1. Conservative and Expansive Theory Extensions

Notation. Given a theory inclusion $T \subseteq T'$, we call $T$ a subtheory of $T'$, and $T'$ an extension of $T$. We will often speak of a subtheory $T \subseteq T'$ (meaning $T$), and an extension $T' \supsetneq T$ (meaning $T'$).

Def. $T' \supsetneq T$ is a conservative extension of $T$ iff

$$\text{th}(T') \cap \text{FO}(\text{sig}(T)) = \text{th}(T).$$

For syntactic theories, $((\Sigma, \Pi), T) \subseteq ((\Sigma', \Pi'), T')$ it's boil down to what one would expect:

$$\forall \varphi \in \text{FO}(\Sigma, \Pi) \quad T \vdash \varphi \iff T' \vdash \varphi.$$

Notation. Given $((\Sigma, \Pi), T) \subseteq ((\Sigma', \Pi'), T')$ and $M' \in \text{Mod}(\Sigma, \Pi)$ we call $M'\upharpoonright (\Sigma, \Pi)$ the $(\Sigma, \Pi)$-reduct of $M'$.

Also, given $M \in \text{Mod}(\Sigma, \Pi)$ we call $M' \in \text{Mod}(\Sigma', \Pi')$ a $(\Sigma', \Pi')$-expansion of $M$ iff $M'\upharpoonright (\Sigma, \Pi) = M$.

This suggest yet another semantic theory operation associated to $((\Sigma, \Pi), T) \subseteq ((\Sigma', \Pi'), T')$ and the function

$$(\Sigma', \Pi') \to \text{Mod}(\Sigma', \Pi')$$

This function $A \mapsto A\upharpoonright (\Sigma, \Pi)$ can be viewed as performing the operation of restricting the theory to the subtheory $(\Sigma, \Pi)$.
where \( A^{\uparrow (\Sigma',\Pi')} = \{ M' \in \text{Mod}(\Sigma',\Pi') \mid M'\mid_{(\Sigma,\Pi)} \in A \} \).

This induces a semantic theory operation \( ((\Sigma,\Pi), A) \mapsto ((\Sigma',\Pi), A^{\uparrow (\Sigma',\Pi')}) \).

**Exercise.** Prove that if \( ((\Sigma,\Pi), A) \cong ((\Sigma,\Pi), \Gamma') \), and \( (\Sigma,\Pi) \cong (\Sigma',\Pi') \),

\( ((\Sigma,\Pi), A^{\uparrow (\Sigma',\Pi')}) \cong ((\Sigma',\Pi'), \Gamma') \).

**Definition.** Call a theory extension \( \Gamma' \supseteq \Gamma \) expandable if

\[
\text{mod}(\Gamma')\mid_{\text{sig}(\Gamma)} = \text{mod}(\Gamma). \;
\text{Call} \; ((\Sigma,\Pi), B) \supseteq ((\Sigma,\Pi), A) \text{ literally expansive if} \;
\text{expansion} \iff (B\mid_{(\Sigma,\Pi)}) = A.
\]

**Examples**

1. \( ((\Sigma,\Pi), \Gamma) \cong ((\Sigma(\gamma),\Pi), \Gamma) \)

   when we have added to \( \Sigma \) a set \( Y \) of new fresh constants in expansive. Indeed, for \( M \in \text{Mod}(\Sigma,\Pi,\Gamma) \)

   \[
   \{ M^{\uparrow (\Sigma',\Pi')} \} = \{ (M, a) \mid a \in \text{Fun}(\Gamma) \}
   \]

   so that,

   \[
   \text{mod}(\Gamma')\mid_{\text{sig}(\Gamma)} = \text{mod}(\Gamma). \;
   \text{Note also, in fact, arguments about satisfiability take place in} \; \text{mod}(\Sigma(\gamma),\Pi,\Gamma) \text{ for}
   \]

   \( Y = \text{Fun}(\Gamma) \).

2. \( \{ +, \ast \}_{AC} \subseteq \{ +, \ast \}_{AC} \) is expansive,

   because for any \( \Theta = (A, \Theta_{\ast}) \), the algebra \( \Theta^{\uparrow} = (A, \Theta_{\ast}, \Theta'_{\ast}) \)

   where \( \Theta'_{\ast} = \Theta_{\ast} \) is such that \( \Theta'\mid_{\{ +, \ast \}} = \Theta \).

