Intuitionistic Reasoning on Hybrid Xpath with Data

Danae Dutto[1]

Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
ddutto@dc.exa.unrc.edu.ar

Abstract. We investigate a fragment of XPath with attribute value (in)equality checks from the perspective of intuitionistic hybrid logic. Our starting point is the work in [2]. There, the same fragment is explored from the perspective of (classical) hybrid logic resulting in a logic we call HXPathD. We replace the classical hybrid logic basis for an intuitionistic one. From this, we obtain a new logic that allows for an intuitionistic reading of the navigational fragment of the XPath with attribute value (in)equality checks. We present the class of models of the new logic and axiomatizability and completeness results.

Keywords: XPath, Hybrid Logic, Intuitionistic Logic, Completeness

1 Introduction

XPath is one of the most popular languages to query and transform XML documents. The language has the ability to navigate a document, check for attribute values, as well as selecting nodes, computing values and processing the content of a document. Here, we focus on the navigational fragment with attribute value (in)equality checks. This fragment allows us to traverse an XML document and compare if attribute values at the end of traversed paths are (or not) equal to each other. In particular, we investigate this fragment from the perspective of intuitionistic hybrid logic [4].

The starting point of our work are the ideas and results reported in [2]. There, the mentioned fragment of XPath is explored from the perspective of classical hybrid logic [1] resulting in a logic we call HXPathD. In HXPathD, the topology of an XML document is formally captured as a relational structure, whereas attribute values (in)equality checks are formally captured as equivalence relations (or complements thereof). This results in a view of XML documents as models of the logic. In turn, the query language of XPath are internalized in a logical language comprised of path expressions (descriptions of how to traverse an XML document) and (in)equality expressions (capturing attribute value (in)equality checks). A query on an XML document can then be seen logically as evaluating a path or (in)equality expression on a model. In [2] it is shown that HXPathD is complete with respect to a proposed axiomatic system, and decidability and satisfiability results are provided. An advantage of the approach taken in [2], in particular in relation to axiomatizability and completeness, is the use of elements from hybrid logic: nominals and the satisfiability operator. These allow for the obtention of a relatively simple axiom system, and accommodate for a Henkin-style proof of completeness. Moreover, they allow for the obtention of more general completeness results for extensions of the logic with pure axioms and existential saturation rules.

We replace the classical basis HXPathD for an intuitionistic one [4]. This replacement yields a new logic we call IHXPathD. The language of IHXPathD is the same as that of HXPathD. The models of IHXPathD take elements from those in HXPathD and are insipired by those found in [4]. In brief, they can be seen as a partially ordered collection of models of HXPathD (which further satisfies some intuitionistic constraints). The caveat: we make a finer grained distinction for (in)equality checks. In concrete, we distinguish equality from inequality — capturing the former as an equivalence relation and the latter as a discrepancy relation [7]. This distinction enables an intuitionistic reading of (in)equality checks and brings into the picture elements from the treatment of equality in first-order intuitionistic logic [5]. For IHXPathD, we present an axiom system and axiomatizability and completeness results. In greater detail, the axiom system for IHXPathD combines: elements from the axiom system for HXPathD (those that correspond to axioms for path expressions and equality checks) and elements from the axiom system for hybrid modal intuitionistic logic (those that correspond to an intuitionistic reading of modalities, nominal, and satisfiability; see [4]). We incorporate new axioms that correspond to the treatment of inequality as a discrepancy relation (i.e., a relation different from the complement of equality). We obtain the completeness of the axiom system for IHXPathD via the construction of a canonical model. In doing this, again, we use ideas present in [2,4] but reformulating and adapting them in order to make them fit into our setting. Finally, we briefly state how the axiom system can be extended with pure axioms and existential saturations rules to obtain a more general completeness result for corresponding classes of frames.

Motivation. We consider our work to be interesting from a logical perspective, but also in terms of modelling real world scenarios. From a logical perspective because it combines existing works in the area of modal approaches to XPath and intuitionistic modal logics (in particular [2] and [4]). From the point of view of modelling real world scenarios because of its potential application in the setting of querying graph databases. This last idea is one of the motivations behind [2]. We consider that an intuitionistic take opens the door to explore what occurs with comparisson with *null* values (are they equal to each other, different from each other, or neither?). Moreover, we consider that viewing the models of IHXPathD as collections of models of HXPathD opens the door to study notions of updates between models.

Results. We present a logic that provides an intuitionistic reading of the navigational fragment of the XPath with attribute values (in)equality checks. We define a class of intuitionistic modal models on which to interpret the language of HXPathD, provide an axiomatization and completeness results.

Structure. Sec. 2 introduces the language and semantics for IHXPathD. Sec. 3 presents its axiom system and the main ideas behind its completeness proof. Sec. 4 shows how to extend the axiom system with pure axioms and existential saturation rules to obtain a more general completeness result. Finally, Sec. 5 offers some final remarks and comment on future lines of research.

2 Intuitionistic Hybrid XPath with Data

We introduce the language and semantics of Intuitionistic Hybrid XPath with Data (IHXPathD for short). We use notation and terminology from [2].

Language. We fix an alphabet of countable sets Prop of symbols for propositions and Nom of symbols for nominals s.t. $\mathsf{Prop} \cap \mathsf{Nom} = \emptyset$. We fix also finite sets Mod of symbols for modalities and Cmp for data comparisons s.t. $\mathsf{Mod} \cap \mathsf{Cmp} = \emptyset$.

Definition 1 (Language). *We define the expressions of IHXPathD as either* path *(indicated α, β) or* node *(indicated φ, ψ) expressions via the grammar:*

$$\alpha, \beta := \mathsf{a} \mid @_i \mid [\varphi] \mid \alpha\beta$$
$$\varphi, \psi := p \mid i \mid \bot \mid \varphi \vee \psi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \langle \alpha * \beta \rangle \mid [\alpha * \beta]$$

for $p \in \mathsf{Prop}$, $i \in \mathsf{Nom}$, $\mathsf{a} \in \mathsf{Mod}$, $\mathsf{c} \in \mathsf{Cmp}$, and $* \in \{=_\mathsf{c}, \neq_\mathsf{c}\}$. We use PE *for the set of all path expressions, and* NE *for the set of all node expressions.*

We call $\langle \alpha * \beta \rangle$ or $[\alpha * \beta]$ *data comparison expressions*; and $[\varphi]$ *test*. We write $=$ and $\neq$ if the data comparison symbol c is not relevant; and $*$ if it is indistinct to use $=$ or $\neq$. We use $\neg\varphi = \varphi \rightarrow \bot$; and $\top = \neg\bot$; Moreover, we use $\epsilon = [\top]$; $\langle \alpha \rangle \varphi = \langle \alpha[\varphi] = \alpha[\varphi] \rangle$; $[\alpha]\varphi = [\alpha[\varphi] = \alpha[\varphi]]$; $@_i\varphi = \langle @_i \rangle \varphi$.

Semantics. We now define the structures and the notion of satisfiability that we use to evaluate expressions of IHXPathD.

