Nested Graph Conditions as String Diagrams

Filippo Bonchi[1], Andrea Corradini[1], and Arend Rensink[2]

[1] University of Pisa, Italy, filippo.bonchi@unipi.it
[2] University of Pisa, Italy, andrea.corradini@unipi.it
[3] University of Twente, Netherlands, arend.rensink@utwente.nl

1 Definitions

- An *interface* I is a discrete graph.
- An *open graph* is an arrow $g\colon I \to G$ where I is an interface; we say that g has *interface* $I_g = I$ and *pattern* $P_g = G$. Open graphs are used for different purposes: within a condition to associate (both upper and lower) interfaces with the patterns, and also as assignments — that is, the objects on which satisfaction is defined.
- An *interface morphism* f from I to J is a graph morphism $k : I \to J$.
- An *open graph morphism* $a\colon g \to h$ a pair of graph morphisms $a^{\mathsf{I}} : I_g \to I_h, a^{\mathsf{P}} : P_g \to P_h$ such that $a^{\mathsf{I}}; h = g; a^{\mathsf{P}}$.

Definition 1 (condition tree). *A* condition tree $\mathcal{T}$ *is a quadruple of the form* $\langle V, \mathrel{\prec\!\!\!-}, \{d_v\}_{v\in V}, \{u_v\}_{v\in V}\rangle$ *in which*

- V *is a non-empty set of vertices;*
- $\mathrel{\prec\!\!\!-} \subseteq V \times V$ *is a parent-child relation, such that one* $v \in P$ *(the* root*, denoted* rt*) has no predecessor, and all other elements of* V *have exactly one predecessor (their* parent*);*
- *for all* $v \in V$*,* d_v *is an open graph. We denote* $I_v = I_{d_v}$ *and* $P_v = P_{d_v}$*.*
- *for all* $v \mathrel{\prec\!\!\!-} w$*,* u_w *is a graph morphism* $I_w \to P_v$*, whereas* $u_{rt} = id_{I_{rt}}$*.*

We use $\prec$ and $\preceq$ to denote the transitive, respectively transitive and reflexive, closure of $\mathrel{\prec\!\!\!-}$. The root of a condition tree $\mathcal{T}$ is denoted $rt_{\mathcal{T}}$; we denote $I_{\mathcal{T}} = I_{rt_{\mathcal{T}}}$, $P_{\mathcal{T}} = P_{rt_{\mathcal{T}}}$ and $d_{\mathcal{T}} = d_{rt_{\mathcal{T}}}$. For an arbitrary vertex $v \in V_{\mathcal{T}}$, $\hat{v}$ will denote the subtree rooted at v. We also use a *sibling* relation, $\smile \subseteq V \times V$, as the smallest equivalence over V such that $v \mathrel{-\!\!\!\succ\!\!\prec\!\!\!-} w$ implies $v \smile w$; i.e., $v \smile w$ if either v and w are (possibly identical) children of the same parent, or if $v = w = rt$.[4] Finally, for an arbitrary vertex $v \in \mathcal{T}$ we use U_v to denote the codomain of u_v; hence $U_w = P_v$ if $w \mathrel{-\!\!\!\succ} v$ and $U_w = I_{\mathcal{T}}$ if $w = rt_{\mathcal{T}}$.

This is new notation; please comment

Definition 2 (satisfaction). *Let* $\mathcal{T}$ *be a condition tree and* $g\colon I_{\mathcal{T}} \to G$ *a graph morphism.* g satisfies $\mathcal{T}$ *denoted* $g \models \mathcal{T}$*, if there is a graph morphism* $h\colon P_{\mathcal{T}} \to G$ *such that* (i) $g = d_{\mathcal{T}}; h$*, and* (ii) $u_v; h \not\models \hat{v}$ *for all children* v *of* $rt_{\mathcal{T}}$*.* h *is then called a* witness *of* $g \models \mathcal{T}$*.*

[4] In fact, this use of the word *sibling* deviates from its meaning in the natural world since there it is not a reflexive relation; however, we use it for lack of a good alternative (*co-child* would be cumbersome).

If g satisfies $\mathcal{T}$, we also call g a $\mathcal{T}$-model.

At this point introduce the connection from condition trees to FOL

Definition 3 (morphism). *Given two condition trees $\mathcal{T}$,$\mathcal{U}$ with $I_{\mathcal{T}} = I_{\mathcal{U}} = I$, a morphism $\mathcal{M} = \langle T, rt \rangle$ from $\mathcal{T}$ to $\mathcal{U}$ is a tuple consisting of*

AC: changed back
AR: note that triples can occur within a single tree; this is required for the sibling rule

- *a set T of triples (v, a, w) where $(v, w) \in (V_{\mathcal{T}} \times V_{\mathcal{U}}) \cup (V_{\mathcal{U}} \times V_{\mathcal{T}})$ and $a\colon d_v \to d_w$ is an open graph morphism;*
- *a distinguished element $rt \in T$, called the* root*, such that $rt = (rt_{\mathcal{T}}, a_{rt}, rt_{\mathcal{U}})$ with $a^{\mathsf{I}}_{rt} = id_I$.*

We typically denote a triple (v, a, w) of the kind above as $v \xrightarrow{a} w$; moreover, we usually associate $\mathcal{M}$ with its component T and write $v \xrightarrow{a} w \in \mathcal{M}$, using $rt_{\mathcal{M}}$ to denote the root of $\mathcal{M}$. Moreover, we also sometimes use the component a (with constituents a^{I} and a^{P}) to refer to the entire triple $v \xrightarrow{a} w$, when that causes no confusion and the identity of v and w is clear from the context. Figure 1 visualises the components of a triple $v \xrightarrow{a} w$.

$$\begin{tikzcd}
I_v \arrow[r, "a^{\mathsf{I}}"] \arrow[d, "d_v"'] & I_w \arrow[d, "d_w"] \\
P_v \arrow[r, "a^{\mathsf{P}}"] & P_w
\end{tikzcd}$$

Fig. 1: Visualisation of a triple $v \xrightarrow{a} w$ as in definition 3. The light blue ellipses represent open graphs corresponding to nodes of $\mathcal{T}$ and $\mathcal{U}$, and the (fat) light blue arrow represents the triple as a whole. For the root triple, $I_v = I_w = I$ and $a^{\mathsf{I}} = id_I$

We could extend the child relation to triples, as follows.

Given a morphism $\mathcal{M}$, we define a parent relation $\prec \subseteq \mathcal{M} \times \mathcal{M}$ as follows:

$$(v, a, w) \prec (y, b, x) \;\Leftrightarrow\; v \prec x \wedge w \prec y \wedge u_y = b^{\mathsf{I}}; u_x; a^{\mathsf{P}} \quad .$$

This might provide a more acceresible and uniform way to define the notions of direct (preservation and reflection) evidence.

Given a condition tree $\mathcal{T}$ with root rt, the identity morphism $\mathcal{I}_{\mathcal{T}}\colon \mathcal{T} \to \mathcal{T}$ is defined as

$$\mathcal{I}_{\mathcal{T}} = \langle \{(v, id_{d_v}, v) \mid v \in V_{\mathcal{T}}\}, (rt, id_{d_{rt}}, rt) \rangle \quad .$$

Given two condition tree morphisms $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}, \mathcal{N}\colon \mathcal{U} \to \mathcal{V}$, their composition $\mathcal{M};\mathcal{N}\colon \mathcal{T} \to \mathcal{V}$ is defined as

$$\mathcal{M};\mathcal{N} = \langle \{(v, a; b, x) \mid \exists w.\ (v, a, w), (w, b, x) \in \mathcal{M} \cup \mathcal{N}\}, (rt_{\mathcal{T}}, a_{rt_{\mathcal{M}}}; a_{rt_{\mathcal{N}}}, rt_{\mathcal{V}}) \rangle \quad .$$

2 Equivalence to First-Order Logic

Nested conditions can express precisely the same properties of graphs as First-Order Logic (FOL). This in fact follows from previous work, but since later on in this paper we strengthen the result to the existence of functors between the respective categories, we start out by recalling the relevant definitions and giving a direct proof of the equivalence. This will also help in developing an intuition about nested conditions.

We restrict to binary predicates a, b, taken from a set *Lab*; furthermore, we use variables x, y taken from a set *Var*. The syntax of FOL that we use is given by the grammar

$$\phi \ ::= \ \mathsf{true} \ \mid \ \mathsf{false} \ \mid \ \mathsf{a}(x_1, x_2) \ \mid \ x_1 \doteq x_2 \ \mid \ \phi_1 \wedge \phi_2 \ \mid \ \phi_1 \vee \phi_2 \ \mid \ \neg\phi \ \mid \ \exists X.\, \phi$$

where $\mathsf{a} \in \mathit{Lab}$ is an arbitrary predicate name, $x_1, x_2 \in \mathit{Var}$ are arbitrary variables and $X \subseteq \mathit{Var}$ an arbitrary set of variables. We use the common $\exists x.\, \phi$ as syntactic sugar for $\exists\{x\}.\, \phi$. We also use the concepts of *free variables* and *bound variables* of ϕ, denoted $\mathit{fv}(\phi)$ and $\mathit{bv}(\phi)$, respectively, both inductively defined in the usual way. Note that unlike $\mathit{fv}(\_)$, $\mathit{bv}(\_)$ is not invariant under α-conversion.

As common practice in categorical logic [?], we consider *formulas in context* of the kind $X \mid \phi$, where the context X is a set of variables containing $\mathit{fv}(\phi)$. As an additional constraint, we require that X is disjoint from $\mathit{bv}(\phi)$: This will simplify the correspondence between formulas and nested conditions without loss of generality, because bound variables can be α-converted if needed.

AC: is x bound in $\exists x.\,\mathsf{true}$? Both answers are reasonable, I think. Here we assume it is. AR: We should make this explicit.

AR: open formulas? enriched formulas? sorted formulas?

