Lecture 7

Translation Schemes: Main definitions and examples

- The framework of translation schemes
  - The induced maps
  - The fundamental lemma
  - Reductions
- The Museum of examples

Definition 1 (Translation Schemes Φ)

- Let τ and $\sigma = \{R_1, \ldots, R_m\}$ be two vocabularies with $\rho(R_i)$ be the arity of R_i.
- Let $\mathcal{L}$ be a fragment of SOL, such as FOL, $MSOL$, $\exists MSOL$, etc.
- Let $\Phi = \langle \phi, \psi_1, \ldots, \psi_m \rangle$ be formulae of $\mathcal{L}(\tau)$ such that ϕ has exactly k distinct free first order variables and each ψ_i has $k\rho(R_i)$ distinct free first order variables.
  We say that Φ is k–**feasible (for σ over τ)**.
- A k-feasible $\Phi = \langle \phi, \psi_1, \ldots, \psi_m \rangle$ is called a k–τ–σ–$\mathcal{L}$–**translation scheme** or, in short, a **translation scheme**, if the parameters are clear in the context.

Distinctions

If $k = 1$ we speak of **scalar** or **non–vectorized** translation schemes.

If $k \geq 2$ we speak of **vectorized** translation schemes.

If ϕ is such that $\forall \bar{x} \phi(\bar{x})$ is a tautology (always true) the translation scheme is **not relativized** otherwise it is **relativized**.

A translation scheme is **simple** if it is neither relativized nor vectorized.

Example 2 (τ_{words_3} and τ_{graphs})

τ_{words_3} consists of $\{R_<, P_0, P_1, P_2\}$ for three letters $\{0, 1, 2\}$.

τ_{graphs} consists of $\{E\}$

Put $k = 1$,
$\phi_1(x) = (P_0(x) \vee P_1(x))$ and
$\psi_E(x, y) = (P_0(x) \wedge P_1(y))$

$$\Phi_1 = \langle \phi_1(x), \psi_E(x, y) \rangle$$

is a **scalar** and **relativized** translation scheme in FOL.

If instead we look at $\phi_2(x) = (x \approx x)$ then

$$\Phi_2 = \langle \phi_2(x), \psi_E(x, y) \rangle$$

is a **simple** translation scheme.

Example 3 (τ_{words_2} and τ_{grids})

τ_{words_2} consists of $\{R_<, P_0, P_1\}$

τ_{grids} consists of $\{E_{NS}, E_{EW}\}$

Put $k = 2$,
$\phi(x) = ((x \approx x) \wedge (y \approx y))$
$\psi_{E_{NS}}(x_1, x_2, y_1, y_2) = (R_<(x_1, x_2) \wedge y_1 \approx y_2)$
$\psi_{E_{ES}}(x_1, x_2, y_1, y_2) = (R_<(y_1, y_2) \wedge x_1 \approx x_2)$

$$\Phi_3 = \langle \phi(x, y), \psi_{E_{NS}}(x_1, x_2, y_1, y_2), \psi_{E_{EW}}(x_1, x_2, y_1, y_2) \rangle$$

is a **vectorized** but **not** relativized translation scheme in FOL.

Definition 4 (The induced transduction $\Phi^\star$)

Given a translation scheme Φ

$$\Phi^\star : Str(\tau) \to Str(\sigma)$$

is a (partial) function from τ–structures to σ–structures defined by $\Phi^\star(\mathcal{A}) = \mathcal{A}_\Phi$ and

1. the universe of $\mathcal{A}_\Phi$ is the set $A_\Phi = \{\bar{a} \in A^k : \mathcal{A} \models \phi(\bar{a})\}$;

2. the interpretation of R_i in $\mathcal{A}_\Phi$ is the set
$$\mathcal{A}_\Phi(R_i) = \{\bar{a} \in {A_\Phi}^{\rho(R_i) \cdot k} : \mathcal{A} \models \psi_i(\bar{a})\}.$$
$\mathcal{A}_\Phi$ is a σ–structure of cardinality at most $\mid A \mid^k$.

As Φ is k–feasible for σ over τ, $\Phi^\star(\mathcal{A})$ is defined iff $\mathcal{A} \models \exists \bar{x} \phi$.

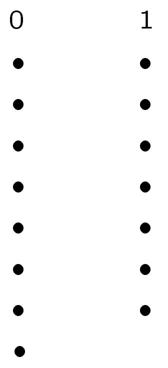
Example 5 (Words and graphs)

Let is compute $\Phi_1^\star$.