Definition 2 (Models). *An intuitionistic hybrid abstract data model (a model) is a tuple*

$$\mathcal{M} = \langle M, \preccurlyeq, \{\langle \mathfrak{M}_m, \approx_m \rangle\}_{m \in M} \rangle$$

such that: $\langle M, \preccurlyeq \rangle$ is a poset; and for all m, $\approx_m$ is a congruence on

$$\mathfrak{M}_m = \langle W_m, \{R^\mathsf{a}_m\}_{\mathsf{a} \in \mathsf{Mod}}, \{\sim^\mathsf{c}_m\}_{\mathsf{c} \in \mathsf{Cmp}}, \{\not\sim^\mathsf{c}_m\}_{\mathsf{c} \in \mathsf{Cmp}}, g_m, V_m \rangle$$

where

1. *a non-empty set of nodes W_m;*
2. *for each $\mathsf{a} \in \mathsf{Mod}$, a binary relation R^a_m on W_m;*
3. *for each $\mathsf{c} \in \mathsf{Cmp}$, an equivalence relation $\sim^\mathsf{c}_m$ on W_m;*
4. *for each $\mathsf{c} \in \mathsf{Cmp}$, a discrepancy relation $\not\sim^\mathsf{c}_m$ on W_m which satisfies that: for $\{w, v, u\} \subseteq W_m$, $(w, w) \notin \not\sim^\mathsf{c}_m$; $(w, v) \in \not\sim^\mathsf{c}_m$ implies $(v, w) \in \not\sim^\mathsf{c}_m$; and $(w, u) \in \not\sim^\mathsf{c}_m$ implies $(w, v) \in \not\sim^\mathsf{c}_m$ or $(v, u) \in \not\sim^\mathsf{c}_m$;*
5. *functions $V_m : \mathsf{Prop} \rightarrow 2^{W_m}$ and $g_m : \mathsf{Nom} \rightarrow W_m$.*

Moreover, $\mathcal{M}$ satisfies the following sets of constraints.

monotonicity *for all* $m \preccurlyeq m'$*: (i)* $W_m \subseteq W_{m'}$*; (ii)* $\approx_m \subseteq \approx_{m'}$*; (iii)* $R^{\mathsf{a}}_m \subseteq R^{\mathsf{a}}_{m'}$*; (iv)* $\not\sim^{\mathsf{c}}_m \subseteq \not\sim^{\mathsf{c}}_{m'}$*; (v) for* $p \in \mathsf{Prop}$*,* $V_m(p) \subseteq V_{m'}(p)$*; and (vi) for* $i \in \mathsf{Nom}$*,* $g_m(i) = g_{m'}(i)$*.*

congruence *for all* $v \approx_m v'$ *and* $u \approx_m u'$*: (i)* $v \in V_m(p)$ *implies* $v' \in V_m(p)$*, (ii)* $vR^{\mathsf{a}}_m u$ *implies* $v'R^{\mathsf{a}}_m u'$*, (iii)* $v \sim^{\mathsf{c}}_m u$ *implies* $v' \sim^{\mathsf{c}}_m u'$*, (iv)* $v \not\sim^{\mathsf{c}}_m u$ *implies* $v' \not\sim^{c}_m u'$*.*

disjointness *for all* $w \sim^{\mathsf{c}}_m v$ *and* $v \not\sim^{\mathsf{c}}_m u$ *implies* $w \not\sim^{\mathsf{c}}_m u$*.*

The notion of a model in Def. 2 deserves a brief explanation. Intuitively, $\mathcal{M}$ may be seen as a collection of the hybrid abstract data models proposed in [2] partially ordered by inclusion. More precisely, borrowing from the terminology used in [4], the idea is that each $m \in M$ represents a 'state of knowledge', indicated by $\mathfrak{M}_m$, and that $\preccurlyeq$ preserves all kinds of knowledge. Each $\mathfrak{M}_m$ in $\mathcal{M}$ is a hybrid abstract data models insofar as *nodes* W_m, *accessibility* relations R^{a}_m, *data equality* relations $\sim^{\mathsf{c}}_m$, and the *valuation* V_m and the *naming* function g_m. The new ingredient corresponds to *data inequality* relations $\not\sim^{\mathsf{c}}_m$. The incorporation of $\not\sim^{\mathsf{c}}_m$ has to do with the treatment of negation of equality in an intuitionistic sense; which may not be simply the complement of equality. In turn, the relations $\approx_m$ correspond to the intuitionistic treatment of nominals (cf. [4]). This treatment yields a congruence on $\mathfrak{M}_m$.

We proceed to define the notion of satisfiability of a node and path expression in a model.

Definition 3 (Satisfiability). *Let* $\mathcal{M}$ *be a model,* m *and* m' *elements of* M*, and* w *and* w' *elements of* W_m*. We define the relation* $\Vdash$ *by induction as follows:*

$$\begin{array}{lll}
\mathcal{M}, m, w, w' \Vdash \mathsf{a} & \textit{iff} & wR^{\mathsf{a}}_m w' \\
\mathcal{M}, m, w, w' \Vdash @_i & \textit{iff} & g_m(i) \approx_m w' \\
\mathcal{M}, m, w, w' \Vdash [\varphi] & \textit{iff} & w \approx_m w' \textit{ and } \mathcal{M}, m, w \Vdash \varphi \\
\mathcal{M}, m, w, w' \Vdash \alpha\beta & \textit{iff} & \textit{exists } v \in W_m \textit{ s.t. } \mathcal{M}, m, w, v \Vdash \alpha \textit{ and } \mathcal{M}, m, v, w' \Vdash \beta
\end{array}$$

$$\begin{array}{lll}
\mathcal{M}, m, w \Vdash p & \textit{iff} & w \in V_m(p) \\
\mathcal{M}, m, w \Vdash \bot & \textit{iff} & \textit{never} \\
\mathcal{M}, m, w \Vdash i & \textit{iff} & w \approx_m g_m(i) \\
\mathcal{M}, m, w \Vdash \varphi \wedge \psi & \textit{iff} & \mathcal{M}, m, w \Vdash \varphi \textit{ and } \mathcal{M}, m, w \Vdash \psi \\
\mathcal{M}, m, w \Vdash \varphi \vee \psi & \textit{iff} & \mathcal{M}, m, w \Vdash \varphi \textit{ or } \mathcal{M}, m, w \Vdash \psi \\
\mathcal{M}, m, w \Vdash \varphi \to \psi & \textit{iff} & \textit{for all } m \preccurlyeq m', \mathcal{M}, m', w \Vdash \varphi \textit{ implies } \mathcal{M}, m', w \Vdash \psi
\end{array}$$

$$\begin{array}{lll}
\mathcal{M}, m, w \Vdash \langle \alpha =_{\mathsf{c}} \beta \rangle & \textit{iff} & \textit{exists } u, v \in W_m \textit{ s.t.} \\
& & \quad \mathcal{M}, m, w, u \Vdash \alpha, \ \mathcal{M}, m, w, v \Vdash \beta \textit{ and } u \sim^{\mathsf{c}}_m v \\
\mathcal{M}, m, w \Vdash \langle \alpha \neq_{\mathsf{c}} \beta \rangle & \textit{iff} & \textit{exists } u, v \in W_m \textit{ s.t.} \\
& & \quad \mathcal{M}, m, w, u \Vdash \alpha, \ \mathcal{M}, m, w, v \Vdash \beta \textit{ and } u \not\sim^{\mathsf{c}}_m v \\
\mathcal{M}, m, w \Vdash [\alpha =_{\mathsf{c}} \beta] & \textit{iff} & \textit{for all } m \preccurlyeq m', \forall\ u, v \in W_{m'} \\
& & \quad \mathcal{M}, m', w, u \Vdash \alpha \ \textit{ and } \ \mathcal{M}, m', w, v \Vdash \beta \textit{ implies } u \sim^{\mathsf{c}}_v v \\
\mathcal{M}, m, w \Vdash [\alpha \neq_{\mathsf{c}} \beta] & \textit{iff} & \textit{for all } m \preccurlyeq m', \forall\ u, v \in W_{m'} \\
& & \quad \mathcal{M}, m', w, u \Vdash \alpha \ \textit{ and } \ \mathcal{M}, m', w, v \Vdash \beta \textit{ implies } u \not\sim^{\mathsf{c}}_v v.
\end{array}$$

