Towards a Generic Model Theory: Automatic Bisimulations for Atomic, Molecular and First-order Logics

Guillaume Aucher

Towards a Generic Model Theory: Automatic Bisimulations for Atomic, Molecular and First–order Logics

Guillaume Aucher
Univ Rennes, CNRS, IRISA, IRMAR
263, Avenue du Général Leclerc
35042 Rennes Cedex, France
guillaume.aucher@univ-rennes1.fr

December 2, 2021

Abstract

After observing that the truth conditions of connectives of non–classical logics are generally defined in terms of formulas of first–order logic (FOL), we introduce protologics, a class of logics whose connectives are defined by arbitrary first–order formulas. Then, we identify two subclasses of protologics which are particularly well-behaved. We call them atomic and molecular logics. Notions of invariance for atomic and molecular logics can be automatically defined from the truth conditions of their connectives, bisimulations do not need to be defined by hand on a case by case basis for each logic. Moreover, molecular logics behave as 'paradigmatic logics': every first–order logic and every protologic is as expressive as a molecular logic. Then, we prove a series of model–theoretical results for molecular logics which characterize them as fragments of FOL and which provide criteria for axiomatizability and definability of a class of models in these logics. In particular, we rediscover van Benthem's theorem for modal logic as a specific instance of our generic theorems and other results for modal intuitionistic logic and temporal logic. We also discover a wide range of novel results, such as for the Lambek calculus. Then, we apply our method and generic results to FOL and find out novel invariance notions for FOL, that we call predicate bisimulation and first–order bisimulation. They refine the usual notions of isomorphism and partial isomorphism. We prove generalizations as well as new versions of the Keisler theorems for countable languages in which isomorphisms are replaced by predicate bisimulations and first–order bisimulations.

Contents

1 Introduction

Non-classical logics are often associated to notions of invariance which relate models satisfying the same formulas. For many non–classical logics, the invariance notion associated to a logic characterizes it as a fragment of first-order logic (FOL for short). For example, modal bisimulation is the notion of invariance of modal logic: every formula of FOL with a free variable whose truth value is always the same in two bisimilar models is equivalent to the translation into FOL of a formula of modal logic. This is the core of the van Benthem characterization theorem. A wide variety of non–classical logics have been introduced over the past decades: modal logics, relevant logics, temporal logics, Lambek calculi, to name just a few. For each of these logics, one can define a notion of invariance and prove by adapting the van Benthem's characterization theorem that this notion of invariance characterizes the given logic as a fragment of FOL. A drawback of this logical pluralistic approach is that this has to be done by hand on a case by case basis for each non–classical logic. Each time the notion of invariance has to be found out and each time the proof of the van Benthem characterization theorem has to be adapted for that specific notion of invariance. For example, a similar characterization theorem has been proved for (modal) intuitionistic logic [33], temporal logic [29], sabotage modal logic [8], the modal μ–calculus [26], graded modal logic [14]. Even without such a full characterization, a number of invariance notions have been defined for numerous logics, like the directed bisimulations of the Lambek calculus [37] or the bisimulations for relation-changing modal logics [3, 20] for example. For these logics, one only has at hand a proposition stating that truth is preserved between two models which are bisimilar. This situation is obviously problematic if one shares the ideal of "universal logic" [9, 10]. Instead, we would prefer to obtain automatically from the definitions of the connectives of a given logic a suitable definition of bisimulation and its associated characterization theorem.

For modal logic and temporal logic, a number of model–theoretical results dealing with the definability of classes of Kripke models by means of a set of formulas or a single formula have been proved by adapting the Keisler theorems of FOL [12, 29]. Basically, the change in the theorems consists in replacing the notion of isomorphism by an appropriate notion of bisimulation (modal or temporal). A natural question that comes up to mind is to wonder whether these model-theoretical results transfer as well to arbitrary non–classical logics. We shall see that they do transfer in general to any non–classical logic once the appropriate notion of bisimulation for the non–classical logic considered has been elicited.

Our objective in this article is to develop in a systematic and generic way the model theory of non–classical logics. The difficulty is to find an appropriate method for that. Our starting point will be the observation that the truth conditions of non–classical connectives are often defined in terms of first-order formulas (without functions). This will lead us to introduce protologics, a class of logics whose connectives are defined by arbitrary first–order formulas. Even if protologics are quite general and capture a wide range of non–classical logics, they do not lend themselves naturally to a systematic exploration of their theoretical properties. The truth conditions of their connectives are arbitrary formulas of first–order logic and, as such, do not yield us much information about their associated theoretical properties. To overcome that difficulty we will introduce atomic and molecular logics, two subclasses of protologics whose connectives are of a specific form and which are particularly well–behaved. They are a generalization of polyadic modal logics that take into account some sort of monotonicity in the truth conditions of their connectives. Atomic and molecular logics are 'paradigmatic' logics in the sense that they can capture a very wide range of non–classical logics: every first–order logic as well as every protologic is as expressive as a molecular logic. The theoretical advantage of atomic logics is that their invariance notions can be automatically defined from the truth conditions of their connectives, bisimulations do not need to be defined by hand on a case by case basis for each atomic or molecular logic. Moreover, these notions of bisimulations come automatically associated with a number of model–theoretical results: a van Benthem characterization theorem as well as two Keisler theorems dealing with conditions under which a class of models is definable by a set of formulas or by a single formula if the molecular conectives are uniform. So, once a non–classical logic is expressed in terms of an atomic or molecular logic, one obtains automatically a notion of bisimulation and these associated model–theoretical results. This generic method applies to a wide range of non–classical logics since all protologics and even first–order logics are expressible in terms of atomic or molecular logics.

We will apply our generic method to first–order logics, since they are as expressive as specific atomic logics. In doing so, we will discover novel invariance notions for FOL, that we call predicate bisimulation and first–order bisimulation. They differ from the usual notions of isomorphism and partial isomorphism. We will prove generalizations as well as new versions of the Keisler–Shelah isomorphism theorem and the Keisler theorem of FOL in which isomorphisms are replaced by predicate bisimulations and first–order bisimulations. We will also generalize these theorems, they will hold for arbitrary sets of first–order formulas and not only for the set of all sentences.

Organization of the article. We start in Section 2 by recalling first–order logics and some of the most well-known non–classical logics. In that section, we will also study and introduce a specific notion of equi-expressivity for logics which are not based on the same classes of models. In Section 3 we introduce "protologics" whose connectives are defined by arbitrary formulas of FOL with free variable(s). We also introduce in that section a class of polyadic logics larger than modal logic and that we call "atomic" and "molecular" logics because molecular connectives are compositions of atomic connectives. In Section 4, we compare the relative expressivity of abstract, atomic and first–order logics. Then, in Section 5, we will show how a suitable notion of bisimulation/invariance can be defined automatically from the definition of the connectives of any atomic or molecular logic. In Section 6, we develop a basic model theory for atomic logics and provide counterparts in these logics of the van Benthem theorem as well as the Keisler-Shelah isomorphism theorem and Keisler theorems. Then, in Section 7, we apply these general results to FOL by means of the translations between FOL and atomic logics that we have established in Section 4.1 and discover new versions and extensions of the Keisler-Shelah and Keisler theorems with predicate bisimulations and first–order bisimulations in place of isomorphisms. In Section 8, we explain the role of Boolean negation in our model–theoretical results and how Boolean negation should be handled in order to apply our methods and results to an arbitrary logic. Finally, we discuss related work in Section 9 and conclude in Section 10.

2 Classical and Non-Classical Logics

In this section, we recall FOL and some of the most well–known non–classical logics. The generic results of the article will be applied to these logics in the sequel. Logics will always be semantically presented by following a tri-partite representation: language, class of models, satisfaction relation.

2.1 Classical Logics

$\mathcal{P} \triangleq \{\mathsf{R}_1, \ldots, \mathsf{R}_n, \ldots\}$ is a set of *predicate symbols* of arity $k_1, \ldots, k_n, \ldots$ respectively (one of them can be the identity predicate = of arity 2), $\mathcal{F} \triangleq \{\mathsf{f}_1, \ldots, \mathsf{f}_n, \ldots\}$ is a set of *function symbols*, $\mathcal{V} \triangleq \{v_1, \ldots, v_n, \ldots\}$ is a set of *variables* and $\mathcal{C} \triangleq \{c_1, \ldots, c_n, \ldots\}$ is a set of *constants*. Each of these sets can be finite or infinite. $v_1, v_2, v_3, \ldots$ are the names of the variables and we use the expressions $x, x_1, x_2, \dots, y, y_1, y_2, \dots, z, z_1, z_2, \dots$ to refer to arbitrary variables or constants, which can be for example $v_{42}, v_5, c_{101}, c_{21}, \dots$ Arity$(\mathcal{P}, \mathcal{F})$ is the set of all arities of predicate and function symbols.

The *first-order language* $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ is defined inductively by the following grammars in BNF:

$$\begin{aligned} \mathcal{L}_{\mathsf{FOL}}^{\mathcal{VC}}: &\ t \ ::= \ x \ | \ \mathsf{c} \ | \ \mathsf{f}t \dots t \\ \mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}: &\ \varphi \ ::= \ \mathsf{R}t \dots t \ | \ \bot t \ | \ (\varphi \to \varphi) \ | \ (\varphi \vee \varphi) \ | \ \forall x \varphi \end{aligned}$$

where $x \in \mathcal{V}$, $\mathsf{c} \in \mathcal{C}$, $\mathsf{f} \in \mathcal{F}$, $t \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{VC}}$ and $\mathsf{R} \in \mathcal{P}$. Elements of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{VC}}$ are called *terms* and elements of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ are called *first–order formulas*. Formulas of the form $\mathsf{R}t_1 \dots t_k$ are called *propositional letteric formulas* and first–order formulas without function symbols are called *pure predicate formulas*. If $\varphi \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$, the *Boolean negation* of φ, denoted $\neg\varphi$, is defined by the abbreviation $\neg\varphi \triangleq (\varphi \to \bot t)$ for an arbitrary $t \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{VC}}$. We also use the abbreviations $\top \triangleq \neg\bot$, $(\varphi \wedge \psi) \triangleq \neg(\neg\varphi \vee \neg\psi)$ and $(\varphi \leftrightarrow \psi) \triangleq (\varphi \to \psi) \wedge (\psi \to \varphi)$ as well as the abbreviations $\exists x \varphi \triangleq \neg\forall x \neg\varphi$, $\forall x_1 \dots x_n \varphi \triangleq \forall x_1 \dots \forall x_n \varphi$, $\exists x_1 \dots x_n \varphi \triangleq \exists x_1 \dots \exists x_n \varphi$ and $\forall \overline{x} \varphi \triangleq \forall x_1 \dots x_n \varphi$ if $\overline{x} = (x_1, \dots, x_n)$ is a tuple of variables.

We also define the *head* $h(t)$ and the *body* $b(t)$ of a term $t \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{VC}}$ inductively as follows: if $t = x$ or $t = \mathsf{c}$ then $h(t) = b(t) = t$; if $t = f(t_1, \dots, t_k)$ then $h(t) = f$ and $b(t) = (t_1, \dots, t_k)$.

Let $\varphi \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$. An occurrence of a variable x in φ is *free* (in φ) if, and only if, x is not within the scope of a quantifier of φ. A variable which is not free (in φ) is *bound* (in φ). We say that a formula of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ is a *sentence* (or is *closed*) when it contains no free variable. We denote by $\varphi(x_1, \dots, x_k)$ a formula of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ whose free variables or constants coincide *exactly* with $x_1, \dots, x_k$. In doing so, we depart from the literature in which this notation means that the free variables of φ are *included* in $\{x_1, \dots, x_k\}$. Free variables may be used to bind elements of two different subformulas. For example, the formula $\mathsf{R}yx \vee \mathsf{R}'xz$ with free variables x, y, z will be evaluated in a structure in such a way that x will be assigned the same element of the domain in the two subformulas $\mathsf{R}yx$ and $\mathsf{R}'xz$.

We denote by $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}$ the fragment of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ whose formulas do not contain function symbols. We denote by $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$ (resp. $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}(\overline{x})$) the fragment of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}$ (resp. $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$) whose formulas all contain at least one free variable or constant and by $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\varnothing)$ (resp. $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}(\varnothing)$) the set of sentences of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}$ (resp. $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$). For all $k \in \mathbb{N}$ and $\overline{x} = (x_1, \dots, x_k) \in \mathcal{V}^k$, we denote by $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x}, k)$ (resp. $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}(\overline{x}, k)$) the fragment of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}$ (resp. $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$) whose formulas all contain *exactly* k free variables or constants and these variables or constants are $\overline{x}$. Note that $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$, $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}(\overline{x})$, $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}$ and $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ can all be partitionned into sublanguages of the form $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x}, k)$. A language $\mathcal{L}_{\mathsf{FOL}} \subseteq \mathcal{L}_{\mathsf{FOL}}^{\mathcal{PF}}$ is *countable* if its set of predicate symbols, function symbols, variables and constants is countable.

A *structure* is a tuple $M \triangleq (W, \{R_1, \dots, R_n, \dots, f_1, \dots, f_n, \dots, c_1, \dots, c_n, \dots\})$ where:

- W is a non-empty set called the *domain*;
- $R_1, \dots, R_n, \dots$ are relations over W with the same arity as $\mathsf{R}_1, \dots, \mathsf{R}_n, \dots$ respectively;
- $f_1, \dots, f_n, \dots$ are functions over W with the same arity as $\mathsf{f}_1, \dots, \mathsf{f}_n, \dots$ respectively;
- $c_1, \dots, c_n, \dots \in W$ are elements of the domain called *distinguished elements*.

An *assignment* over M is a mapping $s : \mathcal{V} \cup \mathcal{C} \to W$ such that for all $\mathsf{c}_i \in \mathcal{C}$, $s(\mathsf{c}_i) = c_i$. If s is an assignment, $s[x := w]$ is the same assignment as s except that the value of the variable $x \in \mathcal{V}$ is assigned to w. A pair of structure and assignement (M, s) is called a *pointed structure*. The class of all pointed structures (M, s) is denoted $\mathcal{M}_{\mathsf{FOL}}$. If K is a class of pointed structures, $\overline{K}$ is $\mathcal{M}_{\mathsf{FOL}} - K$.

If (M, s) is a pointed structure, we extend the assignment s from variables and constants to terms and define the *extended assignment* $\overline{s} : \mathcal{L}^{\mathcal{VC}}_{\mathsf{FOL}} \to W$ inductively as follows:

$$\begin{array}{rcl} \overline{s}(x) & \triangleq & s(x) \\ \overline{s}(\mathsf{c}) & \triangleq & s(\mathsf{c}) \\ \overline{s}(\mathsf{f}t_1 \dots t_k) & \triangleq & f(\overline{s}(t_1), \dots, \overline{s}(t_k)). \end{array}$$

The *satisfaction relation* $\models_{\mathsf{FOL}} \subseteq \mathcal{M}_{\mathsf{FOL}} \times \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ is defined inductively as follows. Below, we write $(M, s) \models \varphi$ for $((M, s), \varphi) \in \models_{\mathsf{FOL}}$.

$$\begin{array}{lll} (M,s) \models \bot t & & \text{never;} \\ (M,s) \models \mathsf{R}_i t_1 \dots t_k & \text{iff} & (\overline{s}(t_1), \dots, \overline{s}(t_k)) \in R_i; \\ (M,s) \models (\varphi \to \psi) & \text{iff} & \text{if } (M,s) \models \varphi \text{ then } (M,s) \models \psi; \\ (M,s) \models (\varphi \vee \psi) & \text{iff} & (M,s) \models \varphi \text{ or } (M,s) \models \psi; \\ (M,s) \models \forall x \varphi & \text{iff} & (M, s[x := w]) \models \varphi \text{ for all } w \in W. \end{array}$$

In the literature [13], $(M, s) \models \varphi(x_1, \dots, x_k)$ is sometimes denoted $M \models \varphi(x_1, \dots, x_k)[w_1, \dots, w_k]$, $M \models \varphi[w_1/x_1, \dots, w_k/x_k]$ or simply $M \models \varphi[w_1, \dots, w_k]$, with $w_1 = s(x_1), \dots, w_k = s(x_k)$. Some other times [19], it is denoted $M \models \varphi(x_1, \dots, x_n)[s]$, $M, s \models \varphi(x_1, \dots, x_n)$ or simply $M \models \varphi[s]$. In that case, we say that (M, s) makes φ *true*. We depart from the literature by treating constants on a par with variables: the denotation of constants is usually not dealt with by means of assignments. Two (pointed) structures are *elementarily equivalent* when they make true the same sentences. If (M, s) and (M', s') are two pointed structures and $\overline{x}$ is a non-empty tuple of variables of size k, we write $(M, s) \equiv_{\overline{x}} (M', s)$ when for all $\varphi \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}, k)$ it holds that $(M, s) \models \varphi$ iff $(M', s') \models \varphi$ and we write $(M, s) \equiv_{\mathcal{F}, \overline{x}} (M', s)$ when for all $\varphi \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}(\overline{x}, k)$ it holds that $(M, s) \models \varphi$ iff $(M', s') \models \varphi$.

We say that the formula $\varphi \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ is *realized in* M when there is an assignment s such that $(M, s) \models \varphi$.

A triple of the form $\left(\mathcal{L}_{\mathsf{FOL}}, \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is called the *first–order logic associated to* $\mathcal{L}_{\mathsf{FOL}}$ *and* $\mathcal{E}_{\mathsf{FOL}}$. If $\mathcal{L}_{\mathsf{FOL}} = \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$, the triple $\left(\mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}, \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is called *first–order logic (associated to* $\mathcal{E}_{\mathsf{FOL}}$*)*, if $\mathcal{L}_{\mathsf{FOL}} = \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$, the triple $\left(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}, \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is called *pure predicate logic (associated to* $\mathcal{E}_{\mathsf{FOL}}$*)*, if $\mathcal{L}_{\mathsf{FOL}} = \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$, the triple $\left(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}), \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is called *pure predicate logic with free variables and constants (associated to* $\mathcal{E}_{\mathsf{FOL}}$*)* and if $\mathcal{L}_{\mathsf{FOL}} = \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}(\overline{x})$, the triple $\left(\mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}(\overline{x}), \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is called *first–order logic with free variables and constants (associated to* $\mathcal{E}_{\mathsf{FOL}}$*)*. When $\mathcal{E}_{\mathsf{FOL}}$ is $\mathcal{M}_{\mathsf{FOL}}$, they are simply called respectively *first–order logic*, *pure predicate logic* and *pure predicate logic with free variables and constants*.

Remark 1. Our choice not to introduce explicitly Boolean negation will become clear in the rest of the article (see in particular Section 8). We have instead introduced the material implication $\to$ and the constant falsum $\bot$. We could also have simply introduced negated atomic formulas, the disjunction and the existential quantifier since the Boolean negation $\neg$ is also definable as abbreviations in terms of these connectives by means of negative normal forms.

Remark 2. We use the symbol $\triangleq$ instead of $=$ to denote the fact that the equality is in fact definitional and $[\![i; j]\!]$ stands for the set of natural numbers from i to j.

Isomorphisms and partial isomorphisms [13] are the traditional notions of invariance of first–order logic.

Definition 1 (Partial isomorphism)**.** A *partial isomorphism* between two structures M_1 and M_2 is a relation Z on the set of pairs of finite sequences $(w_1, \dots, w_n), (v_1, \dots, v_n)$ of elements of M_1 and M_2 of the same length such that:

1. if $(w_1, \ldots, w_n)Z(v_1, \ldots, v_n)$ then (M_1, s_1) and (M_2, s_2) make true the same propositional letteric formulas $\varphi(x_1, \ldots, x_n) \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$, where s_1 and s_2 are such that for all $i \in \{1, \ldots, n\}$, $s_1(x_i) = w_i$ and $s_2(x_i) = v_i$;
2. if $(w_1, \ldots, w_n)Z(v_1, \ldots, v_n)$ then for all $w \in M_1$ there is $v \in M_2$ such that $(w_1, \ldots, w_n, w)Z(v_1, \ldots, v_n, v)$, and vice versa;
3. for all $w \in M_1$ there is $v \in M_2$ such that $(w)Z(v)$, and vice versa. ⊣

Condition 2 is called the *back and forth* condition. The back condition could be equivalently replaced by the assumption that Z is symmetric. Likewise, condition 3 could be equivalently replaced by the condition $\varnothing Z \varnothing$ since they boil down to the same constraints on pairs of structures; this last condition is the one used in [13].

2.2 Non-Classical Logics

The examples of non–classical logics that follow are among the most well–known and most studied non–classical logics.

In this section, $\mathbb{A}$ is a set of *propositional letters* which can be finite or infinite.

2.2.1 Modal Logic

The set $\mathbb{I}$ is a set of indices which can be finite or infinite. The *multi-modal language* $\mathcal{L}_{\mathsf{ML}}$ is defined inductively by the following grammar in BNF:

$$\mathcal{L}_{\mathsf{ML}}: \ \varphi \ ::= \ p \ \mid \ \neg p \ \mid \ (\varphi \wedge \varphi) \ \mid \ (\varphi \vee \varphi) \ \mid \ \Diamond_j \varphi \ \mid \ \Box_j \varphi$$

where $p \in \mathbb{A}$ and $j \in \mathbb{I}$.

We present the so-called *possible world semantics* of modal logic. A *Kripke model* M is a tuple $M \triangleq (W, \{R_1, \ldots, R_m, \ldots, P_1, \ldots, P_n, \ldots\})$ where

- W is a non-empty set whose elements are called *possible worlds*;
- $R_1, \ldots, R_m, \ldots \subseteq W \times W$ are binary relations over W called *accessibility relations*;
- $P_1, \ldots, P_n, \ldots \subseteq W$ are unary relations interpreting the propositional letters of $\mathbb{P}$.

We write $w \in M$ for $w \in W$ by abuse and the pair (M, w) is called a *pointed Kripke model*. The class of all pointed Kripke models is denoted $\mathcal{E}_{\mathsf{ML}}$.

We define the *satisfaction relation* $\models_{\mathsf{ML}} \subseteq \mathcal{E}_{\mathsf{ML}} \times \mathcal{L}_{\mathsf{ML}}$ inductively by the following *truth conditions*. Below, we write $(M, w) \models \varphi$ for $((M, w), \varphi) \in \models_{\mathsf{ML}}$. For all $(M, w) \in \mathcal{E}_{\mathsf{ML}}$, all $\varphi, \psi \in \mathcal{L}_{\mathsf{ML}}$, all $p_i \in \mathbb{P}$ and all $j \in \mathbb{I}$,

$$
\begin{array}{lll}
(M,w) \models p_i & \text{iff} & P_i(w) \text{ holds;} \\
(M,w) \models \neg p_i & \text{iff} & P_i(w) \text{ does not hold;} \\
(M,w) \models (\varphi \wedge \psi) & \text{iff} & (M,w) \models \varphi \text{ and } (M,w) \models \psi; \\
(M,w) \models (\varphi \vee \psi) & \text{iff} & (M,w) \models \varphi \text{ or } (M,w) \models \psi; \\
(M,w) \models \Diamond_j \varphi & \text{iff} & \text{there exists } v \in W \text{ such that } R_j wv \text{ and } (M,v) \models \varphi; \\
(M,w) \models \Box_j \varphi & \text{iff} & \text{for all } v \in W \text{ such that } R_j wv, (M,v) \models \varphi.
\end{array}
$$

The triple $(\mathcal{L}_{\mathsf{ML}}, \mathcal{E}_{\mathsf{ML}}, \models_{\mathsf{ML}})$ forms a logic, that we call *modal logic*. Bisimulations for modal logic can be found in [12].

2.2.2 Lambek Calculus

The *Lambek language* $\mathcal{L}_{LC}$ is the set of formulas defined inductively by the following grammar in BNF:

$$\mathcal{L}_{\mathsf{LC}}: \ \varphi \ ::= \ p \ | \ (\varphi \otimes \varphi) \ | \ (\varphi \subset \varphi) \ | \ (\varphi \supset \varphi)$$

where $p \in \mathbb{P}$. A *Lambek model* is a tuple $M = (W, \{R, P_1, \dots, P_n, \dots\})$ where:

- W is a non-empty set;
- $R \subseteq W \times W \times W$ is a ternary relation over W;
- $P_1, \dots, P_n, \dots \subseteq W$ are unary relations over W.

We write $w \in M$ for $w \in W$ by abuse and (M, w) is called a *pointed Lambek model*. The class of all pointed Lambek models is denoted $\mathcal{E}_{\mathsf{LC}}$. We define the *satisfaction relation* $\models_{\mathsf{Int}} \subseteq \mathcal{E}_{\mathsf{LC}} \times \mathcal{L}_{\mathsf{LC}}$ by the following *truth conditions*. Below, we write $(M, w) \models \varphi$ for $((M, w), \varphi) \in \models_{\mathsf{LC}}$. For all Lambek models $M = (W, \{R, P_1, \dots, P_n, \dots\})$, all $w \in M$, all $\varphi, \psi \in \mathcal{L}_{\mathsf{LC}}$ and all $p_i \in \mathbb{P}$,

$$\begin{array}{lll} (M,w) \models p_i & \text{iff} & P_i(x) \text{ holds;} \\ (M,w) \models (\varphi \otimes \psi) & \text{iff} & \text{there are } v, u \in W \text{ such that } Rvuw, \\ & & (M,v) \models \varphi \text{ and } (M,u) \models \psi; \\ (M,w) \models (\varphi \supset \psi) & \text{iff} & \text{for all } v, u \in W \text{ such that } Rwvu, \\ & & \text{if } (M,v) \models \varphi \text{ then } (M,u) \models \psi; \\ (M,w) \models (\psi \subset \varphi) & \text{iff} & \text{for all } v, u \in W \text{ such that } Rvwu \\ & & \text{if } (M,v) \models \varphi \text{ then } (M,u) \models \psi. \end{array}$$

The triple $(\mathcal{L}_{\mathsf{LC}}, \mathcal{E}_{\mathsf{LC}}, \models_{\mathsf{LC}})$ forms a logic, that we call the *Lambek calculus*. Bisimulations for the Lambek calculus, called directed bisimulations, can be found in [37].