Definition 4 (formulas in context). *Let $X \subseteq \mathit{Var}$ and ϕ be a FOL formula. Then $X \mid \phi$ is a* formula in context *if it can be obtained using the following axioms and formation rules:*

$$\frac{}{\emptyset \mid \mathsf{true}} \qquad \frac{}{\emptyset \mid \mathsf{false}} \qquad \frac{}{\{x_1, x_2\} \mid \mathsf{a}(x_1, x_2)} \qquad \frac{}{\{x_1, x_2\} \mid x_1 \doteq x_2}$$

$$\frac{X \mid \phi_1 \quad X \mid \phi_2}{X \mid \phi_1 \wedge \phi_2}\ [\mathit{ctx}\text{-}\wedge] \qquad \frac{X \mid \phi_1 \quad X \mid \phi_2}{X \mid \phi_1 \vee \phi_2}\ [\mathit{ctx}\text{-}\vee] \qquad \frac{X \mid \phi}{X \mid \neg\phi}\ [\mathit{ctx}\text{-}\neg]$$

$$\frac{Z \mid \phi}{Z \setminus X \mid \exists X.\, \phi}\ [\mathit{ctx}\text{-}\exists] \qquad \frac{X \mid \phi \quad X \subseteq Y \quad \mathit{bv}(\phi) \cap Y = \emptyset}{Y \mid \phi}\ [\mathit{ctx}\text{-}\mathit{weak}]$$

For a formula in context $X \mid \phi$ it is easy to verify that $\mathit{fv}(\phi) \subseteq X$ and, thanks to the condition in the weakening rule, that $\mathit{bv}(\phi) \cap X = \emptyset$. This implies that ϕ (and all its sub-formulas) cannot contain a free variable which is bound in a sub-formula.

In this section we use the following concrete category of graphs.

Definition 5. *An edge-labelled multi-graph is a tuple $\langle N, E, s, t, \ell \rangle$ where*

- *N is a set of nodes;*
- *E is a set of edges;*

- *$s, t \colon E \to N$ are the source and target function, respectively;*
- *$\ell \colon E \to Lab$ is the edge labelling function.*

A multi-graph morphism $f \colon A \to B$ is a tuple $f = \langle f^{\mathsf{N}}, f^{\mathsf{E}} \rangle$ where $f^{\mathsf{N}} \colon N_A \to N_B$ and $f^{\mathsf{E}} \colon E_A \to E_B$ are functions satisfying $f^{\mathsf{E}}; s_B = s_A; f^{\mathsf{N}}$, $f^{\mathsf{E}}; t_B = t_A; f^{\mathsf{N}}$ and $f^{\mathsf{E}}; \ell_B = \ell_A$.

Although our theory mostly abstracts from node and edge identities by relating graphs through morphisms, in the context of FOL interpretation we sometimes have to rely on meaningful, fixed node identities (namely where they correspond to variables). We call A and B *consistent* if their source, target and label functions coincide on $E_A \cap E_B$, and *disjoint* if $N_A \cap N_B = E_A \cap E_B = \emptyset$ (in which case they are certainly consistent). If A and B are disjoint, we use $A \cup B$ to denote their union; also, if $g \colon A \to G$ and $h \colon B \to G$ for disjoint A and B, then $g \cup h \colon (A \cup B) \to G$ denotes the union of g and h. We call A a subgraph of B, denoted $A \subseteq B$, if $B = A \cup B$; likewise, f is a sub-morphism of g, denoted $f \subseteq g$, if $f \cup g = g$. If A and B are consistent, then $B \setminus A$ denotes the largest subgraph of B with node set $N_B \setminus N_A$, and for $g \colon A \to G$ we use $g \upharpoonright B$ and $g \setminus B$ to denote the restriction of g to (the nodes and edges of) $A \cap B$ and $A \setminus B$, respectively. Finally, if X is a set then $\langle X \rangle$ will denote the discrete (i.e., edge-less) graph with node set X.

AC: We use $g \setminus B$ also if B is not a subgraph of A, e.g., in the definition of $g \models \exists X.\ \phi_1$. I think it is actually well-defined. AR: $g \setminus B$ is no longer used there.

For the correspondence of FOL to nested conditions, we equate *Var* to the universe of node identities. A graph morphism $g \colon A \to G$ can then be seen as an assignment of variables (the nodes of A, playing the role of the context) to a given domain (the graph G) in which the existence of an a-labelled edge between $g(x)$ and $g(y)$ means that the predicate $\mathsf{a}(x, y)$ holds. Formally, satisfaction is captured by a relation $\models$ between morphisms and formulas in context: we write $g \models \phi$ as a shorthand for $g \colon A \to G \models N_A \mid \phi$. As a consequence, asserting $g \models \phi$ has an implicit proof obligation, i.e. that $N_A \mid \phi$ is a formula in context. Satisfaction is defined inductively over the structure of the formula. For reasons to become clear below, the source graph A of g is not required to be discrete.

$$\begin{array}{ll}
g \models \mathsf{true} & \text{always} \\
g \models \mathsf{false} & \text{never} \\
g \models \mathsf{a}(x_1, x_2) & :\Leftrightarrow \text{there is an } e \in E_G \text{ with } s(e) = g(x_1), \ell(e) = \mathsf{a}, t(e) = g(x_2) \\
g \models x_1 \doteq x_2 & :\Leftrightarrow g(x_1) = g(x_2) \\
g \models \phi_1 \wedge \phi_2 & :\Leftrightarrow g \models \phi_1 \text{ and } g \models \phi_2 \\
g \models \phi_1 \vee \phi_2 & :\Leftrightarrow g \models \phi_1 \text{ or } g \models \phi_2 \\
g \models \neg\phi & :\Leftrightarrow \text{not } g \models \phi \\
g \models \exists X.\ \phi & :\Leftrightarrow h \models \phi \text{ for some } h \colon B \to G \text{ such that } A \subseteq B, \\
 & \quad X \subseteq N_B, \text{ and } h \upharpoonright A = g \ .
\end{array}$$

Note that the last clause requires that $N_A \mid \exists X.\ \phi$ is a formula in context, which implies that $X \cap bv(\phi) = \emptyset$.

AR: is $X \subseteq N_B$ necessary in that clause?

AC: comment more on this?

The following result states that to check if a morphism $g : A \to G$ satisfies a formula ϕ, it is sufficient to consider the action of the morphism on the free variables of ϕ.

AC: the condion is stronger, using formulas in context. AR: how so?

Proposition 1. *Let ϕ be a FOL-formula and $g\colon A \to G$ such that $N_A \mid \phi$ is a formula in context; then $g \models \phi$ if and only if $g \restriction \langle fv(\phi) \rangle \models \phi$.*

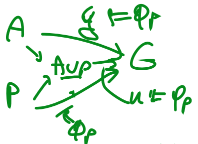
Because of this property, when constructing a morphism $g\colon A \to G$ such that $g \models \phi$, we may assume A to be any graph for which $\langle fv(\phi) \rangle \subseteq A$ and such that $N_A \mid \phi$ is a formula in context.

In the remainder of this section, we show that *(i)* for every nested condition, there is an equivalent FOL formula, and *(ii)* for every FOL formula, there is an equivalent nested condition.

2.1 From nested conditions to FOL

We first define formulas in context for graphs, and then (inductively) for nested conditions. Given a graph P, we define

$$\phi_P = \bigwedge\nolimits_{e \in E_P} \ell(e)\big(s(e), t(e)\big) \tag{1}$$



$K : \underline{A} \to G \models \underline{X} \mid \varphi$

Note that $fv(\phi_P) \subseteq N_P$ and that $N_P \mid \phi_P$ is a formula in context. The essential property of $N_P \mid \phi_P$ is the following.

Proposition 2. *Let P be a graph.*

$\to N_P \mid \phi_P$

$N_A \mid \phi_P$

1. *If $g\colon P \to G$ is a graph morphism, then $g \models \phi_P$;*
2. *If $g : \underline{A} \to G$ is a graph morphism, $\langle N_P \rangle \subseteq A$ and $g \models \phi_P$, then there is a $g' : A' \to G$ such that $P \subseteq A'$, $A \subseteq A'$, $N_{A'} = N_A$, ~~$g' \restriction A$~~ $= g$ and $g' \models \phi_P$.*

$A \cup P$

Doesn't this follow from $g \models \phi_P$?

I think we can simplify this by saying that g' : $(A \cup P) \to G$.

To lift this the definition from graphs to nested conditions, it is convenient to assume that some of the graphs appearing in the latter are disjoint. We can do this without loss of generality, because we can always rename node and edge identities up to isomorphism.

Definition 6. *A nested condition $\mathcal{T}$ is* proper *if for all pair of nodes v, w of $\mathcal{T}$, if w is a descendant of v (i.e. $v \prec w$), then P_v and P_w are disjoint.*

Let $\mathcal{T}$ be a proper nested condition. We define ψ_v, ϕ_v for all nodes $v \in V_{\mathcal{T}}$, inductively on the depth of the subtree $\hat{v}$:

$$\psi_v = \phi_{P_v} \wedge \bigwedge\nolimits_{x \in N_{I_v}} d_v(x) \doteq u_v(x) \wedge \bigwedge\nolimits_{w \succ v} \neg\phi_w \tag{2}$$

$$\phi_v = \exists N_{P_v}.\, \psi_v \tag{3}$$

Finally, we define $\phi_{\mathcal{T}} = \phi_{rt_{\mathcal{T}}}$.

AC: Add examples? eg for some limit cases like true and false?

Proposition 3. *Let $\mathcal{T}$ be a proper nested condition, $v \in V_{\mathcal{T}}$ be a node and ψ_v, ϕ_v be the FOL formula defined in* (2) *and* (3)*. Then $N_{U_v} \cup N_{P_v} \mid \psi_v$ and $N_{U_v} \mid \phi_v$ are formulas in context.*

As a consequence, $N_{I_{\mathcal{T}}} \mid \phi_{\mathcal{T}}$ is a formula in context.

Proof. The second statement immediately follows from the first one, since $\phi_{\mathcal{T}} = \phi_{rt_{\mathcal{T}}}$ and $U_{rt_{\mathcal{T}}} = I_{rt_{\mathcal{T}}} = I_{\mathcal{T}}$.
For the first statement, we proceed by induction, assuming that the property holds for all $w \succ v$ and showing that it then holds for v. Let $v \in V_{\mathcal{T}}$, and assume by inductive hypothesis that $N_{U_w} \mid \phi_w$ is a formula in context for all $w \succ v$.