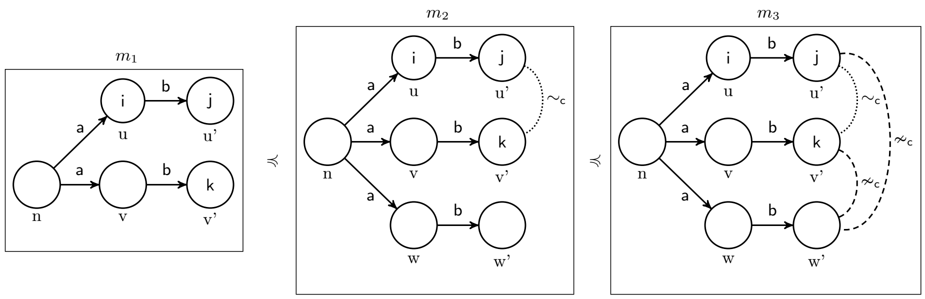

Fig. 1: Intuitionistic Hybrid Abstract Data Model.

Let $\Gamma \cup \{\varphi\}$ be a set of node exressions, we write $\Gamma \vDash \varphi$ iff for all $\mathcal{M}, m, w \Vdash \Gamma$, it follows that $\mathcal{M}, m, w \Vdash \varphi$. We write $\vDash \varphi$, and call φ a tautology*, when $\Gamma = \emptyset$.*

Let us provide a brief example of the concepts introduced thus far.

Example 1. Fig. 1 shows an example of a model $\mathcal{M}$ as per Def. 2. This model consists of three states of knowledge ordered sequentially. These states illustrate how knowledge increases from one state to the next. The state m_1 may be understood as our initial knowledge. This state contains five nodes linked by accessibility relations labelled with a and b. Some nodes in this state, i.e., u, u′, and v′, are named by the nominals i, j, and k. In this state, data equality relations are taken to contain simply the reflexive pairs of nodes. To avoid cluttering we avoided drawing the self-loops. In turn, data inequality relations are taken to be empty.

Taking m_1 as the initial state, we can understand m_2 as a new state in which we incorporated additional nodes and extended the accessibility relations. More interestingly, in m_2 we extended the data equality relation $\sim_{\mathsf{c}}$ with the addition of a new pair, i.e., that between the nodes named by j and k. This pair, together with its symmetric, is depicted by a dotted line. Finally, the state m_3 may be understood as an extension of m_2. In this case, the extension involves adding a data inequality between the node labelled by k and the node w′. The conditions on data inequality in $\mathcal{M}$ force us to further say that the node named by j and the node w′ do not have the same data. In the figure, data inequalities are depicted with a dashed link.

Let us now turn our attention to how to interpret path and node expressions on models. In particular, let us focus on what is the case in $\mathcal{M}$. To begin with, consider the path expression $\mathsf{ab}[\mathsf{j}]$. Intuitively, this path expression indicates a path composed of segments a and b, that ends in a test. We evaluate such a path expression in a model by indicating the current state of knowledge and the start and end points of the path. Then, notice that $\mathcal{M}, m_1, \mathrm{n}, \mathrm{u}' \Vdash \mathsf{ab}[\mathsf{j}]$. In this case, the test may be understood as 'is the end point of the path named by j?' We can also use the language to indicate the starting point of a path. For instance, the path expression $@_{\mathsf{i}}\mathsf{b}$ is intuitively read as a b segment starting at a node named by i. It follows that $\mathcal{M}, m_1, \mathrm{n}, \mathrm{u}' \Vdash @_{\mathsf{i}}\mathsf{b}$. In turn, a node expression $\langle \mathsf{ab}[\mathsf{j}] =_{\mathsf{c}} @_{\mathsf{k}} \rangle$ indicates that there are paths $\mathsf{ab}[\mathsf{j}]$ and $@_{\mathsf{k}}$ whose end points are equal (according to c). Node expressions are evaluated in a state of knowledge at a particular node. Then, we have $\langle \mathsf{ab}[\mathsf{j}] =_{\mathsf{c}} @_{\mathsf{k}} \rangle$ does not hold in any node in m_1, but it does hold in the node n in m_2 and m_3; i.e., $\mathcal{M}, m_1, \mathrm{n} \nVdash \langle \mathsf{ab}[\mathsf{j}] =_{\mathsf{c}} @_{\mathsf{k}} \rangle$; but $\mathcal{M}, m_2, \mathrm{n} \Vdash \langle \mathsf{ab}[\mathsf{j}] =_{\mathsf{c}} @_{\mathsf{k}} \rangle$ and $\mathcal{M}, m_3, \mathrm{n} \Vdash \langle \mathsf{ab}[\mathsf{j}] =_{\mathsf{c}} @_{\mathsf{k}} \rangle$. Data inequality is captured with a node expression such as $\langle \mathsf{ab} \neq_{\mathsf{c}} @_{\mathsf{i}}\mathsf{b} \rangle$. This node expression holds only in m_3.

We conclude this section with some expected properties of our models.

Proposition 1. *Let $\mathcal{M}$ be a model, $m \in M$, and $\{w, w', v, v'\} \subseteq W_m$:*
(1) $\mathcal{M}, m, w \Vdash \varphi$ and $w \approx_m w'$ implies $\mathcal{M}, m, w' \Vdash \varphi$;
(2) $\mathcal{M}, m, w \Vdash \varphi$ and $m \preccurlyeq m'$ implies $\mathcal{M}, m', w \Vdash \varphi$;
(3) $\mathcal{M}, m, w, v \Vdash \alpha$ and $w \approx_m w'$ and $v \approx_m v'$ implies $\mathcal{M}, m, w', v' \Vdash \alpha$;
(4) $\mathcal{M}, m, w, w' \Vdash \alpha$ and $m \preccurlyeq m'$ implies $\mathcal{M}, m', w, w' \Vdash \alpha$.

Proposition 2. *It holds that: (1) $\vDash \langle @_i =_{\mathsf{c}} @_j \rangle \to \neg \langle @_i \neq_{\mathsf{c}} @_j \rangle$. Moreover, (2) $\vDash \langle @_i \neq_{\mathsf{c}} @_j \rangle \to \neg \langle @_i =_{\mathsf{c}} @_j \rangle$. In turn, (1') $\nvDash \neg \langle @_i \neq_{\mathsf{c}} @_j \rangle \to \langle @_i =_{\mathsf{c}} @_j \rangle$; and (2') $\nvDash \neg \langle @_i =_{\mathsf{c}} @_j \rangle \to \langle @_i \neq_{\mathsf{c}} @_j \rangle$.*

Prop. 1 shows our models satisfy: monotonicity and congruence. Namely, if we take m as "what we know at some point", monotonicity states that "such a knowledge only increases". In turn, congruence states that two equivalent nodes satisfy the same expressions. In addition, Prop. 2 shows that equality and inequality are not complementary.