2.2.3 Modal Intuitionistic Logic

The *modal intuitionistic language* $\mathcal{L}_{\mathsf{Int}}$ is defined inductively by the following grammar in BNF:

$$\mathcal{L}_{\mathsf{Int}}: \ \varphi \ ::= \ \top \ | \ \bot \ | \ p \ | \ (\varphi \wedge \varphi) \ | \ (\varphi \vee \varphi) \ | \ (\varphi \Rightarrow \varphi) \ | \ \Diamond\varphi \ | \ \Box\varphi$$

where $p \in \mathbb{A}$. A *modal intuitionistic model* is a tuple $M = (W, \{R, R_\Diamond, P_1, \dots, P_n, \dots\})$ where:

- W is a non-empty set;
- $R \subseteq W \times W$ is a binary relation over W which is reflexive and transitive (R is *reflexive* if for all $w \in W$ Rww and *transitive* if for all $u, v, w \in W$, Ruv and Rvw imply Ruw);
- $R_\Diamond \subseteq W \times W$ is a binary relation over W;
- $P_1, \dots, P_n, \dots \subseteq W$ are unary relations over W such that for all $v, w \in W$, if Rvw and $P_n(v)$ then $P_n(w)$.

We write $w \in M$ for $w \in W$ by abuse and the pair (M, w) is called a *pointed modal intuitionistic model*. The class of all pointed modal intuitionistic models is denoted $\mathcal{E}_{\mathsf{Int}}$. We define the *satisfaction relation* $\models_{\mathsf{Int}} \subseteq \mathcal{E}_{\mathsf{Int}} \times \mathcal{L}_{\mathsf{Int}}$ by the following *truth conditions*. Below, we write $(M, w) \models \varphi$

for $((M,w),\varphi) \in \models_{\mathsf{Int}}$. For all modal intuitionistic models $M = (W, \{R, R_\diamond, P_1, \dots, P_n, \dots\})$, all $w \in M$, all $\varphi, \psi \in \mathcal{L}_{\mathsf{Int}}$ and all $p_i \in \mathbb{P}$,

$(M,w) \models \top$		always;
$(M,w) \models \bot$		never;
$(M,w) \models p_i$	iff	$P_i(x)$ holds;
$(M,w) \models (\varphi \wedge \psi)$	iff	$(M,w) \models \varphi$ and $(M,w) \models \psi$;
$(M,w) \models (\varphi \vee \psi)$	iff	$(M,w) \models \varphi$ or $(M,w) \models \psi$;
$(M,w) \models (\varphi \Rightarrow \psi)$	iff	for all $v \in W$ such that Rwv, if $(M,v) \models \varphi$ then $(M,v) \models \psi$;
$(M,w) \models \Box\varphi$	iff	for all $v \in W$ such that Rwv, for all $u \in W$ such that $R_\diamond vu$, $(M,u) \models \varphi$;
$(M,w) \models \Diamond\varphi$	iff	for all $v \in W$ such that Rwv, there is $u \in W$ such that $R_\diamond vu$ and $(M,u) \models \varphi$.

The triple $(\mathcal{L}_{\mathsf{Int}}, \mathcal{E}_{\mathsf{Int}}, \models_{\mathsf{Int}})$ forms a logic, that we call *modal intuitionistic logic*. Bisimulations for (modal) intuitionistic logic can be found in [32, 33].

2.2.4 Temporal Logic

The *temporal language* $\mathcal{L}_{\mathsf{TL}}$ is defined inductively by the following grammar in BNF:

$$\mathcal{L}_{\mathsf{TL}}: \ \varphi \ ::= \ \top \ | \ \bot \ | \ p \ | \ \neg p \ | \ (\varphi \wedge \varphi) \ | \ (\varphi \vee \varphi) \ | \ U(\varphi,\varphi) \ | \ S(\varphi,\varphi)$$

where $p \in \mathbb{A}$. A *temporal model* is a tuple $M = (W, \{<, P_1, \dots, P_n, \dots\})$ where:

- W is a non-empty set;
- $< \subseteq W \times W$ is a binary relation over W;
- $P_1, \dots, P_n, \dots \subseteq W$ are unary relations over W.

We write $w \in M$ for $w \in W$ by abuse and the pair (M,w) is called a *pointed temporal model*. The class of all pointed temporal models is denoted $\mathcal{E}_{\mathsf{TL}}$. We define the *satisfaction relation* $\models_{\mathsf{TL}} \subseteq \mathcal{E}_{\mathsf{TL}} \times \mathcal{L}_{\mathsf{TL}}$ by the following *truth conditions*. Below, we write $(M,w) \models \varphi$ for $((M,w),\varphi) \in \models_{\mathsf{TL}}$. For all temporal models $M = (W, \{<, P_1, \dots, P_n, \dots\})$, all $w \in M$, all $\varphi, \psi \in \mathcal{L}_{\mathsf{TL}}$ and all $p_i \in \mathbb{P}$,

$(M,w) \models \top$		always;
$(M,w) \models \bot$		never;
$(M,w) \models p_i$	iff	$P_i(x)$ holds;
$(M,w) \models \neg p_i$	iff	$P_i(x)$ does not hold;
$(M,w) \models (\varphi \wedge \psi)$	iff	$(M,w) \models \varphi$ and $(M,w) \models \psi$;
$(M,w) \models (\varphi \vee \psi)$	iff	$(M,w) \models \varphi$ or $(M,w) \models \psi$;
$(M,w) \models U(\varphi,\psi)$	iff	there is $v \in W$ such that $w < v$ and $(M,v) \models \varphi$ and for all $u \in W$ such that $w < u < v$, $(M,u) \models \psi$;
$(M,w) \models S(\varphi,\psi)$	iff	there is $v \in W$ such that $v < w$ and $(M,v) \models \varphi$ and for all $u \in W$ such that $v < u < w$, $(M,u) \models \psi$.

The triple $(\mathcal{L}_{\mathsf{TL}}, \mathcal{E}_{\mathsf{TL}}, \models_{\mathsf{TL}})$ forms a logic, that we call *temporal logic*. Bisimulations for temporal logic can be found in [29].

2.2.5 Many–valued Logics

Our presentation of many–valued logic is inspired by Priest [35] but is slightly different from the usual presentation. V is a set called the *truth values*. Let $D \subseteq V$ be a subset of *designated values* and let $\mathsf{C} = \{\star_1, \ldots, \star_m, \ldots\}$ be a countable set of connectives of arity $k_1, \ldots, k_m, \ldots$. The *many–valued language* $\mathcal{L}_{\mathsf{MV}}^{\mathsf{C}}$ associated to C is defined inductively by the following grammar in BNF:

$$\mathcal{L}_{\mathsf{MV}}^{\mathsf{C}} : \varphi ::= p \mid \star(\varphi, \ldots, \varphi)$$

where $p \in \mathbb{P}$ and $\star \in \mathsf{C}$. A *many–valued model* is a tuple $M = (V, \{R_{\star_1}, \ldots, R_{\star_m}, \ldots, P_1, \ldots, P_n, \ldots\})$ where:

- V is the set of truth values;
- $R_{\star_1}, \ldots, R_{\star_m}, \ldots$ are relations over V of arity $k_1 + 1, \ldots, k_m + 1, \ldots$;
- $P_1, \ldots, P_n, \ldots \subseteq V$ are unary relations over V.

The relations $R_\star$ are obtained from the usual truth functions f of many–valued logics by the connection $R_\star w_1 \ldots w_k w$ iff $f(w_1, \ldots, w_k) = w$. We write $w \in M$ for $w \in W$ and the pair (M, w) is called a *pointed many–valued model*. The class of all pointed many–valued models is denoted $\mathcal{E}_{\mathsf{MV}}$. We also define a *designated many–valued model* as a pair (M, D) where M is a many–valued model (and D is the set of designated values). We define the *satisfaction relation* $\models_{\mathsf{MV}} \subseteq \mathcal{E}_{\mathsf{MV}} \times \mathcal{L}_{\mathsf{MV}}^{\mathsf{C}}$ by the following *truth conditions*. Below, we write $(M, w) \models \varphi$ for $((M, w), \varphi) \in \models_{\mathsf{MV}}$. For all many–valued models $M = (V, \{R_{\star_1}, \ldots, R_{\star_m}, \ldots, P_1, \ldots, P_n, \ldots\})$, all $w \in M$, all $\varphi_1, \ldots, \varphi_k \in \mathcal{L}_{\mathsf{MV}}^{\mathsf{C}}$ and all $p_i \in \mathbb{P}$,

$$\begin{array}{lll} (M,w) \models p_i & \text{iff} & P_i(w) \text{ holds;} \\ (M,w) \models \star(\varphi_1, \ldots, \varphi_k) & \text{iff} & \text{there are } w_1, \ldots, w_k \in V \text{ such that } R_\star w_1 \ldots w_k w \\ & & \text{and } (M, w_1) \models \varphi_1 \text{ and } \ldots \text{ and } (M, w_k) \models \varphi_k \end{array}$$

We extend the satisfaction relation $\models_{\mathsf{ML}}$ to the set of designated many–valued models (M, D) as follows: we set $(M, D) \models \varphi$ iff there is $w \in D$ such that $(M, w) \models \varphi$.

Typically, a many–valued logic is based on a class of designated many–valued models whose truth functions associated to the same connective are the same in every model of the class. A class of designated many–valued models satisfying this condition is called a *many–valued class of designated models*. So, a *many–valued logic* associated to a set of connectives C and designated values D is a triple $\left(\mathcal{L}_{\mathsf{MV}}^{\mathsf{C}}, \mathcal{E}_{\mathsf{MV}}^{D}, \models_{\mathsf{MV}}\right)$ where $\mathcal{E}_{\mathsf{MV}}^{D}$ is a many–valued class of designated models. Fuzzy logic, the 3–valued logics of Kleene and Łukasiewicz are examples of many–valued logics in which the unary predicates P_i are singletons. Our general approach also allows us to capture the logic of first–degree entailment (FDE) since in that case the P_is are not necessarily singletons (see for instance [35] for more details on many–valued logics).

2.3 Common Logical Notions

In the present section, we define a number of notions which are common to all non–classical logics and in particular to the logics introduced beforehand. The way we define logics is different from many proposals considered in universal logic [9] such as pairs of Suzsko's abstract logics, Tarski's consequence operators or logical structures. Often a logic is viewed as a pair of a language together with a consequence relation on this language. Our approach to defining logics is somehow more 'semantic' in that respect than the usual proposals. It corresponds in fact to the "abstract logics" of García-Matos & Väänänen [21].

We will say that a *logic* is a triple $\mathsf{L} \triangleq (\mathcal{L}, \mathcal{E}, \models)$ where

- $\mathcal{L}$ is a *logical language* defined as a set of well-formed expressions built from a set of *connectives* C and a set of *propositional letters* $\mathbb{A}$;
- $\mathcal{E}$ is a *class of pointed models*;
- $\models$ is a *satisfaction relation* which relates in a compositional manner elements of $\mathcal{L}$ to models of $\mathcal{E}$ by means of so-called *truth conditions*.

Let $\mathsf{L} = (\mathcal{L}, \mathcal{E}, \models)$ be a logic and let $\Gamma \subseteq \mathcal{L}$, $\varphi \in \mathcal{L}$ and $M \in \mathcal{E}$. We write $M \models \Gamma$ when for all $\psi \in \Gamma$, we have $M \models \psi$. Then, we say that

- φ is *true* (*satisfied*) at M or M is a *model* of φ when $M \models \varphi$;
- φ is a *logical consequence* of Γ, written $\Gamma \models_{\mathsf{L}} \varphi$, when for all $M \in \mathcal{E}$, if $M \models \Gamma$ then $M \models \varphi$;
- φ is *valid*, written $\models_{\mathsf{L}} \varphi$, when for all models $M \in \mathcal{E}$, we have $M \models \varphi$;
- φ is *satisfiable* when $\neg\varphi$ is not valid in $\mathcal{E}$, *i.e.* when there is a model $M \in \mathcal{E}$ such that $M \models \varphi$.

If Γ is a singleton $\Gamma = \{\psi\}$, we also write by abuse $\psi \models \varphi$ for $\{\psi\} \models \varphi$.

A set of formulas of $\mathcal{L}$ is called a *theory*. A set Δ of formulas of $\mathcal{L}$ is said to be a *set of axioms* for a theory Γ iff Γ and Δ have the same logical consequences. A theory is called *finitely axiomatizable* iff it has a finite set of axioms. A logic L is *axiomatizable* if its set of validities is finitely axiomatizable.

2.4 On the Relative Expressivity of Logics

When two logics $\mathsf{L}_1 = (\mathcal{L}_1, \mathcal{E}, \models_1)$ and $\mathsf{L}_2 = (\mathcal{L}_2, \mathcal{E}, \models_2)$ are interpreted on the same class of models $\mathcal{E}$, there is a standard way to compare their relative expressiveness. We say that L_1 is *at least as expressive as* L_2, denoted $\mathsf{L}_2 \leq \mathsf{L}_1$, when there is a mapping $T : \mathcal{L}_2 \to \mathcal{L}_1$ such that for all $\varphi_2 \in \mathcal{L}_2$ and all $M \in \mathcal{E}$, $M \models \varphi_2$ iff $M \models T(\varphi_2)$. These mappings naturally induce conservative translation morphisms between logics viewed as pairs of language and consequence relation in the sense of Arndt & Al. [4] and yield a category of logics, which are all based on the same class of models $\mathcal{E}$.

When $\mathsf{L}_1 \geq \mathsf{L}_2$ and $\mathsf{L}_2 \geq \mathsf{L}_1$, we say that L_1 is *as expressive as* L_2 and denote it $\mathsf{L}_1 \equiv \mathsf{L}_2$. In that case, the definition rewrites as follows:

- there is a mapping $T_1 : \mathcal{L}_1 \to \mathcal{L}_2$ such that for all $\varphi_1 \in \mathcal{L}_1$ and all $M \in \mathcal{E}$, $M \models \varphi_1$ iff $M \models T_1(\varphi_1)$;
- there is a mapping $T_2 : \mathcal{L}_2 \to \mathcal{L}_1$ such that for all $\varphi_2 \in \mathcal{L}_2$ and all $M \in \mathcal{E}$, $M \models \varphi_2$ iff $M \models T_2(\varphi_2)$.

Now, given a logic $(\mathcal{L}, \mathcal{E}, \models)$, for all $\varphi, \psi \in \mathcal{L}$, we write $\varphi \equiv \psi$ when for all $M \in \mathcal{E}$, it holds that $M \models \varphi$ iff $M \models \psi$ and for all $M, N \in \mathcal{E}$, we write $M \equiv N$ when for all $\varphi \in \mathcal{L}$, it holds $M \models \varphi$ iff $N \models \varphi$. If $\mathcal{M}, \mathcal{N} \subseteq \mathcal{E}$, we write $\mathcal{M} \equiv \mathcal{N}$ when for all $M \in \mathcal{M}$ there is $N \in \mathcal{N}$ such that $M \equiv N$, and vice versa. With these notations, that definition of equi-expressivity entails in particular the following two facts: for all $\varphi_1 \in \mathcal{L}_1$ and all $\varphi_2 \in \mathcal{L}_2$,

$$\{M \in \mathcal{E} \mid M \models \varphi_1\} = \{M \in \mathcal{E} \mid M \models T_1(\varphi_1)\} \qquad \varphi_1 \equiv T_2(T_1(\varphi_1)) \tag{1}$$

$$\{M \in \mathcal{E} \mid M \models \varphi_2\} = \{M \in \mathcal{E} \mid M \models T_2(\varphi_2)\} \qquad \varphi_2 \equiv T_1(T_2(\varphi_2)) \tag{2}$$

However, when two logics $\mathsf{L}_1 = (\mathcal{L}_1, \mathcal{E}_1, \models_1)$ and $\mathsf{L}_2 = (\mathcal{L}_2, \mathcal{E}_2, \models_2)$ are interpreted over different classes of models $\mathcal{E}_1$ and $\mathcal{E}_2$, there is no standard way to compare their relative expressiveness. We are now going to propose some notions to deal with that issue. Our first proposal is the following.

Definition 2 (Essential equi-expressivity)**.** A logic $\mathsf{L}_1 = (\mathcal{L}_1, \mathcal{E}_1, \models_1)$ is *essentially as expressive as* a logic $\mathsf{L}_2 = (\mathcal{L}_2, \mathcal{E}_2, \models_2)$ when the following conditions hold:

1. there is a mapping $T_1^\varphi : \mathcal{L}_1 \to \mathcal{L}_2$ and a mapping $T_1^M : \mathcal{E}_1 \to \mathcal{E}_2$ such that for all $\varphi_1 \in \mathcal{L}_1$ and all $M_1 \in \mathcal{E}_1$, it holds that $M_1 \models \varphi_1$ iff $T_1^M(M_1) \models T_1^\varphi(\varphi_1)$;
2. there is a mapping $T_2^\varphi : \mathcal{L}_2 \to \mathcal{L}_1$ and a mapping $T_2^M : \mathcal{E}_2 \to \mathcal{E}_1$ such that for all $\varphi_2 \in \mathcal{L}_2$ and all $M_2 \in \mathcal{E}_2$, it holds that $M_2 \models \varphi_2$ iff $T_2^M(M_2) \models T_2^\varphi(\varphi_2)$;
3. for all $M_1 \in \mathcal{E}_1$ and all $M_2 \in \mathcal{E}_2$, it holds that $T_2^M(T_1^M(M_1)) \equiv M_1$ and $T_1^M(T_2^M(M_2)) \equiv M_2$. $\dashv$

Our third condition states that T_2^M and T_1^M are inverse bijections of each other (modulo some natural congruence). Hence, our definition is set in such a way that we compare the relative expressivity of each logic by comparing them over their whole class of models, taking into account the specificities of *all* the models of each logic in the comparison. Moreover, if two logics over the same class of models are as expressive in the previous sense, they are also essentially equi-expressive: it suffices to take the identity mappings for T_1^M and T_2^M. Our definition of equi-expressivity entails also the following two facts, which generalize the two previous expressions (1) and (2): for all $\varphi_1 \in \mathcal{L}_1$ and all $\varphi_2 \in \mathcal{L}_2$,

$$T_1^M\left(\{M_1 \in \mathcal{E}_1 \mid M_1 \models \varphi_1\}\right) \equiv \{M_2 \in \mathcal{E}_2 \mid M_2 \models T_1^\varphi(\varphi_1)\} \qquad \varphi_1 \equiv T_2^\varphi(T_1^\varphi(\varphi_1)) \tag{3}$$

$$T_2^M\left(\{M_2 \in \mathcal{E}_2 \mid M_2 \models \varphi_2\}\right) \equiv \{M_1 \in \mathcal{E}_1 \mid M_1 \models T_2^\varphi(\varphi_2)\} \qquad \varphi_2 \equiv T_1^\varphi(T_2^\varphi(\varphi_2)) \tag{4}$$

We can refine our notion of essential equi-expressivity even more as follows.

Definition 3 (Equi-expressivity)**.** A logic $\mathsf{L}_1 = (\mathcal{L}_1, \mathcal{E}_1, \models_1)$ is *as expressive* as a logic $\mathsf{L}_2 = (\mathcal{L}_2, \mathcal{E}_2, \models_2)$ when it is possible to split up $\mathcal{L}_1$ and $\mathcal{L}_2$ into a partition of sublanguages $\mathcal{L}_1 = \biguplus_{i \in I} \mathcal{L}_1^i$ and $\mathcal{L}_2 = \biguplus_{i \in J} \mathcal{L}_2^j$ such that for each $i \in I$ there is $j \in J$ such that $(\mathcal{L}_1^i, \mathcal{E}_1, \models_1)$ is essentially as expressive as $(\mathcal{L}_2^j, \mathcal{E}_2, \models_2)$, and vice versa. In that case, we write $\mathsf{L}_1 \equiv \mathsf{L}_2$. $\dashv$

Essential equi-expressivity obviously entails equi-expressivity (one does not need any splitting in that case) but the converse need not hold. The idea underlying equi-expressivity is that the translation mechanism for formulas may not be the same for the whole language. If it is still possible to split it into sublanguages such that for each of them we can find a translation mechanism, typically different from the others or dedicated to it, then we will still say that the two logics are equally expressive. Expressions (3) and (4) still hold in the case of equi-expressivity.

We canonically extend our definitions of equi–expressivity to classes of logics.

Definition 4 (Equi-expressivity of classes of logics)**.** Let $\mathbb{L}_1$ and $\mathbb{L}_2$ be two classes of logics. We say that $\mathbb{L}_1$ is *as expressive as* $\mathbb{L}_2$, written $\mathbb{L}_1 \equiv \mathbb{L}_2$, when for all $\mathsf{L}_1 \in \mathbb{L}_1$ there is $\mathsf{L}_2 \in \mathbb{L}_2$ such that $\mathsf{L}_1 \equiv \mathsf{L}_2$, and vice versa. $\dashv$

In a sense, our definition of equi-expressivity can be viewed as a partial solution to the so–called "identity problem" of universal logic [9] for logics which are defined semantically by triples like in the previous section. Our proposal is different from the one of García-Matos & Väänänen [21], although they deal with a more general notion of embedding between logics based on different classes of models.

3 Protologics, Atomic and Molecular Logics

Non-classical logics have common features: their syntax is defined by means of connectives, like $\wedge, \vee, \neg, \rightarrow, \square, \Diamond, \otimes, \supset, \subset, \supset, \ldots$; there is not any explicit variable quantification and no variables appear in formulas; they can be given a Kripke-style relational semantics by means of specific structures; the semantics of their connectives are defined by means of truth conditions.

Our overall approach is based on the observation that we can view truth conditions as formulas of first-order logic and that the models considered are very often specific kinds of structures. We revisit below the definitions of non-classical logics of Section 2 and provide the first-order formula $\star(x)$ or $c(x)$ with one free variable x corresponding to the respective truth condition of the non-classical connective. These first-order formulas are written in a specific form that will be explained and become clear later in the article.

- $(M,w) \models \square_i p$ iff for all $v \in M$, if $R_i wv$ then $(M,v) \models p$

  $\star_1(x) \triangleq \forall y(\mathsf{P}y \vee \neg \mathsf{R}_i xy)$
- $(M,w) \models p \otimes q$ iff there are $v,u \in M$ such that $Rvuw$, $(M,v) \models p$ and $(M,u) \models q$

  $\star_2(x) \triangleq \exists yz(\mathsf{P}y \wedge \mathsf{Q}z \wedge \mathsf{R}yzx)$
- $(M,w) \models p \supset q$ iff for all $v,u \in M$ such that $Rwvu$, if $(M,v) \models p$ then $(M,u) \models q$

  $\star_2(x) \triangleq \forall yz(\neg \mathsf{P}y \vee \mathsf{Q}z \vee \neg \mathsf{R}xyz)$
- $(M,w) \models q \subset p$ iff for all $v,u \in M$ such that $Rvwu$ if $(M,v) \models p$ then $(M,u) \models q$

  $\star_2(x) \triangleq \forall yz(\neg \mathsf{P}y \vee \mathsf{Q}z \vee \neg \mathsf{R}yxz)$
- $(M,w) \models \star(\varphi_1,\ldots,\varphi_n)$ iff there are $w_1,\ldots,w_n \in W$ such that $R_\star w_1 \ldots w_n w$ and $(M,w_1) \models \varphi_1$ and ... and $(M,w_n) \models \varphi_n$

  $\star_n(x) \triangleq \exists x_1 \ldots x_n (\mathsf{P}_1 x_1 \wedge \ldots \mathsf{P}_n x_n \wedge \mathsf{R}x_1 \ldots x_n x)$
- $(M,w) \models \square\varphi$ iff for all $v \in W$ such that Rwv, for all $u \in W$ such that $R_\Diamond vu$, $(M,u) \models \varphi$

  $c_1(x) \triangleq \forall y(\forall z(\mathsf{P}z \vee \neg \mathsf{R}_\Diamond yz) \vee \neg \mathsf{R}xy)$
- $(M,w) \models U(\varphi,\psi)$ iff there is $v \in W$ such that $w < v$ and $(M,v) \models \varphi$ and for all $u \in W$ such that $w < u < v$, $(M,u) \models \psi$

  $c_2(x) \triangleq \exists z\,(\mathsf{P}z \wedge \forall y\,(\mathsf{Q}y \vee \neg(x < y < z)) \wedge x < z)$

3.1 Protologics

In this section, $\mathcal{P}$ is a set of predicates symbols and the sets of constants and function symbols of first-order logics are empty, $\mathcal{C} = \mathcal{F} = \emptyset$.