To show that $N_{U_v} \cup N_{P_v} \mid \psi_v$ is a formula in context, by rule [ctx-$\wedge$] it is sufficient to show that (a) $N_{U_v} \cup N_{P_v} \mid \phi_{P_v}$, (b) $N_{U_v} \cup N_{P_v} \mid \bigwedge_{x \in N_{I_v}} d_v(x) \doteq u_v(x)$, and (c) $N_{U_v} \cup N_{P_v} \mid \bigwedge_{w \succ v} \neg\phi_w$ are formulas in context. For (a), by the observation after (1) we know that $N_{P_v} \mid \phi_{P_v}$ is a formula in context, and we can conclude using rule [ctx-weak]: in fact $bv(\phi_{P_v}) = \emptyset$. For (b), note that for all $x \in N_{I_v}$ we have that $\{d_v(x), u_v(x)\} \mid d_v(x) \doteq u_v(x)$ is a formula in context, that $\{d_v(x), u_v(x)\} \subseteq N_{U_v} \cup N_{P_v}$, and $bv(d_v(x) \doteq u_v(x)) = \emptyset$. Therefore using rules [ctx-weak] and [ctx-$\wedge$] we obtain that $N_{U_v} \cup N_{P_v} \mid \bigwedge_{x \in N_{I_v}} d_v(x) \doteq u_v(x)$ is a formula in context. Finally, for (c), for all $w \succ v$ first observe that $U_w = P_v$. Thus by inductive hypothesis we can assume that $N_{P_v} \mid \phi_w$ is a formula in context. Furthermore, since $\mathcal{T}$ is proper, no node of U_v can be bound in ϕ_w, thus applying rules [ctx-$\neg$] and [ctx-weak] we obtain that $N_{U_v} \cup N_{P_v} \mid \neg\phi_w$ is a formula in context, and so is $N_{U_v} \cup N_{P_v} \mid \bigwedge_{w \succ v} \neg\phi_w$ by applying rule [ctx-$\wedge$].

To show that $N_{U_v} \mid \phi_v$ is a formula in context, observe that since $N_{U_v} \cup N_{P_v} \mid \psi_v$ is a formula in context and N_{U_v} and N_{P_v} are disjoint (because $\mathcal{T}$ is proper), we have, as desired:

$$\frac{N_{U_v} \cup N_{P_v} \mid \psi_v}{N_{U_v} \mid \phi_v = \exists N_{P_v}.\, \psi_v}\ \text{[ctx-}\exists\text{]}$$

□

Theorem 1. *Let $\mathcal{T}$ be a proper nested condition, $v \in V_{\mathcal{T}}$ be a node and $g\colon U_v \to G$ be a graph morphism. Then $g \models \phi_v$ if and only if $u_v; g \models \hat{v}$.*

AC: use different simbols for the two notions of satisfaction?

As a consequence, if $g : I_{\mathcal{T}} \to G$, then $g \models \phi_{\mathcal{T}}$ if and only if $g \models \mathcal{T}$.

Proof. The second statement follows from the first one, because $\phi_{\mathcal{T}} = \phi_{rt_{\mathcal{T}}}$ and $U_{rt_{\mathcal{T}}} = I_{rt_{\mathcal{T}}} = I_{\mathcal{T}}$.

For the first statement, as in the previous proposition we proceed by induction, assuming that the property holds for all $w \succ v$, and showing that it then holds for v.

If. Assume $u_v; g \models \hat{v}$, meaning that there is some $h\colon P_v \to G$ such that $d_v; h = u_v; g$ and $u_w; h \not\models \hat{w}$ for all $w \succ v$. Since P_v and U_v are disjoint (because $\mathcal{T}$ is proper), we can construct the graph $A = P_v \cup U_v$ and the graph morphism $k = g \cup h\colon A \to G$. Note that $g = k \restriction U_v$:

By Proposition 2, it follows that $h \models \phi_{P_v}$. Moreover, by the induction hypothesis, $h \not\models \phi_w$, implying $h \models \neg\phi_w$. Proposition 1 then implies $k \models \phi_{P_v}$ and $k \models \neg\phi_w$, given that both $N_{U_v} \cup N_{P_v} \mid \phi_v$ and $N_{U_v} \cup N_{P_v} \mid \neg\phi_w$ are formulas in context by Proposition 3. Also, for any $x \in N_{I_v}$,

$$k(d_v(x)) = h(d_v(x)) = (d_v; h)(x) = (u_v; g)(x) = g(u_v(x)) = k(u_v(x))\ ,$$

hence $k \models d_v(x) \doteq u_v(x)$. All in all, we have $k \models \psi_v$, where $N_{U_v} \cup N_{P_v} \mid \psi_v$ is a formula in context by Proposition 3. Since $g = k \restriction U_v$, by the definition of satisfaction we have $g \models \exists N_{P_v}.\ \psi_v$, i.e. $g \models \phi_v$.

Only if. Assume $g \models \phi_v = \exists N_{P_v}.\ \psi_v$. This means that there is a $k\colon A \to G$ such that $U_v \subseteq A$, $N_{P_v} \subseteq N_A$, $k \restriction U_v = g$ and $k \models \psi_v$. Since $k \models \psi_v$, we know that $k \models \phi_{P_v}$. Then by Proposition 2 there is a $k' : A' \to G$ such that $P_v {\subseteq} A'$, $A \subseteq A'$, $N_{A'} = N_A$, $k' \restriction A = k$ and $k' \models \phi_{P_v}$. Let $h : P_v \to G$ be $P_v \hookrightarrow A' \xrightarrow{k'} G$: we show that h is a witness for $g \models \hat{v}$.

First, note that $k' \models \psi_v = \left(\phi_{P_v} \wedge \bigwedge_{x \in N_{I_v}} d_v(x) \doteq u_v(x) \wedge \bigwedge_{w \succ v} \neg\phi_w\right)$ by Proposition 1, because $k \models \psi_v$ and $N_{A'} = N_A$.

- $k' \models \bigwedge_{x \in N_{I_v}} d_v(x) {\doteq} u_v(x)$ implies that $h(d_v(x)) = k'(d_v(x)) = k'(u_v(x)) = k(u_v(x)) = g(u_v(x))$ for all $x \in N_{I_v}$, and since $I_v = \langle N_{I_v} \rangle$ this implies $d_v; h = u_v; g$.
- $k' \models \neg\phi_w$ (for all $w \succ v$) implies $h \models \neg\phi_w$ (by Proposition 1, since $N_{P_v} = N_{U_w} \mid \phi_w$ is a formula in context by Proposition 3), hence $h \not\models \phi_w$, hence (by the induction hypothesis) $u_w; h' \not\models \hat{w}$.

The formula $\phi_{\mathcal{T}}$ can in practice be simplified by α-converting the variable names such that they are shared as much as possible, rather than being distinct on each layer as required for the definition; depending on the condition tree, this may allow many or even all of the equations $d_v(x) \doteq u_v(x)$ in ϕ_v to be omitted. This is in particular the case if both d_v and u_v are injective. From the logical pont of view, this can be obtained by repeatedly applying the law $(\exists x.\ x = y \wedge P(x)) \equiv P(y)$.

2.2 From formulas in context to nested conditions

AR: At some technical cost, the transformation results in *proper* nested conditions. Do you think that is sufficiently stronger to warrant the cost?

This is shown inductively over the structure of FOL formulas in context. Indeed, we show that

- for every axiom in Definition 4 of the shape $X \mid \phi$ there exists a corresponding proper nested condition $\mathcal{T}_\phi$ with root interface $\langle X \rangle$;
- for every formation rule in Definition 4 there exists a construction over proper nested conditions that mimics it.

Because of the De Morgan duality $\phi_1 \vee \phi_2 \equiv \neg(\neg\phi_1 \wedge \neg\phi_2)$, in fact we only have to provide constructions (of the second kind) for $\wedge$, $\neg$ and $\exists$, as well as for the weakening rule [ctx-weak] .

Definition 7 (from atomic formulas to nested conditions). *The following clauses define for each axiom of Definition 4 a corresponding proper nested condition, having the context as root interface.*

- $\mathcal{T}_{\mathsf{true}}$ *is given by* $V = \{v\}$ *with* $I_v = P_v = \langle\emptyset\rangle$ *(the empty graph), where* $u_v = d_v = id_{I_v}$.

- $\mathcal{T}_{\mathsf{false}}$ *is given by* $V = \{v, w\}$ *with* $v \prec w$, $I_v = P_v = I_w = P_w = \langle \emptyset \rangle$, *where* $u_v = d_v = u_w = d_w = id_{I_v}$.
- $\mathcal{T}_{\mathsf{a}(x_1,x_2)}$ *is given by* $V = \{v\}$ *with* $I_v = \langle \{x_1, x_2\} \rangle$ *and* P_v *having as nodes* $\{x'_1, x'_2\}$ *and a single* a*-labelled edge from* x'_1 *to* x'_2. *Furthermore* $u_v = id_{I_v}$ *and* $d_v \colon I_v \to P_v$ *is the morphism mapping* x_i *to* x'_i *for* $i \in \{1, 2\}$.
- $\mathcal{T}_{x_1 \doteq x_2}$ *is given by* $V = \{v\}$ *with* $I_v = \langle \{x_1, x_2\} \rangle$ *and* $P_v = \langle \{y\} \rangle$, *where* $u_v = id_{I_v}$ *and* $d_v \colon I_v \to P_v$ *is the only available morphism.*

Proposition 4. *Let* $\mathcal{T}$ *be a proper nested condition and* $g \colon I_{\mathcal{T}} \to G$ *a graph morphism.*

1. *If* $\mathcal{T} = \mathcal{T}_{\mathsf{true}}$, *then* $g \models \mathcal{T}$ *if and only if* $g \models \mathsf{true}$.
2. *If* $\mathcal{T} = \mathcal{T}_{\mathsf{false}}$, *then* $g \models \mathcal{T}$ *if and only if* $g \models \mathsf{false}$.
3. *If* $\mathcal{T} = \mathcal{T}_{\mathsf{a}(x_1,x_2)}$, *then* $g \models \mathcal{T}$ *if and only if* $g \models \mathsf{a}(x_1, x_2)$.
4. *If* $\mathcal{T} = \mathcal{T}_{x_1 \doteq x_2}$, *then* $g \models \mathcal{T}$ *if and only if* $g \models x_1 \doteq x_2$.