3 Axiomatization and Completeness

We introduce an axiom system for IHXPathD. It is inspired by the one for the Intuitionistic Hybrid Modal Logic in [4] and that for HXPathD in [2].

Axiom System. The axiom system for IHXPathD is comprised of the axiom schemata in Tab. 1 and the inference rules in Tab. 2. In this axiom system φ, ψ and θ are node expressions; α, β, γ and η are path expressions; and i, j and k are nominals. This axiom system gives rise to a Hilbert-style notion of deduction of a node expression φ from a set of node expressions Γ, written $\Gamma \vdash \varphi$, in the obvious way. The theorems and derived rules in Tab. 3 are obtained immediately.

Completeness. For soundness, we check that the axioms are valid and that the rules preserve validity. We obtain completeness by showing that consistent sets of node expressions are satisfiable in some model. We use a Henkin-style construction similar to that for Hybrid Logic with the satisfaction operator (see [6,3,4]).

Notation We use $\mathsf{PE}(\mathsf{Nom}')$, $\mathsf{NE}(\mathsf{Nom}')$, and $\mathsf{Exp}(\mathsf{Nom}')$ –a.k.a. PE', NE', and Exp'– for the paths, nodes, and expressions with nominals in a set Nom'.

Definition 4. *Let $\mathsf{Nom}' \subset \mathsf{Nom}''$; a set $\Gamma'' \subseteq \mathsf{NE}''$ is Nom''-saturated iff:*

Table 1: Axioms for IHXPathD

Basic	
(IPL)	Theorems of Intuitionistic Prop. Logic
(K)	$[\alpha](\varphi \to \psi) \to ([\alpha]\varphi \to [\alpha]\psi)$
Satisfaction	
($\mathrm{Distr}^{@}_{\wedge}$)	$@_i(\varphi \wedge \psi) \leftrightarrow (@_i\varphi \wedge @_i\psi)$
($\mathrm{Distr}^{@}_{\vee}$)	$@_i(\varphi \vee \psi) \leftrightarrow (@_i\varphi \vee @_i\psi)$
($\mathrm{Distr}^{@}_{\to}$)	$@_i(\varphi \to \psi) \leftrightarrow (@_i\varphi \to @_i\psi)$
(Falsum)	$@_i\bot \to \bot$
(Refl@)	$@_i i$
Comparisons	
($\mathrm{Refl}_{=}$)	$\langle @_i = @_i \rangle$
($\mathrm{Sym}_{\langle * \rangle}$)	$\langle \alpha * \beta \rangle \leftrightarrow \langle \beta * \alpha \rangle$
($\mathrm{Sym}_{[*]}$)	$[\alpha * \beta] \leftrightarrow [\beta * \alpha]$
(Trans)	$\langle @_i = @_j \rangle \wedge \langle @_j = @_k \rangle \to \langle @_i = @_k \rangle$
(Irefl)	$\neg\langle @_i \neq @_i \rangle$
(CoTrans)	$\langle @_i \neq @_j \rangle \to \langle @_i \neq @_k \rangle \vee \langle @_j \neq @_k \rangle$
($\mathrm{Disj}^{=}_{\neq}$)	$\langle @_i = @_j \rangle \wedge \langle @_j \neq @_k \rangle \to \langle @_i \neq @_k \rangle$

Paths	
(Cat)	$\langle \alpha \rangle \langle \beta \rangle \varphi \leftrightarrow \langle \alpha\beta \rangle \varphi$
(Id_ϵ)†	$\langle \alpha\beta * \gamma \rangle \leftrightarrow \langle \alpha\epsilon\beta * \gamma \rangle$
(Dist1)	$@_i\langle \alpha\beta * \gamma \rangle \leftrightarrow @_i(\langle \alpha \rangle \langle \beta * @_i\gamma \rangle)$
(Dist2)	$\langle \alpha \rangle \langle \beta * \gamma \rangle \to \langle \alpha\beta * \alpha\gamma \rangle$
(Dist3)	$\langle @_i\alpha * @_i\beta \rangle \to @_i\langle \alpha * \beta \rangle$
(Test)	$\langle [\psi]\alpha \rangle \varphi \leftrightarrow \psi \wedge \langle \alpha \rangle \varphi$
(Nom@)	$\langle \alpha[i]\beta * \gamma \rangle \to \langle @_i\beta * \gamma \rangle$
(Assoc)	$\langle (\alpha\beta)\gamma * \eta \rangle \leftrightarrow \langle \alpha(\beta\gamma) * \eta \rangle$
(Dist3)	$\langle \alpha\beta \rangle \varphi \to \langle \alpha \rangle \langle \beta \rangle \varphi$
(SPath)	$\langle \alpha\beta * \gamma \rangle \to \langle \alpha \rangle \top$
(Scope)	$\langle @_j\alpha * \beta \rangle \to \langle @_i @_j \alpha * \beta \rangle$
(Back)	$\langle \alpha @_i \beta * \gamma \rangle \to \langle @_i\beta * \gamma \rangle$
($\langle * \rangle$I)	$(\langle \alpha \rangle i \wedge \langle @_i * \beta \rangle) \to \langle \alpha * \beta \rangle$
($[*]$E)	$(\langle \alpha \rangle i \wedge \langle \beta \rangle j \wedge [\alpha * \beta]) \to \langle @_i * @_j \rangle$

† α or β can be omitted (but not both).

Table 2: Rules of Inference for IHXPathD

Basic

$$\frac{\varphi \quad \varphi \to \psi}{\psi}\ (\mathrm{MP}) \qquad \frac{\varphi}{[\alpha]\varphi}\ (\mathrm{NEC})$$

Satisfaction

$$\frac{\varphi}{@_i\varphi}\ (@\mathrm{I'}) \qquad \frac{@_i j \quad @_i\varphi}{@_j\varphi}\ (\mathrm{Nom})$$

$$\frac{@_i\varphi}{\varphi}\ (@\mathrm{E})^\dagger \qquad \frac{@_i j \quad @_i\langle \alpha \rangle k}{@_j\langle \alpha \rangle k}\ (\mathrm{Nom}_2)$$

† i does not occur in φ.

Paths

$$\frac{\varphi \to @_i\langle \alpha\beta * \gamma \rangle \wedge (\varphi \wedge @_i\langle \alpha \rangle j \wedge @_i\langle @_j\beta * \gamma \rangle) \to \psi}{\varphi \to \psi}\ (\langle * \rangle \mathrm{E})^\dagger$$

$$\frac{(\varphi \wedge @_i(\langle \alpha \rangle j \wedge \langle \beta \rangle k)) \to \langle @_j * @_k \rangle}{\varphi \to @_i[\alpha * \beta]}\ ([*]\mathrm{I})^\ddagger$$

† j does not occur in α, β, γ, χ nor ψ.
‡ j and k do not occur in α nor β.

Table 3: Theorems and Derived Rules.

Theorems

(Id@)	$@_i\langle @_j\alpha * @_k\beta \rangle \leftrightarrow \langle @_j\alpha * @_k\beta \rangle$
(Agree)	$@_i @_j \varphi \leftrightarrow @_j\varphi$

Derived Rules

$$\frac{\varphi \to \chi \quad \psi \to \chi}{(\varphi \vee \psi) \to \chi}\ (\vee\mathrm{E}) \qquad \frac{i \to \varphi}{\varphi}\ (\mathrm{named'})^\ddagger$$

‡ i does not appear in φ.