Definition 5 (Abstract connectives)**.** The *abstract propositional letters* $\mathbb{P}^a$ are a subset $\mathcal{Q} \subseteq \mathcal{P}$ of the predicate symbols $\mathcal{P}$ and the *abstract connectives* $\mathbb{C}^a$ are the formulas of $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ together with a sequence of distinct predicate symbols that occur in each formula. That is,

$$\mathbb{P}^a \triangleq \mathcal{Q}$$

$$\mathbb{C}^a \triangleq \left\{ (\chi(\overline{x}), (\mathsf{Q}_1, \dots, \mathsf{Q}_n)) \mid \chi \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}) \text{ and } \mathsf{Q}_1, \dots, \mathsf{Q}_n \in \mathcal{Q} \text{ all distinct appear in } \chi \right\}$$

Let p be an abstract propositional letter whose arity as a predicate symbol is k. Then, the *arity* of p is 0 and its *output type* is k. Let $\star = (\chi(x_1, \dots, x_k), (\mathsf{Q}_1, \dots, \mathsf{Q}_n)) \in \mathbb{C}^a$ where χ is a formula with k free variables and such that the predicates $\mathsf{Q}_1, \dots, \mathsf{Q}_n$ are of arity $k_1, \dots, k_n$ respectively. Then, the *arity* of $\star$ is n, its *type* is $(k, k_1, \dots, k_n)$ and its *output type* is k, denoted $k(\star)$. The predicate symbols of $\chi(\overline{x})$ which do not belong to $\{\mathsf{Q}_1, \dots, \mathsf{Q}_n\}$ are called the *parameter predicates*. ⊣

Example 1. Let us consider the formula $\chi(x) \triangleq \forall y(\forall z(\mathsf{Q}z \vee \neg \mathsf{R}_{\diamond}yz) \vee \neg \mathsf{R}xy)$ or equivalently $\chi(x) \triangleq \forall y(\mathsf{R}xy \to \forall z(\mathsf{R}_{\diamond}yz \to \mathsf{Q}(z)))$ corresponding to the truth condition of the box operator of modal intuitionistic logic. The connective $(\chi(x), (\mathsf{Q}))$ of $\mathbb{C}^a$ of arity 1, of type $(1, 1)$ and of output type 1 corresponds to the connective of modal intuitionistic logic. Its parameter predicates are R and $\mathsf{R}_{\diamond}$. We could define other connectives based on $\chi(x)$ such as $(\chi(x), (\mathsf{R}_{\diamond}, \mathsf{Q}))$ and $(\chi(x), (\mathsf{R}, \mathsf{R}_{\diamond}, \mathsf{Q}))$ of arities 2 and 3 and of types $(1, 1, 1)$ and $(1, 1, 1, 1)$ respectively, possibly with the predicates ordered differently in the tuples. The parameter predicate in $(\chi(x), (\mathsf{R}_{\diamond}, \mathsf{Q}))$ is R and there is none in $(\chi(x), (\mathsf{R}, \mathsf{R}_{\diamond}, \mathsf{Q}))$. ⊣

Definition 6 (Protolanguage)**.** The *protolanguage* $\mathcal{L}$ is the smallest set that contains the abstract propositional letters and that is closed under the other abstract connectives, while respecting the type constraints:

- $\mathbb{P}^a \subseteq \mathcal{L}$;
- for all $\star \in \mathbb{C}^a$ of type $(k, k_1, \dots, k_n)$ and for all $\varphi_1, \dots, \varphi_n \in \mathcal{L}$ of respective output types $k_1, \dots, k_n$, we have $\star(\varphi_1, \dots, \varphi_n) \in \mathcal{L}$. The *output type* of $\star(\varphi_1, \dots, \varphi_n)$ is k.

If $\mathsf{C}^a \subseteq \mathbb{C}^a$ is such that $\mathsf{C}^a \cap \mathbb{P}^a \neq \emptyset$, then an element of $\mathcal{L}_{\mathsf{C}^a}$ is an element of $\mathcal{L}$ that contains only connectives of C^a. *In the sequel, we always assume that all $\mathsf{C}^a \subseteq \mathbb{C}^a$ are such that $\mathsf{C}^a \cap \mathbb{P}^a \neq$* $\emptyset$. Elements of $\mathcal{L}$ are called *protoformulas* and are generally denoted φ, ψ, α. ⊣

Example 2. If we want to recover the language of modal intuitionistic logic with only the box modality then we consider the set of connectives $\mathsf{C} = \{p, (\chi(x), (\mathsf{Q})) \mid p \in \mathcal{Q} \text{ is of arity } 1\}$ where $\chi(x)$ is the formula of the previous example. ⊣

Definition 7 (C^a–model)**.** Let $\mathsf{C}^a \subseteq \mathbb{C}^a$ be a finite set of connectives. A *C^a–model* is a structure $M = (W, \mathcal{R})$ for $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ where W is a non-empty set and $\mathcal{R}$ is a set of relations over W such that each predicate $\mathsf{Q} \in \mathcal{Q}$ is associated to a relation Q of the same arity as Q and such that the parameter predicates $\{\mathsf{R}_1, \dots, \mathsf{R}_m\}$ of each connective $\star = (\chi(\overline{x}), (\mathsf{Q}_1, \dots, \mathsf{Q}_n)) \in \mathsf{C}^a$ can be associated to a subset of the relations $\mathcal{R}_\star = \{R_1, \dots, R_m\} \subseteq \mathcal{R}$ interpreting the predicates $\mathsf{R}_1, \dots, \mathsf{R}_m$ (possibly with some overlap for different connectives).

A *C^a–assignment for M* is a tuple of W^k, generally denoted $\overline{w}$, where $k = k(\star)$ for some $\star \in \mathsf{C}^a$. The set of all C^a–assignments for M is denoted $\overline{w}(M, \mathsf{C}^a)$. A *pointed C^a–model* $(M, \overline{w})$ is a C^a–model M together with a C^a–assignment $\overline{w}$ for M. The class of all pointed $\mathbb{C}^a$–models is denoted $\mathcal{M}^a$. ⊣

Example 3. If we resume our previous example, a modal intuitionistic model $M = (W, \{R, R_\Diamond, P_1, \ldots, P_n, \ldots\})$ is a C–model. The parameter predicates $\{\mathsf{R}, \mathsf{R}_\Diamond\}$ of $(\chi(x), (\mathsf{Q}))$ are associated to the relations $\{R, R_\Diamond\}$ and the predicates $p_n \in \mathcal{Q}$ are associated to the relations P_n. ⊣

Definition 8 (Model $M[P_i := W_i]$)**.** Let $\mathsf{C}^a \subseteq \mathbb{C}^a$ be a finite set of connectives, let M be a C–model containing relations $Q_1, \ldots, Q_n$ of arity $k_1, \ldots, k_n$ respectively and let $W_1 \in \mathcal{P}(W^{k_1}), \ldots, W_k \in \mathcal{P}(W^{k_n})$. We define the C–model $M[Q_i := W_i]$ as the C–model M where (the interpretation of the predicates $\mathsf{Q}_1, \ldots, \mathsf{Q}_n$ by) the relations $Q_1, \ldots, Q_n$ are replaced by the relations $W_1, \ldots, W_n$ (viewed as sets) respectively, all the rest being the same. ⊣

Definition 9 (Truth function associated to a connective of $\mathbb{C}^a$)**.** Let $\star = (\chi(\overline{x}), (\mathsf{Q}_1, \ldots, \mathsf{Q}_n)) \in \mathbb{C}^a$ be a connective of type $(k, k_1, \ldots, k_n)$ and let $M = (W, \mathcal{R})$ be a C^a–model such that $(\chi(\overline{x}), (\mathsf{Q}_1, \ldots, \mathsf{Q}_n)) \in \mathsf{C}^a$. The $(k, k_1, \ldots, k_n)$*–ary truth function* $f_\star : \mathcal{P}(W^{k_1}) \times \ldots \times \mathcal{P}(W^{k_n}) \to \mathcal{P}(W^k)$ *associated to* $\star$ on M is defined as follows:

- if $n = 0$ and $\star = \mathsf{Q}$, $f_\star \triangleq Q$;
- if $n > 0$, then for all $W_1 \in \mathcal{P}(W^{k_1}), \ldots,$ all $W_n \in \mathcal{P}(W^{k_n})$ we define

$$f_\star(W_1, \ldots, W_n) \triangleq \left\{(w_1, \ldots, w_k) \in W^k \mid M[Q_i := W_i] \models \chi[x_1/w_1, \ldots, x_k/w_k]\right\}$$ ⊣

In the above definition, M is really taken as a structure of first-order logic. The choice of the assignment s in the evaluation does not play a role in the determination of $f_\star$ since the only value of variables that matter for that, $(x_1, \ldots, x_k)$, are given by the definition.

Example 4. One can easily check that the truth functions associated to the connectives $(\star_1(x), (\mathsf{Q})), (\star_2(x), (\mathsf{Q}_1, \mathsf{Q}_2)), (\star_3(x), (\mathsf{Q}_1, \mathsf{Q}_2))$ and $(\star_4(x), (\mathsf{Q}_1, \mathsf{Q}_2))$ at the begining of Section 3 correspond to the truth conditions of the connectives defined above each of them respectively. For example, for the case of modal intuitionistic logic, if $\star = (\chi(x), (\mathsf{Q}))$ then $f_\star(W_1) = \{w \in W \mid M[P := W_1] \models \chi(x)\} = \{w \in W \mid M[P := W_1] \models \forall y(\forall z(\mathsf{Q}z \lor \neg \mathsf{R}_\Diamond yz) \lor \neg \mathsf{R}xy)[x/w]\} = \{w \in W \mid$ for all $v \in W$ such that Rwv, for all $u \in W$ such that $R_\Diamond vu$, $u \in W_1\}$. ⊣

Definition 10 (Protologic)**.** Let $\mathsf{C}^a \subseteq \mathbb{C}^a$ and let $M = (W, \mathcal{R})$ be a C^a–model. The *extension function of* $\mathcal{L}_{\mathsf{C}^a}$ *in* M, denoted $[\![\cdot]\!]^M : \mathcal{L}_{\mathsf{C}^a} \to \overline{w}(M, \mathsf{C}^a)$, is defined inductively as follows: for all $p \in \mathsf{C}^a \cap \mathbb{P}^a$ and all $\star \in \mathsf{C}^a$ of arity $n > 0$ and type $(k, k_1, \ldots, k_n)$, for all $\varphi_1, \ldots, \varphi_n \in \mathcal{L}_{\mathsf{C}^a}$ of respective output types $k_1, \ldots, k_n$,

$$\begin{aligned} [\![p]\!]^M &\triangleq f_p \\ [\![\star(\varphi_1, \ldots, \varphi_n)]\!]^M &\triangleq f_\star\left([\![\varphi_1]\!]^M, \ldots, [\![\varphi_n]\!]^M\right) \end{aligned}$$

where $f_\star$ is the truth function associated to the formula $\star \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$.

If $\mathcal{E}_{\mathsf{C}^a}$ is a class of pointed C^a–models, we define the *satisfaction relation* $\Vdash \subseteq \mathcal{E}_{\mathsf{C}^a} \times \mathcal{L}_{\mathsf{C}^a}$ as follows: for all $\varphi \in \mathcal{L}_{\mathsf{C}^a}$ and all $(M, \overline{w}) \in \mathcal{E}_{\mathsf{C}^a}$, we set $(M, \overline{w}) \Vdash \varphi$ iff $\overline{w} \in [\![\varphi]\!]^M$. The triple $(\mathcal{L}_{\mathsf{C}^a}, \mathcal{E}_{\mathsf{C}^a}, \Vdash)$ is called the *protologic associated to* $\mathcal{E}_{\mathsf{C}^a}$ *and* C^a. ⊣

When the type of φ does not match the size of the assignment $\overline{w} \in \overline{w}(M, \mathsf{C})$ of a pointed C^a–model, then it is not the case that $((M, \overline{w}), \varphi) \in \Vdash$. That is, $(M, \overline{w}) \Vdash \varphi$ does not hold. In that case, we do not say that φ is "undefined" or "false" at the pointed C^a–model $(M, \overline{w})$, what holds is that the pair $((M, \overline{w}), \varphi)$ does not belong to the relation $\Vdash$.

One may argue that protologics do not really deserve their qualification of being somehow 'primal' since they seem at first sight to be able to encode only two–valued logics. This is not the case, as the following example shows.

Example 5 (Many–valued logics)**.** Many–valued logics associated to a set of connectives C are examples of protologics. The first–order formulas $\varphi(x) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ on which they are based and that define the connectives are of the form $\exists x_1 \dots x_n(\mathsf{Q}_1 x_1 \wedge \dots \wedge \mathsf{Q}_n x_n \wedge \mathsf{R}_\star x_1 \dots x_n x)$. ⊣

3.2 Atomic Logics

Atomic logics are protologics whose connective skeletons are defined by first-order formulas of the form $\forall x_1 \dots x_n(\pm_1 \mathsf{Q}_1 x_1 \vee \dots \vee \pm_n \mathsf{Q}_n x_n \vee \pm \mathsf{R} x_1 \dots x_n x)$ or $\exists x_1 \dots x_n(\pm_1 \mathsf{Q}_1 x_1 \wedge \dots \wedge \pm_n \mathsf{Q}_n x_n \wedge \pm \mathsf{R} x_1 \dots x_n x)$ where the $\pm_i$s are either empty or $\neg$. We will represent the structure of these formulas by means of so–called *skeletons* whose various arguments capture the different features and patterns from which they can be redefined completely.

We recall that $\mathbb{N}^*$ denotes the set of natural numbers minus 0 and that for all $n \in \mathbb{N}^*$, $\mathfrak{S}_n$ denotes the group of permutations over the set $\{1, \dots, n\}$. Permutations are generally denoted σ, τ, the identity permutation Id is sometimes denoted 1 as the neutral element of every permutation group and σ^- stands for the inverse permutation of the permutation σ (see for instance [38] for more details).

Definition 11 (Atomic skeletons and connectives)**.** The sets of *atomic skeletons* $\mathbb{P}$ and $\mathbb{C}$ are defined as follows:

$$\mathbb{P} \triangleq \mathfrak{S}_1 \times \{+,-\} \times \{\forall, \exists\} \times \mathbb{N}^*$$
$$\mathbb{C} \triangleq \mathbb{P} \cup \bigcup_{n \in \mathbb{N}^*} \left\{ \mathfrak{S}_{n+1} \times \{+,-\} \times \{\forall, \exists\} \times \mathbb{N}^{*n+1} \times \{+,-\}^n \right\}.$$

$\mathbb{P}$ is called the set of *propositional letter skeletons* and $\mathbb{C}$ is called the set of *connective skeletons*. They can be represented by tuples $(\sigma, \pm, Æ, \overline{k}, \overline{\pm_j})$ or $(\sigma, \pm, Æ, k)$ if it is a propositional letter skeleton, where $Æ \in \{\forall, \exists\}$ is called the *quantification signature* of the skeleton, $\overline{k} = (k, k_1, \dots, k_n) \in \mathbb{N}^{*n+1}$ is called the *type signature* of the skeleton and $\overline{\pm_j} = (\pm_1, \dots, \pm_n) \in \{+,-\}^n$ is called the *tonicity signature* of the skeleton; $(Æ, \overline{k}, \overline{\pm_j})$ is called the *signature* of the skeleton. The *arity* of a propositional letter skeleton is 0 and its *type* is k. The *arity* of a skeleton $\star \in \mathbb{C}$ is n, its *input types* are $k_1, \dots, k_n$ and its *output type* is k.

A *(atomic) connective* or *(atomic) propositional letter* is an object to which is associated a (atomic) skeleton. Its arity, signature, quantification signature, type signature, tonicity signature, input and output types are the same as its skeleton. By abuse, we sometimes identify a connective with its skeleton. We also introduce the *Boolean connectives* called *conjunctions and disjunctions*:

$$\mathbb{B} \triangleq \{\wedge_k, \vee_k \mid k \in \mathbb{N}^*\}$$

The type signatures of $\wedge_k$ and $\vee_k$ are (k, k, k) and their arity is 2.

We say that a set of atomic connectives C *is complete for conjunction and disjunction* when it contains all conjunctions and disjunctions $\wedge_k, \vee_k$, for k ranging over all input types and output types of the atomic connectives of C. The set of atomic skeletons associated to C is denoted $\star(\mathsf{C})$, its propositional letters is denoted $\mathbb{P}(\mathsf{C})$.

Propositional letters are denoted p, p_1, p_2, *etc.* and connectives are denoted $\star, \star_1, \star_2$, *etc.* ⊣

We need to distinguish between connectives and skeletons because in general we need a countable number of propositional letters or connectives of the same skeleton, like in some modal logics, where we need multiple modalities of the same (similarity) type/skeleton.

Remark 3. The permutations σ mentioned in atomic skeletons do not really play a role in this article. Permutations play an important role in the proof theory of atomic logics, which is dealt with in [6, 7].

Definition 12 (Atomic language)**.** Let C be a set of atomic connectives. The *(typed) atomic language* $\mathcal{L}_{\mathsf{C}}$ associated to C is the smallest set that contains the propositional letters and that is closed under the atomic connectives. That is,

- $\mathbb{P}(\mathsf{C}) \subseteq \mathcal{L}_{\mathsf{C}}$;
- for all $\star \in \mathsf{C}$ of arity $n > 0$ and of type signature $(k, k_1, \ldots, k_n)$ and for all $\varphi_1, \ldots, \varphi_n \in \mathcal{L}_{\mathsf{C}}$ of types $k_1, \ldots, k_n$ respectively, we have that $\star(\varphi_1, \ldots, \varphi_n) \in \mathcal{L}_{\mathsf{C}}$ and $\star(\varphi_1, \ldots, \varphi_n)$ is of *type* k.

The *Boolean atomic language* $\mathcal{L}_{\mathsf{C}}^{\mathbb{B}}$ is the smallest set that contains the propositional letters and that is closed under the atomic connectives of C as well as the Boolean connectives $\mathbb{B}$:

- for all $\varphi, \psi \in \mathcal{L}_{\mathsf{C}}^{\mathbb{B}}$ of type k, we have that $(\varphi \wedge_k \psi), (\varphi \vee_k \psi) \in \mathcal{L}_{\mathsf{C}}^{\mathbb{B}}$.

Elements of $\mathcal{L}_{\mathsf{C}}$ are called *atomic formulas* and are denoted $\varphi, \psi, \alpha, \ldots$ The *type of a formula* $\varphi \in \mathcal{L}_C$ is denoted $k(\varphi)$. For all $\varphi_1, \ldots, \varphi_n \in \mathcal{L}$ of type k, $\bigwedge \{\varphi_1, \ldots, \varphi_n\}$ and $\bigvee \{\varphi_1, \ldots, \varphi_n\}$ stand for $((\varphi_1 \wedge_k \varphi_2) \wedge_k \ldots \wedge_k \varphi_n)$ and $((\varphi_1 \vee_k \varphi_2) \vee_k \ldots \vee_k \varphi_n)$ respectively. When it is clear from the context, we will omit the subscript k in $\wedge_k, \vee_k$ and write them $\wedge, \vee$.

The *skeleton syntactic tree* of a formula $\varphi \in \mathcal{L}_{\mathsf{C}}$ is the syntactic tree of the formula φ in which the nodes labeled with subformulas of φ are replaced by the skeleton of their outermost connective.

In the sequel, we assume that all sets of connectives C *are such that they contain at least a propositional letter.* ⊣

Definition 13 (C–models)**.** Let C be a set of atomic connectives. A *C–model* is a tuple $M = (W, \mathcal{R})$ where W is a non-empty set and $\mathcal{R}$ is a set of relations over W such that each n–ary connective $\star \in \mathsf{C}$ which is not a Boolean connective of type signature $(k, k_1, \ldots, k_n)$ is associated to a $k_1 + \ldots + k_n + k$–ary relation $R_\star \in \mathcal{R}$.

An *assignment* is a tuple $(w_1, \ldots, w_k) \in W^k$ for some $k \in \mathbb{N}^*$, generally denoted $\overline{w}$. A *pointed C–model* $(M, \overline{w})$ is a C–model M together with an assignment $\overline{w}$. In that case, we say that $(M, \overline{w})$ *is of type* k. The class of all pointed C–models is denoted $\mathcal{M}_{\mathsf{C}}$. ⊣

Note that a C–model can be canonically seen as a structure, for some appropriate set of predicates $\mathcal{P}$ associated to the relations of $\mathcal{R}$.

Definition 14 (Atomic logics)**.** Let C be a set of atomic connectives and let $M = (W, \mathcal{R})$ be a C–model. We define the *interpretation function of $\mathcal{L}_{\mathsf{C}}$ in M*, denoted $[\![\cdot]\!]^M : \mathcal{L}_{\mathsf{C}} \to \bigcup\limits_{k \in \mathbb{N}^*} W^k$, inductively as follows: for all propositional letters $p \in \mathsf{C}$, all connectives $\star \in \mathsf{C}$ of skeleton $(\sigma, \pm, \text{Æ}, (k, k_1, \ldots, k_n), (\pm_1, \ldots, \pm_n))$ of arity $n > 0$ and all $k \in \mathbb{N}^*$, for all $\varphi, \psi, \varphi_1, \ldots, \varphi_n \in \mathcal{L}_{\mathsf{C}}$, if $k(\varphi) = k(\psi) = k$,

$$
\begin{aligned}
[\![p]\!]^M &\triangleq \pm R_p \\
[\![(\varphi \wedge_k \psi)]\!]^M &\triangleq [\![\varphi]\!]^M \cap [\![\psi]\!]^M \\
[\![(\varphi \vee_k \psi)]\!]^M &\triangleq [\![\varphi]\!]^M \cup [\![\psi]\!]^M \\
[\![\star(\varphi_1, \ldots, \varphi_n)]\!]^M &\triangleq f_\star([\![\varphi_1]\!]^M, \ldots, [\![\varphi_n]\!]^M)
\end{aligned}
$$

where the function $f_\star$ is defined as follows: for all $W_1 \in \mathcal{P}(W^{k_1}), \ldots, W_n \in \mathcal{P}(W^{k_n})$, $f_\star(W_1, \ldots, W_n) \triangleq \{\overline{w}_{n+1} \in W^k \mid \mathcal{C}^\star(W_1, \ldots, W_n, \overline{w}_{n+1})\}$ where $\mathcal{C}^\star(W_1, \ldots, W_n, \overline{w}_{n+1})$ is called the *truth condition* of $\star$ and is defined as follows:

- if Æ = ∀: "$\forall \overline{w}_1 \in W^{k_1} \ldots \overline{w}_n \in W^{k_n} (\overline{w}_1 \pitchfork_1 W_1 \vee \ldots \vee \overline{w}_n \pitchfork_n W_n \vee R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w}_{n+1})$";

Permutations of $\mathfrak{S}_2$	unary signatures
$\tau_1 = (1,2)$	$t_1 = (\exists,(1,1),+)$
$\tau_2 = (2,1)$	$t_2 = (\forall,(1,1),+)$
	$t_3 = (\forall,(1,1),-)$
	$t_4 = (\exists,(1,1),-)$

Permutations of $\mathfrak{S}_3$	binary signatures
$\sigma_1 = (1,2,3)$	$s_1 = (\exists,(1,1,1),(+,+))$
$\sigma_2 = (3,2,1)$	$s_2 = (\forall,(1,1,1),(+,-))$
$\sigma_3 = (3,1,2)$	$s_3 = (\forall,(1,1,1),(-,+))$
$\sigma_4 = (2,1,3)$	$s_4 = (\forall,(1,1,1),(+,+))$
$\sigma_5 = (2,3,1)$	$s_5 = (\exists,(1,1,1),(+,-))$
$\sigma_6 = (1,3,2)$	$s_6 = (\exists,(1,1,1),(-,+))$
	$s_7 = (\exists,(1,1,1),(-,-))$
	$s_8 = (\forall,(1,1,1),(-,-))$

Figure 1: Permutations of $\mathfrak{S}_2$ and $\mathfrak{S}_3$ and 'families' of unary and binary signatures

- if Æ $= \exists$: "$\exists \overline{w}_1 \in W^{k_1} \ldots \overline{w}_n \in W^{k_n} \; (\overline{w}_1 \pitchfork_1 W_1 \wedge \ldots \wedge \overline{w}_n \pitchfork_n W_n \wedge R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w}_{n+1})$";

where, for all $j \in [\![1;n]\!]$, $\overline{w}_j \pitchfork_j W_j \triangleq \begin{cases} \overline{w}_j \in W_j & \text{if } \pm_j = + \\ \overline{w}_j \notin W_j & \text{if } \pm_j = - \end{cases}$ and $R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_{n+1}$ holds iff $\pm R_\star \overline{w}_{\sigma^-(1)} \ldots \overline{w}_{\sigma^-(n+1)}$ with the notations $+R_\star \triangleq R_\star$ and $-R_\star \triangleq W^{k+k_1+\ldots+k_n} - R_\star$. If $\mathcal{E}_{\mathsf{C}}$ is a class of pointed C–models, the *satisfaction relation* $\Vdash \subseteq \mathcal{E}_{\mathsf{C}} \times \mathcal{L}_{\mathsf{C}}$ is defined as follows: for all $\varphi \in \mathcal{L}_{\mathsf{C}}$ and all $(M, \overline{w}) \in \mathcal{E}_{\mathsf{C}}$, $((M, \overline{w}), \varphi) \in \Vdash$ iff $\overline{w} \in [\![\varphi]\!]^M$. We usually write $(M, \overline{w}) \Vdash \varphi$ instead of $((M, \overline{w}), \varphi) \in \Vdash$ and we say that φ is *true* in $(M, \overline{w})$.