To encode negation, according to rule [ctx-$\neg$] given a proper nested condition $\mathcal{T}$ for the formula in context $X \mid \phi$, we must provide a proper nested condition for $X \mid \neg\phi$, thus having the same interface $\langle X \rangle$. We introduce a construction $\neg\mathcal{T}$ that essentially "pushes" $\mathcal{T}$ one level down by introducing a new root, suitably renaming the nodes of the old root. More explicitly, $\mathcal{U} = \neg\mathcal{T}$ is given by $V_{\mathcal{U}} = \langle V_{\mathcal{T}} \uplus \{rt\} \rangle$[5] with $\prec_{\mathcal{U}} = \prec_{\mathcal{T}} \cup \{(rt, rt_{\mathcal{T}})\}$, $I_{rt} = I_{\mathcal{T}}$, $u_{rt} = id_{I_{\mathcal{T}}}$, $P_{rt} = \langle Z \rangle$ with Z a set of fresh node identities with $i : N_{I_{\mathcal{T}}} \to Z$ a bijection, $d_{rt} = i$, and u_u for all $rt_{\mathcal{T}} \prec u$ adjusted by post-composing them with i, whereas for all remaining $v \in V_{\mathcal{T}}$, d_v and u_v remain as they were in $\mathcal{T}$.

u is fixed for the root and need not be defined.

Proposition 5. *Let* $\mathcal{T}$ *be a proper nested condition and* $g \colon I_{\mathcal{T}} \to G$ *a graph morphism; then* $g \models \neg\mathcal{T}$ *if and only if* $g \not\models \mathcal{T}$.

To encode conjunction, according to rule [ctx-*wedge*], given two proper nested conditions $\mathcal{T}_1$ and $\mathcal{T}_2$ for the formulas in context $X \mid \phi_1$ and $X \mid \phi_2$, therefore with the same root interface $\langle X \rangle$, we must provide a proper nested condition having the same root interface for the conjunction $\phi_1 \wedge \phi_2$. This construction is based on the pushout of the root morphisms.

Let $\mathcal{T}_i$ for $i = 1, 2$ be proper nested conditions with $V_1 \cap V_2 = \emptyset$ and $I_{\mathcal{T}_1} = I_{\mathcal{T}_2} = I$, and denote $P_i = P_{rt_{\mathcal{T}_i}}$ and $d_i = d_{rt_{\mathcal{T}_i}}$ for $i = 1, 2$. The pushout of d_1 and d_2 consists of a pushout object $P = P_1 \times_I P_2$ and morphisms $d'_i \colon P_i \to P$ for $i = 1, 2$, and for every morphism $f \colon A \to P_i$ there is an extended morphism $f; d'_i \colon A \to P$; in particular, we denote $d = d_1; d'_1 = d_2; d'_2 \colon I \to P$. Now we define $\mathcal{U} = \mathcal{T}_1 \times \mathcal{T}_2$ with $V_{\mathcal{U}} = ((V_1 \setminus \{rt_{\mathcal{T}_1}\}) \cup (V_2 \setminus \{rt_{\mathcal{T}_2}\})) \uplus \{rt\}$,[6]5 $\prec_{\mathcal{U}} = (\prec_{\mathcal{T}_1} \cup \prec_{\mathcal{T}_2}) \restriction (V_{\mathcal{U}} \times V_{\mathcal{U}}) \cup \{(rt, v) \mid rt_{\mathcal{T}_1} \prec_1 v \vee rt_{\mathcal{T}_2} \prec_2 v\}$ and

- $I^{\mathcal{U}}_{rt} = I$, $P^{\mathcal{U}}_{rt} = P$ and $d^{\mathcal{U}}_{rt} = d$;
- $u^{\mathcal{U}}_w = u^{\mathcal{T}_i}_w; d'_i$ for all $rt \prec_{\mathcal{U}} w$.

I don't think we have to define I_{rt} and P_{rt}, as they are derived from d_{rt}. Come to think of it, this holds for all constructions in this section.

[5] The notation means that we assume to choose a node identity rt not appearing in $V_{\mathcal{T}}$.

Proposition 6. *Let $\mathcal{T}_1, \mathcal{T}_2$ be proper nested conditions with $V_1 \cap V_2 = \emptyset$ and $I_{\mathcal{T}_1} = I_{\mathcal{T}_2} = I$. For all $g \colon I \to G$, $g \models \mathcal{T}_1 \times \mathcal{T}_2$ if and only if $g \models \mathcal{T}_1$ and $g \models \mathcal{T}_2$.*

A formula in context can be obtained by weakening, using rule [ctx-weak]. If the premises of the rule hold, namely $X \mid \phi$ is a formula in context, $X \subseteq Y$ and $bv(\phi) \cap Y = \emptyset$ we know that the variables in $Y \setminus X$ do not appear either free nor bound in ϕ. Therefore given a proper nested condition for $X \mid \phi$, we can obtain one for $X \mid \phi$ with the following auxiliary construction. For any discrete graph $\langle Z \rangle$ disjoint from $\mathcal{T}$, $\mathcal{U} = \mathcal{T} \oplus \langle Z \rangle$ is equal to $\mathcal{T}$ except that $\langle Z \rangle$ is "added" to the root; hence

- $I^{\mathcal{U}}_{rt} = I^{\mathcal{T}}_{rt} \cup \langle Z \rangle$
- $P^{\mathcal{U}}_{rt} = P^{\mathcal{T}}_{rt} \cup \langle Z' \rangle$, where Z' are fresh node indentifiers with a bijection $i : Z \to Z'$. Let ϵ be the embedding $\epsilon \colon P^{\mathcal{T}}_{rt} \to P^{\mathcal{U}}_{rt}$.
- $d^{\mathcal{U}}_{rt} = d^{\mathcal{T}}_{rt} \cup i$;
- $u^{\mathcal{U}}_{w} = u^{\mathcal{T}}_{w}; \epsilon$ for all $w \succ rt$.

This seems to assume that the nodes in the derived proper nested condition are always precisely the variables in the formula. That is not always true, see the construction for negation.

Do you mean disjoint from $I_{\mathcal{T}}$, or also from all patterns occurring in $\mathcal{T}$ so that the construction is proper?

Do we need fresh node identifiers, if Z is already known to be disjoint from all patterns of $\mathcal{T}$?

Proposition 7. *Let $\mathcal{T}$ be a proper nested condition and $\langle Z \rangle$ a discrete graph disjoint from $\mathcal{T}$. Then $\mathcal{T} \oplus \langle Z \rangle$ is a proper nested condition and for all $g \colon I_{rt} \cup \langle Z \rangle \to G$, $g \models \mathcal{T} \oplus \langle Z \rangle$ if and only if $g \setminus \langle Z \rangle \models \mathcal{T}$.*

Finally, the encoding of the existential quantification is pretty simple. By rule [ctx-$\exists$], given a nested condition for $Z \mid \phi$, i.e. $\mathcal{T}_\phi$ with root interface $\langle Z \rangle$ we need to provide a nested condition for $(Z \setminus X) \mid \exists X.\, \phi$. We obtain it using the following operation of *restriction* on nested conditions: given $\mathcal{T}$ and a set X, nested condition $\mathcal{T} \restriction X$ is identical to $\mathcal{T}$ but for the root rt': $I_{rt'} = \langle N_{I_{\mathcal{T}}} \setminus X \rangle$, $u_{rt'} = id_{I_{rt'}}$, and $d_{rt'} = i; d_{\mathcal{T}}$, where $i : \langle N_{I_{\mathcal{T}}} \setminus X \rangle \to I_{\mathcal{T}}$ is the inclusion.

Summarizing, for a formula in context $X \mid \phi$ we define the corresponding nested condition $\mathcal{T}_\phi$ with root $\langle X \rangle$ inductively as follows:

I think you mean to replace the original root rt by this rt', but that hasn't been completely defined. Alternatively, you can just change the d-morphism of the root (and hence the interface).

AC: Denote it everywhere as $\mathcal{T}_\phi \colon \langle X \rangle$? AR: Do you mean for the next proposition, or really everywhere in the paper? I think I would rather use $\mathcal{T}_{X \mid \phi}$ as a notation.

- $\mathcal{T}_{\mathsf{true}}$, $\mathcal{T}_{\mathsf{false}}$, $\mathcal{T}_{\mathsf{a}(x_1,x_2)}$ and $\mathcal{T}_{x_1 \doteq x_2}$ are as in Definition 7.
- $\mathcal{T}_{\mathsf{true}} \colon \langle \emptyset \rangle$, $\mathcal{T}_{\mathsf{false}} \colon \langle \emptyset \rangle$, $\mathcal{T}_{\mathsf{a}(x_1,x_2)} \colon \langle x_1, x_2 \rangle$ and $\mathcal{T}_{x_1 \doteq x_2} \colon \langle x_1, x_2 \rangle$ are as in Definition 7.
- $\mathcal{T}_{\neg\phi} \colon \langle X \rangle = \neg \mathcal{T}_\phi \colon \langle X \rangle$
- $\mathcal{T}_{\phi_1 \wedge \phi_2} \colon \langle X \rangle = \mathcal{T}_{\phi_1} \colon \langle X \rangle \times \mathcal{T}_{\phi_2} \colon \langle X \rangle$
- If $X \subseteq Y$ and $bv(\phi) \cap Y = \emptyset$, then $\mathcal{T}_\phi \colon \langle Y \rangle = \mathcal{T}_\phi \colon \langle X \rangle \oplus \langle Y \setminus X \rangle$
- $\mathcal{T}_{\exists X.\, \phi} \colon \langle Z \setminus X \rangle = \mathcal{T}_\phi \colon \langle Z \rangle \restriction Z \setminus X$
- $\mathcal{T}_{\phi_1 \vee \phi_2} \colon \langle X \rangle = \mathcal{T}_{\neg(\neg\phi_1 \wedge \neg\phi_2)} \colon \langle X \rangle$

AC: either this one or the next item.

The main result of this sections states that the proposed encoding is precise in terms of satisfaction.

Proposition 8. *Let $X \mid \phi$ be a formula in context, and let $\mathcal{T}_\phi \colon \langle X \rangle$ be the corresponding nested condition. Then for any graph morphisms $g : \langle X \rangle \to G$ we have*

$$g \models \phi \qquad \Longleftrightarrow \qquad g \models \mathcal{T}_\phi$$

Proof (Sketch).