1. $\Gamma'' = \{ \varphi \mid \Gamma'' \vdash \varphi \} \subset \mathsf{NE}''$;
2. $@_i(\varphi \vee \psi) \in \Gamma''$ *implies* $@_i\varphi \in \Gamma''$ *or* $@_i\psi \in \Gamma''$;
3. *there is* $i \in \mathsf{Nom}''$ *such that* $i \in \Gamma''$;

4. $@_i\langle @_j \mathsf{a}\alpha * \beta\rangle \in \Gamma''$ *implies exists* $k \in \mathsf{Nom}''$ *s.t.* $\{@_j\langle \mathsf{a}\rangle k, @_i\langle @_k \alpha * \beta\rangle\} \subseteq \Gamma''$.

The conditions above have the following names: 1. $\vdash$*-closed; 2. the disjunction property; 3. named; 4. pasted.*

Lemma 1 (Extended Lindenbaum Lemma). *Let* $\mathsf{Nom}' \subset \mathsf{Nom}''$*; and consider* $\Gamma' \subseteq \mathsf{NE}'$ *s.t.* $\Gamma' \nvdash \psi$*. Then, there exists* $\Gamma'' \subseteq \mathsf{NE}''$ *such that: (1)* $\Gamma' \subseteq \Gamma''$*; (2)* Γ'' *is* Nom''*-saturated; and (3)* $\Gamma'' \nvdash \psi$*.*

Proof. Enumerate all node expressions in NE'' and let $k \in (\mathsf{Nom}'' \setminus \mathsf{Nom}')$ be the first nominal in this enumeration. Define $\Sigma_0 = \Gamma' \cup \{k\}$. Now, suppose that we have defined Σ_n, for $n \geq 0$. Let $\varphi_{(n+1)}$ be the $(n+1)^{\text{th}}$ node expression in the enumeration. If $\Sigma_n \cup \{\varphi_{(n+1)}\} \vdash \psi$, then, define $\Sigma_{(n+1)} = \Sigma_n$. Otherwise, i.e., if $\Sigma_n \cup \{\varphi_{(n+1)}\} \nvdash \psi$, then, define $\Sigma_{(n+1)} = \Sigma_n \cup \{\varphi_{(n+1)}\} \cup \Sigma'$ where:

$$\Sigma' = \begin{cases} \emptyset & \text{if } \varphi_{(n+1)} \notin \{@_i(\theta \vee \chi), @_i\langle \mathsf{a}\rangle\varphi, @_i\langle @_j \mathsf{a}\alpha * \beta\rangle\} \\ \{@_i\theta\} & \text{if } \varphi_{(n+1)} \text{ is } @_i(\theta \vee \chi) \text{ and } \Sigma_n \cup \{\varphi_{(n+1)}, @_i\theta\} \nvdash \psi \\ \{@_i\chi\} & \text{if } \varphi_{(n+1)} \text{ is } @_i(\theta \vee \chi) \text{ and } \Sigma_n \cup \{\varphi_{(n+1)}, @_i\theta\} \vdash \psi \\ \{@_j\langle \mathsf{a}\rangle k, @_i\langle @_k\alpha * \beta\rangle\} & \text{if } \varphi_{(n+1)} \text{ is } @_i\langle @_j \mathsf{a}\alpha * \beta\rangle \\ & \quad \text{and } k \in \mathsf{Nom}'' \text{ does not appear in } \Sigma_n \cup \{\varphi_{(n+1)}\}. \end{cases}$$

Let $\Sigma = \bigcup_{n \geqslant 0} \Sigma_n$. It follows by induction that $\Sigma \nvdash \psi$. The proof is concluded if Σ is saturated. We prove only the case for pasted.

(pasted) Let $@_i\langle @_j \mathsf{a}\alpha * \beta\rangle \in \Sigma$ and, w.l.o.g., $\varphi_{(n+1)} = @_i\langle @_j \mathsf{a}\alpha * \beta\rangle$. It follows that, $\{@_i\langle \mathsf{a}\rangle j, @_i\langle @_k\alpha * \beta\rangle\} \subseteq \Sigma_{(n+1)} \subseteq \Sigma$ for $j \in \mathsf{Nom}''$.

Lem. 1, states a crucial result: consistent sets can be extended to *saturated* sets (enriching the language with new symbols for nominals). This enables us to build the model needed for completeness (Def. 5).

Definition 5 (Extracted Model). *Let* $\{\mathsf{Nom}'_i\}_{i \in \mathbb{N}}$ *be a family of pairwise disjoint denumerable sets of nominals. Moreover, let* $\mathsf{Nom}^*_n = \bigcup_{i=1}^{n} \mathsf{Nom}'_i$ *and* $\mathsf{Exp}^*_n = \mathsf{Exp}(\mathsf{Nom} \cup \mathsf{Nom}^*_n)$*. For any consistent set* $\Gamma \subseteq \mathsf{NE}$*; define*

$$\mathcal{M}_\Gamma = \langle M, \subseteq, \{\langle \mathfrak{M}_{\Gamma'}, \approx_{\Gamma'}\rangle\}_{\Gamma' \in M}\rangle$$

where: $\langle W_{\Gamma'}, \{R^{\mathsf{a}}_{\Gamma'}\}_{\mathsf{a} \in \mathsf{Mod}}, \{\sim^{\mathsf{c}}_{\Gamma'}\}_{\mathsf{c} \in \mathsf{Cmp}}, \{\nsim^{\mathsf{c}}_{\Gamma'}\}_{\mathsf{c} \in \mathsf{Cmp}}, g_{\Gamma'}, V_{\Gamma'}\rangle$ *and*

1. $M = \{\, \Gamma' \subseteq \mathsf{Exp}^*_n \mid n \in \mathbb{N}$ *and* $\Gamma \subseteq \Gamma'$ *and* Γ' *is* Nom^*_n*-saturated* $\}$*;*
2. $W_{\Gamma'} = \{\, i \mid i$ *is a nominal appearing in* $\Gamma' \,\}$*;*
3. $\approx_{\Gamma'} = \{\, (i,j) \mid @_i j \in \Gamma' \,\}$ *is an equivalence relation;*
4. $R^{\mathsf{a}}_{\Gamma'} = \{\, (i,j) \mid @_i\langle \mathsf{a}\rangle j \in \Gamma' \,\}$ *is a binary relation;*
5. $\sim^{\mathsf{c}}_{\Gamma'} = \{\, (i,j) \mid @_i\langle \epsilon =_{\mathsf{c}} @_j\rangle \in \Gamma' \,\}$ *is an equivalence relation;*
6. $\nsim^{\mathsf{c}}_{\Gamma'} = \{\, (i,j) \mid @_i\langle \epsilon \neq_{\mathsf{c}} @_j\rangle \in \Gamma' \,\}$ *is a discrepancy relation;*
7. $V_{\Gamma'} : \mathsf{Prop} \to 2^{W_{\Gamma'}}$*, it follows that* $V_{\Gamma'}(p) = \{\, i \mid @_i p \in \Gamma' \,\}$*; and*
8. $g_{\Gamma'} : \mathsf{Nom} \to W_{\Gamma'}$*, it follows that* $g_{\Gamma'}(i) = i$*.*

Lastly, we state Lem. 2 and Thm. 1.