For the word

1001020102001022111

we get the graph



(1)

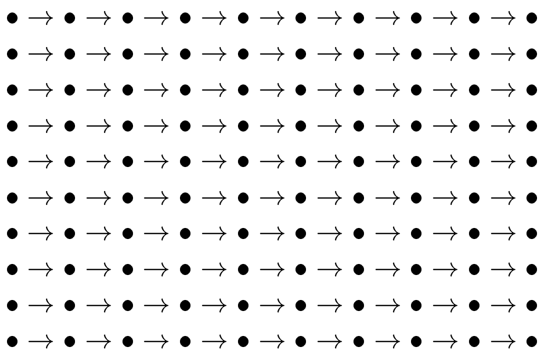
Example 6 (Words and grids)

Let is compute $\Phi_3^\star$.

For a word

0110101001

we get



This is independent of the letters ¶$0, 1\}$.

Definition 7 (The induced translation $\Phi^\sharp$)

Given a translation scheme Φ we define a function $\Phi^\sharp : \mathcal{L}(\sigma) \to \mathcal{L}(\tau)$ from $\mathcal{L}(\sigma)$–formulae to $\mathcal{L}(\tau)$–formulae inductively as follows:

- For $R_i \in \sigma$ and $\theta = R_i(x_1, \ldots, x_m)$ let $x_{j,h}$ be new variables with $i \leq m$ and $h \leq k$ and denote by $\bar{x}_i = \langle x_{i,1}, \ldots, x_{i,k} \rangle$. We put
$$\Phi^\sharp(\theta) = \left(\psi_i(\bar{x}_1, \ldots, \bar{x}_m) \wedge \bigwedge_i \phi(\bar{x}_i)\right)$$
- This also works for equality and relation variables U instead of relation symbols R.

Definition 7 (Continued: booleans)

For the boolean connectives, the translation distributes, i.e.

- if $\theta = (\theta_1 \vee \theta_2)$ then
$$\Phi_\sharp(\theta) = (\Phi_\sharp(\theta_1) \vee \Phi_\sharp(\theta_2))$$
- if $\theta = \neg\theta_1$ then
$$\Phi_\sharp(\theta) = \Phi_\sharp(\neg\theta_1)$$
- similarly for $\wedge$ and $\to$.

Definition 7 (Continued: quantification)

- For the existential quantifier, we use relativization to ϕ:
If $\theta = \exists y \theta_1$, let $\bar{y} = \langle y_1, \ldots, y_k \rangle$ be new variables. We put
$$\theta_\Phi = \exists \bar{y}(\phi(\bar{y}) \wedge (\theta_1)_\Phi).$$
This concludes the inductive definition for first order logic FOL.
- For second order quantification of variables U of arity ℓ and $\bar{a}$ a vector of length ℓ of first order variables or constants, we translate $U(\bar{a})$ by treating U as a relation symbol above and put
$$\theta_\Phi = \exists V(\forall \bar{v}(V(\bar{v}) \to (\phi(\bar{v_1}) \wedge \ldots \phi(\bar{v}_\ell) \wedge (\theta_1)_\Phi)))$$

Example 8 (Computing $\Phi_1^\sharp$)

Recall
$$\Phi_1 = \langle \phi_1(x), \psi_E(x,y) \rangle$$
with $k = 1$,
$\phi_1(x) = (P_0(x) \vee P_1(x))$ and
$\psi_E(x,y) = (P_0(x) \wedge P_1(y))$

Let θ_{conn} be the formula which says the graph is connected:

$$\neg\left(\exists U \left(\exists x \neg U(x) \wedge \forall x \forall y (U(x) \wedge E(x,y) \to U(y))\right)\right)$$

Example 8 (Continued)

- $U(x)$ is replaced by
$$(\phi_1(x) \wedge U(x)) = ((P_0(x) \vee P_1(x)) \wedge U(x))$$
- $E(x, y)$ is replaced by
$$(\phi_1(x) \wedge \phi_1(y) \wedge E(x, y)) =$$
$$((P_0(x) \vee P_1(x)) \wedge (P_0(y) \vee P_1(y)) \wedge E(x, y))$$
- $(x \approx y)$ is replaced by
$$(\phi_1(x) \wedge \phi_1(y) \wedge (x \approx y)) =$$
$$((P_0(x) \vee P_1(x)) \wedge (P_0(y) \vee P_1(y)) \wedge (x \approx y))$$
- Then we proceed inductively.

$(x \approx y)$ does not occur in θ_{conn}.