The class of *atomic logics*, denoted $\mathbb{L}_{\mathsf{GGL}}$, is defined by $\mathbb{L}_{\mathsf{GGL}} \triangleq \{(\mathcal{L}_{\mathsf{C}}, \mathcal{E}_{\mathsf{C}}, \Vdash) \mid \mathsf{C} \text{ is a finite set of atomic connectives and } \mathcal{E}_{\mathsf{C}} \text{ is a class of } \mathsf{C}\text{–models}\}$. The atomic logic $(\mathcal{L}_{\mathsf{C}}, \mathcal{E}_{\mathsf{C}}, \Vdash)$ is the *atomic logic associated to $\mathcal{E}_{\mathsf{C}}$ and C.* The logics of the form $(\mathcal{L}_{\mathsf{C}}, \mathcal{M}_{\mathsf{C}}, \Vdash)$ are called *basic atomic logics*. We call them *Boolean (basic) atomic logics* when their language includes the Boolean connectives $\mathbb{B}$. ⊣

We stress that the $\pm$ sign in $R_\star^{\pm\sigma}$ is the $\pm$ sign in $(\sigma, \pm, \text{Æ}, (k, k_1, \ldots, k_n), (\pm_1, \ldots, \pm_n))$.

Example 6 (Lambek calculus, modal logic, many–valued logics)**.** The Lambek calculus, where $\mathsf{C} = \{p, \circ, \backslash, /\}$ is defined in Section 2.2, is an example of atomic logic. Here $\circ, \backslash, /$ are the connectives of skeletons $(\sigma_1, +, s_1), (\sigma_5, -, s_3), (\sigma_3, -, s_2)$. Another example of atomic logic is modal logic where $\mathsf{C} = \{p, \top, \bot, \wedge, \vee, \Diamond, \Box\}$ is such that

- $\top, \bot$ are connectives of skeletons $(1, +, \exists, 1)$ and $(1, -, \forall, 1)$ respectively;
- $\wedge, \vee, \Diamond, \Box$ are connectives of skeletons $(\sigma_1, +, s_1)$, $(\sigma_1, -, s_4)$, $(\tau_2, +, t_1)$ and $(\tau_2, -, t_2)$ respectively;
- the C-models $M = (W, \mathcal{R}) \in \mathcal{E}_{\mathsf{C}}$ are such that $R_\wedge = R_\vee = \{(w, w, w) \mid w \in W\}$, $R_\Diamond = R_\Box$ and $R_\top = R_\bot = W$.

Indeed, one can easily show that, with these conditions on the C–models of $\mathcal{E}_{\mathsf{C}}$, we have that for all $(M, w) \in \mathcal{E}_{\mathsf{C}}$, $(M, w) \Vdash \wedge(\varphi, \psi)$ iff $(M, w) \Vdash \varphi$ and $(M, w) \Vdash \psi$, and $(M, w) \Vdash \vee(\varphi, \psi)$ iff $(M, w) \Vdash \varphi$ or $(M, w) \Vdash \psi$. The Boolean conjunction and disjunction $\wedge$ and $\vee$ are defined using the connectives of $\mathbb{C}$ by means of special relations $R_\wedge$ and $R_\vee$. However, they could obviously be defined directly. Many–valued logics are also examples of atomic logics. Many more examples of atomic connectives are given in Figures 2 and 3. They are all of type signature $(1, 1, \ldots, 1)$. ⊣

Atomic Connective	Truth condition	Non–classical connective in the literature
The existentially positive orbit		
$(\tau_1, +, t_1)\ \varphi$	$\exists v\,(v \in [\![\varphi]\!] \wedge Rvw)$	$\Diamond^{-}\varphi$ [36] $\Diamond_{\downarrow}$ [15]
$(\tau_2, -, t_2)\ \varphi$	$\forall v\,(v \in [\![\varphi]\!] \vee -Rwv)$	$\Box\varphi$ [27]
The abstractly positive orbit		
$(\tau_1, +, t_2)\ \varphi$	$\forall v\,(v \in [\![\varphi]\!] \vee Rvw)$	$+_{\downarrow}\varphi$ [15] [17, p. 401]
$(\tau_2, -, t_1)\ \varphi$	$\exists v\,(v \in [\![\varphi]\!] \wedge -Rwv)$	[15]
The existentially negative orbit		
$(\tau_1, +, t_4)\ \varphi$	$\exists v\,(v \notin [\![\varphi]\!] \wedge Rvw)$	$?\varphi$ [15][17, p. 402] $\boxminus_1\varphi$ [15][11, Def. 10.7.7]
$(\tau_2, +, t_4)\ \varphi$	$\exists v\,(v \notin [\![\varphi]\!] \wedge Rwv)$	$?_{\downarrow}\varphi$ [15][18] [17, p. 402] $\boxminus_2\varphi$ [11, Def. 10.7.7]
The abstractly negative orbit		
$(\tau_1, +, t_3)\ \varphi$	$\forall v\,(v \notin [\![\varphi]\!] \vee Rvw)$	$\varphi^{\perp}$ [15, 16] $\varphi^{\mathbf{0}}$ [23] $\diamondminus_1\varphi$ [11, Def. 10.7.2]
$(\tau_2, +, t_3)\ \varphi$	$\forall v\,(v \notin [\![\varphi]\!] \vee Rwv)$	$\sim\varphi$ [22] ${}^{\perp}\varphi$ [15, 16] ${}^{\mathbf{0}}\varphi$ [23] $\diamondminus_2\varphi$ [11, Def. 10.7.2]
The symmetrical existentially positive orbit		
$(\tau_1, -, t_1)\ \varphi$	$\exists v\,(v \in [\![\varphi]\!] \wedge -Rvw)$	[15]
$(\tau_2, +, t_2)\ \varphi$	$\forall v\,(v \in [\![\varphi]\!] \vee Rwv)$	$+\varphi$ [15] [17, p. 402] φ^{*} [11, Def. 7.1.19]
The symmetrical abstractly positive orbit		
$(\tau_1, -, t_2)\ \varphi$	$\forall v\,(v \in [\![\varphi]\!] \vee -Rvw)$	$\Box^{-}\varphi$ [36] $\Box_{\downarrow}$ [15]
$(\tau_2, +, t_1)\ \varphi$	$\exists v\,(v \in [\![\varphi]\!] \wedge Rwv)$	$\Diamond\varphi$ [27]
The symmetrical existentially negative orbit		
$(\tau_1, -, t_4)\ \varphi$	$\exists v\,(v \notin [\![\varphi]\!] \wedge -Rvw)$	$?\varphi$ [15][11, Ex. 1.4.5] $\varphi^{\mathbf{1}}$ [23]
$(\tau_2, -, t_4)\ \varphi$	$\exists v\,(v \notin [\![\varphi]\!] \wedge -Rwv)$	$?_{\downarrow}\varphi$ [15] [11, Ex. 1.4.5] ${}^{\mathbf{1}}\varphi$ [23]
The symmetrical abstractly negative orbit		
$(\tau_1, -, t_3)\ \varphi$	$\forall v\,(v \notin [\![\varphi]\!] \vee -Rvw)$	[15]
$(\tau_2, -, t_3)\ \varphi$	$\forall v\,(v \notin [\![\varphi]\!] \vee -Rwv)$	$\neg_h\varphi$ [28, 37] $\perp\varphi$ [18]

Figure 2: Unary connectives of atomic logics of type $(1, 1)$

Atomic connective	Truth condition	Non–classical connective in the literature
	The conjunction orbit	
$\varphi\ (\sigma_1, +, s_1)\ \psi$	$\exists vu\, (v \in [\![\varphi]\!] \wedge u \in [\![\psi]\!] \wedge Rvuw)$	$\varphi \circ \psi$ [30], $\varphi \otimes_3 \psi$ [5]
$\varphi\ (\sigma_2, -, s_2)\ \psi$	$\forall vu\, (v \in [\![\varphi]\!] \vee u \notin [\![\psi]\!] \vee -Rwuv)$	
$\varphi\ (\sigma_3, -, s_2)\ \psi$	$\forall vu\, (v \in [\![\varphi]\!] \vee u \notin [\![\psi]\!] \vee -Ruwv)$	/ [30], $\varphi \subset_2 \psi$ [5]
$\varphi\ (\sigma_4, +, s_1)\ \psi$ $= \psi\ (\sigma_1, +, s_1)\ \varphi$	$\exists vu\, (v \in [\![\varphi]\!] \wedge u \in [\![\psi]\!] \wedge Ruvw)$	
$\varphi\ (\sigma_5, -, s_3)\ \psi$ $= \psi\ (\sigma_2, -, s_2)\ \varphi$	$\forall vu\, (v \notin [\![\varphi]\!] \vee u \in [\![\psi]\!] \vee -Rwvu)$	\ [30], $\varphi \supset_1 \psi$ [5]
$\varphi\ (\sigma_6, -, s_3)\ \psi$ $= \psi\ (\sigma_3, -, s_2)\ \varphi$	$\forall vu\, (v \notin [\![\varphi]\!] \vee u \in [\![\psi]\!] \vee -Rvwu)$	
	The not–but orbit	
$\varphi\ (\sigma_1, +, s_6)\ \psi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \in [\![\psi]\!] \wedge Rvuw)$	$\varphi \succ_3 \psi$ [5]
$\varphi\ (\sigma_2, +, s_6)\ \psi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \in [\![\psi]\!] \wedge Rwuv)$	
$\varphi\ (\sigma_3, -, s_4)\ \psi$	$\forall vu\, (v \in [\![\varphi]\!] \vee u \in [\![\psi]\!] \vee -Ruwv)$	$\varphi \oplus_2 \psi$ [5]
$\varphi\ (\sigma_4, +, s_5)\ \psi$ $= \psi\ (\sigma_1, +, s_6)\ \varphi$	$\exists vu\, (v \in [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Ruvw)$	
$\varphi\ (\sigma_5, +, s_5)\ \psi$ $= \psi\ (\sigma_2, +, s_6)\ \varphi$	$\exists vu\, (v \in [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rwvu)$	$\varphi \prec_1 \psi$ [5]
$\varphi\ (\sigma_6, -, s_4)\ \psi$ $= \psi\ (\sigma_3, -, s_4)\ \varphi$	$\forall vu\, (v \in [\![\varphi]\!] \vee u \in [\![\psi]\!] \vee -Rvwu)$	
	The but–not orbit	
$\varphi\ (\sigma_1, +, s_5)\ \psi$	$\exists vu\, (v \in [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rvuw)$	$\varphi \prec_3 \psi$ [5]
$\varphi\ (\sigma_2, -, s_4)\ \psi$	$\forall vu\, (v \in [\![\varphi]\!] \vee u \in [\![\psi]\!] \vee -Rwuv)$	
$\varphi\ (\sigma_3, +, s_6)\ \psi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \in [\![\psi]\!] \wedge Ruwv)$	$\varphi \succ_2 \psi$ [5]
$\varphi\ (\sigma_4, +, s_6)\ \psi$ $= \psi\ (\sigma_1, +, s_5)\ \varphi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \in [\![\psi]\!] \wedge Ruvw)$	$\varphi \obslash \psi$ [24, 31]
$\varphi\ (\sigma_5, -, s_4)\ \psi$ $= \psi\ (\sigma_2, -, s_4)\ \varphi$	$\forall vu\, (v \in [\![\varphi]\!] \vee u \in [\![\psi]\!] \vee -Rwvu)$	$\varphi \oplus \psi$ [24, 31] $\varphi \oplus_1 \psi$ [5]
$\varphi\ (\sigma_6, +, s_5)\ \psi$ $= \psi\ (\sigma_3, +, s_6)\ \varphi$	$\exists vu\, (v \in [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rvwu)$	$\varphi \oslash \psi$ [24, 31]
	The stroke orbit	
$\varphi\ (\sigma_1, +, s_7)\ \psi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rvuw)$	$\varphi \mid_3 \psi$ [1, 23]
$\varphi\ (\sigma_2, +, s_7)\ \psi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rwuv)$	
$\varphi\ (\sigma_3, +, s_7)\ \psi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Ruwv)$	
$\varphi\ (\sigma_4, +, s_7)\ \psi$ $= \psi\ (\sigma_1, +, s_7)\ \varphi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Ruvw)$	
$\varphi\ (\sigma_5, +, s_7)\ \psi$ $= \psi\ (\sigma_2, +, s_7)\ \varphi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rwvu)$	$\varphi \mid_1 \psi$ [1, 23]
$\varphi\ (\sigma_6, +, s_7)\ \psi$ $= \psi\ (\sigma_3, +, s_7)\ \varphi$	$\exists vu\, (v \notin [\![\varphi]\!] \wedge u \notin [\![\psi]\!] \wedge Rvwu)$	$\varphi \mid_2 \psi$ [1, 23]

Figure 3: Some binary connectives of atomic logics of type $(1, 1, 1)$

3.3 Molecular Logics

Molecular logics are basically logics whose primitive connectives are compositions of atomic connectives.

Definition 15 (Molecular skeleton and connective)**.** The class $\mathbb{C}^*$ of *molecular skeletons* is the smallest set such that:

- $\mathbb{P} \cup \mathbb{B} \subseteq \mathbb{C}^*$ and $\mathbb{C}^*$ contains for each $k \in \mathbb{N}^*$ a symbol id_k of *type signature* (k, k) and *arity* 1;
- for all $\star \in \mathbb{C}$ of type signature $(k, k_1, \ldots, k_n)$ and all $c_1, \ldots, c_n \in \mathbb{C}^*$ of type signatures $(k_1, k_1^1, \ldots, k_{a_1}^1), \ldots, (k_n, k_1^n, \ldots, k_{a_n}^n)$ respectively, the connective $\star(c_1, \ldots, c_n)$ belongs to $\mathbb{C}^*$, its *type signature* is $(k, k_1^1, \ldots, k_{a_1}^1, \ldots, k_1^n, \ldots, k_{a_n}^n)$ and its *arity* is $a_1 + \ldots + a_n$.

We define the *quantification signature* $Æ(c)$ of $c = \star(c_1, \ldots, c_n)$ by $Æ(c) \triangleq Æ(\star)$.

If $c \in \mathbb{C}^*$, we define its *decomposition tree* as follows. If $c = \star \in \mathbb{C}$ is of arity $n > 0$, then its decomposition tree T_c is the tree of root $\star$ with n children–leaves labeled by id. If $c = \star(c_1, \ldots, c_n) \in \mathbb{C}^*$ then its decomposition tree T_c is a tree labeled with atomic connectives defined inductively as follows: the root of T_c is c and it is labeled with $\star$ and one sets edges between that root and the roots $c_1, \ldots, c_n$ of the decomposition trees $T_{c_1}, \ldots, T_{c_n}$ respectively.

A *molecular connective* is an object to which is associated a molecular skeleton. Its arity, quantification signature and decomposition tree are the same as its skeleton.

The set of *atomic connectives associated to a set* C *of molecular connectives* is the set of labels different from id of the decomposition trees of the molecular connectives of C. ⊣

According to that definition, if c_i is a propositional letter of type k_i, the type signature of $\star(c_1, \ldots, c_n)$ is $(k, k_1^1, \ldots, k_{a_1}^1, \ldots, k_1^{i-1}, \ldots, k_{a_{i-1}}^{i-1}, k_1^{i+1}, \ldots, k_{a_{i+1}}^{i+1}, \ldots, k_1^n, \ldots, k_{a_n}^n)$ and its arity is $a_1 + \ldots + a_{i-1} + a_{i+1} + \ldots + a_n$.

Obviously, atomic connectives are (specific) molecular connectives. Note that the same label (atomic connective) may appear several times in a decomposition tree. Note also that the vertices of a decomposition tree are molecular connectives. One needs to introduce the connective id to deal with molecular connectives whose skeletons are for example of the form $\star(p, id_1)$, where $p \in \mathbb{P}$.

Example 7 (Modal intuitionistic logic)**.** Let us consider the connectives defined by the following first–order formulas:

$$\begin{aligned} c(x) &\triangleq \forall y\,(\mathsf{R}xy \to \forall z\,(\mathsf{R}_\Diamond yz \to \mathsf{P}(z))) \\ c'(x) &\triangleq \forall y\,(\mathsf{R}xy \to \exists z\,(\mathsf{R}_\Diamond yz \wedge \mathsf{P}(z))) \\ \star_1(x) &\triangleq \forall y\,(\mathsf{R}xy \to \mathsf{P}(y)) \\ \star_2(x) &\triangleq \forall z\,(\mathsf{R}_\Diamond yz \to \mathsf{P}(z)) \\ \star_3(x) &\triangleq \exists z\,(\mathsf{R}_\Diamond yz \wedge \mathsf{P}(z)) \end{aligned}$$

Then, $\star_1, \star_2, \star_3$ are atomic connectives and the connectives associated to c, c' are molecular connectives. Indeed, c is the composition of $\star_1$ and $\star_2$, $c = \star_1(\star_2)$, and c' is the composition of $\star_1$ and $\star_3$, $c' = \star_1(\star_3)$. Equivalently, c and c' will have the same semantics as $c = \star_1(\star_2(id_1))$ and $c' = \star_1(\star_3(id_1))$. The connective associated to c corresponds to the connective $\Box$ of modal intuitionistic logic and the connective associated to c' corresponds to the connective $\Diamond$ of modal intuitionistic logic [33] defined in Section 2.2.3. ⊣

Example 8 (Temporal logic)**.** Let us consider the connectives defined by the following first–order formulas:

$$\star_1(x) \triangleq \exists y z z' \left(\mathsf{P}y \wedge \mathsf{Q}zz' \wedge \mathsf{R}yzz'x\right)$$
$$\star_1'(x) \triangleq \exists y z z' \left(\mathsf{P}y \wedge \mathsf{Q}zz' \wedge \mathsf{R}xzz'y\right)$$
$$\star_2(x, x') \triangleq \forall y \left(\mathsf{P}y \vee \neg \mathsf{S}yxx'\right)$$

$\star_1$, $\star_1'$ and $\star_2$ are atomic connectives (this is independent from the definitions of R and S). The connectives of skeletons $c = \star_1(id_1, \star_2)$ and $c' = \star_1'(id_1, \star_2)$ are molecular connectives. We will see in Example 15 that they correspond to the molecular connectives of temporal logic if we choose the class of C–models appropriately. ⊣

Definition 16 (Molecular language)**.** Let C be a set of molecular connectives. The *(typed) molecular language* $\mathcal{L}_{\mathsf{C}}$ associated to C is the smallest set that contains the propositional letters and that is closed under the molecular connectives while respecting the type constraints. That is,

- the propositional letters of C belong to $\mathcal{L}_{\mathsf{C}}$;
- for all $\star \in \mathsf{C}$ of type signature $(k, k_1, \ldots, k_n)$ and for all $\varphi_1, \ldots, \varphi_n \in \mathcal{L}_{\mathsf{C}}$ of types $k_1, \ldots, k_n$ respectively, we have that $\star(\varphi_1, \ldots, \varphi_n) \in \mathcal{L}_{\mathsf{C}}$ and $\star(\varphi_1, \ldots, \varphi_n)$ is of *type* k.

The *Boolean molecular language* $\mathcal{L}_{\mathsf{C}}^{\mathbb{B}}$ is the smallest set that contains the propositional letters and that is closed under the molecular connectives of C as well as the Boolean connectives $\mathbb{B}$:

- for all $\varphi, \psi \in \mathcal{L}_{\mathsf{C}}^{\mathbb{B}}$ of type k, we have that $(\varphi \wedge_k \psi), (\varphi \vee_k \psi) \in \mathcal{L}_{\mathsf{C}}^{\mathbb{B}}$.

We say that C *is complete for conjunction and disjunction* when its associated set of atomic connectives is complete for conjunction and disjunction.

Elements of $\mathcal{L}_{\mathsf{C}}$ are called *molecular formulas* and are denoted $\varphi, \psi, \alpha, \ldots$ The *type of a formula* $\varphi \in \mathcal{L}_{\mathsf{C}}$ is denoted $k(\varphi)$. We use the same abbreviations as for the atomic language. ⊣

Definition 17 (Molecular logic)**.** If C is a set of molecular connectives, then a *C–model M* is a C'–model M where C' is the set of atomic connectives associated to C. We also define $\overline{w}(M, \mathsf{C}) \triangleq \overline{w}(M, \mathsf{C}')$. The truth conditions for molecular connectives are defined naturally from the truth conditions of atomic connectives, inductively as follows: for all $k \in \mathbb{N}^*$,

$$\begin{aligned} [\![\star(c_1, \ldots, c_n)\left(\varphi_1^1, \ldots, \varphi_1^{k_1}, \ldots, \varphi_n^1, \ldots, \varphi_n^{k_n}\right)]\!]^M &\triangleq f_\star\left([\![c_1(\varphi_1^1, \ldots, \varphi_1^{k_1})]\!]^M, \ldots, [\![c_n(\varphi_n^1, \ldots, \varphi_n^{k_n}]\!]^M\right) \\ [\![id_k(\varphi)]\!]^M &\triangleq [\![\varphi]\!]^M \end{aligned}$$

If $\mathcal{E}_{\mathsf{C}}$ is a class of pointed C–models, the triple $(\mathcal{L}_{\mathsf{C}}, \mathcal{E}_{\mathsf{C}}, \Vdash)$ is a logic called the *molecular logic associated to $\mathcal{E}_{\mathsf{C}}$ and C.* ⊣

Any atomic logic is obvisouly a molecular logic.

3.4 Boolean Negation

Note that atomic logics do not include Boolean negation as a primitive connective. It turns out that Boolean negation can be defined systematically for each atomic connective by applying a transformation on it. The Boolean negation of a formula then boils down to taking the Boolean negation of the outermost connective of the formula. This transformation is defined as follows.

Definition 18 (Boolean negation)**.** Let $\star$ be a n–ary connective of skeleton$(\sigma, \pm, Æ, \overline{k}, \pm_1, \dots, \pm_n)$. The *Boolean negation of* $\star$ is the connective $-\star$ of skeleton $(\sigma, -\pm, -Æ, \overline{k}, -\pm_1, \dots, -\pm_n)$ where $-Æ \triangleq \exists$ if $Æ = \forall$ and $-Æ \triangleq \forall$ otherwise, which is associated in any C–model to the same relation as $\star$. If $\varphi = \star(\varphi_1, \dots, \varphi_n)$ is a atomic formula, the *Boolean negation of* φ is the formula $-\varphi \triangleq - \star (\varphi_1, \dots, \varphi_n)$. ⊣

Proposition 1. *Let C be a set of atomic connectives such that $-\star \in C$ for all $\star \in C$. Let $\varphi \in \mathcal{L}_C$ and M be a C–model. Then, for all $\overline{w} \in \overline{w}(M, C)$, $\overline{w} \in [\![-\varphi]\!]^M$ iff $\overline{w} \notin [\![\varphi]\!]^M$.*

4 Relative Expressivity of Protologics, Molecular and First–order Logics

In this section, we are going to investigate the relative expressivity of atomic, abstract and first–order logics: atomic logics versus FOL in Section 4.1 and protologics versus atomic logics in Section 4.2.

4.1 Atomic Logics versus FOL

4.1.1 Pure Predicate Logics in Atomic Logics

Definition 19 (Predicate atomic connectives)**.** The set of *predicate atomic connectives* is $\mathsf{C}^{\mathcal{P}} \triangleq \mathbb{B} \cup \{\bot\} \cup \left\{\mathsf{R}_{f_l^k} \mid \mathsf{R} \in \mathcal{P} \text{ of arity } k \text{ and } f_l^k : [\![1;k]\!] \to [\![1;l]\!] \text{ is surjective with } l \leq k\right\} \cup \{[\sigma_k] \mid k \in \mathbb{N}^*, \sigma_k \in \mathfrak{S}_k\} \cup \{\|_{k_1,k_2} \mid k_1, k_2 \in \mathbb{N}^*\} \cup \{\supset_k, \forall_k \mid k \in \mathbb{N}^*\} \cup \{\forall_0\}$ where

- $\bot$ has skeleton $(Id, -, \forall, 1)$;
- $\mathsf{R}_{f_l^k}$ has skeleton $(Id, +, \forall, l)$ for all $k, l \in \mathbb{N}^*$ (such that $l \leq k$);
- $[\sigma_k]$ has skeleton $(Id, -, \forall, (k,k), +)$, for all $k \in \mathbb{N}^*$;
- $\|_{k_1,k_2}$ has skeleton $(Id, -, \forall, (k_1 + k_2, k_1, k_2), (+,+))$;
- $\supset_k$ has skeleton $(Id, -, \forall, (k,k,k), (-,+))$;
- $\forall_k$ has skeleton $((2,3,1), -, \forall, (k, 1, k+1), (+,+))$, for all $k \in \mathbb{N}^*$;
- $\forall_0$ has skeleton $(1, -, \forall, (1,1), +)$.

