR^{\mathfrak{W}}v)\} \qquad \text{(antidomain)}$$

$$\Delta(R) := \{w \in \underline{\mathfrak{W}} \mid wR^{\mathfrak{W}}w\} \qquad \text{(diagonal)}$$

Examples of modes:

$$?X := \{\langle x, x\rangle \mid x \in X\} \qquad \text{(testing)}$$

$$!X := \{\langle w, x\rangle \mid w \in \underline{\mathfrak{W}}, x \in X\} \qquad \text{(realizing)}$$

Examples of projections:

$$\langle R\rangle := \{w \in \underline{\mathfrak{W}} \mid \exists v \in \underline{\mathfrak{W}}. wR^{\mathfrak{W}}v\} \qquad \text{(domain)}$$

$$\pi^{-1}(R) := \{w \in \underline{\mathfrak{W}} \mid \exists v \in \underline{\mathfrak{W}}. vR^{\mathfrak{W}}w\} \qquad \text{(codomain)}$$

$$\sim R := \{w \in \underline{\mathfrak{W}} \mid \forall v \in \underline{\mathfrak{W}}. \neg(wR^{\mathfrak{W}}v)\} \qquad \text{(antidomain)}$$

$$\Delta(R) := \{w \in \underline{\mathfrak{W}} \mid wR^{\mathfrak{W}}w\} \qquad \text{(diagonal)}$$

NOTE THAT:

$$\begin{aligned} \langle R\rangle &= \sim\sim R \\ &= R/R^{\smile} \cap \cdot \\ \Delta(R) &= R \cap \cdot \\ \pi^{-1}(R) &= \langle R^{\smile}\rangle \end{aligned}$$

From ML to DRA via testing mode

We propose the following translation:

$$
\begin{aligned}
P^{\tau ML \to DRA} &:= ?P \\
\neg\phi^{\tau ML \to DRA} &:= \sim\phi^{\tau ML \to DRA} \\
(\phi \vee \psi)^{\tau ML \to DRA} &:= \phi^{\tau ML \to DRA} \cup \psi^{\tau ML \to DRA} \\
\langle S \rangle \phi^{\tau ML \to DRA} &:= \sim\sim(S/\phi^{\tau ML \to DRA})
\end{aligned}
$$

From ML to DRA via testing mode

We propose the following translation:

$$
\begin{aligned}
P^{\tau ML \to DRA} &:= ?P \\
\neg\phi^{\tau ML \to DRA} &:= \sim\phi^{\tau ML \to DRA} \\
(\phi \vee \psi)^{\tau ML \to DRA} &:= \phi^{\tau ML \to DRA} \cup \psi^{\tau ML \to DRA} \\
\langle S \rangle \phi^{\tau ML \to DRA} &:= \sim\sim(S / \phi^{\tau ML \to DRA})
\end{aligned}
$$

Lemma

For any $\mathfrak{W} \in \mathrm{Str}(\Sigma, \Pi)$, $w \in \underline{\mathfrak{W}}$, $\phi \in ML_{\Sigma,\Pi}$

$$\mathfrak{W}, w \models \phi \quad \textit{iff} \quad \mathfrak{W}, w, w \models \phi^{\tau ML \to DRA}$$

Converse direction

Theorem

For any $\mathfrak{W} \in \mathsf{Str}(\Sigma, \Pi)$ *and any* $X \subseteq W$*, the following are equivalent*

- *there is* $\phi \in DRA_{\Sigma,\Pi}$ *s.t.* $?X = \langle \phi^{\mathfrak{W}} \rangle$
- *there is* $\psi \in ML_{\Sigma,\Pi}$ *s.t.* $X = \psi^{\mathfrak{W}}$

Proof.

Based on the fact that all sets of the form $\langle \phi^{\mathfrak{W}} \rangle$ for $\phi \in DRA_{\Sigma,\Pi}$ can be defined using only formulas in the image of *ML* via $(\cdot)^{T_{ML \to DRA}}$. □

Theorem (Van Benthem Safety Theorem)

A formula $\phi \in FOL^2_{\Sigma,\Pi}$ is equivalent to $ST^{DRA}(\psi)$ for some $\psi \in DRA_{\Sigma,\Pi}$ if it is safe for bisimulations

Safe for bisimulations: if $w, w' \in \underline{\mathfrak{W}}$, $v \in \mathfrak{V}$, wZv and $\mathfrak{W} \models \phi[0/w, 1/w']$, then there is $v' \in \underline{\mathfrak{V}}$ s.t. $\mathfrak{V} \models \phi[0/v, 1/v']$

Exercise

1. Is intersection safe for bisimulations?

On labelled structures:

- *ML* is the bisimulation invariant fragment of *FOL*[1]
- *DRA* is the bisimulation safe fragment of *FOL*[2]

Is being more expressive always better?

Is being more expressive always better?

Axiomatization of equivalence/validity

On arbitrary (possibly infinite) structures:

- *ML* and *DRA* have nice equational axiomatization and are decidable
- *TRA* and *FOL* have no equational axiomatization, but are axiomatizable
  and hence recursively enumerable
- *MSO* is not axiomatizable and not even recursively enumerable

On finite structures . . .

Axiomatization of equivalence/validity

On finite structures:

- *ML* and *DRA* have nice equational axiomatization and are decidable
- *TRA*, *FOL* and *MSO* are not axiomatizable and not even recursively enumerable