- For the atomic formulas this holds by Proposition 4.
- For $\neg\phi$ this holds by Proposition 5.
- For $\phi_1 \wedge \phi_2$ this holds by Proposition 6.
- For a formula in context obtained by weakening, this holds by Proposition 7.
- We prove it explicitly for existential quantification. Given $Z \mid \phi$, assume by induction hypothesis that for each $f : \langle Z \rangle \to G$ we have that $f \models \phi$ if and only if $f \models \mathcal{T}_\phi$.
  (if: sketchy) Assume that $g \models \mathcal{T}_{\exists X.\, \phi}$, that is $g \models \mathcal{T}_\phi \colon \langle Z \rangle \upharpoonright Z \setminus X$. Therefore there is a witness $h : P \to G$ s.t... Consider $h' = h \upharpoonright \langle Z \rangle$. Then $h' \models \mathcal{T}_\phi$, and by induction hypothesis $h' \models \phi$. Now show that $h \models \exists X.\, \phi$ using the definition of satisfaction.
  (only if: missing)

3 Model reflection

With the above in mind, we first define reflection of models, this being the more straightforward of the two directions.

Definition 8 (model reflection). *Let $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be a morphism.*

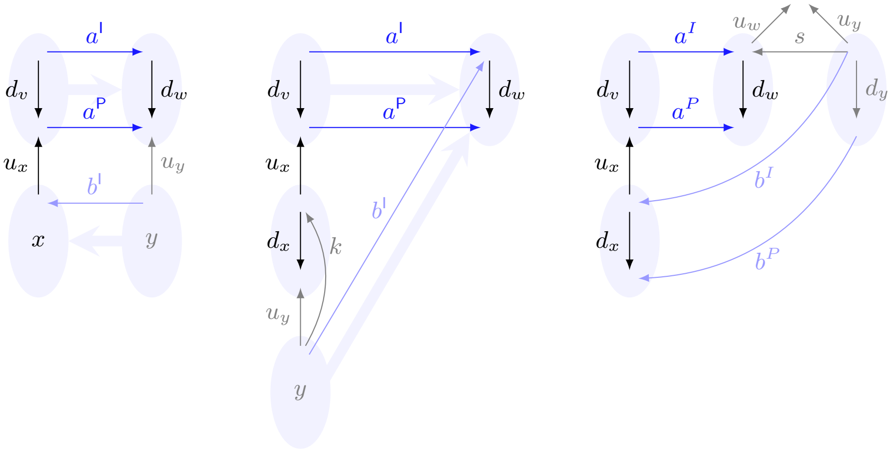
- $v \xrightarrow{a} w \in \mathcal{M}$ reflects a $\hat{w}$-model g *if there is a sibling $v' \smile v$ with a morphism $s : I_{v'} \to I_v$ such that $u_{v'} = s; u_v$ and $s; a^{\mathsf{I}}; g \models \hat{v}'$. We say that* $v \xrightarrow{a} w$ reflects models *if it reflects all $\hat{w}$-models.*
- $\mathcal{M}$ reflects models *if $rt_{\mathcal{M}}$ reflects models.*

Note that, in case $v \xrightarrow{a} w = rt_{\mathcal{M}}$, we have $a^{\mathsf{I}} = id_I$, and $v' \smile v$ implies $v' = v$; hence the condition of model reflection implies that, if g is a model of $\mathcal{U} = \hat{w}$, then $g = a^{\mathsf{I}}; g$ is a model of $\hat{v} = \mathcal{T}$; in other words, the models of $\mathcal{U}$ form a subset of those of $\mathcal{T}$.

A given triple $v \xrightarrow{a} w$ can be shown to reflect models if the morphism as a whole provides enough internal *evidence* for reflection. In particular, for a candidate model-reflecting triple $v \xrightarrow{a} w \in \mathcal{M}$, for every child x of v we require the existence of further triples in $\mathcal{M}$ (the evidence). To obtain a rich enough framework, we introduce diverse forms of evidence.

Definition 9 (reflection evidence). *Let $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be an CT-morphism. For a given $v \xrightarrow{a} w \in \mathcal{M}$ and child x of v, we define the following kinds of* reflection evidence.

- (Direct reflection evidence.) $y \xrightarrow{b} x \in \mathcal{M}$ *provides* direct reflection evidence, *denoted $b \in \mathsf{re\text{-}dir}(x, a)$, if $w -\!\!\!\prec y$ and $u_y = b^{\mathsf{I}}; u_x; a^{\mathsf{P}}$.*
- (Child-based reflection evidence.) $y \xrightarrow{b} w \in \mathcal{M}$ *provides* child-based reflection evidence, *denoted $b \in \mathsf{re\text{-}chd}(x, a)$, if $x -\!\!\!\prec y$ and there is a mediating morphism $k\colon I_y \to I_x$ such that $k; d_x = u_y$ and $k; u_x; a^{\mathsf{P}} = b^{\mathsf{I}}; d_w$.*
- (Sibling-based reflection evidence.) $y \xrightarrow{b} x \in \mathcal{M}$ *provides* sibling-based reflection evidence, *denoted $b \in \mathsf{re\text{-}sib}(x, a)$, if $y \smile w$ and there is a mediating morphism $s\colon I_y \to I_w$ such that $u_y = s; u_x$ and $s; d_w = b^{\mathsf{I}}; u_x; a^{\mathsf{P}}$.*



Direct
($b \in \mathsf{re\text{-}dir}(x, a)$)

Child-based
($b \in \mathsf{re\text{-}chd}(x, a)$)

Sibling-based
($b \in \mathsf{re\text{-}sib}(x, a)$)

Fig. 2: The notions of reflection evidence of definition 9

For a visualisation see fig. 2.

Alternatively: for all $v \xrightarrow{a} w \in \mathcal{M}$ and child x of v, we can define the sets $\mathsf{re\text{-}dir}(a, x)$, $\mathsf{re\text{-}chd}(a, x)$ and $\mathsf{re\text{-}sib}(a, x)$ as below, as well as the set $\mathsf{re}(a, x)$ of (generalised) reflection evidence as their union.

$$\begin{aligned}
\mathsf{re\text{-}dir}(a, x) &= \{y \xrightarrow{b} x \in \mathcal{M} \mid w \prec y, b^{\mathsf{I}}; u_x; a^{\mathsf{P}} = u_y\} \\
\mathsf{re\text{-}chd}(a, x) &= \{y \xrightarrow{b} w \in \mathcal{M} \mid x \prec y, \exists k : I_y \to I_x.\ k; d_x = u_y \wedge k; u_x; a^{\mathsf{P}} = b^{\mathsf{I}}; d_w\} \\
\mathsf{re\text{-}sib}(a, x) &= \{y \xrightarrow{b} x \in \mathcal{M} \mid w \smile y, \exists s : I_y \to I_w.\ u_y = s; u_x \wedge s; d_w = b^{\mathsf{I}}; u_x; a^{\mathsf{P}}\} \\
\mathsf{re}(a, x) &= \mathsf{re\text{-}dir}(a, x) \cup \mathsf{re\text{-}chd}(a, x) \cup \mathsf{re\text{-}sib}(a, x) \ .
\end{aligned}$$

These evidence arrows satisfy the following (rather technical) properties.

Lemma 1. *Let $\mathcal{M} \colon \mathcal{T} \to \mathcal{U}$ be a condition tree morphism, let $v \xrightarrow{a} w \in \mathcal{M}$ and $x \succ v$. Let $g \models \hat{w}$ with witness h.*

1. *If $y \xrightarrow{b} x \in \mathsf{re\text{-}dir}(a, x)$ and b reflects models, then $u_x; a^{\mathsf{P}}; h \not\models \hat{x}$.*
2. *If $y \xrightarrow{b} w \in \mathsf{re\text{-}chd}(a, x)$ and b reflects models, then $u_x; a^{\mathsf{P}}; h \not\models \hat{x}$.*
3. *If $y \xrightarrow{b} x \in \mathsf{re\text{-}sib}(a, x)$ and b reflects models, then either there is a sibling $w' \smile w$ with a morphism $s' \colon I_{w'} \to I_w$ such that $u_{w'} = s'; u_w$ and $s'; g \models \hat{w}'$, or $u_x; a^{\mathsf{P}}; h \not\models \hat{x}$.*

Proof. In each case, assume $u_x; a^{\mathsf{P}}; h \models \hat{x}$.

1. $y \xrightarrow{b} x \in \mathsf{re\text{-}dir}(a, x)$ means that $w \prec y$ and $b^{\mathsf{I}}; u_x; a^{\mathsf{P}} = u_y$. Since b reflects models, $u_x; a^{\mathsf{P}}; h \models \hat{x}$ implies there is a sibling $y' \smile y$ with a morphism $s \colon I_{y'} \to I_y$ such that $u_{y'} = s; u_y$ and $s; b^{\mathsf{I}}; u_x; a^{\mathsf{P}}; h \models \hat{y}$, implying (due to $u_{y'}; h = s; u_y; h = s; b^{\mathsf{I}}; u_x; a^{\mathsf{P}}; h$) that $u_{y'}; h \models \hat{y}$, which contradicts $g \models \hat{w}$.