We also use the following abbreviations: $\bot_1 \triangleq \bot$, and for all $k > 1$, $\bot_k \triangleq \|_{k-1,1} (\bot_{k-1}, \bot_1)$. ⊣

Note that all predicate atomic connectives have the quantification signature $\forall$ and that all tonicity signatures are positive +, except for the $\supset$ operator which contains a negative tonicity −. This exception is crucial. It somehow encodes the whole Boolean negation. Existential quantification signatures may then indirectly reappear in the formula through a combination of this tonicity − and a connective of universal quantification signature $\forall$. Moreover, we could have replaced our connective $\|_{k_1,k_2}$ with a connective $\&_{k_1,k_2}$ whose skeleton would be $(1, +, \exists, (k_1 + k_2, k_1, k_2), (+,+))$ and with the same associated relation, and likewise for other connectives. Doing so, we would obtain the same results.

Definition 20 (Predicate atomic logic)**.** A *predicate $\mathsf{C}^{\mathcal{P}}$–model* is a $\mathsf{C}^{\mathcal{P}}$–model $M = (W, \mathcal{R})$ such that:

- for all $l \in \mathbb{N}^*$, the connectives $\mathsf{R}_{f_l^k}$ are associated to l–ary relations $R_{f_l^k}$ over W;

- the connective $\perp$ is associated to the 1–ary relation $R_\perp \triangleq W$;
- for all $k \in \mathbb{N}^*$ and all $\sigma_k \in \mathfrak{S}_k$, the connectives $[\sigma_k]$ are associated to the $2k$–ary relation R^{σ_k} such that $R^{\sigma_k} w_1 \dots w_k w'_1 \dots w'_k$ iff for all $i \in [\![1;k]\!]$, $w'_i = w_{\sigma_k(i)}$;
- for all $k_1, k_2 \in \mathbb{N}^*$, the connectives $\Vert_{k_1,k_2}$ are associated to the $2(k_1 + k_2)$–ary relation $R_{k_1,k_2} \subseteq W^{2(k_1+k_2)}$ such that for all $\overline{w}_1 \in W^{k_1}$, all $\overline{w}_2 \in W^{k_2}$ and all $\overline{w}_3 \in W^{k_1+k_2}$, we have that $R_{k_1,k_2}\overline{w}_1\overline{w}_2\overline{w}_3$ iff $\overline{w}_3 = (\overline{w}_1, \overline{w}_2)$;
- for all $k \in \mathbb{N}^*$, the connectives $\supset_k$ are associated to the $3k$–ary relation $R_k \subseteq W^{3k}$ such that for all $\overline{w}_1 \in W^k$, all $\overline{w}_2 \in W^k$ and all $\overline{w}_3 \in W^k$, we have that $R_k\overline{w}_1\overline{w}_2\overline{w}_3$ iff $\overline{w}_1 = \overline{w}_2 = \overline{w}_3$;
- for all $k \in \mathbb{N}^*$, the connectives $\forall_k$ are associated to the $2(k+1)$–ary relation $R_{k,1}$ as defined for $\Vert_{k_1,k_2}$;
- $\forall_0$ is associated to the 2–ary relation $R_{\forall_0} \triangleq W^2$.

The class of all pointed predicate $\mathsf{C}^{\mathcal{P}}$–models is denoted $\mathcal{M}_{\mathsf{C}^{\mathcal{P}}}$. The satisfaction relation $\Vdash \subseteq \mathcal{M}_{\mathsf{C}^{\mathcal{P}}} \times \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}$ is then defined following Definition 14. If $\mathcal{E}_{\mathsf{C}^{\mathcal{P}}}$ is a specific class of pointed $\mathsf{C}^{\mathcal{P}}$–model, the triple $\left(\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}, \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \Vdash\right)$ is called the *predicate atomic logic associated to* $\mathcal{E}_{\mathsf{C}^{\mathcal{P}}}$.

We also define $\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k) \triangleq \left\{\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P}}} \mid k(\varphi) = k\right\}$ and for all $(M, \overline{w}), (N, \overline{v}) \in \mathcal{M}_{\mathsf{C}^{\mathcal{P}}}$, we write $(M, \overline{w}) \equiv_k (N, \overline{v})$ when for all $\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k)$ it holds that $(M, \overline{w}) \Vdash \varphi$ iff $(N, \overline{v}) \Vdash \varphi$. $\dashv$

Definition 21 (Translation from FOL to predicate atomic logics)**.**
Syntax. For all $k \in \mathbb{N}$ and all $\overline{x} = (x_1, \dots, x_k) \in \mathcal{V}^k$, we define the mappings $T_{\overline{x}} : \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}, k) \to \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k)$ and $T_\varnothing : \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\varnothing) \to \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(1)$ inductively on the formula $\varphi(x_1, \dots, x_k) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$ (with or without free variables) as follows:

- if φ is $\perp t$ then we define $T_{\overline{x}}(\perp t) \triangleq \perp$ (t can only be a single variable or constant since we do not have functions in the language);
- if φ is $\mathsf{R}y_1 \dots y_l$ and $\{y_1, \dots, y_l\} = \{x_{i_1}, \dots, x_{i_k}\}$ is of cardinality $k \leq l$ (some variables can be the same) with $x_{i_1}, \dots, x_{i_k}$ all distinct and in the same order of appearance as $y_1, \dots, y_l$, then there is a unique surjective function $f^k_l : [\![1;l]\!] \to [\![1;k]\!]$ such that $(y_1, \dots, y_l) = (x_{i_{f^k_l(1)}}, \dots, x_{i_{f^k_l(l)}})$. Then, we define the permutation σ_k on $[\![1;k]\!]$ by for all $j \in [\![1;k]\!]$, $i_{\sigma_k(j)} \triangleq j$. Finally, we define
  $T_{\overline{x}}(\mathsf{R}y_1 \dots y_l) \triangleq [\sigma_k]\mathsf{R}_{f^k_l}$;
- if $\varphi = \varphi_1 \vee \varphi_2$ where φ_1 and φ_2 are both sentences, then we define
  $T_\varnothing(\varphi) \triangleq (T_\varnothing(\varphi_1) \vee_1 T_\varnothing(\varphi_2))$
  if $\varphi = \varphi_1 \vee \varphi_2(x)$ where φ_1 is a sentence, then we define
  $T_{(x)}(\varphi) \triangleq \left(T_\varnothing(\varphi_1) \vee_1 T_{(x)}(\varphi_2(x))\right)$
  if $\varphi = \varphi_1(x) \vee \varphi_2$ where φ_2 is a sentence, then we define
  $T_{(x)}(\varphi) \triangleq \left(T_{(x)}(\varphi_1(x)) \vee_1 T_\varnothing(\varphi_2)\right)$
  if $\varphi = \varphi_1 \vee \varphi_2(x_1, \dots, x_k)$ where φ_1 is a sentence and $k > 1$, then we define
  $T_{\overline{x}}(\varphi) \triangleq \left(\Vert_{1,k-1}\left(T_\varnothing(\varphi_1), \perp_{k-1}\right) \vee_k T_{(x_1,\dots,x_k)}(\varphi_2(x_1, \dots, x_k))\right)$
  if $\varphi = \varphi_1(x_1, \dots, x_k) \vee \varphi_2$ where φ_2 is a sentence and $k > 1$, then we define
  $T_{\overline{x}}(\varphi) \triangleq \left(T_{(x_1,\dots,x_k)}(\varphi_1(x_1, \dots, x_k)) \vee_k \Vert_{k-1,1}\left(\perp_{k-1}, T_\varnothing(\varphi_2)\right)\right)$

if $\varphi = \varphi_1(x_{i_1}, \ldots, x_{i_{k_1}}) \vee \varphi_2(y_{j_1}, \ldots, y_{j_{k_2}})$ where $i_1 < \ldots < i_{k_1}$ and $j_1 < \ldots < j_{k_2}$ and $k_1, k_2 \leq k$, then we define

$T_{\overline{x}}(\varphi) \triangleq \|_{k_1,k_2} \left(\mathrm{EXP}_{\overline{x}}(\varphi_1(x_{i_1}, \ldots, x_{i_{k_1}})), \mathrm{EXP}_{\overline{x}}(\varphi_2(x_{j_1}, \ldots, x_{j_{k_2}}))\right)$ where

$$\mathrm{EXP}_{\overline{x}}(\varphi_1(x_{i_1}, \ldots, x_{i_{k_1}})) \triangleq \begin{cases} T_{(x_1,\ldots,x_k)}(\varphi_1(x_{i_1}, \ldots, x_{i_{k_1}})) & \text{if } k_1 = k \\ [\sigma_k] \|_{k_1,k-k_1} \left(T_{(x_{i_1},\ldots,x_{i_{k_1}})}(\varphi_1(x_{i_1}, \ldots, x_{i_{k_1}})), \bot_{k-k_1}\right) & \text{if } k_1 < k \end{cases}$$

where σ_k is a permutation such that for all $j \in [\![1; k_1]\!]$, $i_{\sigma_k(j)} = j$. The definition is similar for $\mathrm{EXP}_{\overline{x}}(\varphi_2(x_{j_1}, \ldots, x_{j_{k_2}}))$, one only needs to replace i by j and k_1 by k_2.

- if $\varphi = \varphi_1 \to \varphi_2$ where φ_1 and φ_2 are both sentences, then we define

  $T_\varnothing(\varphi) \triangleq \supset_1 (T_\varnothing(\varphi_1), T_\varnothing(\varphi_2))$

  if $\varphi = \varphi_1 \to \varphi_2(x)$ where φ_1 is a sentence, then we define

  $T_{(x)}(\varphi) \triangleq \supset_1 \left(T_\varnothing(\varphi_1), T_{(x)}(\varphi_2(x))\right)$

  if $\varphi = \varphi_1(x) \to \varphi_2$ where φ_2 is a sentence, then we define

  $T_{(x)}(\varphi) \triangleq \supset_1 \left(T_{(x)}(\varphi_1(x)), T_\varnothing(\varphi_2)\right)$

  if $\varphi = \varphi_1 \to \varphi_2(x_1, \ldots, x_k)$ where φ_1 is a sentence and $k > 1$, then we define

  $T_{\overline{x}}(\varphi) \triangleq \supset_k \left(\|_{1,k-1} (T_\varnothing(\varphi_1), \bot_{k-1}), T_{(x_1,\ldots,x_k)}(\varphi_2(x_1, \ldots, x_k))\right)$

  if $\varphi = \varphi_1(x_1, \ldots, x_k) \to \varphi_2$ where φ_2 is a sentence and $k > 1$, then we define

  $T_{\overline{x}}(\varphi) \triangleq \supset_k \left(T_{(x_1,\ldots,x_k)}(\varphi_1(x_1, \ldots, x_k)), \|_{k-1,1} (\bot_{k-1}, T_\varnothing(\varphi_2))\right)$;

  if $\varphi = \varphi_1(x_{i_1}, \ldots, x_{i_{k_1}}) \to \varphi_2(y_{j_1}, \ldots, y_{j_{k_2}})$ where $i_1 < \ldots < i_{k_1}$ and $j_1 < \ldots < j_{k_2}$ and $k_1, k_2 \leq k$, then we define

  $T_{\overline{x}}(\varphi) \triangleq \supset_k \left(\mathrm{EXP}_{\overline{x}}(\varphi_1(x_{i_1}, \ldots, x_{i_{k_1}})), \mathrm{EXP}_{\overline{x}}(\varphi_2(x_{j_1}, \ldots, x_{j_{k_2}}))\right)$

  where $\mathrm{EXP}_{\overline{x}}(\varphi_1(x_{i_1}, \ldots, x_{i_{k_1}}))$ and $\mathrm{EXP}_{\overline{x}}(\varphi_2(x_{j_1}, \ldots, x_{j_{k_2}}))$ are defined as above.

- if $\varphi = \forall x \psi(x)$ where φ is a sentence, then we define

  $T_\varnothing(\varphi) \triangleq \forall_0 T_{(x)}(\psi(x))$;

  if $\varphi = \forall x \psi(x_1, \ldots, x_k, x)$ with $k \geq 1$, then we define

  $T_{\overline{x}}(\varphi) \triangleq \forall_k(\bot, T_{(x_1,\ldots,x_k,x)}(\psi(x_1, \ldots, x_k, x)))$.

Semantics. Let $M = (W, \{R_1, \ldots, R_n, \ldots, c_1, \ldots, c_n, \ldots\})$ be a structure without functions. We define the $\mathsf{C}^{\mathcal{P}}$–model $T(M) = (W, \mathcal{R})$ as follows:

- the k–ary relation $R_{f_l^k}$ is defined from the l–ary relation R of M associated to each $\mathsf{R} \in \mathcal{P}$ as follows: for all $w_1, \ldots, w_k \in W$, we have that $R_{f_l^k} w_1 \ldots w_k$ iff $R w_{f_l^k(1)} \ldots w_{f_l^k(l)}$;
- the other relations of $\mathcal{R}$ are defined like in Definition 20.

If $\overline{x} = (x_1, \ldots, x_k)$ is a tuple of k variables then we define $T_{\overline{x}}(M, s) \triangleq (T(M), (s(x_1), \ldots, s(x_k)))$ and $T_\varnothing(M, s) \triangleq (T(M), (s(x)))$ for an arbitrary $x \in \mathcal{V}$. $\dashv$

In the syntactic part, the definitions may leave some freedom concerning the exact determination of the permutations σ_k for the disjunction and implication cases. This is not problematic and does not impact the results. Permutations are introduced so that the same variables which appear in different places in a formula be evaluated at the same points in the domain of the structure. Moreover, our definitions are set in such a way that for any sentence φ, we will have that $T_\varnothing(\varphi)$ will always be a formula of type 1.

Example 9. We provide two examples of translations.

- If P is a 8–ary predicate, then $T_{(x_1,x_2,x_3)}(\mathsf{P}x_3x_3x_1x_1x_2x_1x_2x_2) = [\sigma_3]\mathsf{P}_f$ where $f(1) = 1, f(2) = 1, f(3) = 2, f(4) = 2, f(5) = 3, f(6) = 2, f(7) = 3, f(8) = 3$ and $\sigma_3 = (2,3,1)$ because here $(x_{i_1}, x_{i_2}, x_{i_3}) = (x_3, x_1, x_2)$.
- $T_{(x,y)}(\forall z(\mathsf{P}zx \to Ryzx) \vee \forall z\mathsf{Q}zz)$
  $= \left(T_{(x,y)}(\forall z(\mathsf{P}zx \to Ryzx)) \vee_2 \|_{1,1}(\bot_1, T_\varnothing(\forall z\mathsf{Q}zz))\right)$
  $= \left(\forall_2\left(\bot, T_{(x,y,z)}(\mathsf{P}zx \to Ryzx)\right) \vee_2 \|_{1,1}\left(\bot_1, \forall_0 T_{(z)}(\mathsf{Q}zz)\right)\right)$
  $= \left(\forall_2\left(\bot, \supset_3\left([(1,3,2)]\,\|_{2,1}\left(T_{(x,z)}(\mathsf{P}zx), \bot_1\right), T_{(x,y,z)}(Ryzx)\right)\right) \vee_2\right.$
  $\left.\|_{1,1}\left(\bot_1, \forall_0 T_{(z)}(\mathsf{Q}zz)\right)\right)$
  $= \left(\forall_2(\bot, \supset_3([(1,3,2)]\,\|_{2,1}([(2,1)]\mathsf{P}, \bot_1), [(3,1,2)]R)) \vee_2 \|_{1,1}\left(\bot_1, \forall_0[Id]\mathsf{Q}_f\right)\right)$
  where $f(1) = 1$ and $f(2) = 1$. $\dashv$

Lemma 1. *Let $\varphi \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$, let $\overline{x}$ be the tuple of free variables of φ (possibly empty) and let (M,s) be a pointed structure. Then, we have that*

$$(M,s) \models \varphi \text{ iff } T_{\overline{x}}(M,s) \Vdash T_{\overline{x}}(\varphi).$$

Proof. By induction on φ. The difficulty lies in the way that variables should be handled. The same variable appearing at different places in a formula should be evaluated at the same element of the domain, which does not seem at first sight to be possible without assignments. In order to overcome this difficulty, we identify the same variables via f_l^k and change the predicate R into $\mathsf{R}_{f_l^k}$ accordingly. Alternatively, we could use the identity predicate to identify the same variables but we now show that we can in fact do without it. Those indices i, j that correspond to the same variable $x_i = x_j$ will be such that $f_l^k(i) = f_l^k(j)$. For example, the formula $\mathsf{R}xyx$ will be translated by $T_{(x,y)}$ into the formula $[Id]\mathsf{R}_{f_2^3}xy$ with $f_2^3(1) = 1$, $f_2^3(2) = 2$ and $f_2^3(3) = 1$. In doing so, we will not need to keep track of the names of variables, they will be encoded by the surjective function f_l^k. This said, note that for sentences φ, the evaluation point does not play any role in the interpretation of a formula of the form $T_\varnothing(\varphi)$ because it will begin either by the quantifier $\forall$ or it will simply be the constant $\bot t$. As for the induction steps, we first detail the connectives $\vee$ and $\forall$. For the connective $\vee$, we only consider the last most complex case, the previous ones are without particular difficulty. What we do is that we complete the two conjuncts φ_1 and φ_2 with the remaining variables (those that do not occur both in φ_1 and φ_2 but in only one of the conjuncts) thanks to constants $\bot_k$ using the $\|$ connective. Then, we arrange the order of variables so that the variables are affected to the right elements of the domain by means of the connective $[\sigma_k]$, with appropriate permutations σ_k, if such reordering is necessary. Indeed, if $k = k_1$ then no reordering is necessary since $(i_1, \ldots, i_{k_1}) = (1, \ldots, k)$. Finally, we can take the Boolean disjunction $\vee_k$ since the two disjuncts are now of the same type, k. As for the connective $\forall$, the treatment splits up into two cases depending on whether $\varphi = \forall x\psi$ is a sentence or a formula with free variable(s). We first deal with the case where φ contains free variables. The result stems from the definition of R_{k_1,k_2} and $\forall_k$. The quantification is captured by the abstract quantification inherent to the atomic connective $\forall_k$. The definition of $\forall_k$ ensures that this quantification bears on the right elements of the domain, by means of the permutation $(2,3,1)$. Second, we deal with the case where φ is a sentence. In that case, the quantification is dealt with by means of a universal modality $\forall_0$ since we do not have to account for free variables (that do not appear in φ). Finally, for the connective $\to$, the treatment is similar to $\vee$. We have to reorganize the order of the variables appearing in the antecedent and consequent of φ using $[\sigma_k]$ so that the variables are evaluated at the right elements. Then, we apply the $\supset_k$ connective to these transformed formulas. The relation associated to $\supset_k$, which is of the form $R_k\overline{w}\overline{w}\overline{w}$, combined with the truth condition of $\supset_k$, will be such that we will be able to capture the usual semantics of the material implication. $\square$

Definition 22 (Translation from predicate atomic logics to FOL)**.**
Syntax. For all $k \in \mathbb{N}^*$ and all tuples $\overline{x} = (x_1, \dots, x_k)$ of variables and constants, we define the mappings $ST_{\overline{x}} : \mathcal{L}^k_{\mathsf{C}^{\mathcal{P}}} \to \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$, where $\mathcal{L}^k_{\mathsf{C}^{\mathcal{P}}}$ is the set of formulas of $\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}$ of type k, inductively as follows:

$$\begin{array}{rcll}
ST_{\overline{x}}(\mathsf{R}_{f^k_l}) & \triangleq & \mathsf{R}x_{f^k_l(1)} \dots x_{f^k_l(l)} & \\
ST_{\overline{x}}(\bot) & \triangleq & \bot t & \\
ST_{\overline{x}}([\sigma_k]\varphi) & \triangleq & ST_{(x_{\sigma_k(1)}, \dots, x_{\sigma_k(k)})}(\varphi) & \\
ST_{\overline{x}}(\Vert_{k_1,k_2}(\varphi_1, \varphi_2)) & \triangleq & \left(ST_{(x_1, \dots, x_{k_1})}(\varphi_1) \vee ST_{(x_{k_1+1}, \dots, x_{k_1+k_2})}(\varphi_2)\right) & \text{if } k = k_1 + k_2 \\
ST_{\overline{x}}(\supset_k(\varphi_1, \varphi_2)) & \triangleq & (ST_{\overline{x}}(\varphi_1) \to ST_{\overline{x}}(\varphi_2)) & \\
ST_{\overline{x}}(\forall_k \varphi) & \triangleq & \forall x ST_{(\overline{x}, x)}(\varphi) &
\end{array}$$

where t is an arbitrary term of $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$.
Semantics. Let $(M, (w_1, \dots, w_k))$ be a pointed $\mathsf{C}^{\mathcal{P}}$–model of type k and let $\overline{x} = (x_1, \dots, x_k)$ be a tuple of free variables or constants of size k. The *(pointed) structure associated to* $(M, \overline{w})$, denoted $ST_{\overline{x}}(M, \overline{w}) \triangleq (ST(M), s^{\overline{w}}_{\overline{x}})$, is defined as follows. The assignment $s^{\overline{w}}_{\overline{x}}$ is such that for all $i \in \{1, \dots, k\}$, $s(x_i) = w_i$ and for all $x \in \mathcal{V} - \{x_1, \dots, x_k\}$, $s(x) = w_1$ and $ST(M)$ is the structure $ST(M) = (W, ST(\mathcal{R}))$ where $ST(\mathcal{R})$ is the set $\mathcal{R}$ to which we remove the relations of the form $R^{\sigma_k}, R_{k_1}, R_{k_1,k_2}$ and $R_{\forall_0}$ and replace the relations $R_{f^k_l}$ with the l–ary relations R associated to each $\mathsf{R} \in \mathcal{P}$, which are defined as follows: for all $w_1, \dots, w_l \in W$, we have that $\{w_1, \dots, w_l\} = \{v_1, \dots, v_k\}$ for some $k \leq l$ with $v_1, \dots, v_k$ all distinct and in the same order as $w_1, \dots, w_l$. There is a unique surjective function $f^k_l : [\![1;l]\!] \to [\![1;k]\!]$ such that $(w_1, \dots, w_l) = \left(v_{f^k_l(1)}, \dots, v_{f^k_l(l)}\right)$. Then, we set $Rw_1 \dots w_l$ iff $R_{f^k_l} v_1 \dots v_k$. $\dashv$

Lemma 2. *Let $(M, \overline{w})$ be a pointed predicate $\mathsf{C}^{\mathcal{P}}$–model, let $\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}$ of type k and let $\overline{x} \in \mathcal{V}^k$. Then,*

$$(M, \overline{w}) \Vdash \varphi \textit{ iff } ST_{\overline{x}}(M, \overline{w}) \Vdash ST_{\overline{x}}(\varphi).$$

Moreover, for all pointed structures (M, s) without functions and distinguished elements, we have that $ST_{\overline{x}}(T_{\overline{x}}(M, s)) \equiv_{\overline{x}} (M, s)$ and for all pointed $\mathsf{C}^{\mathcal{P}}$–model $(M, \overline{w})$, we have that $T_{\overline{x}}(ST_{\overline{x}}(M, \overline{w})) \equiv_k (M, \overline{w})$.

Proof sketch. By induction on φ. The only difficult case is for the propositional letters. It suffices to check the corresponding conditions in Definitions 21 and 22 to see that it works. $\square$

Theorem 1. *The class of pure predicate logics is as expressive as the class of predicate atomic logics.*

Proof. The class of pure predicate logics with free variables and constants is in fact as expressive as the class of pure predicate logics: every sentence φ is equivalent to the formula with free variable $\varphi \vee \bot x$. So, we are going to prove that every pure predicate logic with free variables and constants is as expressive as a predicate atomic logic, and vice versa.

If $\left(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}), \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is a pure predicate logic with free variables and constants then $\left(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}), \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}}\right)$ is as expressive as $\left(\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}, \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \Vdash\right)$ where $\mathcal{E}_{\mathsf{C}^{\mathcal{P}}} = \{T_{\overline{x}}(M, s) \mid (M, s) \in \mathcal{E}_{\mathsf{FOL}}$ and $\overline{x}$ ranges over all tuples of variables$\}$. We use the following partition for the pure predicate logic $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}} = \biguplus_{k \in \mathbb{N}^*} \biguplus_{\overline{x} \in \mathcal{V}^k} \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}, k)$ and the following partition for the predicate atomic logic $\mathcal{L}_{\mathsf{C}^{\mathcal{P}}} = \biguplus_{k\in\mathbb{N}^*} \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k)$ where we recall that $\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k) \triangleq \{\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P}}} \mid k(\varphi) = k\}$. Then, for all $k \in \mathbb{N}^*$ and all $\overline{x} \in \mathcal{V}^k$ we have that $(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}, k), \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}})$ is essentially as expressive as $(\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k), \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \Vdash)$; the translations are $T_{\overline{x}}, T_{\overline{x}}$ (for one direction) and $ST_{\overline{x}}, ST_{\overline{x}}$ (for the other direction). The result follows from Lemmas 1 and 2. Vice versa, to prove that $(\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}(k), \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \Vdash)$ is essentially as expressive as some $(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}, k), \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}})$, we pick an arbitrary tuple of k variables $\overline{x}$ and use the same translations.