Proposition 9
(Preservation of tautologies I)

Let $\mathcal{L}$ be First Order Logic FOL.

$$\Phi = \langle \phi, \psi_1, \ldots, \psi_m \rangle$$

be a k–$(\tau - \sigma)$—$\mathcal{L}$–translation scheme, which is not relativizing, i.e. $\forall \bar{x} \phi(\bar{x})$ is a tautology. Let θ a σ-formula.

- If θ is a tautology (not satisfiable), so is $\Phi^\sharp(\theta)$.
- If ϕ is not a tautology, this is not true.
- There are formulas θ which are not tautologies (are satsifiable), such that $\Phi^\sharp(\theta)$ is a tautology (is not satisfiable).

Proof of proposition 9

Proof:
For FOL, the first two parts are by straight induction using the completeness theorem. What we observe is that proof sequences translate properly using $\Phi^\sharp$.

Generalizing to other logics needs regularity conditions.

If ϕ is not a tautology, $\exists x(x = x)$ is a tautology, but $\Phi^\sharp(\exists x(x = x)) = \exists x \phi(x) \wedge x = x$ is not a tautology.

Now let $\Phi = \langle \psi_R, \psi_S \rangle$ be defined by

$$\psi_R(x) = P(x) \text{ and } \psi_S(x) = \neg P(x).$$

$\exists x \theta_1$ be $R(x) \wedge S(x)$ and $\exists x \theta_2$ be $R(x) \vee S(x)$ are both satisfiable but not tautolgies. But $\Phi^\sharp(\theta_1)$ is not satisfiable and $\Phi^\sharp(\theta_2)$ is a tautology. $Q.E.D.$

Theorem 10 (Fundamental Property)

Let $\Phi = \langle \phi, \psi_1, \ldots, \psi_m \rangle$ be a k–$(\tau - \sigma)$–translation scheme in a logic $\mathcal{L}$. Then the transduction $\Phi^\star$ and the translation $\Phi^\sharp$ are in linked in $\mathcal{L}$.

In other words, given

- $\mathcal{A}$ be a τ-structure and
- θ be a $\mathcal{L}(\sigma)$–formula.

Then

$$\boxed{\mathcal{A} \models \Phi^\sharp(\theta) \text{ iff } \Phi^\star(\mathcal{A}) \models \theta}$$

Translation Scheme and its induced maps

in the Fundamental Property of theorem 10

Translation scheme Φ		
	$\Phi^\star$	
τ-structure	$\longrightarrow$	σ-structure
$\mathcal{A}$		$\Phi^\star(\mathcal{A})$
τ-formulae	$\longleftarrow$	σ-formulae
	$\Phi^\sharp$	
$\Phi^\sharp(\theta)$		θ
$\mathcal{A} \models \Phi^\sharp(\theta)$	iff	$\Phi^\star(\mathcal{A}) \models \theta$

Definition 11 ($\mathcal{L}$–Reductions)

Let $\mathcal{L}$ be a regular logic and Φ be a $(\tau_1 - \tau_2)$ translation scheme. We are given

- two classes K_1, K_2 of $\tau_1(\tau_2)$-structures closed under isomorphism

We say

1. $\Phi^\star$ is a *weak reduction* of K_1 to K_2 if for every τ_1-structure $\mathfrak{A}$ with $\mathfrak{A} \in K_1$ we have $\Phi^\star(\mathfrak{A}) \in K_2$.

2. $\Phi^\star$ is a *reduction* of K_1 to K_2 if for every τ_1-structure $\mathfrak{A}$, $\mathfrak{A} \in K_1$ iff $\Phi^\star(\mathfrak{A}) \in K_2$.

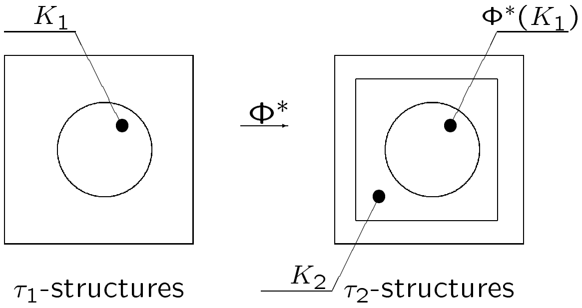
Definition 11(Continued)

---

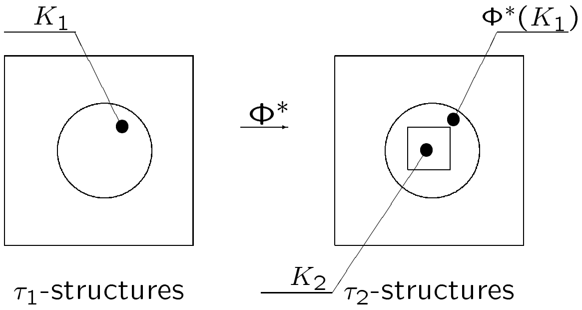
3. $\Phi^\star$ of K_1 to K_2 is *onto* if (additionally) for every $\mathfrak{B} \in K_2$ there is an $\mathfrak{A} \in K_1$ with $\Phi^\star(\mathfrak{A})$ isomorphic to $\mathfrak{B}$.