3. \( \{ \ast, 1 \}, \text{MON} \} \subseteq \{ \ast, 1 \}, \text{MON} \cup \{ (x,y) \in \text{Fun} \mid x \ast y = 1 \} \) with \( \text{MON} \) the theory of monoids in not an expansive extension. For example,

   \( 2 = (2, \text{or}, 0) \) with \( 2 = \{ 0, 1 \}, \ast = \text{or}, 1 = 0 \) has no expansion to a group.
Suppose \( (\Sigma, \Pi) \) and \( (\Sigma \cup \Delta, \Pi \cup \Pi_\Delta) \) are order-sorted signatures with same sort \((S, \prec)\) of sorts and \(\Delta\) and \(\Pi_\Delta\) declare new function and predicate symbols not appearing in \(\Sigma\), resp. \(\Pi\).

Assume, further, for simplicity, that each function symbol \(f\) has a type \(f : s_1 \ldots s_n \to s\), so that for any other typing \(f : s'_1 \ldots s'_n \to s'\) we have \(s \preceq s'_1, \ldots, s \preceq s'_n\), and each predicate symbol in \(\Pi_\Delta\) has likewise a top typing \(p : s_1 \ldots s_n\) so that any other typing \(p : s'_1 \ldots s'_n\) in such that \(s'_1 \preceq s_1, \ldots, s'_n \preceq s_n\). Call \(T'\) the inclusion \(T' = ((\Sigma, \Pi, \Pi_\Delta), T) \subseteq ((\Sigma \cup \Delta, \Pi \cup \Pi_\Delta), T \cup \text{del}(\Delta) \cup \text{del}(\Pi_\Delta))\) a 

\[\text{definitional extension of } T \text{ if:}\]

\[\text{(i) } \text{def}(\Delta) \text{ has an equation } f(x_1 : s_1, \ldots, x_n : s_n) = t_f(x_1 : s_1, \ldots, x_n : s_n) \text{ where } f : s_1 \ldots s_n \to s \text{ is the top typing, and for any typing } f : s'_1 \ldots s'_n \to s', \quad f(x_1 : s'_1, \ldots, x_n : s'_n) \in T_\Sigma(X)_s'.\]

\[\text{(ii) } \text{def}(\Pi_\Delta) \text{ has equivalences } p(x_1 : s_1, \ldots, x_n : s_n) \iff p_{\text{def}}, \text{ where for each } p \in \Pi_\Delta, \text{ where } p : s_1 \ldots s_n \text{ is the top typing.}\]

Then \(T \subseteq T'\) is expansive. Indeed, for each \(M \in \text{mod}(T')\) there is a unique expansion \((M, T')\), where for each \(f : s_1 \ldots s_n \to s\) in \(\Delta\) and \(a \in [x_1 : s_1, \ldots, x_n : s_n] \to M\)

\[f_{\text{def}}(a(x_1), \ldots, a(x_n)) = t_f a\]

and \((a_1(x_1), \ldots, a_n(x_n)) \in p_{\text{def}} \iff M, a \models p\). Thus \(T \subseteq T'\) is expansive.
4. For \((\Sigma', E') \models (\Sigma, E)\) a \underline{proving} extension of \underline{equational theory}, \underline{Init}(\Sigma', E') \supseteq \underline{Init}(\Sigma, E)\) is \underline{literally} expansive.

In the \underline{Vas} paper we have \underline{systematically} exploited such \underline{literally} expansive extensions
\[
(\Sigma, \{ \Sigma_{\text{EUV}_n} \}) \supseteq (\Sigma, \{ \Sigma_{\text{EUV}_n} \})
\]

5. For \((\{X^3\}, \emptyset) \subseteq (\Sigma, \Gamma)\), where \((\Sigma, \Gamma)\) are, for example, \underline{the theory of lists, countable lists, multisets, sets, and HF-sets parameterised by} the \underline{sort} \(X\), the \underline{extension}
\[
(\Sigma, \bigcup_{\pi \in \Pi} \underline{IF}(\underline{NoSet})) \supseteq (\{X^3, \underline{NoSet}\})
\]

are \underline{literally} expansive extensions, since for each \underline{non-empty} \underline{set} \(A\) we have the \underline{parameter-preservation property}:
\[
\underline{IF}(A) \upharpoonright_{\{X^3\}} \equiv A
\]

For example, \(\underline{List}(\underline{NN})\upharpoonright_{\{X^3\}} \equiv \underline{NN}\).