For the other direction, if $(\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}, \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \Vdash)$ is a predicate atomic logic, then $(\mathcal{L}_{\mathsf{C}^{\mathcal{P}}}, \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \Vdash)$ is as expressive as $(\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}, \mathcal{E}_{\mathsf{FOL}}, \models_{\mathsf{FOL}})$ where this time $\mathcal{E}_{\mathsf{FOL}} = \{ST_{\overline{x}}(M, \overline{w}) \mid (M, \overline{w}) \in \mathcal{E}_{\mathsf{C}^{\mathcal{P}}}, \overline{x} \in \mathcal{V}^k \text{ and } k = |\overline{w}|\}$. We use the same partitions and the same translations. □

4.1.2 First–order Logics in Atomic Logics

We recall that $\mathcal{P}$ is a set of predicate symbols, $\mathcal{F}$ is a set of function symbols, $\mathcal{V}$ is a set of variables and $\mathcal{C}$ is a set of constants (see Section 2.1).

Definition 23 (First–order atomic connectives)**.** The set of *propositional letter atomic connectives* is $\mathsf{C}^{\mathcal{F}} \triangleq \{p_{\mathsf{R}} \mid \mathsf{R} \in \mathcal{P}\} \cup \{[h_1, \dots, h_k] \mid k \in \text{Arity}(\mathcal{P}, \mathcal{F}) \text{ and } h_1, \dots, h_k \in \mathcal{V} \cup \mathcal{C} \cup \mathcal{F}\}$ where:

- p_{R} has skeleton $(Id, +, \forall, k)$ with k the arity of $\mathsf{R} \in \mathcal{P}$;
- $[h_1, \dots, h_k]$ has skeleton $(Id, -, \forall, (n_1 + \ldots + n_k, k), +)$ in which n_i is the arity of the function h_i (it is 1 if $h_i \in \mathcal{V} \cup \mathcal{C}$).

The set of *first–order atomic connectives* $\mathsf{C}^{\mathcal{P},\mathcal{F}}$ is $\mathsf{C}^{\mathcal{F}}$ to which we add $\mathsf{C}^{\mathcal{P}}$ except that we remove the propositional letters $\left\{\mathsf{R}_{f^k_1}, \dots, \mathsf{R}_{f^k_l}, \dots, \mathsf{R}_{f^k_k} \mid \mathsf{R} \in \mathcal{P} \text{ of arity } k \text{ and } f^k_l : [\![1;k]\!] \to [\![1;l]\!] \text{ is surjective}\right\}$ of $\mathsf{C}^{\mathcal{P}}$ and replace them with the propositional letters $\{p_\chi \mid \chi \in \mathcal{L}_{\mathsf{C}^{\mathcal{F}}}\}$ of skeletons $(1, +, \forall, k)$, where k is the type of $\chi \in \mathcal{L}_{\mathsf{C}^{\mathcal{F}}}$. ⊣

Definition 24 (First–order atomic logic)**.** A *first–order $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–model* is a $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–model $M = (W, \mathcal{R})$ such that:

- the relations associated to the connectives of $\mathsf{C}^{\mathcal{P}}$ satisfy the conditions of Definition 20;
- for all $\mathsf{R} \in \mathcal{P}$ of arity n, the connectives p_{R} are associated to n–ary relations $R_{p_{\mathsf{R}}}$;
- Each connective $[h_1, \dots, h_k]$ is associated to a relation $R_{[h_1,\dots,h_k]}$ over W^l where $l = k + n_1 + \ldots + n_k$ with n_i the arity of the function h_i (it is 1 if $n_i \in \mathcal{V} \cup \mathcal{C}$);
- for all $\chi \in \mathcal{L}_{\mathsf{C}^{\mathcal{F}}}$, the connectives p_χ are associated to k–ary relations R_{p_χ}, where k is the type of χ.

The class of all pointed first–order $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–models is denoted $\mathcal{M}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$. We define the two–tiered language $\mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$ as follows:

$$\begin{aligned} \mathcal{L}_{\mathsf{C}^{\mathcal{F}}} : \ \chi \ &::= \ p_{\mathsf{R}} \ \mid \ [h_1, \dots, h_k]\chi \\ \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}} : \ \varphi \ &::= \ p_\chi \ \mid \ \star(\varphi, \dots, \varphi) \end{aligned}$$

where $\mathsf{R} \in \mathcal{P}$, $[h_1, \dots, h_k] \in \mathsf{C}^{\mathcal{F}}$, $\star \in \mathsf{C}^{\mathcal{P},\mathcal{F}}$ (different from propositional letters) and χ is of type k in $[h_1, \dots, h_k]\chi$. The satisfaction relation $\Vdash \subseteq \mathcal{M}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}} \times \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$ is then defined following Definition 14. If $\mathcal{E}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$ is a specific class of abstract $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–models, the triple $(\mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}, \mathcal{E}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}, \Vdash)$

is called the *first–order atomic logic associated to* $\mathcal{E}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$. The triple $\left(\mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}, \mathcal{M}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}, \Vdash\right)$ is called *first–order atomic logic*.

We also define $\mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}(k) \triangleq \left\{\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}} \mid k(\varphi) = k\right\}$ and for all $(M,\overline{w}), (N,\overline{v}) \in \mathcal{M}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$, we write $(M,\overline{w}) \equiv_{\mathcal{F},k} (N,\overline{v})$ when for all $\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}(k)$ it holds that $(M,\overline{w}) \Vdash \varphi$ iff $(N,\overline{v}) \Vdash \varphi$. ⊣

Definition 25 (Translation from FOL to first–order atomic logic)**.**
Syntax. We define the mapping $T^+ : \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}(\overline{x}) \to \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$ inductively on the formula $\varphi \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ as follows.

- if $\varphi = \mathsf{R}(t_1,\dots,t_k)$ then we define $T^+(\varphi) \triangleq T^+(t_1,\dots,t_k)p_{\mathsf{R}}$ where $T^+(t_1,\dots,t_k)$ is defined inductively as follows:
  - if $(t_1,\dots,t_k)$ is a tuple of only variables and constants then $T^+(t_1,\dots,t_k) \triangleq [t_1,\dots,t_k]$;
  - otherwise, if one of the terms t_i contains a function symbol f_i, then $T^+(t_1,\dots,t_k) \triangleq T^+\left(b(t_1),\dots,b(t_k)\right)[h(t_1),\dots,h(t_k)]$.

  Note that $T^+(\varphi)$ belongs to $\mathcal{L}_{\mathsf{C}^{\mathcal{F}}}$.
- if φ is of one of the other forms then the translation T^+ is the same as T.

Semantics. Let $M = \left(\left(W,\{R_1,\dots,R_n,\dots,f_1,\dots,f_n,\dots\}\right),s\right)$ be a pointed structure. We define the $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–model $T^+(M) \triangleq (W,\mathcal{R})$ as follows:

- the n–ary relation $R_{p_{\mathsf{R}}}$ is the n–ary relation R associated to $\mathsf{R} \in \mathcal{P}$;
- for all $h_1,\dots,h_k \in \mathcal{V} \cup \mathcal{C} \cup \mathcal{F}$, the connectives $[h_1,\dots,h_k] \in \mathsf{C}^{\mathcal{F}}$ are associated to the relation $R_{[h_1,\dots,h_k]} \triangleq \{(v_1,\dots,v_k,\overline{w}_1,\dots,\overline{w}_k) \mid \text{ for all } i \in [\![1;k]\!], \text{ if } h_i = f_i \text{ then } f_i(\overline{w}_i) = v_i, \text{ if } h_i \in \mathcal{V}\cup\mathcal{C} \text{ then } \overline{w}_i = v_i\}$;
- the other relations of $\mathcal{R}$ for the other connectives are defined like in Definition 24.

If $\overline{x} = (x_1,\dots,x_k) \in \mathcal{V}^k$ then we define $T^+_{\overline{x}}(M,s) \triangleq (T^+(M),(s(x_1),\dots,s(x_k)))$. ⊣

Lemma 3. *Let $k \in \mathbb{N}^*$, let $\varphi = \varphi(x_1,\dots,x_k) \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ and let (M,s) be a pointed structure. Then we have that*

$$(M,s) \models \varphi \text{ iff } T^+_{\overline{x}}(M,s) \Vdash T^+(\varphi).$$

Proof. The proof follows the same reasoning as the proof of Lemma 1. The only real new case is for $\varphi = \mathsf{R}(t_1,\dots,t_n)$. It follows from the truth condition for that formula: we have that $(M,s) \models \mathsf{R}(t_1,\dots,t_n)$ iff $R(w_1,\dots,w_n)$ and $w_1 = \overline{s}(t_1),\dots,w_n = \overline{s}(t_n)$. The condition $R(w_1,\dots,w_n)$ is captured by p_{R} and the condition $w_1 = \overline{s}(t_1),\dots,w_n = \overline{s}(t_n)$ is captured by the composition of appropriate connectives $[h_1,\dots,h_k]$. □

Definition 26 (Translation from first–order atomic logic to FOL)**.**
Syntax. For all $n \in \mathbb{N}^*$ and all tuples $\overline{t} = (\overline{t}_1,\dots,\overline{t}_k)$ of tuples of terms such that each tuple of terms $\overline{t}_i$ is of size n_i, we define the mappings $ST^+_{\overline{t}} : \mathcal{L}^k_{\mathsf{C}^{\mathcal{F}}} \to \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$, where $\mathcal{L}^k_{\mathsf{C}^{\mathcal{F}}}$ is the set of formulas of $\mathcal{L}_{\mathsf{C}^{\mathcal{F}}}$ of type k, inductively as follows. For all $[h_1,\dots,h_k]$ of type $(n_1+\dots+n_k,k)$, where $n_1,\dots,n_k$ are the arities of $h_1,\dots,h_k$, we define

$$\begin{aligned} ST^+_{\overline{t}}(p_{\mathsf{R}}) &\triangleq \mathsf{R}\overline{t} \\ ST^+_{\overline{t}}([h_1,\dots,h_k]\chi) &\triangleq ST^+_{(h_1(\overline{t_1}),\dots,h_k(\overline{t}_k))}(\chi). \end{aligned}$$

Then, we define the mapping $ST^+_{\overline{t}} : \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}} \to \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ inductively as follows.

$$ST^+_{\overline{t}}(p_\chi) \quad \triangleq \quad ST^+_{\overline{t}}(\chi).$$

where $\overline{t}$ is an appropriate tuple of tuples of terms. The clauses for the other connectives of $\mathsf{C}^{\mathcal{P},\mathcal{F}}$ are defined like in Definition 22, tuples of variables $\overline{x}$ only have to be replaced by tuples of terms $\overline{t}$.

Semantics. Let $(M, \overline{w})$ be a pointed $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–model of type k and let $\overline{x}$ be a tuple of free variables or constants of size k. The *pointed structure associated to* $(M, \overline{w})$, denoted $ST^+_{\overline{x}}(M, \overline{w}) \triangleq (ST^+(M), s^{\overline{w}}_{\overline{x}})$, is defined as follows. The assignment $s^{\overline{w}}_{\overline{x}}$ is defined like in Definition 22 and $ST^+(M)$ is M to which we remove the relations of the form $R^{\sigma_k}, R_{k_1}, R_{k_1,k_2}$ and $R_{\forall_0}$ and we replace relations of the form $R_{[h_1,\ldots,h_n]}$ with functions f defined as follows: $f(w_1, \ldots, w_k) = w$ iff $R_{[f]}ww_1, \ldots w_k$. ⊣

Lemma 4. *Let* $(M, \overline{w})$ *be a pointed first–order* $\mathsf{C}^{\mathcal{P},\mathcal{F}}$*–model, let* $\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P},\mathcal{F}}}$ *of type* k *and let* $\overline{x}$ *be a tuple of* k *variables. Then, we have that*

$$(M, \overline{w}) \Vdash \varphi \textit{ iff } ST^+_{\overline{x}}(M, \overline{w}) \Vdash ST^+_{\overline{x}}(\varphi)$$

Moreover, for all pointed structures (M, s)*, we have that* $ST^+_{\overline{x}}\left(T^+_{\overline{x}}(M, s)\right) \equiv_{\mathcal{F},\overline{x}} (M, s)$ *and for all pointed* $\mathsf{C}^{\mathcal{P},\mathcal{F}}$*–model* $(M, \overline{w})$ *of type* k*, we have that* $T^+_{\overline{x}}\left(ST^+_{\overline{x}}(M, \overline{w})\right) \equiv_{\mathcal{F},k} (M, \overline{w})$.

Proof sketch. By induction on φ. The tricky case is when φ is of the form $[h_1, \ldots, h_n]\,\chi$. This is where we need to index the translation by a tuple of terms. These terms keep track of the terms or subterms of the formula which is being translated. □

Theorem 2. *The class of first–order logics is as expressive as the class of first–order atomic logics.*

Proof. The class of first-order logics with free variables and constants is in fact as expressive as the class of first-order logics: every sentence φ is equivalent to the formula with free variable $\varphi \vee \bot x$. We have to prove that every first–order logic with free variables and constants is as expressive as a first–order atomic logic, and vice versa. It follows from Lemmas 3 and 4. The proof is the same as the proof of Theorem 1. □

4.2 Molecular Logics versus Protologics

In this section, $\mathcal{Q}$ and $\mathcal{P}$ are sets of predicates such that $\mathcal{Q} \subseteq \mathcal{P}$. We will also use the equality predicate = so as to avoid formulas of the form $\mathsf{Q}xyx$. That is, we want all variables in the scope of a predicate to be different. For example, $\mathsf{Q}_i xyx$ will be translated into the logically equivalent formula $\forall z(x = z \to \mathsf{Q}_i xyz)$. This is because we do not want to change the predicate symbols P into $\mathsf{P}_{f^k_l}$ like in Definition 20. Note that this preprocessing with the equality predicate could also be applied in the translation from first-order formulas to predicate atomic logics in order to avoid the complication with the introduction of $\mathsf{P}_{f^k_l}$ in Definition 20. These two approaches are in fact equivalent in that case.

Definition 27 (Translation from protologics to molecular logics)**.** Let C^a be a set of abstract connectives.
Syntax. We define the mapping t from the abstract connectives of C^a to molecular connectives as follows:

- $t(\mathsf{Q}) = \mathsf{Q}$ for all predicate $\mathsf{Q} \in \mathcal{Q}$;

- For abstract connectives of the form $(\chi(\overline{x}), (\mathsf{Q}_1, \dots, \mathsf{Q}_n))$, we proceed as follows. We first translate $\chi(\overline{x})$ into a logically equivalent formula of first–order logic such that each predicate $\mathsf{P}_i x_1 \dots x_k$ which occurs in χ has distinct variables $x_1, \dots, x_k$. This is possible using the equality predicate $=$. For example, $\mathsf{Q}_i xyx$ is translated into the logically equivalent formula $\forall z(x = z \to \mathsf{Q}_i xyz)$. We obtain a logically equivalent formula denoted $\chi_{=}(\overline{x})$ in which the equality predicate may occur. This first transformation is injective because the equality predicate was not present in the initial formulas.

  Then, we translate $\chi_{=}(\overline{x})$ into predicate atomic logic using the translation $T_{\overline{x}}$ of Definition 21 (the equality is simply viewed as a predicate $\mathsf{R}_{=}$). We obtain a formula $\chi' = T_{\overline{x}}(\chi_{=}(\overline{x}))$ of predicate atomic logic. That second transformation is also injective.

  That formula χ' is finally transformed into a molecular connective c. The skeleton decomposition tree of that molecular connective is the skeleton syntactic tree of χ' where the leafs labeled with the skeleton of a predicate Q_i are all replaced by the skeleton symbol id_{k_i} where k_i is the arity of each Q_i. The resulting molecular connective c of arity n and type signature $(k, k_1, \dots, k_n)$ is denoted $t(\chi(\overline{x}), (\mathsf{Q}_1, \dots, \mathsf{Q}_n))$.

The resulting set of molecular connectives is denoted $t(\mathsf{C}^a)$. Then, this translation t is extended to the whole language as follows: for all $\star \in \mathsf{C}^a$ and all $\varphi_1, \dots, \varphi_n \in \mathcal{L}_{\mathsf{C}^a}$ of appropriate types, $t(\star(\varphi_1, \dots, \varphi_n)) = t(\star)(t(\varphi_1), \dots, t(\varphi_n))$. By construction, this transformation t is injective.
Semantics. Let $(M, \overline{w})$ be a pointed C^a–model with $M = (W, \mathcal{R})$. The pointed $t(\mathsf{C}^a)$–model $t(M, \overline{w})$ is the $t(\mathsf{C}^a)$–model $((W, \mathcal{R}'), \overline{w})$ where $\mathcal{R}'$ is $\mathcal{R}$ together with the relations of Definition 20 (except those of the first item) as well as the equality predicate $R_{=} \triangleq \{(w, w) \mid w \in W\}$. ⊣

Lemma 5. *Let C^a be a set of abstract connectives, let $\varphi \in \mathcal{L}_{\mathsf{C}^a}$ and let $(M, \overline{w})$ be a pointed C^a–model where $\overline{w}$ is of size the type of φ. Then, we have that*

$$(M, \overline{w}) \Vdash \varphi \text{ iff } t(M, \overline{w}) \Vdash t(\varphi)$$

Proof. By induction on φ. The base case holds trivially. For the induction step, that is for abstract connectives of the form $(\chi(\overline{x}), (\mathsf{Q}_1, \dots, \mathsf{Q}_n))$, each transformation of the formula χ, into $\chi_{=}(\overline{x})$ and then χ', preserves its truth in any assignment $\overline{w}$ by Lemma 1. □

Definition 28 (Translation from molecular logics to protologics)**.** Let C be a set of molecular connectives.
Syntax. We define the mapping st from molecular connectives to abstract connectives inductively as follows:

- $st(p) \triangleq p$ for all propositional letters p of C.

- For all molecular connectives of the form $c = \star(c_1, \dots, c_m)$ we proceed as follows. First, we replace all symbols id_k appearing in c by fresh and distinct propositional letters $(p_1, \dots, p_n)$. This yields a formula $\varphi \in \mathcal{L}_{\mathsf{C}}$ of some type k. Then, we pick a tuple of free variables $\overline{x}$ of size k and we define the first–order formula $st_{\overline{x}}(\varphi)$ inductively as follows. If φ is a propositional letter p then $st_{\overline{x}}(p) \triangleq \mathsf{Q}\overline{x}$, where Q is a predicate symbol of $\mathcal{Q}$. If φ is of the form $\star(\varphi_1, \dots, \varphi_m)$ then

  - if $\star = (\sigma, \pm, \forall, \overline{k}, (\pm_1, \dots, \pm_m))$ then
    $st_{\overline{x}}(\star(\varphi_1, \dots, \varphi_m)) \triangleq \forall \overline{y}_1 \dots \overline{y}_m \left(\pm_1 st_{\overline{y}_1}(\varphi_1) \vee \dots \vee \pm_n st_{\overline{y}_m}(\varphi_m) \vee \pm \mathsf{R}^\sigma \overline{y}_1 \dots \overline{y}_m \overline{x}\right)$;
  - if $\star = (\sigma, \pm, \exists, \overline{k}, (\pm_1, \dots, \pm_m))$ then
    $st_{\overline{x}}(\star(\varphi_1, \dots, \varphi_m)) \triangleq \exists \overline{y}_1 \dots \overline{y}_m \left(\pm_1 st_{\overline{y}_1}(\varphi_1) \wedge \dots \wedge \pm_n st_{\overline{y}_n}(\varphi_m) \wedge \pm \mathsf{R}^\sigma \overline{y}_1 \dots \overline{y}_m \overline{x}\right)$;

where $\overline{y}_1, \ldots, \overline{y}_n$ are fresh tuples of free variables and R is a predicate symbol of $\mathcal{P} - \mathcal{Q}$. We recall that for all formulas ψ, $\pm_i \psi$ stands for ψ if $\pm_i = +$ and for $\neg\psi$ if $\pm_i = -$, and that $\mathsf{R}^{\sigma}\overline{y}_1 \ldots \overline{y}_m \overline{y}_{m+1} \triangleq \mathsf{R}\overline{y}_{\sigma^-(1)} \cdots \overline{y}_{\sigma^-(m)} \overline{y}_{\sigma^-(m+1)}$.

Finally, we define the abstract connective $st(c) \triangleq (st_{\overline{x}}(\varphi), (\mathsf{Q}_1, \ldots, \mathsf{Q}_n))$ where for all $i \in [\![1; n]\!]$, $\mathsf{Q}_i = st(p_i)$.

The resulting set of abstract connectives is denoted $st(\mathsf{C})$. Then, this translation st is extended to the whole language as follows: for all $\star \in \mathsf{C}$ and all $\varphi_1, \ldots, \varphi_n \in \mathcal{L}_{\mathsf{C}}$ of appropriate types, $st(\star(\varphi_1, \ldots, \varphi_n)) = st(\star)(st(\varphi_1), \ldots, st(\varphi_n))$. By construction, this transformation st is injective. $\dashv$

Lemma 6. *Let C be a set of molecular connectives, let $\varphi \in \mathcal{L}_{\mathsf{C}}$ and let $(M, \overline{w})$ be a pointed C–model where $\overline{w}$ is of size the type of φ. Then, we have that*

$$(M, \overline{w}) \Vdash \varphi \text{ iff } (M, \overline{w}) \Vdash st(\varphi)$$

Proof. By induction on φ. The base case holds trivially. The inductive case is just a reformulation in first–order logic of the truth conditions associated to each molecular connective. We do not need to introduce extra predicates like in the previous case, the ones needed are all already present in the initial C–model. Therefore we keep the same C–model. $\square$

Theorem 3. *The class of protologics is as expressive as the class of molecular logics.*

Proof. We have to prove that every protologic is as expressive as a molecular logic, and vice versa. If $(\mathcal{L}_{\mathsf{C}}, \mathcal{E}_{\mathsf{C}}, \Vdash)$ is a molecular logic, it is as expressive as the protologic $(st(\mathcal{L}_{\mathsf{C}}), \mathcal{E}_{\mathsf{C}}, \Vdash)$, where $st(\mathcal{L}_{\mathsf{C}}) \triangleq \{st(\varphi) \mid \varphi \in \mathcal{L}_{\mathsf{C}}\}$. In that case, the mappings for models are the identity mappings, the mapping from $\mathcal{L}_{\mathsf{C}}$ to $st(\mathcal{L}_{\mathsf{C}})$ is st and the mapping from $st(\mathcal{L}_{\mathsf{C}})$ to $\mathcal{L}_{\mathsf{C}}$ is the inverse mapping of st. It exists because st is injective. We then obtain the result thanks to Lemma 6.

Conversely, if $(\mathcal{L}_{\mathsf{C}^a}, \mathcal{E}_{\mathsf{C}^a}, \Vdash)$ is a protologic, it is as expressive as the molecular logic $(t(\mathcal{L}_{\mathsf{C}^a}), t(\mathcal{E}_{\mathsf{C}^a}), \Vdash)$, where $t(\mathcal{L}_{\mathsf{C}^a}) \triangleq \{t(\varphi) \mid \varphi \in \mathcal{L}_{\mathsf{C}^a}\}$ and $t(\mathcal{E}_{\mathsf{C}^a}) \triangleq \{t(M, \overline{w}) \mid (M, \overline{w}) \in \mathcal{E}_{\mathsf{C}^a}\}$. We define the mapping t^- from $t(\mathcal{E}_{\mathsf{C}^a})$ to $\mathcal{E}_{\mathsf{C}^a}$ by removing from any $t(\mathsf{C}^a)$–model all the relations which do not appear in the initial C^a–model before the translation t, we thus obtain the initial C^a–model. The mappings t and t^- for models are therefore inverse bijections of each other. The mappings for formulas are t and the inverse of t; this inverse also exists in that case because t is injective. We then obtain the result thanks to Lemma 5. $\square$

Example 10. We resume our example of modal intuitionistic logic. We consider the abstract connective $(\chi(x), (\mathsf{Q}))$ where $\chi(x) \triangleq \forall y(\mathsf{R}xy \to \forall z(\mathsf{R}_{\Diamond}yz \to \mathsf{Q}(z)))$. We show how this abstract connective is transformed into a molecular connective. Applying $T_{(x)}$, we obtain

$$\begin{aligned} T_{(x)}(\chi(x)) &= \forall_1(\bot_1, \supset_2 (T_2(\mathsf{R}xy), \|_{1,1} (\bot_1, T_1(\forall z(\mathsf{R}_{\Diamond}yz \to \mathsf{Q}(z)))))) \\ &= \forall_1(\bot_1, \supset_2 (\mathsf{R}, \|_{1,1} (\bot_1, \forall_1(\bot_1, T_1(\mathsf{R}_{\Diamond}yz \to \mathsf{Q}(z)))))) \\ &= \forall_1(\bot_1, \supset_2 (\mathsf{R}, \|_{1,1} (\bot_1, \forall_1(\bot_1, \supset_2 (\mathsf{R}_{\Diamond}, \|_{1,1} (\bot_1, \mathsf{Q})))))) \end{aligned}$$

Then we replace Q by id_1 and we obtain the following molecular connective:

$$\forall_1(\bot_1, \supset_2 (\mathsf{R}, \|_{1,1} (\bot_1, \forall_1(\bot_1, \supset_2 (\mathsf{R}_{\Diamond}, \|_{1,1} (\bot_1, id_1))))))$$

It turns out that it is quite different from the molecular connective that was introduced in Example 8. Yet, they both have the same effect, the former on C–models and the latter on $\mathsf{C}^{\mathcal{P}}$–models when we consider only the part of the information present in the initial C–model. $\dashv$

So, it is possible that a protologic corresponds naturally to a molecular logic but that its translation into a molecular logic does not yield the expected natural outcome. However, the two logics, the one expected and the one obtained thanks to our translation, will be equi–expressive. The pieces of information which are added in $\mathsf{C}^{\mathcal{P}}$–models (corresponding to the connectives $\|, \supset, \forall_k, \sigma_k$) actually do not increase the expressive power of the logic because the object language cannot refer to their associated relations. Yet, they allow us to reformulate it systematically under the form of an atomic or molecular logic.