2. $y \xrightarrow{b} w \in \mathsf{re\text{-}chd}(a,x)$ means that $x \prec y$ and there is a mediating morphism $k\colon I_y \to I_x$ such that $k;d_x = u_y$ and $k;u_x;a^{\mathsf{P}} = d_y;b^{\mathsf{P}}$. Since $u_x;a^{\mathsf{P}};h \models \hat{x}$, there must be some witness $f\colon P_x \to G$ such that $u_x;a^{\mathsf{P}};h = d_x;f$ and $u_{y'};f \not\models \hat{y}'$ for all siblings $y' \smile y$. Since b reflects models, there is a sibling $y' \smile y$ with a morphism $s : I_{y'} \to I_y$ such that $u_{y'} = s;u_y$ and $s;b^{\mathsf{I}};g \models \hat{y}$; since $s;b^{\mathsf{I}};g = s;b^{\mathsf{I}};d_w;h = s;k;u_x;a^{\mathsf{P}};h = s;k;d_x;f = s;u_y;f = u_{y'};f$, this gives rise to a contradiction.
3. $y \xrightarrow{b} x \in \mathsf{re\text{-}sib}(a,x)$ means that there is a morphism $s\colon I_y \to I_w$ such that $u_y = s;u_w$ and $s;d_w = b^{\mathsf{I}};u_x;a^{\mathsf{P}}$. Since $u_x;a^{\mathsf{P}};h \models \hat{x}$ and b reflects models, there is a sibling $y' \smile y$ with a morphism $s''\colon I_{y'} \to I_y$ such that $u_{y'} = s'';u_y$ and $s'';b^{\mathsf{I}};u_x;a^{\mathsf{P}};h \models \hat{y}'$. Let $v' = y'$ and $s' = s'';s$; then $v' \smile v$, $u_{v'};s' = u_y;s = u_v$ and $s';a^{\mathsf{I}};g = s';a^{\mathsf{I}};d_w;h = s'';s;d_v;a^{\mathsf{P}};h = s'';b^{\mathsf{I}};u_x;a^{\mathsf{P}};h \models \hat{v}'$.

continue

Based on these notions of evidence, we define corresponding notions of *syntactic reflection*:

Definition 10 (syntactic reflection). *Let $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be a condition tree morphism, and let $v \xrightarrow{a} w \in \mathcal{M}$.*

- (Direct reflection.) *$a \in \mathsf{r\text{-}dir}$ if for every child x of v, there is some $b \in \mathsf{r\text{-}dir}$ such that $b \in \mathsf{re\text{-}dir}(x,a)$.*
- (Child-based reflection.) *$a \in \mathsf{r\text{-}chd}$ if for every child x of v, there is some $b \in \mathsf{r\text{-}chd}$ such that $b \in \mathsf{re\text{-}dir}(x,a)$ or $b \in \mathsf{re\text{-}chd}(x,a)$.*
- (Sibling-based reflection.) *$a \in \mathsf{r\text{-}sib}$ if for every child x of v, there is some $b \in \mathsf{r\text{-}sib}$ such that $b \in \mathsf{re\text{-}dir}(x,a)$ or $b \in \mathsf{re\text{-}sib}(x,a)$.*
- (General reflection.) *$s \in \mathsf{r}$ if for every child x of v, there is some $b \in \mathsf{r}$ such that $b \in \mathsf{re\text{-}dir}(x,a)$, $b \in \mathsf{re\text{-}chd}(x,a)$ or $b \in \mathsf{re\text{-}sib}(x,a)$.*

Alternatively: $\mathsf{r\text{-}dir}$, $\mathsf{r\text{-}chd}$, $\mathsf{r\text{-}sib}$ and r are the smallest sets satisfying the following recursive equations:

$$\begin{aligned}
\mathsf{r\text{-}dir} &= \{v \xrightarrow{a} w \in \mathcal{M} \mid \forall x \succ v.\ \mathsf{r\text{-}dir} \cap \mathsf{re\text{-}dir}(a,x) \neq \emptyset\} \\
\mathsf{r\text{-}chd} &= \{v \xrightarrow{a} w \in \mathcal{M} \mid \forall x \succ v.\ \mathsf{r\text{-}chd} \cap (\mathsf{re\text{-}dir}(a,x) \cup \mathsf{re\text{-}chd}(a,x)) \neq \emptyset\} \\
\mathsf{r\text{-}sib} &= \{v \xrightarrow{a} w \in \mathcal{M} \mid \forall x \succ v.\ \mathsf{r\text{-}sib} \cap (\mathsf{re\text{-}dir}(a,x) \cup \mathsf{re\text{-}sib}(a,x)) \neq \emptyset\} \\
\mathsf{r} &= \{v \xrightarrow{a} w \in \mathcal{M} \mid \forall x \succ v.\ \mathsf{r} \cap \mathsf{re}(a,x) \neq \emptyset\}\ .
\end{aligned}$$

Clearly, $\mathsf{r\text{-}dir} \subseteq \mathsf{r\text{-}chd} \cap \mathsf{r\text{-}sib}$ and $\mathsf{r\text{-}dir} \cup \mathsf{r\text{-}chd} \cup \mathsf{r\text{-}sib} \subseteq \mathsf{r}$; also, if v does not have children then $v \xrightarrow{a} w \in \mathsf{r\text{-}dir}$ for all $v \xrightarrow{a} w \in \mathcal{M}$. The latter observation will form the base case of our induction proofs.

Lemma 2 (syntactic reflection implies model reflection). *If $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ is a condition tree morphism, then all $a \in \mathsf{r}_{\mathcal{M}}$ reflect models.*

Proof. By induction, using the fact that r is the smallest set satisfying the equation above, which implies that $\mathsf{r} = \bigcup_{i \geq 0} \mathsf{r}^i$ where

$$\begin{aligned}
\mathsf{r}^0 &= \{v \xrightarrow{a} w \in \mathcal{M} \mid \neg\exists x.\ x \succ v\} \\
\mathsf{r}^{i+1} &= \{v \xrightarrow{a} w \in \mathcal{M} \mid \forall x \succ v.\ \mathsf{r}^i \cap \mathsf{re}(a,x) \neq \emptyset\}\ .
\end{aligned}$$

Consider $v \xrightarrow{a} w \in \mathsf{r}$ and let $g \models \hat{w}$; we set out to prove that $a^{\mathsf{I}}; g \models \hat{v}$. Let $h\colon P_w \to G$ be the witness for $g \models \hat{w}$. It follows that $a^{\mathsf{I}}; g = a^{\mathsf{I}}; d_w; h = d_v; a^{\mathsf{P}}; h$, hence $a^{\mathsf{P}}; h$ is a prospective witness for $a^{\mathsf{I}}; g \models \hat{v}$.

Assume that $a \in \mathsf{r}^i$ and the property has been proved for all $a \in \mathsf{r}^j$ with $j < i$. If $i = 0$, then v has no children, meaning that there is nothing left to prove. Otherwise, let $x \succ v$. By definition of r^i, there is some $b \in \mathsf{r}^{i-1} \cap \mathsf{re}(a, x)$. By the induction hypothesis, b preserves models; hence one of the clauses of lemma 1 applies. It follows that either $u_x; a^{\mathsf{P}}; h \not\models \hat{x}$ or (in case of Clause 3) poresibly there is a sibling v' of v with a morphism $s : I_{v'} \to I_v$ such that $u_{v'} = s; u_v$ and $s; a^{\mathsf{I}}; g \models \hat{v}'$. In either case a reflects g.

Corollary 1. *If $\mathcal{M}$ is a condition tree morphism, then $rt_{\mathcal{M}} \in \mathsf{r}$ implies that $\mathcal{M}$ reflects models.*

4 Model preservation

In order to reason concisely about model preservation, we call a graph morphism $f : I \to P$ *e-prefixed*, for some graph morphism $e : I \to P'$, if $f = e; f'$ for some graph morphism $f' : P' \to P$. In practice, we often use this for e that are epi, in which case f' is in fact uniquely defined.

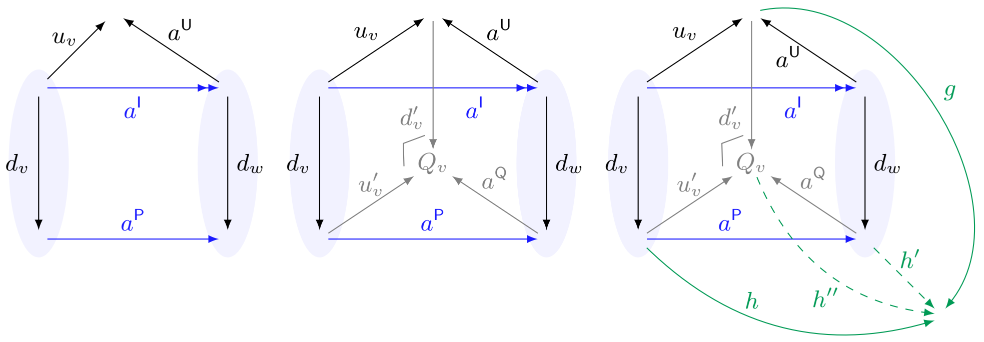
To define model preservation, we have to take into account that (as observed earlier) a triple $v \xrightarrow{a} w$ in general cannot be expected to carry over any $\hat{v}$-model g to some $\hat{w}$-model: rather, g should at least be a^{I}-prefixed for this to make sense, at which point the $\hat{w}$-model is the 'remnant" of g after a^{I}. In fact, we go one step further and only consider preservation for u_v-prefixed models of $\hat{v}$, where u_v itself is required to be a^{I}-prefixed. (Such a u_v-prefixed model is thus certainly an a^{I}-prefixed model.) This additional complication is required to make our induction proofs work.

Definition 11 (model preservation). *Let $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be a morphism.*

- $v \xrightarrow{a} w \in \mathcal{M}$ preserves a $\hat{v}$-model g *if $g = a^{\mathsf{I}}; g'$ implies $g' \models \hat{w}$.*
- $v \xrightarrow{a} w \in \mathcal{M}$ preserves models *if a^{I} is epi, u_v is a^{I}-prefixed, and $v \xrightarrow{a} w$ preserves all u_v-prefixed $\hat{v}$-models.*
- *$\mathcal{M}$ preserves models if $rt_{\mathcal{M}}$ preserves models.*

We will use $\mathcal{M}_{\mathsf{p}} \subseteq \mathcal{M}$ to denote the subset of triples $v \xrightarrow{a} w \in \mathcal{M}$ such that a^{I} is epi and u_v is a^{I}-prefixed. If $v \xrightarrow{a} w \in \mathcal{M}_{\mathsf{p}}$, then the remnant of u_v after a^{I} is uniquely defined, and it is useful to have a default notation for it. Henceforth, we write a^{U} for the unique graph morphism such that $u_v = a^{\mathsf{I}}; a^{\mathsf{U}}$. For a visualisation see Figure 3a.

Note that for the root triple $(v, a, w) = rt_{\mathcal{M}}$, we have $a^{\mathsf{I}} = id_I$, which is epi and implies that $g = a^{\mathsf{I}}; g'$ if and only if $g = g'$; and also $u_v = id_I$, which is itself a^{I}-prefixed and implies that every $\hat{v}$-model is u_v-prefixed. It follows that $rt_{\mathcal{M}} \in \mathcal{M}_{\mathsf{p}}$ and model preservation (of $\mathcal{M}$) simply means that all $\mathcal{T}$-models are also $\mathcal{U}$-models, as expected. Like for model reflection, we prove model preservation for morphisms with a certain "syntactic" structure. First, we introduce the concept of a *fusion*, illustrated in Figure 3b.