**Lemma.** If \(T' \supset T\) is \underline{expansive} or \underline{literally expansive}, then \(T\) is a \underline{conservative} extension of \(T\).

**Proof.** In both cases we have \(\theta_{(\Sigma, \Pi)}(\underline{mod}(T')|\underline{sig}(T)) = \theta_{(\underline{mod}(T')|\underline{sig}(T))}
\]
but \(\theta_{(\Sigma, \Pi)}(\underline{mod}(T')|\underline{sig}(T)) = \theta(T') \cap \underline{FO}(\underline{sig}(T))\). \(\square\)
The converse of this lemma is not true: there are conservative extensions that are not expansive. Furthermore, there are literally expansive extensions that are not expansive, and expansive extensions that are not literally expansive.

Most of these points can be illustrated by the following exercise:

**Exercise.** Let $\mathbb{R}$ denote the set of real numbers (no operations).

Let $\Sigma_{\mathbb{R}}$ be the signature with a single sort $\mathbb{R}$,

$\Sigma_{\mathbb{R},0} = \mathbb{R},$ (constants), and $\Sigma_{\mathbb{R},n} = \emptyset,$ $n > 0$.

Consider the literally expansive extension

$T' = (\Sigma_{\mathbb{R}}, \{ (\mathbb{R}, + \mathbb{R}) \}) \supset (\{ \text{Elt} \}, \{ \mathbb{R} \}) = T$

where $+_{\mathbb{R}} : \mathbb{R} \rightarrow \mathbb{R}$ in the interpretation of each constant $a \in \Sigma_{\mathbb{R},0}$ by itself. Prove that the, obviously conservative, extension (by alone lemma)

$(\Sigma_{\mathbb{R}}, \text{th}(T')) \supset (\{ \text{Elt} \}, \text{th}(T))$

is not expansive.

**Hint.** Use the upward and downward Löwenheim–Skolem theorems in their sharp formulation in, e.g., Wilfrid Hodges, "A Shorter Model Theory", Cambridge UP, (as Corollary 3.1.4, and Corollary 5.1.4 there).
Signature Maps and Theory Interpretations

In SMT solving, extensibility is a crucial goal: we want the methods, tools, and algorithms to apply as widely as possible to as many applications as possible. For this:

1. theory-generic satisfiability decision procedures such as those in the VanLei paper, and

2. ways of shifting our ground, so that a satisfiability procedure for a theory $T$ can be used to decide the satisfiability of a different theory $T'$.

are both crucial. As an example of (2), in the VanLei paper we have already encountered the method of descent maps. But, more generally, how can we "shift our ground"?

One possible answer is: by theory interpretations $\text{H: } T \rightarrow T'$, which can often translate (as opposed to just include, as in the case of a theory inclusion $T \subseteq T'$) the language of $T$ into that of $T'$.

But what is a theory interpretation? These are increasingly more general notions of "theory interpretation," so a single answer cannot cover them all. For many users a good, quite general notion is the notion of a view $\text{H}$ from a theory $T$ to a theory $T'$,
which indeed defines a theory interpretation $H : T \rightarrow T'$.

But underlying the notion of theory interpretation we have the simpler, more basic notion of a signature map. So, not to begin the house by the roof, I will first explain signature maps. I will focus on a notion somewhat less general of signature map than that used in Maude views, and will leave the extension to that more general notion as a non-trivial exercise.

I first need some detailed notation. To simplify life (and notation) I will replace the notation $(\Sigma, T)$ for an ordered-sorted first-order theory with function symbols $\Sigma$ and predicate symbols $T$ by the simpler notation

$$\Sigma = (\langle S, \preceq \rangle, F, P)$$

where:

1. $\langle S, \preceq \rangle$ is the ground sort

2. $F$ is a mapping

$$F : \text{fun}(\Sigma) \rightarrow \mathcal{P}(S \times S)$$

from a set of function symbols $\text{fun}(\Sigma)$ to a set of ranks for such symbols such that for each $f \in \text{fun}(\Sigma)$,

- 2.1 $F(f) \neq \emptyset$

2.4 (subsort polymorphism) if $(w, s), (w', s') \in F(f)$
then, (i) $|w| = |w'|$, and (ii) $w \equiv s \text{ and } s \equiv s'$.