5 Automatic Bisimulations for Atomic and Molecular Logics

In this section, we are going to see that notions of bisimulations can be automatically defined for atomic logics on the basis of the definition of the truth conditions of their connectives, not only for plain atomic logics but also for molecular logics. These notions are such that they preserve the truth of the formulas of the atomic logic considered between models. We will illustrate these results on modal logic, the Lambek calculus, intuitionistic logic, modal intuitionistic logic and temporal logic of Section 2.2. The bisimulation notions that we will find out by applying our generic definitions on these atomic logics will correspond to the bisimulation notions introduced in the literature for these logics.

5.1 Atomic Logics

Definition 29 (C–bisimulation)**.** Let C be a set of atomic connectives, let $\star \in \mathsf{C}$ and let $M_1 = (W_1, \mathcal{R}_1)$ and $M_2 = (W_2, \mathcal{R}_2)$ be two C–models. A binary relation $Z \subseteq \bigcup_{k \in \mathbb{N}^*} (W_1^k \times W_2^k) \cup (W_2^k \times W_1^k)$ is a *C–bisimulation* between M_1 and M_2 when for all $\star \in \mathsf{C}$, if $\{M, M'\} = \{M_1, M_2\}$, then for all $\overline{w}_1, \ldots, \overline{w}_n, \overline{w'}_1, \ldots, \overline{w'}_n, \overline{w}, \overline{w'} \in \overline{w}(M, \mathsf{C}) \cup \overline{w}(M', \mathsf{C})$,

1. if $\star$ is a propositional letter p then, if $\overline{w} Z \overline{w'}$ and $\overline{w} \in [\![p]\!]$ then $\overline{w'} \in [\![p]\!]$;
2. if $\star$ has skeleton $(\sigma, \pm, \exists, \overline{k}, (\pm_1, \ldots, \pm_n))$ and we have $\overline{w} Z \overline{w'}$ and $R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w}$, then $\exists \overline{w'}_1, \ldots, \overline{w'}_n \left(\overline{w}_1 \bowtie \overline{w'}_1 \wedge \overline{w}_2 \bowtie \overline{w'}_2 \wedge \ldots \wedge \overline{w}_n \bowtie \overline{w'}_n \wedge R_\star'^{\pm\sigma} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'} \right)$;
3. if $\star$ has skeleton $(\sigma, \pm, \forall, \overline{k}, (\pm_1, \ldots, \pm_n))$ and we have $\overline{w} Z \overline{w'}$ and $-R_\star'^{\pm\sigma} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'}$, then $\exists \overline{w}_1, \ldots, \overline{w}_n \left(\overline{w}_1 \bowtie \overline{w'}_1 \wedge \overline{w}_2 \bowtie \overline{w'}_2 \wedge \ldots \wedge \overline{w}_n \bowtie \overline{w'}_n \wedge -R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w} \right)$;

where, for all $j \in [\![1; n]\!]$, we define $\overline{w}_j \bowtie \overline{w'}_j \triangleq \begin{cases} \overline{w}_j Z \overline{w'}_j & \text{if } \pm_j = + \\ \overline{w'}_j Z \overline{w}_j & \text{if } \pm_j = - \end{cases}$.

When such a C–bisimulation Z exists and $\overline{w} Z \overline{w'}$, we say that $(M, \overline{w})$ and $(M', \overline{w'})$ are *C–bisimilar* and we write it $(M, \overline{w}) \rightarrow_{\mathsf{C}} (M', \overline{w'})$. ⊣

Note that case 1. is a particular instance of cases 2. and 3. with $n = 0$.

Definition 30. Let C be a set of atomic connectives. Let $(M, \overline{w})$ and $(M', \overline{w'})$ be two pointed C–models. We write $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}} (M', \overline{w'})$ when for all $\varphi \in \mathcal{L}_{\mathsf{C}}$, $(M, \overline{w}) \Vdash \varphi$ implies $(M', \overline{w'}) \Vdash \varphi$. ⊣

Proposition 2. *Let C be a set of atomic connectives and let $M_1 = (W_1, \mathcal{R}_1)$ and $M_2 = (W_2, \mathcal{R}_2)$ be two C–models. Let Z be a C–bisimulation between M_1 and M_2. Then, if $\{M, M'\} = \{M_1, M_2\}$ then for all $\overline{w} \in \overline{w}(M, \mathsf{C})$, all $\overline{w'} \in \overline{w}(M', \mathsf{C})$, if $\overline{w} Z \overline{w'}$ then $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}} (M', \overline{w'})$.*

Proof. We prove it by induction on a formula φ. If $\varphi = p$ is a propositional letter then the result holds by definition of Z. Assume that $\varphi = \star(\varphi_1, \ldots, \varphi_n)$.

If $\star$ has skeleton $(\sigma, \pm, \exists, (k, k_1, \ldots, k_n), (\pm_1, \ldots, \pm_n))$, then assume that $(M, \overline{w}) \models \varphi$. Then, there are $\overline{w}_1, \ldots, \overline{w}_n$ such that $\overline{w}_1 \pitchfork_1 [\![\varphi_1]\!]$ and $\ldots$ and $\overline{w}_n \pitchfork_n [\![\varphi_n]\!]$ and $R_\star^{\pm\sigma}\overline{w}_1 \ldots \overline{w}_n\overline{w}$. Now, $\overline{w}Z\overline{w'}$, so by definition of Z, there are $\overline{w'}_1, \overline{w'}_2 \ldots, \overline{w'}_n$ such that $\overline{w}_1 \bowtie_1 \overline{w'}_1$ and $\ldots$ and $\overline{w}_n \bowtie_n \overline{w'}_n$ and $R_\star^{'\pm\sigma}\overline{w'}_1 \ldots \overline{w'}_n\overline{w'}$ where $\overline{w}_j \bowtie \overline{w'}_j \triangleq \begin{cases} \overline{w}_j Z \overline{w'}_j & \text{if } \pm_j = + \\ \overline{w'}_j Z \overline{w}_j & \text{if } \pm_j = - \end{cases}$. Now, by Induction Hypothesis, for all $i \in \{1, \ldots, n\}$, if $\pm_i = +$ then $\overline{w}_i \in [\![\varphi_i]\!]$ and $\overline{w}_i Z \overline{w'}_i$, therefore $\overline{w'}_i \in [\![\varphi_i]\!]$ and if $\pm_i = -$ then $\overline{w}_i \notin [\![\varphi_i]\!]$ and $\overline{w}_i Z \overline{w'}_i$, therefore $\overline{w'}_i \notin [\![\varphi_i]\!]$. So, in all cases, $\overline{w'}_i \pitchfork_i [\![\varphi_i]\!]$. Moreover, $R_\star^{'\pm\sigma}\overline{w'}_1 \ldots \overline{w'}_n\overline{w'}$. Thus, there are $\overline{w'}_1, \ldots, \overline{w'}_n$ such that $\overline{w'}_1 \pitchfork_1 [\![\varphi_1]\!]$ and $\ldots$ and $\overline{w'}_n \in [\![\varphi_n]\!]$ and $R_\star^{'\pm,\sigma}\overline{w'}_1 \ldots \overline{w'}_n\overline{w'}$. Hence, $(M', \overline{w'}) \Vdash \varphi$.

If $\star$ has skeleton $(\sigma, \pm, \forall, (k, k_1, \ldots, k_n), (\pm_1, \ldots, \pm_n))$, then assume by contraposition that it is not the case that $(M', \overline{w'}) \Vdash \varphi$. Then, there are $\overline{w'}_1, \ldots, \overline{w'}_n$ such that not $\overline{w'}_1 \pitchfork_1 [\![\varphi_1]\!]$ and $\ldots$ and not $\overline{w'}_n \pitchfork_n [\![\varphi_n]\!]$ and $-R_\star^{'\pm,\sigma}\overline{w'}_1 \ldots \overline{w'}_n\overline{w'}$. Now, $\overline{w}Z\overline{w'}$, so by definition of Z, there are $\overline{w}_1, \ldots, \overline{w}_n$ such that $\overline{w}_1 \bowtie_1 \overline{w'}_1$ and $\ldots$ and $\overline{w}_n \bowtie_n \overline{w'}_n$ and $-R_\star^{\pm\sigma}\overline{w}_1 \ldots \overline{w}_n\overline{w}$ where for all $i \in \{1, \ldots, n\}$, $\overline{w}_i \bowtie \overline{w'}_i \triangleq \begin{cases} \overline{w}_i Z \overline{w'}_i & \text{if } \pm_i = + \\ \overline{w'}_i Z \overline{w}_i & \text{if } \pm_i = - \end{cases}$. Let $i \in \{1, \ldots, n\}$. If $\pm_i = +$ then $\overline{w'}_i \notin [\![\varphi_i]\!]$ and $\overline{w}_i Z \overline{w'}_i$. Therefore, by Induction Hypothesis, $\overline{w}_i \notin [\![\varphi_i]\!]$. If $\pm_i = -$, then $\overline{w'}_i \in [\![\varphi_i]\!]$ and $\overline{w'}_i Z \overline{w}_i$. Therefore, by Induction Hypothesis, $\overline{w}_i \in [\![\varphi_i]\!]$. So, in all cases, not $\overline{w}_i \pitchfork_i [\![\varphi_i]\!]$. Thus, there are $\overline{w}_1, \ldots, \overline{w}_n$ such that not $\overline{w}_1 \pitchfork_1 [\![\varphi_1]\!]$ and $\ldots$ and not $\overline{w}_n \pitchfork_n [\![\varphi_n]\!]$ and $-R_\star^{\pm\sigma} w_1 \ldots w_n w$. That is, not $(M, \overline{w}) \Vdash \varphi$. So, we have proved that if not $(M', \overline{w'}) \Vdash \varphi$ and $\overline{w}Z\overline{w'}$ then not $(M, \overline{w}) \Vdash \varphi$. Hence we obtain the result by contraposition. □

Example 11 (Modal logic)**.** Let us consider the connectives of modal logic: $\mathsf{C} = \{p, \neg p, \wedge, \vee, \Diamond, \Box\}$ where p has skeleton $(Id, +, \exists, 1)$, $\neg p$ has skeleton $(Id, -, \forall, 1)$, $\Diamond$ has skeleton $(\tau_2, +, t_1)$ and $\Box$ has skeleton $(\tau_2, -, t_2)$. Let $M_1 = (W_1, \{R_1, P_1\})$ and $M_2 = (W_2, \{R_2, P_2\})$ be two Kripke models (they are also C-models). A binary relation Z between M_1 and M_2 is a C–bisimulation between M_1 and M_2 when for all $M, M' \in \{M_1, M_2\}$ with $M = (W, \{R, P\})$ and $M' = (W', \{R', P'\})$, all $w, v \in M$ and all $w', v' \in M'$,

- if wZw' and $w \in [\![p]\!]$ then $w' \in [\![p]\!]$ (condition for p);
- if wZw' and $w' \in [\![p]\!]$ then $w \in [\![p]\!]$ (condition for $\neg p$);
- if wZw' and Rwv then there is $v' \in W'$ such that vZv' and $R'w'v'$ (condition for $\Diamond = (\tau_2, +, t_1)$);
- if wZw' and $R'w'v'$ then there is $v \in W$ such that vZv' and Rwv (condition for $\Box = (\tau_2, -, t_2)$).

Note that every C–bisimulation can be canonically extended into a *symmetric* C–bisimulation: one sets $w'Zw$ when wZw' already holds. ⊣

Example 12 (Lambek calculus)**.** Let us consider the connectives of the Lambek calculus: $\mathsf{C} = \{p, \circ, \backslash, /\}$ where p has skeleton $(Id, +, \exists, 1)$, $\circ$ has skeleton $(\sigma_1, +, s_1)$, $\backslash$ has skeleton $(\sigma_5, -, s_3)$ and / has skeleton $(\sigma_3, -, s_2)$. Let $M_1 = (W_1, \{R_1, P_1\})$ and $M_2 = (W_2, \{R_2, P_2\})$ be two Lambek models (they are also C–models). A binary relation Z between M_1 and M_2 is a C–bisimulation between M_1 and M_2 when for all $M, M' \in \{M_1, M_2\}$ with $M = (W, \{R, P\})$ and $M' = (W', \{R', P'\})$, all $w, v, u \in M$ and all $w', v', u' \in M'$,

- if wZw' and $w \in [\![p]\!]$ then $w' \in [\![p]\!]$ (condition for p);

- if wZw' and $Rvuw$ then there are $v', u' \in W'$ such that vZv', uZu' and $Rv'u'w'$ (condition for $\circ$);
- if vZv' and $Rv'u'w'$ then there are $u, w \in W$ such that $u'Zu$, wZw' and $Rvuw$ (condition for $\backslash = (\sigma_5, -, s_3)$);
- if uZu' and $Rv'u'w'$ then there are $v, w \in W$ such that $v'Zv$, wZw' and $Rvuw$ (condition for $/ = (\sigma_3, -, s_2)$). ⊣

The following proposition shows that the notions of C–bisimulation for the Lambek calculus and directed bisimulation coincide (directed bisimulations are defined for example in [37, Definition 13.2]) and likewise for modal logic.

Proposition 3.

- *Let* $\mathsf{C} = \{p, \neg p, \wedge, \vee, \Diamond, \Box\}$ *be the connectives of Example 11 and let* M *and* M' *be two* C*–models. Then, a* C*–bisimulation between* M *and* M' *is a modal bisimulation between* M *and* M' *and vice versa.*
- *Let* $\mathsf{C} = \{p, \circ, \backslash, /\}$ *be the connectives of Example 12 and let* M *and* M' *be two* C*–models. Then, a* C*–bisimulation between* M *and* M' *is a directed bisimulation between* M *and* M' *and vice versa.*

Proof. It suffices to compare the conditions of Examples 11 and 12 with the definitions of modal bisimulation [12, Def. 2.16] and directed bisimulation. This said, we obtain a notion of bisimulation which is slightly more general than the usual definition of modal bisimulation. However, they are in fact equivalent because both definitions of bisimulation can be extended to obtain symmetric bisimulation relations. □

Example 13 (Intuitionistic logic)**.** Let us consider the connectives of intuitionistic logic: $\mathsf{C} = \{p, \bot, \top, \wedge, \vee, \Rightarrow\}$ where p has skeleton $\{(Id, +, \exists, 1)$, $\top$ has skeleton $(Id, +, \exists, 1)$, $\bot$ has skeleton $(Id, -, \forall, 1)$, $\wedge$ and $\vee$ are Boolean connectives and $\Rightarrow$ has skeleton $(\sigma_5, -, s_3)$ (here, $\top$ and $\bot$ are represented by specific propositional letters of respective signatures $(Id, +, \exists, 1)$ and $(Id, -, \forall, 1)$). Let $M_1 = (W_1, R_1, P)$ and $M_2 = (W_2, R_2, P)$ be two intuitionistic models. Following the results of [5, Section 8], we represent these intuitionistic models by the C–models $M_1^{\Rightarrow} = (W_1, R_{1,\Rightarrow}, P)$ and $M_2^{\Rightarrow} = (W_2, R_{2,\Rightarrow}, P)$ respectively such that for all $u_1, v_1, w_1 \in W_1$ and all $u_2, v_2, w_2 \in W_2$,

$$R_{1,\Rightarrow}u_1v_1w_1 \text{ iff } R_1u_1w_1 \text{ and } R_1v_1w_1 \tag{5}$$

$$R_{2,\Rightarrow}u_2v_2w_2 \text{ iff } R_2u_2w_2 \text{ and } R_2v_2w_2 \tag{6}$$

One can show [5] that for all $\varphi \in \mathcal{L}_\mathsf{C}$ and all $w_1 \in W_1$, $M_1, w_1 \models \varphi$ iff $M_1^{\Rightarrow}, w_1 \Vdash \varphi$ (and likewise for M_2 and $M_2^{\Rightarrow}$). Now, a binary relation Z between $M_1^{\Rightarrow}$ and $M_2^{\Rightarrow}$ is a C–bisimulation between $M_1^{\Rightarrow}$ and $M_2^{\Rightarrow}$ iff for all $M, M' \in \{M_1^{\Rightarrow}, M_2^{\Rightarrow}\}$, all $w, w', v', u' \in \overline{w}(M, \mathsf{C}) \cup \overline{w}(M', \mathsf{C})$ and all $p \in \mathbb{P}$,

- if wZw' and $w \in [\![p]\!]$ then $w' \in [\![p]\!]$ (condition for p);
- if vZv' and $R'_{\Rightarrow}v'u'w'$ then there are $u, w \in W$ such that $u'Zu$, wZw' and $R_{\Rightarrow}vuw$ $(*)$ (condition for $\Rightarrow$);
- conditions for $\top$ and $\bot$ trivially hold because of their semantics.

Using Expressions (5) and (6), one can easily show that condition $(*)$ is equivalent to the following condition:

- if vZv' and $R'v'w'$ and $R'u'w'$ then there are $u, w \in W$ such that $u'Zu$, wZw' and Rvw and Ruw $(**)$.

We will show in Section 9 that condition $(**)$ is equivalent on ω–saturated models to Olkhovikov's condition "step" of [33, Definition 1] of his "basic asimulation". ⊣

5.2 Molecular Logics

Definition 31 (C–bisimulation for molecular connectives)**.** Let C be a set of molecular connectives and let $M_1 = (W_1, \mathcal{R}_1)$ and $M_2 = (W_2, \mathcal{R}_2)$ be two C–models. For all $c_0 \in$ C, let V_{c_0} be the vertices of the decomposition tree T_{c_0}. We associate to each vertex $c \in V_{c_0}$ a binary relation $Z_c \subseteq \bigcup_{k \in \mathbb{N}^*} (W_1^k \times W_2^k) \cup (W_2^k \times W_1^k)$. The set of such binary relations is denoted $\{Z\} \cup \bigcup_{c_0 \in \mathsf{C}} \{Z_c \mid c \in V_{c_0}\}$ and is such that if c is id_k for some $k \in \mathbb{N}^*$ then Z_c is Z and we have that $Z \subseteq \bigcap \{Z_c \mid c \in \mathsf{C}\}$. We say that this set of binary relations is a *C–bisimulation* between M_1 and M_2 when for all $c_0 \in$ C, all vertices $c \in V_{c_0}$, if $\{M, M'\} = \{M_1, M_2\}$ then for all $\overline{w}_1, \ldots, \overline{w}_n, \overline{w'}_1, \ldots, \overline{w'}_n, \overline{w}, \overline{w'} \in \overline{w}(M, \mathsf{C}) \cup \overline{w}(M', \mathsf{C})$,

1. if c is a propositional letter p then, $\overline{w} Z_c \overline{w'}$ and $\overline{w} \in [\![p]\!]$ imply $\overline{w'} \in [\![p]\!]$;
2. if c has skeleton $\star(c_1, \ldots, c_n)$ with $\star = (\sigma, \pm, \exists, \overline{k}, (\pm_1, \ldots, \pm_n))$ and we have $\overline{w} Z_c \overline{w'}$ and $R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w}$, then
$\exists \overline{w'}_1 \overline{w'}_2 \ldots \overline{w'}_n \left(\overline{w}_1 \bowtie_{c_1} \overline{w'}_1 \wedge \overline{w}_2 \bowtie_{c_2} \overline{w'}_2 \wedge \ldots \wedge \overline{w}_n \bowtie_{c_n} \overline{w'}_n \wedge R_\star'^{\pm\sigma} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'} \right)$;
3. if c has skeleton $\star(c_1, \ldots, c_n)$ with $\star = (\sigma, \pm, \forall, \overline{k}, (\pm_1, \ldots, \pm_n))$ and we have $\overline{w} Z_c \overline{w'}$ and $-R_\star'^{\pm\sigma} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'}$, then
$\exists \overline{w}_1 \overline{w}_2 \ldots \overline{w}_n \left(\overline{w}_1 \bowtie_{c_1} \overline{w'}_1 \wedge \overline{w}_2 \bowtie_{c_2} \overline{w'}_2 \wedge \ldots \wedge \overline{w}_n \bowtie_{c_n} \overline{w'}_n \wedge -R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w} \right)$;

where for all $j \in [\![1; n]\!]$, we have $\overline{w}_j \bowtie_{c_j} \overline{w'}_j \triangleq \begin{cases} \overline{w}_j Z_{c_j} \overline{w'}_j & \text{if } \pm_j = + \\ \overline{w'}_j Z_{c_j} \overline{w}_j & \text{if } \pm_j = - \end{cases}$.

When such a set of binary relations exists and is such that $\overline{w} Z \overline{w'}$, we say that $(M, \overline{w})$ and $(M', \overline{w'})$ are *C–bisimilar* and we write it $(M, \overline{w}) \to_{\mathsf{C}} (M', \overline{w'})$. ⊣

Note that case 1. is a particular instance of cases 2. and 3. with $n = 0$.

Definition 32. Let C be a set of molecular connectives. For all $c_0 \in$ C and all vertex c of the decomposition tree T_{c_0}, we define the language $\mathcal{L}_{c\mathsf{C}}$ as follows:

$$\mathcal{L}_{c\mathsf{C}} \triangleq \begin{cases} \{c(\varphi_1, \ldots, \varphi_n) \mid \varphi_1, \ldots, \varphi_n \in \mathcal{L}_\mathsf{C}\} & \text{if } c \text{ is of arity } n > 0 \\ \{p\} & \text{if } c = p \text{ is a propositional letter} \\ \mathcal{L}_\mathsf{C} & \text{if } c \text{ is } id_k \text{ for some } k \in \mathbb{N}^*. \end{cases}$$

Let $(M, \overline{w})$ and $(M', \overline{w'})$ be two pointed C–models. We write $(M, \overline{w}) \rightsquigarrow_{c\mathsf{C}} (M', \overline{w'})$ when for all $\varphi \in \mathcal{L}_{c\mathsf{C}}$, $(M, \overline{w}) \Vdash \varphi$ implies $(M', \overline{w'}) \Vdash \varphi$. We also write $(M, \overline{w}) \rightsquigarrow_\mathsf{C} (M', \overline{w'})$ when for all $\varphi \in \mathcal{L}_\mathsf{C}$, $(M, \overline{w}) \Vdash \varphi$ implies $(M', \overline{w'}) \Vdash \varphi$. ⊣

Proposition 4. *Let C be a set of molecular connectives and let $M_1 = (W_1, \mathcal{R}_1)$ and $M_2 = (W_2, \mathcal{R}_2)$ be two C–models. Let $\mathsf{C}_0 \subseteq \mathsf{C}$ and for all $c \in \mathsf{C}_0$, let V_c be the vertices of the decomposition tree T_c. Let $\{Z\} \cup \bigcup_{c_0 \in \mathsf{C}_0} \{Z_c \mid c \in V_{c_0}\}$ be a C_0–bisimulation between M_1 and M_2. If $\{M, M'\} = \{M_1, M_2\}$ then for all $c_0 \in \mathsf{C}_0$ and all $c \in V_{c_0}$, for all $\overline{w} \in \overline{w}(M, \mathsf{C})$ and all $\overline{w'} \in \overline{w}(M', \mathsf{C})$, if $\overline{w} Z_c \overline{w'}$ then $(M, \overline{w}) \rightsquigarrow_{c\mathsf{C}_0} (M', \overline{w'})$. In particular, if $\overline{w} Z \overline{w'}$ then $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}_0} (M', \overline{w'})$.*

Proof. We prove by induction on c that for all vertices c of the decomposition tree of a molecular connective $c_0 \in \mathbb{C}^*$, if $\overline{w} Z_c \overline{w'}$ then $(M, \overline{w}) \rightsquigarrow_{c\mathsf{C}} (M', \overline{w'})$.

If c is a propositional letter then it follows straightforwardly from condition 1 of Definition 31. Assume now that $\overline{w} Z_c \overline{w'}$ with c not a propositional letter. We prove by induction on $\varphi \in \mathcal{L}_{c\mathsf{C}}$ that $(M, \overline{w}) \Vdash \varphi$ implies $(M', \overline{w'}) \Vdash \varphi$. Let $\varphi = c(\varphi_1, \ldots, \varphi_m)$ with c of skeleton $\star(c_1, \ldots, c_n)$ where $c_1, \ldots, c_n \in \mathbb{C}^*$ are of arity $k_1, \ldots, k_n$ respectively and such that $k_1 + \ldots + k_n = m$. Then $\varphi = c(\varphi_1, \ldots, \varphi_m) = \star(c_1(\varphi_1^1, \ldots, \varphi_1^{k_1}), \ldots, c_n(\varphi_n^1, \ldots, \varphi_n^{k_n}))$ (here we confuse connectives with their skeletons to ease the presentation of the ideas).