(a) a^{I} is epi, u_v is a^{I}-prefixed

(b) $a \in \mathsf{fuse}$

(c) "Only if" of Lemma 3

Fig. 3: Various preservation-related visualisations

Definition 12. *Let $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be a condition tree morphism. A triple $v \xrightarrow{a} w \in \mathcal{M}_{\mathsf{p}}$ is a* fusion *if the pushout of u_v over d_v is a^{P}-prefixed. The set of fusions is denoted* fuse.

The term *fusion* is chosen because, essentially, w tests for the same structure as v, except that some of the "parent" pattern (i.e., the target graph of u_v) is *fused* into the pattern P_w.

Fusions satisfy the following technical property:

Lemma 3. *If $v \xrightarrow{a} w \in \mathsf{fuse}$, then $u_v; g = d_v; h$ if and only if $a^{\mathsf{U}}; g = d_w; h'$ for some h' with $h = a^{\mathsf{P}}; h'$.*

This can be interpreted as follows: if we ignore the children of v and w, then $u_v; g$ is a $\hat{v}$-model with witness h if and only if $a^{\mathsf{U}}; g$ is a $\hat{w}$-model with some witness h' such that $h = a^{\mathsf{P}}; h'$.

Proof. Let $v \xrightarrow{a} w \in \mathsf{fuse}$, meaning that a^{I} is epi and we have a commuting diagram as in Figure 3b.

If. Assume $a^{\mathsf{U}}; g = d_w; h'$ and let $h = a^{\mathsf{P}}; h'$; then $u_v; g = a^{\mathsf{I}}; a^{\mathsf{U}}; g = a^{\mathsf{I}}; d_w; h' = d_v; a^{\mathsf{P}}; h' = d_v; h$.

Only if. (Visualised in Figure 3c.) Assume $u_v; g = d_v; h$; then by the pushout property, there is a unique morphism h'' such that $d'_v; h'' = g$ and $u'_v; h'' = h$. Let $h' = a^{\mathsf{Q}}; h''$; then the latter equation implies $h = u'_v; h'' = a^{\mathsf{P}}; a^{\mathsf{Q}}; h'' = a^{\mathsf{P}}; h'$. It follows that

$$a^{\mathsf{I}}; a^{\mathsf{U}}; g = u_v; g = u_v; d'_v; h'' = d_v; u'_v; h'' = d_v; h = d_v; a^{\mathsf{P}}; h' = a^{\mathsf{I}}; d_w; h' \enspace .$$

Since a^{I} is epi, we may conclude $a^{\mathsf{U}}; g = d_w; h'$.

Definition 13 (preservation evidence). *Let $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be a condition tree morphism and let $v \xrightarrow{a} w \in \mathcal{M}$.*

- (Direct preservation evidence.) *Let* $y \succ w$. *A triple* $y \xrightarrow{b} x$ *provides* direct preservation evidence for y, *denoted* $b \in \mathsf{pe\text{-}dir}(y, a)$, *if* $x \succ v$ *and* $b^{\mathsf{I}}; u_x; a^{\mathsf{P}} = u_y$.
- (Universal preservation evidence.) *A triple* $y \xrightarrow{b} w$ *provides* universal preservation evidence for a, *denoted* $b \in \mathsf{pe\text{-}univ}(a)$, *if* $y \succ x \succ v$ *for some* x *and there is a mediating morphism* $k\colon I_y \to I_x$ *such that* $k; d_x = u_y$ *and* $k; u_x; a^{\mathsf{P}} = b^{\mathsf{I}}; d_w$.

The visualisation is the same as for r-chd, see Figure 2

Alternatively and more succinctly, we can define these sets as follows. Let $v \xrightarrow{a} w \in \mathcal{M}$, $y \succ w$ and $z \succ x \succ v$:

$$\begin{aligned} \mathsf{pe\text{-}dir}(a, y) &= \{y \xrightarrow{b} x \in \mathcal{M} \mid x \succ v, b^{\mathsf{I}}; u_x; a^{\mathsf{P}} = u_y\} \\ \mathsf{pe\text{-}univ}(a) &= \{y \xrightarrow{b} w \in \mathcal{M} \mid \exists k\colon I_y \to I_x.\ k; d_x = u_y \wedge k; u_x; a^{\mathsf{P}} = b^{\mathsf{I}}; d_w\} \end{aligned}$$

The following lemma captures the essential properties.

Lemma 4. *Let* $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ *be a condition tree morphism, and let* $v \xrightarrow{a} w \in \mathcal{M}_{\mathsf{p}}$.

1. *Assume* $a \in \mathsf{fuse}$, *let* $u_v; g \models \hat{v}$ *with witness* h, *and let* $y \succ w$. *If there is some* $y \xrightarrow{b} x \in \mathsf{pe\text{-}dir}(a, y)$ *that preserves models, then* $b^{\mathsf{I}}; u_x; h \not\models \hat{y}$.
2. *If* $a \in \mathsf{fuse}$ *and for all* $y \succ w$ *there is some* $y \xrightarrow{b} x \in \mathsf{pe\text{-}dir}(a, y)$ *that preserves models, then* $v \xrightarrow{a} w$ *preserves models.*
3. *If there is some* $x \succ v$ *such that for all* $y \succ x$ *there is some* $y \xrightarrow{b} v \in \mathsf{pe\text{-}univ}(a)$ *that preserves models, then* $v \xrightarrow{a} w$ *preserves models.*

Proof.

To be done

Let $\mathsf{p} \subseteq \mathcal{M}$ be the smallest set satisfying the following recursive equation:

$$\begin{aligned} \mathsf{p} = {} & \{v \xrightarrow{a} w \in \mathcal{M}_{\mathsf{p}} \mid \forall y \succ w.\ \mathsf{p} \cap \mathsf{pe\text{-}dir}(a, y) \neq \emptyset\} \\ & \cup \{v \xrightarrow{a} w \in \mathcal{M}_{\mathsf{p}} \mid \exists x \succ v.\ \forall y \succ x.\ \mathsf{p} \cap \mathsf{pe\text{-}univ}(a) \neq \emptyset\} \ . \end{aligned}$$

The following can now be proved by induction, analogously to Lemma 2.

Lemma 5 (syntactic preservation implies model preservation). *If* $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ *is a condition tree morphism, then all* $a \in \mathsf{p}_{\mathcal{M}}$ *preserve models.*

Corollary 2. *If* $\mathcal{M}$ *is a condition tree morphism, then* $rt_{\mathcal{M}} \in \mathsf{p}$ *implies that* $\mathcal{M}$ *preserves models.*

5 Examples

In this section we give various examples of the concepts introduced earlier.

6 Typing morphisms

In the previous section after introducing the category **CT** of nested conditions we discussed several structural properties that morphisms of **CT** may satisfy. In this section we will consider several wide sub-categories of **CT**, i.e. categories having all the objects of **CT** but only some proper subsets of morphisms, and we will relate them by obvious functors.

The properties of morphisms we have considered are structural, thus in a sense syntactical. But the notion of satisfaction of conditions naturally induces a relation of *entailment* among conditions having the same root, that gives rise to a "semantic" category, $\mathbf{CT}^{\models}$, which is actually a preoder.

Definition 14 (semantic entailment and equivalence). *Given two conditions $\mathcal{T}$ and $\mathcal{U}$ having the same root, we define* semantic entailment $\mathcal{T} \models \mathcal{U}$ *and* semantic equivalence $\mathcal{T} \equiv \mathcal{U}$ *as follows:*

$$\mathcal{T} \models \mathcal{U} \text{ if for all arrows } g\text{: } g \models \mathcal{T} \text{ implies } g \models \mathcal{U}$$
$$\mathcal{T} \equiv \mathcal{U} \text{ if for all arrows } g\text{: } g \models \mathcal{T} \text{ if and only if } g \models \mathcal{U} \ .$$

Category $\mathbf{CT}^{\models}$ has the same objects of **CT***, and for each pair of conditions $\mathcal{T}, \mathcal{U}$ one arrow from $\mathcal{T}$ to $\mathcal{U}$ iff $\mathcal{T} \models \mathcal{U}$. Since there is at most one arrow between two objects, $\mathbf{CT}^{\models}$ is a preorder.*

One main concern will be to relate the syntactic categories of conditions, based on structural morphisms, with the semantic category $\mathbf{CT}^{\models}$.

In order to deal with the several kinds of morphisms introduced before, we will consider a typing system over arrows of **CT**. For a property *prop* on such arrows, we will write $(\mathcal{M}\colon \mathcal{T} \to \mathcal{V})\colon$*prop* to state that arrow $\mathcal{M}$ satisfies it, or simply $\mathcal{M}\colon$*prop* if the source and the target conditions of $\mathcal{M}$ are understood. We will summarize various results relating properties of arrows through inference rules, with the usual meaning: if the typing judgements of the premises are true then also the typing judgement in the consequence is provably true. Therefore every inference rule represents a proof obligation.

Let us start introducing the properties of interest.

Definition 15 (properties of arrows of CT).

Let $\mathcal{T}$ and $\mathcal{U}$ be conditions with the same root ($I_{\mathcal{T}} = I_{\mathcal{U}}$) and $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ be an arrow of **CT**. Then we will write:

[r-dir] $\mathcal{M}\colon$r-dir if $\mathcal{M}$ is source-saturated using only the direct rule;
[r-chd] $\mathcal{M}\colon$r-chd if $\mathcal{M}$ is source-saturated using the direct and the child rules (but not the sibling rule);
[r-sib] $\mathcal{M}\colon$r-sib if $\mathcal{M}$ is source-saturated using the direct and the sibling rules (but not the child rule);
[p-dir] $\mathcal{M}\colon$p-dir if $\mathcal{M}$ is target-saturated using only the direct rule;
[p-chd] $\mathcal{M}\colon$p-chd if $\mathcal{M}$ is target-saturated using the direct and the child rules;

Based on the results of the previous section, we can state the following facts.