Ranks such as, say, $(s_1, \ldots, s_n, s) \in F(f)$ are usually denoted: $f : s_1 \ldots s_n \to s$. Strictly polymorphic means that if $f : s_1 \ldots s_n \to s$ and $f : s'_1 \ldots s'_n \to s'$ are typing for $f$, then $[s] = [s']$, and $[s] = [s']$, where $[s]$ denotes the connected component of $s$ in the poset $(S, \leq)$, that is, the equivalence class of $s$ under the equivalence relation $\equiv \leq \equiv = (\leq u \geq)^+$. For example, we may have $\text{Nat} < \text{Int} < \text{Rat}$, and $F(+) = \{(\text{Nat}, \text{Nat}, \text{Nat}), (\text{Int}, \text{Int}, \text{Int}), (\text{Rat}, \text{Rat}, \text{Rat})\}$. This notion is slightly restrictive, but not much. It excludes:

(A) ad hoc polymorphism, as in $\to\to : \text{Nat} \to \text{Nat}$ for addition, and $\to\to : \text{Bool} \to \text{Bool}$ for exclusive or, but such ad hoc polymorphism can always be renamed away.

(B) using the same symbol $f$ with different numbers of arguments, as $f : s s' \to s''$, and $f : s s' s'' \to s''$, but we can always distinguish them either by renaming, or by marking argument positions, as subrubs: $f(-,-)$ and $f(-,-,-)$. 
3. \( P \) is a mapping

\[ P : \text{pred}(\Sigma) \to \mathcal{P}(\Sigma^*) \]

from a set of predicate symbols (here denoted \( \text{pred}(\Sigma) \)) to a set of arities for such symbols such that for each \( p \in \text{pred}(\Sigma) \)

3.1 \( P(p) \neq \emptyset \)

3.2 (Subset polymorphism) if \( w, w' \in P(p) \)

then \( |w| = |w'| \) and \( w \preceq w' \). If \( w \in P(p) \) we write \( p : w \).

Recall from van Dantzig the notion of a \( \Sigma \)-model : \( M \). It is: (a) an \underline{order-sorted algebra} for the signature \( ((\Sigma, \preceq), F) \), and (b) together with for each \( p : w \) a subset \( P^w_m \subseteq M^w \), where \( M^E = \{ 0, 1 \} \), and \( M^1 \times \cdots \times M^m = M^1_1 \times \cdots \times M^m_m \), such that if \( p : w \) and \( p : w' \), and \( (a_1, \ldots, a_n) \in M^w \cap M^{w'} \), then \( (a_1, \ldots, a_n) \in P^w_m \iff (a_1, \ldots, a_n) \in P^{w'}_m \).

So, altogether \( M = (M, \preceq_m, \rightarrow_m) \), where

\[ (f : s_1, \ldots, s_n \to s)_m = f : M^1_1 \times \cdots \times M^m_m \to M^s_s, \] and

\[ (p : w)_m = P^w_m \subseteq M^w \]

(given the subset polymorphism we can omit the \( w \) in \( P^w_m \)).
Definition. Given ordered signatures $\Sigma = (S, \leq, F, P)$ and $\Sigma' = (S', \leq', F', P')$, a signature map $H : \Sigma \to \Sigma'$ is specified by:

1. a monotonic function $H : (S, \leq) \to (S', \leq')$

2. a function $H : \text{fun}(\Sigma) \to \text{fun}(\Sigma')$ such that:
   
   (i) if $f : s_1 \ldots s_n \to t$ in $\Sigma$, and $(w, s) \in F'(H(f))$
   then $H(s_1) \ldots H(s_n) \equiv_{\leq'} w'$ and $H(t) \equiv_{\leq'} s'$.

   (ii) For each $f : s_1 \ldots s_n \to t$ in $\Sigma',
   H(f)(x_1 : H(s_1), \ldots, x_n : H(s_n)) \in \bigcap_{\Sigma'} (X)_{H(t)}$

   Note that condition (ii) is more flexible than just requiring that $H(f) : H(s_1) \ldots H(s_n) \to H(t)$ is rank for $H(f)$ in $F'$.