Lemma 2 (Truth Lemma). *Let* $\mathcal{M}_\Gamma$ *be as in Def. 5; it follows that:*

(1) $\mathcal{M}_\Gamma, \Gamma', i, j \Vdash \alpha$ *iff* $@_i\langle\alpha\rangle j \in \Gamma'$ (2) $\mathcal{M}_\Gamma, \Gamma', i \Vdash \varphi$ *iff* $@_i\varphi \in \Gamma'$.

Proof. The proof is by mutual induction on path and node expressions.

Ind. hyp.: (A) $\mathcal{M}_\Gamma, \Gamma', i, j \Vdash \alpha$ iff $@_i\langle\alpha\rangle j \in \Gamma'$; (B) $\mathcal{M}_\Gamma, \Gamma', i \Vdash \varphi$ iff $@_i\varphi \in \Gamma'$.

We prove the inductive case for $[\alpha =_{\mathsf{c}} \beta]$ as one of the most interesting. For this case, we need to prove $\mathcal{M}_\Gamma, \Gamma', i \Vdash [\alpha =_{\mathsf{c}} \beta]$ iff $@_i[\alpha =_{\mathsf{c}} \beta] \in \Gamma'$.

($\Rightarrow$) The proof proceeds by contradiction. Suppose: (a) $\mathcal{M}_\Gamma, \Gamma', i \Vdash [\alpha =_{\mathsf{c}} \beta]$ and (b) $@_i[\alpha =_{\mathsf{c}} \beta] \notin \Gamma'$. We prove that (c) $\Gamma' \cup \{@_i\langle\alpha\rangle j, @_i\langle\beta\rangle k\} \nvdash @_i\langle @_j =_{\mathsf{c}} @_k\rangle$ for j, k arbitrary in $W_{\Gamma'}$. From not (c), $\Gamma' \cup \{@_i\langle\alpha\rangle j, @_i\langle\beta\rangle k\} \vdash @_i\langle @_j =_{\mathsf{c}} @_k\rangle$, we get $\Gamma' \vdash @_i((\langle\alpha\rangle j \wedge \langle\beta\rangle k) \rightarrow \langle @_j =_{\mathsf{c}} @_k\rangle)$; and using ([$*$]I) we obtain that $\Gamma' \vdash @_i[\alpha =_{\mathsf{c}} \beta]$; this contradicts (b). Then, from Lem. 1,

(d) exists Γ'' such that: $\Gamma' \cup \{@_i\langle\alpha\rangle j, @_i\langle\beta\rangle k\} \subseteq \Gamma''$ and $@_i\langle @_j =_{\mathsf{c}} @_k\rangle \notin \Gamma''$.

The claim is: (d) contradicts (a). Suppose that exists Γ'' such that $\Gamma' \subseteq \Gamma''$, $\{j, k\} \subseteq W_{\Gamma''}$, and $j \not\sim^{\mathsf{c}}_{\Gamma''} k$. Using (A), $\mathcal{M}_\Gamma, \Gamma'', i, j \Vdash \alpha$, $\mathcal{M}_\Gamma, \Gamma'', i, k \Vdash \beta$. This means that $\mathcal{M}_\Gamma, \Gamma', i \nVdash [\alpha =_{\mathsf{c}} \beta]$; which is a contradiction. Therefore, $\mathcal{M}_\Gamma, \Gamma', i \Vdash [\alpha =_{\mathsf{c}} \beta]$ implies $@_i[\alpha =_{\mathsf{c}} \beta] \in \Gamma'$.

($\Leftarrow$) Let $@_i[\alpha =_{\mathsf{c}} \beta] \in \Gamma'$. Proving $\mathcal{M}_\Gamma, \Gamma', i \Vdash [\alpha =_{\mathsf{c}} \beta]$ is the same to proving that for all Γ'', $\Gamma' \subseteq \Gamma''$ and all $\{j, k\} \subseteq W_{\Gamma''}$, if $\mathcal{M}_\Gamma, \Gamma'', i, j \Vdash \alpha$ and $\mathcal{M}_\Gamma, \Gamma'', i, k \Vdash \beta$, then, $j \sim^{\mathsf{c}}_{\Gamma''} k$. Let Γ'', where $\Gamma' \subseteq \Gamma''$ and $\{j, k\} \subseteq W_{\Gamma''}$ be s.t. $\mathcal{M}_\Gamma, \Gamma'', i, j \Vdash \alpha$ and $\mathcal{M}_\Gamma, \Gamma'', i, k \Vdash \beta$. The proof is concluded if $j \sim^{\mathsf{c}}_{\Gamma''} k$, i.e., $\langle @_j =_{\mathsf{c}} @_k\rangle \in \Gamma''$. From (A) $\{@_i\langle\alpha\rangle j, @_i\langle\beta\rangle k\} \subseteq \Gamma''$. Since $\Gamma' \subseteq \Gamma''$, $@_i[\alpha =_{\mathsf{c}} \beta] \in \Gamma''$. Using ([$*$]E), we get $@_i\langle @_j =_{\mathsf{c}} @_k\rangle \in \Gamma''$. Thus, $j \sim^{\mathsf{c}}_{\Gamma''} k$.

Theorem 1 (Completeness). $\Gamma \vDash \varphi$ *implies* $\Gamma \vdash \varphi$

Proof. We prove $\Gamma \nvdash \varphi$ implies $\Gamma \nvDash \varphi$. Let $\mathcal{M}_\Gamma$ be as in Def. 5. From Lem. 1 we know that exists Γ', such that: $\Gamma \subseteq \Gamma'$; $\Gamma' \in \mathcal{M}_\Gamma$; and $\varphi \notin \Gamma'$. From Lem. 2, it is clear that for some nominal $i \in W_{\Gamma'}$, it is satisfies $\mathcal{M}_\Gamma, \Gamma', i \Vdash \Gamma$ and $\mathcal{M}_\Gamma, \Gamma', i \nVdash \varphi$. This demonstrates $\Gamma \nvDash \varphi$.

4 Extended Axiomatic Systems

We briefly cover extensions of IHXPathD with *pure axioms* and *existential saturation rules* and completeness for corresponding classes of models. Pure axioms and existential saturation rules may allow us to characterize classes of models that are not definable otherwise. Adding pure axioms and existential saturation rules into the axiom system for IHXPathD is interesting because it automatically yields strong completeness proofs for the extended axioms systems for corresponding classes of models (similarly to what is done in [2,3,4]).

Standard Translation. We define a *standard translation* into Intuitionistic FOL with equality (IFOL). This translation is needed to characterize the frame conditions that the new axioms and rules define.