Proposition 9 (relations among properties of CT arrows). *The following implications, presented as inference rules, hold for all arrows $\mathcal{M}\colon \mathcal{T} \to \mathcal{V}$ and $\mathcal{N}\colon \mathcal{V} \to \mathcal{U}$:*

$$\frac{}{\mathcal{I}_{\mathcal{T}}\colon \textsf{r-dir}}\ [1] \quad \frac{\mathcal{M}\colon \textsf{r-dir}}{\mathcal{M}\colon \textsf{r-chd}}\ [2] \quad \frac{\mathcal{M}\colon \textsf{r-dir}}{\mathcal{M}\colon \textsf{r-sib}}\ [3] \quad \frac{\mathcal{M}\colon \textsf{r-chd} \quad \mathcal{N}\colon \textsf{r-chd}}{\mathcal{M};\mathcal{N}\colon \textsf{r-chd}}\ [4] \quad \frac{\mathcal{M}\colon \textsf{r-sib} \quad \mathcal{N}\colon \textsf{r-sib}}{\mathcal{M};\mathcal{N}\colon \textsf{r-sib}}\ [5]$$

$$\frac{}{\mathcal{I}_{\mathcal{T}}\colon \textsf{p-dir}}\ [9] \quad \frac{\mathcal{M}\colon \textsf{p-dir}}{\mathcal{M}\colon \textsf{p-chd}}\ [10] \quad \frac{\mathcal{M}\colon \textsf{p-chd} \quad \mathcal{N}\colon \textsf{p-chd}}{\mathcal{M};\mathcal{N}\colon \textsf{p-chd}}\ [11]$$

Definition 16 (properties rfl and prs). *The properties* rfl *(provably reflects satisfaction) and* prs *(provably preserves satisfaction) are defined by the following rules:*

$$\frac{\mathcal{M}\colon \textsf{r-chd}}{\mathcal{M}\colon \textsf{rfl}}\ [6] \quad \frac{\mathcal{M}\colon \textsf{r-sib}}{\mathcal{M}\colon \textsf{rfl}}\ [7] \quad \frac{\mathcal{M}\colon \textsf{rfl} \quad \mathcal{N}\colon \textsf{rfl}}{\mathcal{M};\mathcal{N}\colon \textsf{rfl}}\ [8]$$

$$\frac{\mathcal{M}\colon \textsf{p-chd}}{\mathcal{M}\colon \textsf{prs}}\ [12] \quad \frac{\mathcal{M}\colon \textsf{prs} \quad \mathcal{N}\colon \textsf{prs}}{\mathcal{M};\mathcal{N}\colon \textsf{prs}}\ [13]$$

The properties just summarized allow us to define easily some wide subcategories of **CT**.

Definition 17 (subcategories of CT).

- $\mathbf{CT}_{\textsf{r-chd}}$ *includes all arrows $\mathcal{M}$ such that $\mathcal{M}\colon \textsf{r-chd}$;*
- $\mathbf{CT}_{\textsf{r-sib}}$ *includes all arrows $\mathcal{M}$ such that $\mathcal{M}\colon \textsf{r-sib}$;*
- $\mathbf{CT}_{\textsf{rfl}}$ *includes all arrows $\mathcal{M}$ such that $\mathcal{M}\colon \textsf{rfl}$;*
- $\mathbf{CT}_{\textsf{p-chd}}$ *includes all arrows $\mathcal{M}$ such that $\mathcal{M}\colon \textsf{p-chd}$;*
- $\mathbf{CT}_{\textsf{prs}}$ *includes all arrows $\mathcal{M}$ such that $\mathcal{M}\colon \textsf{prs}$.*

The above categories are well defined: in fact each class of arrows considered includes identities and is closed under composition. The unit and associativity laws are automatically satisfied because all arrows belong to category **CT**.

Proposition 10 (functors among categories of condition trees). *There are obvious inclusion functors*

- $\mathbf{CT}_{\textsf{r-chd}} \hookrightarrow \mathbf{CT}_{\textsf{rfl}}$
- $\mathbf{CT}_{\textsf{r-sib}} \hookrightarrow \mathbf{CT}_{\textsf{rfl}}$
- $\mathbf{CT}_{\textsf{p-chd}} \hookrightarrow \mathbf{CT}_{\textsf{prs}}$
- $\mathbf{CT}_{\textsf{rfl}} \hookrightarrow \mathbf{CT}$
- $\mathbf{CT}_{\textsf{prs}} \hookrightarrow \mathbf{CT}$

Furthermore, the following functors relate syntactic categories of structural morphisms among condition trees with the preorder of entailment among conditions:

- $\mathbf{CT}^{op}_{\mathsf{rfl}} \to \mathbf{CT}^{\models}$
- $\mathbf{CT}_{\mathsf{prs}} \to \mathbf{CT}^{\models}$

The last two functors state a form of soundness: (1) if there is an rfl-arrow from $\mathcal{U}$ to $\mathcal{T}$, then $\mathcal{T} \models \mathcal{U}$, and (2) if there is a prs-arrow from $\mathcal{T}$ to $\mathcal{U}$, then $\mathcal{T} \models \mathcal{U}$. The main goal of the next section is enrich this "soundness" result to achieve "completeness". That is, we introduce further syntactic categories of condition trees such that there is an arrow between two conditions *if and only if* the second entails the first one.

6.1 Path categories of condition trees

Definition 18 (path category of condition trees). *Let $\mathbf{CT}^*$ be the category having all the objects of $\mathbf{CT}$, and undirected paths of arrows of $\mathbf{CT}$ as arrows, denoted $\mathcal{P}\colon \mathcal{T} \rightsquigarrow \mathcal{U}$.*

More explicitly, undirected paths between objects are defined by the following rules, where if $\mathcal{M}\colon \mathcal{T} \to \mathcal{U}$ is an arrow of $\mathbf{CT}$ by $\mathcal{M}^{\mathrm{op}} : \mathcal{U} \rightsquigarrow \mathcal{T}$ we denote the correponding path going the other way around:

$$\frac{\mathcal{I}_{\mathcal{T}}\colon \mathcal{T} \to \mathcal{T}}{\mathcal{I}_{\mathcal{T}}\colon \mathcal{T} \rightsquigarrow \mathcal{T}} \qquad \frac{\mathcal{M}\colon \mathcal{T} \to \mathcal{U}}{\mathcal{M}\colon \mathcal{T} \rightsquigarrow \mathcal{U}} \qquad \frac{\mathcal{M}\colon \mathcal{T} \to \mathcal{U}}{\mathcal{M}^{\mathrm{op}}\colon \mathcal{U} \rightsquigarrow \mathcal{T}} \qquad \frac{\mathcal{P}\colon \mathcal{T} \rightsquigarrow \mathcal{U} \quad \mathcal{Q}\colon \mathcal{U} \rightsquigarrow \mathcal{V}}{\mathcal{P} \cdot \mathcal{Q}\colon \mathcal{T} \rightsquigarrow \mathcal{V}}$$

Paths are subject to the obvious equations making $\mathbf{CT}^*$ a category:

$$\mathcal{I}_{\mathcal{T}} \cdot \mathcal{P} = \mathcal{P} = \mathcal{P} \cdot \mathcal{I}_{\mathcal{U}} \quad \textit{if} \quad \mathcal{P}\colon \mathcal{T} \rightsquigarrow \mathcal{U}$$
$$\mathcal{P} \cdot (\mathcal{Q} \cdot \mathcal{R}) = (\mathcal{P} \cdot \mathcal{Q}) \cdot \mathcal{R}$$

AC: there should be a general name for this kind of operation

The $\_^{\mathrm{op}}$ operation extends to paths in the obvious way, so that if $\mathcal{P}\colon \mathcal{T} \rightsquigarrow \mathcal{U}$ then $\mathcal{P}^{\mathrm{op}}\colon \mathcal{U} \rightsquigarrow \mathcal{T}$:

$$\mathcal{I}_{\mathcal{T}}^{\mathrm{op}} = \mathcal{I}_{\mathcal{T}} \qquad (\mathcal{M}^{\mathrm{op}})^{\mathrm{op}} = \mathcal{M} \qquad (\mathcal{P} \cdot \mathcal{Q})^{\mathrm{op}} = \mathcal{Q}^{\mathrm{op}} \cdot \mathcal{P}^{\mathrm{op}}$$

We extend now the properties of Definition 16 to arrows of $\mathbf{CT}^*$.

Definition 19 (properties of paths).

$$\frac{(\mathcal{M}\colon \mathcal{T} \to \mathcal{U})\colon \mathsf{rfl}}{(\mathcal{M}\colon \mathcal{T} \rightsquigarrow \mathcal{U})\colon \mathsf{rfl}} \qquad \frac{(\mathcal{M}\colon \mathcal{T} \to \mathcal{U})\colon \mathsf{prs}}{(\mathcal{M}^{\mathrm{op}}\colon \mathcal{U} \rightsquigarrow \mathcal{T})\colon \mathsf{rfl}} \qquad \frac{\mathcal{P}\colon \mathsf{rfl} \quad \mathcal{Q}\colon \mathsf{rfl}}{\mathcal{P} \cdot \mathcal{Q}\colon \mathsf{rfl}} \qquad \frac{\mathcal{P}\colon \mathsf{rfl}}{\mathcal{P}^{\mathrm{op}}\colon \mathsf{prs}}$$

Furhter, let $\mathbf{CT}^*_{\mathsf{prs}}$ be the wide subcategory of $\mathbf{CT}^*$ containing as arrows all the paths $\mathcal{P}$ such that $\mathcal{P}\colon \mathsf{prs}$, and similarly $\mathbf{CT}^*_{\mathsf{rfl}}$ be the one containing all paths $\mathcal{P}$ such that $\mathcal{P}\colon \mathsf{rfl}$.

Theorem 2 (soundness and completeness). *There are identity-on-objects functors $(\mathbf{CT}^*_{\mathsf{rfl}})^{\mathrm{op}} \to \mathbf{CT}^{\models}$ and $\mathbf{CT}^*_{\mathsf{prs}} \to \mathbf{CT}^{\models}$. Furthermore, both functors are sujective on arrows.*

That is, for conditions $\mathcal{T}, \mathcal{U}$ having the same root, if $\mathcal{T} \models \mathcal{U}$ then there is an undirected path $\mathcal{P} : \mathcal{U} \rightsquigarrow \mathcal{T}$satisfying property rfl, ...