3. a function $H : \text{pred}(\Sigma) \to \text{pred}(\Sigma')$ such that:
   
   (i) if $p : w$ in $\Sigma$ and $H(p) : w'$ in $\Sigma'$,
   then $|w| = |w'|$ and $H(w) \equiv_{\leq'} w'$, where
   $H(e) = H(\varepsilon)$, and $H(sw) = H(s) H(w)$.

   (ii) for each $p : w$ in $\Sigma$ there is $H(p) : w'$ in $\Sigma'$
   such that $H(w) \equiv' w'$.

   Again, this is more flexible than requiring $H(w) \in P(H(p))$. 
Exercise. Prove that signature maps compose, i.e., if we have \( \Sigma \xrightarrow{H} \Sigma' \xrightarrow{G} \Sigma' \), then \( H; G : \Sigma \to \Sigma' \) is also a signature map.

Therefore, ordered FO-signatures and signature maps define a category \( \text{OS-FOSign} \).

The key model-theoretic property of signature maps is that the mapping \( \text{Mod} : \Sigma \to \text{Mod}(\Sigma) \) extends to a contravariant functor \( \text{Mod} : \text{OS-FOSign} \to \text{Cat} \), where we use the notation:

\[
\text{Mod}(\Sigma \xrightarrow{H} \Sigma') = \text{Mod}(\Sigma') \xrightarrow{-1H} \text{Mod}(\Sigma)
\]

and call \( \text{Mod}(H) = -1H \) the \( H \) reduct functor.

In the case when \( H \) is a theory inclusion \( \Sigma \subseteq \Sigma' \), \(-1H\) becomes the usual reduct functor \(-1\Sigma\).

Let us define \(-1H : \text{Mod}(\Sigma') \to \text{Mod}(\Sigma)\) in detail for the objects. Given \( M' \in \text{Mod}(\Sigma')\), say \( M' = (M', \bar{f}', \bar{g}') \), we define \( M' \mid H \) as follows:
1. using the monotonic function \( H : (S, \leq) \to (S', \leq') \) we define \( M' |_H = (M' |_H, ^F_{M' |_H}, ^P_{M' |_H}) \) with

\( S \)-sorted family of sets \( M' |_H = \{ M' |_H, s \} \) for \( s \in S \),

where \( M' |_H, s = M' |_{H(s)} \). Note that, by monotonicity of \( H \), if \( s \leq s' \), then we have \( M' |_H, s \subseteq M' |_H, s' \).

2. \( ^F_{M' |_H} \) maps each \( f : s_1, \ldots, s_n \to s \) in \( \Sigma \) to the function

\( \mathcal{M} |_{H(s_1)} \times \cdots \times \mathcal{M} |_{H(s_n)} \ni (a_1, \ldots, a_n) \mapsto H(f)(a_1, \ldots, a_n) \in \mathcal{M} |_{H(s)} \)

which is well-defined since \( H(f)(x_1 : H(s_1), \ldots, x_n : H(s_n)) \in \mathcal{T}_\Sigma(x) |_{H(s)} \).

3. \( ^P_{M' |_H} \) maps each \( p : w \) in \( \Sigma \) to the subset

\[ M' |_{H(w)} \cap H(p) w' \]

which is well-defined by choosing an \( H(p) : w' \) in \( \Sigma' \) such that \( H(w) \leq w' \), and does not depend of the choice of \( w' \) because of subterm polymorphism.

In what sense does a signature map \( H : \Sigma \to \Sigma' \) induce a translation \( H : \text{FO}(\Sigma) \to \text{FO}(\Sigma') \) of the language of \( \Sigma \) into that of \( \Sigma' \)?
In the full type sense. Let \( X \) be an infinite set of names.
Define \( X_S = \{ X_s \}_{s \in S} \), where \( X_s = X \times \{ s \} \), and we write the pair \( (x, s) \) as the typed variable \( x : s \).