Table 4: Standard Translation into IFOL

Propositional	Paths
$\mathsf{ST}'_x(\bot) = \bot$	$\mathsf{ST}'_x(\langle\alpha\rangle\varphi) = \exists y(\mathsf{ST}'_{x,y}(\alpha) \wedge \mathsf{ST}'_y(\varphi))$
$\mathsf{ST}'_x(p) = \mathsf{P}(x)$	$\mathsf{ST}'_x([\alpha]\varphi) = \forall y(\mathsf{ST}'_{x,y}(\alpha) \to \mathsf{ST}'_y(\varphi))$
$\mathsf{ST}'_x(i) = x = x_i$	$\mathsf{ST}'_{x,y}(\mathsf{a}) = \mathsf{R}_\mathsf{a}(x,y)$
$\mathsf{ST}'_x(\varphi \vee \psi) = \mathsf{ST}'_x(\varphi) \vee \mathsf{ST}'_x(\psi)$	$\mathsf{ST}'_{x,y}(@_i) = y = x_i$
$\mathsf{ST}'_x(\varphi \wedge \psi) = \mathsf{ST}'_x(\varphi) \wedge \mathsf{ST}'_x(\psi)$	$\mathsf{ST}'_{x,y}([\varphi]) = (x = y) \wedge \mathsf{ST}'_y(\varphi)$
$\mathsf{ST}'_x(\varphi \to \psi) = \mathsf{ST}'_x(\varphi) \to \mathsf{ST}'_x(\psi)$	$\mathsf{ST}'_{x,y}(\alpha\beta) = \exists z(\mathsf{ST}'_{x,z}(\alpha) \wedge \mathsf{ST}'_{z,y}(\beta))$
Comparisons	
$\mathsf{ST}'_x(\langle\alpha =_\mathsf{c} \beta\rangle) = \exists y \exists z(\mathsf{ST}'_{x,y}(\alpha) \wedge \mathsf{ST}'_{x,z}(\beta) \wedge \mathsf{E}_\mathsf{c}(y,z))$	
$\mathsf{ST}'_x(\langle\alpha \neq_\mathsf{c} \beta\rangle) = \exists y \exists z(\mathsf{ST}'_{x,y}(\alpha) \wedge \mathsf{ST}'_{x,z}(\beta) \wedge \mathsf{E}_\mathsf{c}(y,y) \wedge \mathsf{E}_\mathsf{c}(z,z) \wedge \mathsf{D}_\mathsf{c}(y,z))$	
$\mathsf{ST}'_x([\alpha =_\mathsf{c} \beta]) = \forall y \forall z(\mathsf{ST}'_{x,y}(\alpha) \wedge \mathsf{ST}'_{x,z}(\beta) \to \mathsf{E}_\mathsf{c}(y,z))$	
$\mathsf{ST}'_x([\alpha \neq_\mathsf{c} \beta]) = \forall y \forall z(\mathsf{ST}'_{x,y}(\alpha) \wedge \mathsf{ST}'_{x,z}(\beta) \to (\mathsf{E}_\mathsf{c}(y,y) \wedge \mathsf{E}_\mathsf{c}(z,z) \wedge \mathsf{D}_\mathsf{c}(y,z)))$	
Equality	
$\rho_\mathsf{c} = \forall x(\mathsf{E}_\mathsf{c}(x,x))$	$\tau_\mathsf{c} = \forall x \forall y \forall z(\mathsf{E}_\mathsf{c}(x,y) \wedge \mathsf{E}_\mathsf{c}(y,z) \to \mathsf{E}_\mathsf{c}(x,z))$
$\sigma_\mathsf{c} = \forall x \forall y(\mathsf{E}_\mathsf{c}(x,y) \to \mathsf{E}_\mathsf{c}(x,y))$	
Inequality	
$\rho'_\mathsf{c} = \forall x(\neg \mathsf{D}_\mathsf{c}(x,x))$	$\tau'_\mathsf{c} = \forall x \forall y \forall z(\mathsf{D}_\mathsf{c}(x,y) \to \mathsf{D}_\mathsf{c}(y,z) \vee \mathsf{D}_\mathsf{c}(x,z))$
$\sigma'_\mathsf{c} = \forall x \forall y(\mathsf{D}_\mathsf{c}(x,y) \to \mathsf{D}_\mathsf{c}(x,y))$	
Disjointness	
$\delta_\mathsf{c} = \forall x \forall y \forall z(\mathsf{E}_\mathsf{c}(x,y) \wedge \mathsf{D}_\mathsf{c}(y,z) \to \mathsf{D}_\mathsf{c}(x,z))$	

Definition 6. *The language of IFOL is defined by the grammar:*

$$\varphi, \psi := \mathsf{P}(t) \mid \mathsf{R}(t,t') \mid \bot \mid t = t' \mid \varphi \vee \psi \mid \varphi \wedge \psi \mid \varphi \to \psi \mid \exists x \varphi \mid \forall x \varphi$$

where: P *is a predicate symbol,* R *is a relation symbol, and* t *and* t' *are terms –i.e., symbols for constants or variables. We call* $\mathsf{P}(t)$, $\mathsf{R}(t,t')$, *and* $t = t'$ atomic. *Free and bounded variables are as usual; and so are* closed *and* open *formulas.*

We establish two correspondences from IHXPathD to IFOL. The first –in Def. 7– is at the level of languages. The second –in Def. 9– is at the level of models. We show –in Prop. 3– that such correspondences preserve satisfiability.

Definition 7. *The* standard translation *of a node expression* φ*, is defined as:*

$$\mathsf{ST}_x(\varphi) = \left(\bigwedge\nolimits_{\mathsf{c}\in\mathsf{Cmp}}((\rho_\mathsf{c} \wedge \sigma_\mathsf{c} \wedge \tau_\mathsf{c}) \wedge (\rho'_\mathsf{c} \wedge \sigma'_\mathsf{c} \wedge \tau'_\mathsf{c}) \wedge \delta_\mathsf{c})\right) \wedge \mathsf{ST}'_x(\varphi).$$

The definition of ST_x *makes use of the auxiliary functions in Tab. 4.*

We recall the definition of satisfiability of a formula in a model for IFOL.

Definition 8. *A* model *for IFOL is a structure* $\mathcal{K} = \langle K, \preccurlyeq, \{\langle D_k, \Sigma_k\rangle\}_{k\in K}\rangle$ *where:* $\langle K, \preccurlyeq\rangle$ *is a poset; and for all* k

- D_k *is a non-empty set of symbols for constants*

– Σ_k *is a set of closed atomic formulas with constant symbols in* D_k *such that:*

for all $\{a, b, c\} \subseteq D_k$	*for all* $a = b \in \Sigma_k$
$a = a \in \Sigma_k$;	$\mathsf{P}(a) \in \Sigma_k$ *implies* $\mathsf{P}(b) \in \Sigma_k$;
$a = b \in \Sigma_k$ *implies* $b = a \in \Sigma_k$;	$\mathsf{R}(a, c) \in \Sigma_k$ *implies* $\mathsf{R}(b, c) \in \Sigma_k$;
$\{a = b, b = c\} \subseteq \Sigma_k$ *implies* $a = c \in \Sigma_k$;	$\mathsf{R}(c, a) \in \Sigma_k$ *implies* $\mathsf{P}(c, b) \in \Sigma_k$.

Moreover, for all $k \preccurlyeq k'$, $D_k \subseteq D_{k'}$ *and* $\Sigma_k \subseteq \Sigma_{k'}$.

We now state clearly how to build models of IFOL from models of IHXPathD.