4. By abuse of language we say $\Phi^\star$ is a *translation of* K_1 *onto* K_2 also if $\Phi^\star$ is not a weak reduction but only $K_2 \subseteq \Phi^\star(K_1)$.

5. We say that Φ induces a reduction (a weak reduction) of K_1 to K_2, if $\Phi^\star$ is a reduction (a weak reduction) of K_1 to K_2. For simplicity, we also say Φ is a reduction (a weak reduction) instead of saying that Φ induces a reduction (a weak reduction).



Weak reduction



ONTO

Definition 12 ($\mathcal{L}$–Reducibility)

1. Let $k \in \mathbf{N}$.
   We say that K_1 is *$\mathcal{L}$-k-reducible* to K_2 ($K_1 \lhd_{\mathcal{L}-k} K_2$), if there is a $\mathcal{L}$-k-translation scheme Φ for τ_2 over τ_1, such that $\Phi^\star$ is a reduction of K_1 to K_2.

2. We say that K_1 is *$\mathcal{L}$-reducible* to K_2 ($K_1 \lhd_{\mathcal{L}} K_2$), if $K_1 \lhd_{\mathcal{L}-k} K_2$ for some $k \in \mathbf{N}$.

3. We say that K_1 is *$\mathcal{L}$-bi-reducible* to K_2 and write $K_1 \bowtie_{\mathcal{L}} K_2$, if $K_1 \lhd_{\mathcal{L}-k} K_2$ and $K_2 \lhd_{\mathcal{L}-k} K_1$ for some $k \in \mathbf{N}$.
   Clearly, bi-reducibility is a symmetric relation.

Theorem 13 (Definability and Reducibility)

Let $\Phi^\star$ be an $\mathcal{L}$-reduction of K_1 to K_2.
If K_2 is $\mathcal{L}$-definable then K_1 is -definable.

Recall that a class of τ-structures K_2 is $\mathcal{L}$-definable if there is a $\mathcal{L}(\tau)$-sentence θ such that $K_2 = Mod(\theta)$.

Proof:

We use the Fundamental Property of Φ.

If K_2 is defined by θ, so K_1 is defined by $\Phi^\sharp(\theta)$.

Proposition 14

Hamiltonian graphs are not $MSOL$-definable (both in τ_{graphs_1} and τ_{graphs_2}).

Proof:

We use Φ_2 from example 2.

$\Phi_2^\star$ is a reduction from words 0^n1^m over $\{0,1\}$ to complete bipartite graphs $K_{n,m}$, which are $MSOL$-defined by θ_{co-bi}.

$K_{n,m}$ is Hamiltonian iff $n = m$.

So, if θ_{hamil} defined all Hamiltonian graphs,
$$\Phi_2^\sharp(\theta_{hamil} \land \theta_{co-bi})$$
defined the language $\{0^n1^n\}$.

But $\{0^n1^n\}$ is not regular, and hence, by Büchi's theorem, not $MSOL$-definable.

Q.E.D.

Proposition 15

Eulerian graphs are not $MSOL$-definable (both in τ_{graphs_1} and τ_{graphs_2}).

Proof: Let SET be the class of finite sets and $ODD \subseteq SET$ those of odd cardinality.
Let $CLIQUE$ be the class of complete graphs.
$CLIQUE$ is FOL-definable by some θ_{clique}.

Let the simple FOL translation scheme Φ be given by
$\phi(x) = (x \approx x)$ and $\psi_E(x,y) = (\neg x \approx y)$.

$\Phi^\star$ is a reduction from SET to $CLIQUE$.

Now assume that there is $\theta_{euler} \in MSOL$, with $EULER = Mod(\theta_{euler})$.
Put $\theta = (\theta_{clique} \land \theta_{euler})$.
$\Phi^\sharp(\theta)$ is equivalent to $\theta_{odd} \in MSOL$.

But this contradicts the fact that ODD ($EVEB$) is not $MSOL$-definable.

Q.E.D.

Proof of theorem 10

We use induction over the construction of θ.