Now note that \( \mathcal{T}_S' (X_{S'}) \upharpoonright_H \) is an \( F \)-algebra and we have an \( S \)-rooted function \( H : X_S \to X_{S'} \upharpoonright_H \),
\[
H = \left\{ \begin{array}{l}
H_s : X_S \ni x : s \mapsto x : H(i) \in X_{H(i)} = X_{S'} \upharpoonright_H, s \end{array} \right\}_{s \in S}.
\]

Therefore there is a unique \( F \)-homomorphism \( H : \mathcal{T}_S (X_S) \to \mathcal{T}_S' (X_{S'}) \upharpoonright_H \) extending \( H \).

This also extends to predicate atoms in the obvious sense:
\[
H : p(t_1, \ldots, t_n) \mapsto H(p)(H(t_1), \ldots, H(t_n)).
\]

H : \( FO(\Sigma) \to FO(\Sigma') \) is then defined in the obvious recursive way:

(i) atoms \( t = t' \) and \( p(t_1, \ldots, t_n) \) are mapped to \( H(t) = H(t') \)
and \( H(p(t_1, \ldots, t_n)) \)
(ii) \( H(\neg \psi) = \neg H(\psi) \), \( H(\psi \lor \psi) = H(\psi) \lor H(\psi) \)
\( \lor \in \{ \land, \lor, \rightarrow \} \)
(iii) \( H(\forall x : s \psi) = \forall x : H(i) H(\psi) \), and
\( H(\exists x : s \psi) = \exists x : H(i) H(\psi) \).
Theorem. Let \( H : \Sigma \rightarrow \Sigma' \) be an OS-FO-signature map. Then for each \( M' \in \text{Mod}(\Sigma') \), \( \varphi \in \text{FO}(\Sigma) \), and \( a \in \text{fo}(\varphi) \rightarrow M' \mid H \) we have
\[
M' \mid H, a \models \varphi \iff M, \tilde{a} \models H(\varphi)
\]
where \( \tilde{a} \in [H(\text{fo}(\varphi)) \rightarrow M'] \)
maps: \( \chi : H(\lambda) \mapsto a(\chi ; t) \in M' \mid H(\lambda) \).

Proof (Sketch) [Filling in the details is a highly recommended exercise!].
The crucial step is to prove this for atoms. That is, for \( t = t' \) a \( \Sigma \)-equation with \( Y = \text{var}(t = t') \) we need to show (1) \( M' \mid H, a \models t = t' \iff M', \tilde{a} \models H(t) = H(t') \)
and for \( \varphi(t_1, \ldots, t_n) \) a predicate atom with \( Y = \text{var}(\varphi(t_1, \ldots, t_n)) \)
need to show (2) \( M' \mid H, a \models \varphi(t_1, \ldots, t_n) \iff M', \tilde{a} \models H(\varphi)(H(t_1), \ldots, H(t_n)) \).

Let us show (1) and leave (2) as an exercise.
The key observations to prove (1) are the following:
(a) For each \( a \in [Y \rightarrow M' \mid H] \), \( \tilde{a} \in [H(Y) \rightarrow M'] \),
\( \tilde{a} : \Sigma, (H(Y)) \rightarrow M' \) extends uniquely to an \( F' \)-homomorphism.
(b) since \( \bar{a}(x: H(y)) = a(x: t) \), a \( \alpha \) refactors as:

\[
\begin{array}{c}
Y \xrightarrow{\eta_Y} T_\Sigma(Y) \\
\downarrow \quad \downarrow \quad \downarrow \\
H \quad T_\Sigma(H(Y)) \quad \tilde{a} \quad H \\
\quad \downarrow \quad \downarrow \\
M \quad M \quad M
\end{array}
\]

which, by the Frechen Corollary (Cf. CS 476 lecture notes), gives us the identity of \( \Sigma \)-homomorphisms:

\[
\begin{array}{c}
T_\Sigma(Y) \xrightarrow{-\alpha} M_H \\
\downarrow \quad \downarrow \\
H \quad \tilde{a}_H \\
\quad \downarrow \\
M_H
\end{array}
\]

Therefore, for any \( \Sigma \)-equation \( t = t' \) we have

\[ t \bar{a} = t \bar{a}' \iff H(t) \tilde{a} = H(t') \tilde{a}, \text{ and} \]

\[
\text{Therefore } \quad M' \models t = t' \quad \iff \quad M', \tilde{a} \models H(t) = H(t')
\]

as desired. \( \square \)

Remark. The above theorem expresses the "invariance of truth under translation" and is crucial for many applications, including applications to SMT solving.