Definition 9. *Let* $\mathcal{M} = \langle M, \preccurlyeq, \{\langle \mathfrak{M}_m, \approx_m \rangle\}_{m \in M}\rangle$ *as in Def. 2; define*

$$\mathcal{K}_{\mathcal{M}} = \langle M, \preccurlyeq, \{\langle D_m, \Sigma_m \rangle\}_{m \in M}\rangle$$

such that: $D_m = \{\, \bar{w} \mid w \in W_m \,\}$; $\Sigma_m = \mathsf{P}_m \cup \mathsf{R}_m \cup \mathsf{E}_m \cup \mathsf{D}_m \cup \mathrm{Eq}_m$; *and*

$$\mathsf{P}_m = \{\, \mathsf{P}(\bar{w}) \mid w \in V_m(p) \text{ and } p \in \mathsf{Prop} \,\} \quad \mathsf{R}_m = \{\, \mathsf{R}_\mathsf{a}(\bar{w}, \bar{v}) \mid (w, v) \in R^{\mathsf{a}}_m \text{ and } \mathsf{a} \in \mathsf{Mod} \,\}$$
$$\mathsf{E}_m = \{\, \mathsf{E}_\mathsf{c}(\bar{w}, \bar{v}) \mid w \sim^{\mathsf{c}}_m v \text{ and } \mathsf{c} \in \mathsf{Cmp} \,\} \quad \mathsf{D}_m = \{\, \mathsf{D}_\mathsf{c}(\bar{w}, \bar{v}) \mid w \nsim^{\mathsf{c}}_m v \text{ and } \mathsf{c} \in \mathsf{Cmp} \,\}$$
$$\mathrm{Eq}_m = \{\, \bar{w} = \bar{v} \mid w \approx_m v \,\}$$

where $\bar{w}$ *is a constant symbol identifying uniquely a* $w \in \bigcup\{\, W_m \mid m \in M \,\}$.

The last piece we need is the definition of satisfiability in IFOL.

Definition 10. *Let* $\mathcal{K}$ *be as in Def. 8, and* $k \in K$; *define* $\mathcal{K}, k \Vdash \varphi$ *as:*

$\mathcal{K}, k \Vdash \varphi$	*iff* $\varphi \in \Sigma_k$	*for* φ *atomic*
$\mathcal{K}, k \Vdash \varphi \wedge \psi$	*iff* $\mathcal{K}, k \vDash \varphi$ *and* $\mathcal{K}, k \Vdash \psi$	
$\mathcal{K}, k \Vdash \varphi \vee \psi$	*iff* $\mathcal{K}, k \vDash \varphi$ *or* $\mathcal{K}, k \Vdash \psi$	
$\mathcal{K}, k \Vdash \varphi \rightarrow \psi$	*iff for all* $k \preccurlyeq k'$, $\mathcal{K}, k' \Vdash \varphi$ *implies* $\mathcal{K}, k' \Vdash \psi$	
$\mathcal{K}, k \Vdash \exists x \varphi$	*iff there is* $a \in D_k$ *s.t.* $\mathcal{K}, k \Vdash \varphi[x/a]$	
$\mathcal{K}, k \Vdash \forall x \varphi$	*iff for all* $k \preccurlyeq k'$ *and* $a \in D_k$, $\mathcal{K}, k' \Vdash \varphi[x/a]$	

where: $\varphi[x/a]$ *is the result of replacing all free occurrences of* x *in* φ *for* a.

Proposition 3. $\mathcal{M}, m, w \Vdash \varphi$ *iff* $\mathcal{K}_{\mathcal{M}}, m \Vdash \mathsf{ST}_x(\varphi)[x/\bar{w}][\mathbf{x}/\bar{\mathbf{w}}]$; *where* $\mathbf{x}$ *is the list of variables* x_i *for the nominals* i *in* φ, *and* $\mathbf{w}$ *is the list of symbols for constants* $\bar{v}$ *such that* $g_m(i) = v$.

Pure Axioms and Existential Saturation Rules. We introduce pure axioms and existential saturation rules, and their associated frame conditions.

Definition 11 (Pure Expressions). *A* pure *expression is built using only nominals. Let* $\bar{i} = i_1 \ldots i_n$ *be a list of nominals; we use* $\varphi(\bar{i})$ *for the pure node expression with nominals in* $\bar{i}$ *and* $\forall \bar{i} = \forall x_{i_1} \ldots \forall x_{i_n}$. *A set* Π *of pure node expressions defines a* frame condition

$$\mathsf{FC}(\Pi) = \bigwedge\{\, (\forall x)(\forall \bar{i})\mathsf{ST}_x(\varphi(\bar{i})) \mid \varphi(\bar{i}) \in \Pi \,\}.$$

Definition 12 (Existential Saturation Rules). *Consider disjoint lists of nominals* $\bar{i} = i_1, \ldots, i_n$ *and* $\bar{j} = j_1, \ldots, j_m$*; an* existential saturation rule *is a rule of the form*

$$\rho = \varphi(\bar{i}, \bar{j}) \to \psi / \psi$$

where $\bar{j}$ *does not occur in* ψ*. Set* $\mathrm{hd}(\rho) = \varphi(\bar{i}, \bar{j})$*; and* $\exists \bar{j} = \exists x_{j_1} \ldots \exists x_{j_m}$.

A set P *of existential saturation rules defines a* frame condition

$$\mathsf{FC}(\mathrm{P}) = \bigwedge \{ (\forall x)(\forall \bar{i})(\exists \bar{j})\mathsf{ST}_x(\mathrm{hd}(\rho)) \mid \rho \in \mathrm{P} \text{ and } \mathrm{hd}(\rho) = \varphi(\bar{i}, \bar{j}) \}.$$

Example 2. The axiom $p \to \neg\langle \mathsf{a} \rangle p$ is not pure, since it contains the propositional symbol p. On the other hand, $\Pi = \{i \to \neg\langle \mathsf{a} \rangle i\}$ is a singleton set of pure axioms. Its corresponding frame condition $\mathsf{FC}(\Pi)$, i.e., $\forall x. \neg \mathsf{R}_{\mathsf{a}}(x, x)$, indicates that R_{a} is irreflexive.

Theorem 2. *Let* Π *and* P *be sets of pure node expressions and existential saturation rules. We use IHXPathD*$+\Pi+\mathrm{P}$ *for the axiomatic system IHXPathD extended with* Π *and* P *as additional axioms and rules. Then, IHXPathD*$+\Pi+\mathrm{P}$ *is strongly complete w.r.t. the class of models whose frames satisfy* $\mathsf{FC}(\Pi) \wedge \mathsf{FC}(\mathrm{P})$.

5 Final Remarks

We presented a logic –IHXPathD– that provides an intuitionistic reading of the navigational fragment of XPath with attribute values (in)equality checks. For this logic, we defined a corresponding class of models, and provided an axiomatization and completeness results. There is much work yet to be done. In particular, we would like to explore additional comparison operators (e.g., less than, greater than, etc). Moreover, we would like to explore decidability problems for IHXPathD, and their computational complexity.

Acknowledgments

We thank the reviewers and the PC member for their constructive comments and suggestions. This work is partially supported by ANPCyT-PICT-2020-3780, CONICET PIP 11220200100812CO, the EU Grant Agreement 101008233 (MISSION), and by the Laboratoire International Associé SINFIN.

References

1. Areces, C., ten Cate, B.: Hybrid logics. In: Handbook of Modal Logic, pp. 821–868. Elsevier (2006)
2. Areces, C., Fervari, R.: Axiomatizing hybrid XPath with data. LMCS **17**(3) (2021)
3. Blackburn, P., ten Cate, B.: Pure extensions, proof rules, and hybrid axiomatics. Studia Logica **84**(2), 277–322 (2006)
4. Braüner, T.: Hybrid Logics and its Proof-Theory, Applied Logics Series, vol. 37. Springer (2011)
5. van Dalen, D.: Logic and structure. Universitext, Springer, 5 edn. (2013)
6. Goldblatt, R.: An abstract setting for Henkin proofs. Topoi **3**(1), 37–41 (1984)
7. Romano, D.: Equality and coequality relations on the cartesian product of sets. Mathematical Logic Quarterly **34**(5), 471–480 (1988)