- If all the formulas ϕ, ψ_i of Φ and θ are atomic, both $\Phi^\star(\mathfrak{A}) = \mathfrak{A}$ and $\Phi^\sharp(\theta) = \theta$.

- Next we keep θ atomic and assume
$$\Phi = \langle \phi(\bar{x}), \psi_{S_1}(\bar{x}), \ldots \psi_{S_m}(\bar{x}) \rangle$$
$\Phi^\star(\mathfrak{A}) \models S_i(\bar{a})$ iff $\mathfrak{A} \models \psi_{S_i}(\bar{a})$ by definition of $\Phi^\star$.

- Now the induction on θ uses that $\Phi^\sharp$ commutes with the logical constructs.

Q.E.D.

Proposition 16 (Preservation of tautologies II)

Let $\mathcal{L}$ be First Order Logic FOL.
$$\Phi = \langle \phi, \psi_1, \ldots, \psi_m \rangle$$
be a k–$(\tau - \sigma)$—$\mathcal{L}$–translation scheme. Let θ a σ-formula.

Assume that $\Phi^\star$ is onto all σ-structures, i.e. for every σ-structure $\mathfrak{B}$ there is a τ-structure $\mathfrak{A}$ such that $\Phi^\star(\mathfrak{A}) = cong\mathfrak{B}$

- If θ is a tautology, so is $\Phi^\sharp(\theta)$.

- If additionally $\exists \bar{x} \phi(\bar{x})$ is a tautology and $\Phi\sharp(\theta)$ is a tautology then θ is a tautology.

Proof:
Use the fundamental property. $Q.E.D.$
Note that here the proof is semantical.

Example 17 (Renaming)

One of the simplest translations encountered in logic is the renaming of basic relations.

Let $\tau_1 = \{R_i : i \leq k\}$ and $\tau_2 = \{S_i : i \leq k\}$, where R_i and S_i are of the same arity, respectively.

Let Φ be the (τ_1, τ_2) translation scheme given by $\Phi = \langle x = x, R_1(\bar{u}), \ldots, R_k(\bar{v}) \rangle$.

Such a translation scheme and as well as its induced maps $\Phi^\star$ and $\Phi^\sharp$ are called **renaming**.

Example 18 (Cartesian Product)

Let us consider one example of vectorized translation scheme that defines Cartesian Product.

For simplicity, we assume that $k = 2$.

Let $\tau_1 = \{R_1(x_1, x_2)\}$ with R_1 binary and $\tau_2 = \{R_2(x_1, x_2)\}$ with R_2 binary.

$$\Phi = \langle (x_1 = x_1 \vee x_2 = x_2), \\ (R_1(x_1, x_2) \wedge R_2(x_3, x_4)) \rangle$$

It is easy to see that $\Phi^\star(\mathcal{A})$ is isomorphic to the Cartesian product $\mathcal{A}^2$.

The n-hold Cartesian product is defined in the same way.

Example 19 (Graphs)

$Graphs_1$ is the class of structures of the form $\langle V, E\rangle$ where E is a binary irreflexiv relation on the set of vertices V.

$Graphs_2$ is the class of structures of the form $\langle V \sqcup E; Src(v,e), Tgt(v,e)\rangle$ with the universe consisting of **disjoint** sets of vertices and edges and $Src(v,e)$ ($Tgt(v,e)$) indicates that v is the source (target) of the directed edge e.

For a graph G we denote its representations by G_i for $G_i \in Graphs_i$ respectively.

We define a scalar translation scheme $\Phi = \langle \phi, \psi_E \rangle$ from $Graphs_2$ to $Graphs_1$ by

$$\phi(v) = (\exists e(Src(v,e) \lor eTgt(v,e))\lor$$
$$(v = v \land \neg \exists x(Src(x,v) \lor Tgt(x,v))$$
$$\phi_E(x,y) = \exists e\,((Src(x,e) \land Tgt(y,e))$$

Clearly, for every graph G we have

$$\Phi^\star(G_2) \cong G_1$$

Theorem 20 (Complexity of transductions)

If Φ is in FOL (or $\exists HornSOL$) then $\Phi^\star$ is computable in polynomial time.

Proof:

We test all k-tuples $\bar{a}$ in $\mathfrak{A}$ of size n for

$$\mathfrak{A} \models \phi(\bar{a})$$

This takes $n^k \cdot TIME(\mathfrak{A}, \phi)$ time.

But we know that $TIME(\mathfrak{A}, \phi)$ is a polynomial in n.

For the ψ_{S_i} this is the same.

Q.E.D.

By a theorem of Grädel, this also holds for $HornSOL$, cf. the project page.