1. Assume that $\star$ is of the form $(\sigma, \pm, \exists, \overline{k}, (\pm_1, \ldots, \pm_n))$.

Assume that $(M, \overline{w}) \Vdash c(\varphi_1, \ldots, \varphi_m)$. Then, $(M, \overline{w}) \Vdash \star (c_1(\varphi_1^1, \ldots, \varphi_1^{k_1}), \ldots, c(\varphi_n^1, \ldots, \varphi_n^{k_n}))$, *i.e.* there are $\overline{w}_1, \ldots, \overline{w}_n$ such that $R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w}$ and $\overline{w}_1 \pitchfork_1 [\![c_1(\varphi_1^1, \ldots, \varphi_1^{k_1})]\!]$ and $\ldots$ and $\overline{w}_n \pitchfork_n [\![c_n(\varphi_n^1, \ldots, \varphi_n^{k_n})]\!]$ (1). Then, by definition of Z_c, there are $\overline{w'}_1, \ldots \overline{w'}_n$ such that $R_\star^{\pm\sigma} \overline{w'}_1 \ldots \overline{w'}_m \overline{w'}$ (4) and $\overline{w}_1 \bowtie_{c_1} \overline{w'}_1$ and $\ldots$ and $\overline{w}_n \bowtie_{c_n} \overline{w'}_n$ (2) where for all $j \in [\![1; n]\!]$, we have $\overline{w}_j \bowtie_{c_j} \overline{w'}_j = \begin{cases} \overline{w}_j Z_{c_j} \overline{w'}_j & \text{if } \pm_j = + \\ \overline{w'}_j Z_{c_j} \overline{w}_j & \text{if } \pm_j = - \end{cases}$

Since $c_1(\varphi_1^1, \ldots, \varphi_1^{k_1}) \in \mathcal{L}_{c_1\mathsf{C}}, \ldots, c_n(\varphi_n^1, \ldots, \varphi_n^{k_n}) \in \mathcal{L}_{c_n\mathsf{C}}$, by Induction Hypothesis, for all $\overline{w} \in \overline{w}(M, \mathsf{C})$, all $\overline{w'} \in \overline{w}(M', \mathsf{C})$ and all $j \in [\![1; n]\!]$, if $(M, \overline{w}_j) \Vdash c(\varphi_j^1, \ldots, \varphi_j^{k_j})$ and $\overline{w}_j Z_{c_j} \overline{w'}_j$ then $(M', \overline{w'}_j) \Vdash c_j(\varphi_j^1, \ldots, \varphi_j^{k_j})$; and if $(M', \overline{w'}_j) \Vdash c(\varphi_j^1, \ldots, \varphi_j^{k_j})$ and $\overline{w'}_j Z_{c_j} \overline{w}_j$ then $(M, \overline{w}_j) \Vdash c_j(\varphi_j^1, \ldots, \varphi_j^{k_j})$ (3).

From (1) and (2), we derive by means of (3) that $\overline{w'}_1 \pitchfork_1 [\![c_1(\varphi_1^1, \ldots, \varphi_1^{k_1})]\!]$ and $\ldots$ and $\overline{w'}_n \pitchfork_n [\![c_n(\varphi_n^1, \ldots, \varphi_n^{k_n})]\!]$. Then, with (4), it follows that $(M', \overline{w'}) \Vdash \star (c_1(\varphi_1, \ldots, \varphi_1^{k_1}), \ldots, c_n(\varphi_n^1, \ldots, \varphi_n^{k_n}))$. That is, $(M', \overline{w'}) \Vdash c(\varphi_1, \ldots \varphi_m)$.

2. Assume that $\star$ is of the form $(\sigma, \pm, \forall, \overline{k}, (\pm_1, \ldots, \pm_n))$.

By contraposition, assume that it is not the case that $(M', \overline{w'}) \Vdash c(\varphi_1, \ldots, \varphi_m)$. That is, it is not the case that $(M', \overline{w'}) \Vdash \star (c_1(\varphi_1^1, \ldots, \varphi_1^{k_1}), \ldots, c_n(\varphi_n^1, \ldots, \varphi_n^{k_n}))$, *i.e.* there are $\overline{w'}_1, \ldots, \overline{w'}_n \in \overline{w}(M, \mathsf{C})$ such that $-R_\star^{\pm\sigma} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'}$ and not $\overline{w'}_1 \pitchfork_1 [\![c_1(\varphi_1^1, \ldots, \varphi_1^{k_1})]\!]$ and $\ldots$ and not $\overline{w'}_n \pitchfork_n [\![c_n(\varphi_n^1, \ldots, \varphi_n^{k_n})]\!]$ (1). Now, by definition of Z_c and since $\overline{w} Z_c \overline{w'}$ there are $\overline{w}_1, \ldots, \overline{w}_n$ such that $-R_\star^{\pm\sigma} \overline{w}_1 \ldots \overline{w}_n \overline{w}$ (2) and $\overline{w}_1 \bowtie_{c_1} \overline{w'}_1$ and $\ldots$ and $\overline{w}_n \bowtie_{c_n} \overline{w'}_n$ (2) where for all $j \in [\![1; n]\!]$, $\overline{w}_j \bowtie_{c_j} \overline{w'}_j = \begin{cases} \overline{w}_j Z_{c_j} \overline{w'}_j & \text{if } \pm_j = + \\ \overline{w'}_j Z_{c_j} \overline{w}_j & \text{if } \pm_j = - \end{cases}$. Like for the previous case, combining (1) and (3) with the Induction Hypothesis, we obtain that it holds that not $\overline{w}_1 \pitchfork_1 [\![c_1(\varphi_1^1, \ldots, \varphi_1^{k_1})]\!]$ and $\ldots$ and not $\overline{w}_n \pitchfork_n [\![c_n(\varphi_n^1, \ldots, \varphi_n^{k_n})]\!]$. Together with (2), we obtain that it is not the case that $(M, \overline{w}) \Vdash \star (c_1(\varphi_1^1, \ldots, \varphi_1^{k_1}), \ldots, c_n(\varphi_n^1, \ldots, \varphi_n^{k_n}))$. That is, it is not the case that $(M, \overline{w}) \Vdash c(\varphi_1, \ldots, \varphi_m)$.

Finally, assume that $\overline{w} Z_c \overline{w'}$ with $c = id_k$. Then, Z_c is Z and since by assumption $Z \subseteq \bigcap \{Z_c \mid c \in \mathsf{C}\}$, we obtain the result for formulas of the form $c(\varphi_1, \ldots, \varphi_m)$. □

Definition 33 (Uniform connective)**.** A *uniform connective* is a molecular connective c whose skeleton is of the form $\star(c_1, \ldots, c_n)$ with $\star = (\sigma, \pm, \text{Æ}, \overline{k}, (\pm_1, \ldots, \pm_n)) \in \mathbb{C}$ such that

1. $n \geq 1$ and $c_1, \ldots, c_n$ are molecular skeletons of arity 1;
2. for all $j \in [\![1; n]\!]$ such that $c_j \neq id_k$ for all $k \in \mathbb{N}^*$, $\text{Æ}(c_j) = \begin{cases} \forall & \text{if } \pm_j = + \\ \exists & \text{if } \pm_j = - \end{cases}$;

3. if c_0 is a molecular skeleton appearing in the decomposition tree of c of the form $c_0 = \star_0(c'_1, \ldots, c'_m)$ such that the tonicity signature of $\star_0$ is $(\pm_1, \ldots, \pm_m)$, then for all $i \in [\![1; m]\!]$, $Æ(c'_i) = \pm_i Æ(c_0)$. ⊣

According to our definition, molecular connectives of the form $\star(c(c'(c'_1, c'_2)))$ cannot be uniform connectives, unless c'_1 or c'_2 is a propositional letter. This is due to our first condition: in that case, $c(c'(c'_1, c'_2))$ should be of arity 1, which is possible only if c'_1 or c'_2 is a propositional letter. Hence, uniform connectives can be reduced to the composition of compound subconnectives $c_i^1, \ldots, c_i^{m_i}$, each of arity 1, so that molecular connectives are essentially of the form $\star(c_1^1(\ldots c_1^{m_1-1}(c_1^{m_1})), \ldots, c_m^1(\ldots c_m^{m_n-1}(c_m^{m_n})))$. Basically, uniform connectives are such that the quantification patterns of their successive internal connectives are essentially of the form $\exists \ldots \exists \ldots$ or $\forall \ldots \forall \ldots$

Example 14 (Modal intuitionistic logic)**.** Let $\mathsf{C} = \{p, \top, \bot, \wedge, \vee, \Rightarrow, \star, -\star'\}$ where $\star, \star' \in \mathbb{C}^*$ are the molecular connectives c, c' of Example 8 and where $\{p, \top, \bot, \wedge, \vee, \Rightarrow\}$ are defined in Example 13. The connectives of C are all uniform connectives. Note that $\star'$ is not a uniform connective and that is why we consider $-\star'$ for the moment, which is a uniform connective. We are going to see that we can easily get the bisimilarity condition for $\star'$ from the bisimilarity condition for $-\star'$.

Let $M_1 = (W_1, \{R_1, R_{1,\diamond}, P\})$ and $M_2 = (W_2, \{R_2, R_{2,\diamond}, P\})$ be two modal intuitionistic models. The set of binary relations $\{Z, Z^{\star}_{\star_2}, Z^{-\star'}_{\star_3}\}$ is a C–bisimulation iff for all $M, M' \in \{M_1, M_2\}$ with $M = (W, \{R, R_\diamond, P\})$ and $M' = (W', \{R', R'_\diamond, P\})$, all $w, v, u, w', v', u' \in \overline{w}(M, \mathsf{C}) \cup \overline{w}(M', \mathsf{C})$ and all $p \in \mathbb{P}$,

- if wZw' and $w \in [\![p]\!]$ then $w' \in [\![p]\!]$ (condition for p, like in Example 13);
- if vZv' and $R'v'w'$ and $R'u'w'$ then there are $u, w \in W$ such that $u'Zu$, wZw' and Rvw and Ruw (condition for $\star$, like in Example 13);
- if $R'w'v'$ and wZw' then there is $v \in W$ such that $vZ^{\star}_{\star_2}v'$ and Rwv,
  if $wZ^{\star}_{\star_2}w'$ and $R'_\diamond w'v'$ then there is $v \in W$ such that vZv' and Rwv
  (condition for $\star = \star_1(\star_2)$);
- if Rwv and wZw' then there is $v' \in W'$ such that $v'Z^{-\star'}_{\star_3}v$ and $R'w'v'$,
  if $wZ^{\star'}_{\star_3}w'$ and $R_\diamond wv$ then there is $v' \in W'$ such that vZv' and $Rw'v'$
  (condition for $-\star' = -\star_1(\star_3)$).

To obtain the bisimilarity condition for $\star'$, it suffices to observe that for all $c \in \mathbb{C}^*$, it holds that $(M, w) \rightsquigarrow_{\{-c\}} (M', w')$ iff $(M', w') \rightsquigarrow_{\{c\}} (M, w)$. So, we just have to replace wZw' by $w'Zw$ in the condition above. We obtain:

($\ast$) if $w'Zw$ and Rwv then there is $v' \in W'$ such that $v'Z_{\star_3}v$ and $R'w'v'$,
if $wZ_{\star_3}w'$ and $R_\diamond wv$ then there is $v' \in W'$ such that vZv' and $Rw'v'$.

It turns out that the Conditions of ($\ast$) are the conditions (diam–2(1)) and (diam–2(2)) of Olkhovikov [33, Definition 9], as expected. ⊣

Example 15 (Temporal logic)**.** Let us consider the connectives defined by the following first–order formulas:

$$\star_1(x) \triangleq \exists yzz' \left(\mathsf{P}y \wedge \mathsf{Q}zz' \wedge \mathsf{R}yzz'x\right)$$
$$\star'_1(x) \triangleq \exists yzz' \left(\mathsf{P}y \wedge \mathsf{Q}zz' \wedge \mathsf{R}xzz'y\right)$$
$$\star_2(x, x') \triangleq \forall y \left(\mathsf{P}y \vee \neg\mathsf{S}yxx'\right)$$

$\star_1$, $\star'_1$ and $\star_2$ are atomic connectives (this is independent from the definitions of R and S). Moreover, the molecular connectives of skeletons $c = \star_1(id_1, \star_2)$ and $c' = \star'_1(id_1, \star_2)$ are uniform connectives. We consider in this example the set of connectives $\mathsf{C} = \{p, \neg p, \top, \bot, \wedge, \vee, c, c'\}$.

Let $M_1 = (W_1, \{<_1, P\})$ and $M_2 = (W_2, \{<_2, P\})$ be two temporal models. We represent these temporal models by the C–models $M_1^{U,S} = (W_1, \{R_1, S_1, P\})$ and $M_2^{U,S} = (W_2, \{R_2, S_2, P\})$ respectively such that for all $v_1, u_1, u'_1, w_1, w'_1 \in W_1$,

$$R_1 v_1 u_1 u'_1 w_1 \text{ iff } w_1 <_1 v_1, w_1 = u_1 \text{ and } v_1 = u'_1 \tag{7}$$

$$S_1 v_1 w_1 w'_1 \text{ iff } w_1 <_1 v_1 <_1 w'_1 \tag{8}$$

and likewise for R_2 and S_2 of M_2. One can show that for all $\varphi, \psi \in \mathcal{L}_{\mathsf{C}}$ and all $w_1 \in W_1$, $(M_1, w_1) \models \varphi$ iff $(M_1^{U,S}, w_1) \Vdash \varphi$ (and likewise for M_2 and $M_2^{U,S}$) because the standard translation of the until and since operators are:

$$ST_x(U(\varphi, \psi)) = \exists y(x < y \wedge ST_y(\varphi) \wedge \forall z(x < z < y \to ST_z(\psi)))$$
$$ST_x(S(\varphi, \psi)) = \exists y(y < x \wedge ST_y(\varphi) \wedge \forall z(y < z < x \to ST_z(\psi)))$$

Let $Z \subseteq (W_1 \times W_2) \cup (W_2 \times W_1)$ and $Z_{\star_2} \subseteq ((W_1 \times W_1) \times (W_2 \times W_2)) \cup ((W_2 \times W_2) \times (W_1 \times W_1))$. Then, by Definition 31, $\{Z, Z_{\star_2}\}$ is a C–bisimulation iff for all $w, v \in W$, all $w', v' \in W'$ and all $p \in \mathbb{P} \cap \mathsf{C}$,

- if wZw' and $w \in [\![p]\!]$ then $w' \in [\![p]\!]$ (condition for p);
- if wZw' and $w' \in [\![p]\!]$ then $w \in [\![p]\!]$ (condition for $\neg p$);
- if $w < v$ and wZw' then there is $v' \in W'$ such that $w' < v'$ and vZv' and $(w, v)Z_{\star_2}(w', v')$, if $(w, v)Z_{\star_2}(w', v')$ and $w' < u' < v'$ then there is $u \in W$ such that uZu' and $w < u < v$ (condition for Until c);
- if $v < w$ and wZw' then there is $v' \in W'$ such that $v' < w'$ and vZv' and $(v, w)Z_{\star_2}(v', w')$, if $(v, w)Z_{\star_2}(v', w')$ and $v' < u' < w'$ then there is $u \in W$ such that uZu' and $v < u < w$ (condition for Since c').

We have rediscovered the notion of bisimulation for temporal logic introduced by Kurtonina & de Rijke [29]. The relation $Z_{\star_2}$ is presented differently in [29], it is split up into two relations $Z_1 \subseteq ((W_1 \times W_1) \times (W_2 \times W_2))$ and $Z_2 \subseteq ((W_2 \times W_2) \times (W_1 \times W_1))$ but the two formal definitions boil down to the same. $\dashv$

6 Basic Model Theory of Atomic and Molecular Logics

In this section, we are going to develop in a systematic and generic fashion the model theory of atomic and molecular logics.

6.1 Ultrafilters, Ultraproducts and Ultrapowers

In this section, we are going to recall and generalize to atomic logics a number of key notions and results of model theory, such as ultrafilter, ultraproducts, the Łoś theorem or an ultraproduct version of the compactness theorem. We will also recall the definition of ω–saturation which will play an important role in the sequel.

Definition 34 (Filter and ultrafilter)**.** Let I be a non–empty set. A *filter F over I* is a set $F \subseteq \mathcal{P}(I)$ such that

1. $I \in F$;
2. if $X, Y \in F$ then $X \cap Y \in F$;
3. if $X \in F$ and $X \subseteq Z \subseteq I$ then $Z \in F$.

A filter is called *proper* if it is distinct from $\mathcal{P}(I)$. An *ultrafilter over I* is a proper filter U such that for all $X \in \mathcal{P}(I)$, $X \in U$ iff $I - X \notin U$. A *countably incomplete ultrafilter* is an ultrafilter which is not closed under countable intersections. ⊣

In the rest of this section, I is a non-empty set and U is an ultrafilter over I.

Definition 35 (Ultraproduct of sets)**.** For each $i \in I$, let W_i be a non-empty set. For all $(w_i)_{i\in I}, (v_i)_{i\in I} \in \prod_{i\in I} W_i$, we say that $(w_i)_{i\in I}$ and $(v_i)_{i\in I}$ are *U-equivalent*, written $(w_i)_{i\in I} \sim_U (v_i)_{i\in I}$, if $\{i \in I \mid w_i = v_i\} \in U$. Note that $\sim_U$ is an equivalence relation on $\prod_{i\in I} W_i$. The equivalence class of $(w_i)_{i\in I}$ under $\sim_U$ is denoted $\prod_U w_i \triangleq \left\{(v_i)_{i\in I} \in \prod_{i\in I} W_i \mid (v_i)_{i\in I} \sim_U (w_i)_{i\in I}\right\}$.

The *ultraproduct of $(W_i)_{i\in I}$ modulo U* is $\prod_U W_i \triangleq \left\{\prod_U w_i \mid (w_i)_{i\in I} \in \prod_{i\in I} W_i\right\}$. When $W_i = W$ for all $i \in I$, the ultraproduct is called the *ultrapower* of W modulo U, written $\prod_U W$. ⊣

Definition 36 (Ultraproduct and ultrapower)**.** Let C be a set of molecular connectives and let $(M_i, \overline{w}_i)_{i\in I}$ be a family of pointed C–models. The *ultraproduct* $\prod_U (M_i, \overline{w}_i)$ of $(M_i, \overline{w}_i)$ modulo U is the pointed C–model $\left(\prod_U M_i, \prod_U \overline{w}_i\right)$ where $\prod_U M_i = (W_U, \mathcal{R}_U)$ and $\prod_U \overline{w}_i$ are defined by:

- $W_U = \prod_U W_i$;
- for all $n+1$–ary relations $R^i_\star$ of M_i, the $n+1$–ary relation $\prod_U R_\star \in \mathcal{R}_U$ is defined for all $\prod_U w^1_i, \dots, \prod_U w^{n+1}_i \in W_U$ by $\prod_U R_\star \prod_U w^1_i \dots \prod_U w^{n+1}_i$ iff $\left\{i \in I \mid R^i_\star w^1_i \dots w^{n+1}_i\right\} \in U$;
- $\prod_U \overline{w}_i \triangleq \left(\prod_U w^1_i, \dots, \prod_U w^k_i\right)$ if $(\overline{w}_i)_{i\in I} = (w^1_i, \dots, w^k_i)_{i\in I}$.

If $(M_i, s_i)_{i\in I}$ is a family of pointed structures, the *ultraproduct* $\prod_U (M_i, s_i)$ is the pointed structure $\left(\prod_U M_i, \prod_U s_i\right)$ where $\prod_U M_i$ is defined as above (the M_i are viewed as C–models) and $\prod_U s_i : \mathcal{V} \to \prod_U W_i$ is the assignment such that for all $x \in \mathcal{V}$, $\left(\prod_U s_i\right)(x) = \prod_U s_i(x)$.

If for all $i \in I$, $(M_i, \overline{w}_i) = (M, \overline{w})$ (and $(M_i, s_i) = (M, s)$) then $\prod_U (M_i, \overline{w}_i)$ is also called an *ultrapower* of $(M, \overline{w})$ (resp. (M, s)) modulo U, also denoted $\prod_U (M, \overline{w})$ (resp. $\prod_U (M, s)$). ⊣

Proposition 5. *Let* C *be a set of molecular connectives and let* $(M_i, \overline{w}_i)_{i\in I}$ *be a family of* C*–models. Let* $\prod_U (M_i, \overline{w}_i)$ *be an ultraproduct of* $(M_i, \overline{w}_i)_{i\in I}$*. Then, for all* $\varphi \in \mathcal{L}_{\mathsf{C}}$*,* $\prod_U (M_i, \overline{w}_i) \Vdash \varphi$ *iff* $\{i \in I \mid (M_i, \overline{w}_i) \Vdash \varphi\} \in U$*.*

Let $(M_i, s_i)_{i\in I}$ *be a family of pointed structures and let* $\prod_U (M_i, s_i)$ *be an ultraproduct of* $(M_i, s_i)_{i\in I}$*. Then, for all* $\varphi \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$*,* $\prod_U (M_i, s_i) \models \varphi$ *iff* $\{i \in I \mid (M_i, s_i) \models \varphi\} \in U$*.*

Proof. We consider the completion of C under Boolean negation: C can always be extended to include the Boolean negation of each connective (defined in Definition 18). Now, we prove it for this extended language by induction on φ.

1) The base case $\varphi = p$ holds by definition.

2) a) If $\varphi = \star(\varphi_1, \dots, \varphi_n)$ with $\star = (\sigma, \pm, \exists, (k, k_1, \dots, k_n), (\pm_1, \dots, \pm_n))$, then $\prod_U (M_i, \overline{w}_i) \Vdash \star(\varphi_1, \dots, \varphi_n)$

iff there are $\prod_U \overline{w}_i^1, \dots, \prod_U \overline{w}_i^n$ such that $\prod_U R_\star^{\pm\sigma} \prod_U \overline{w}_i^1 \dots \prod_U \overline{w}_i^n \prod_U \overline{w}_i$ and $\prod_U \overline{w}_i^1 ⫛_1 [\![\varphi_1]\!]$ and ... and $\prod_U \overline{w}_i^n ⫛_n [\![\varphi_n]\!]$

iff there are $\prod_U \overline{w}_i^1, \dots, \prod_U \overline{w}_i^n$ such that $\{j \in I \mid R_\star^{\pm\sigma} \overline{w}_j^1 \dots \overline{w}_j^n \overline{w}_j\} \in U$ and $\{j \in I \mid \overline{w}_j^1 ⫛_1 [\![\varphi_1]\!]\} \in U$ and ... and $\{j \in I \mid \overline{w}_j^n ⫛_n [\![\varphi_n]\!]\} \in U$ by Induction Hypothesis

iff there are $\prod_U \overline{w}_i^1, \dots, \prod_U \overline{w}_i^n$ such that $\{j \in I \mid R_\star^{\pm\sigma} \overline{w}_j^1 \dots \overline{w}_j^n \overline{w}_j$ and $\overline{w}_j^1 ⫛_1 [\![\varphi_1]\!]$ and $\overline{w}_j^n ⫛_n [\![\varphi_n]\!]\} \in U$ by closure under intersection of the filter definition (from left to right) and by closure by superset of the filter definition (from right to left)

iff $\{j \in I \mid$ there are $\overline{w}_j^1, \dots, \overline{w}_j^n$ such that $R_\star^{\pm\sigma} \overline{w}_j^1 \dots \overline{w}_j^n \overline{w}_j$ and $\overline{w}_j^1 ⫛_1 [\![\varphi_1]\!]$ and ... and $\overline{w}_j^n ⫛_n [\![\varphi_n]\!]\} \in U$

iff $\{j \in I \mid (M_j, \overline{w}_j) \Vdash \star(\varphi_1, \dots, \varphi_n)\} \in U$

b) If $\varphi = \star(\varphi_1, \dots, \varphi_n)$ with $\star = (\sigma, \pm, \forall, (t, t_1, \dots, t_n), (\pm_1, \dots, \pm_n))$, then $\prod_U (M_i, \overline{w}_i) \Vdash \varphi$ iff $\{i \in I \mid (M_i, \overline{w}_i) \Vdash \varphi\} \in U$

iff $\left(\left(\text{not } \prod_U (M_i, \overline{w}_i) \Vdash \varphi\right) \text{ iff } \{i \in I \mid (M_i, \overline{w}_i) \Vdash \varphi\} \notin U\right)$

iff $\left(\prod_U (M_i, \overline{w}_i) \Vdash -\varphi \text{ iff } \{i \in I \mid \text{ not } (M_i, \overline{w}_i) \Vdash \varphi\} \in U\right)$ by Proposition 1 and because U is an *ultra*filter (we recall that $-\varphi$ is defined in Definition 18)

iff $\left(\prod_U (M_i, \overline{w}_i) \Vdash -\varphi \text{ iff } \{i \in I \mid (M_i, \overline{w}_i) \Vdash -\varphi\} \in U\right)$ by Proposition 1.

This case thus boils down to the case 2) a).

c) If $\varphi = \varphi_1 \wedge \varphi_2$ or $\varphi = \varphi_1 \vee \varphi_2$ then the result follows directly from the definition of an ultrafilter.

As for the second part of the proposition, for pointed structures, the proof is similar and follows the main lines of the proof of [13, Theorem 4.1.9]. □

Proposition 6. *Let* C *be a set of molecular connectives and let* M *be a* C*–model. Let* $\prod_U M$ *be an ultrapower of* M*. Then, for all* $\varphi \in \mathcal{L}_{\mathsf{C}}$*, we have* $(M, \overline{w}) \Vdash \varphi$ *iff* $\prod_U (M, \overline{w}) \Vdash \varphi$*. Likewise, if* (M, s) *is a pointed structure then for all* $\varphi \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$*, we have* $(M, s) \models \varphi$ *iff* $\prod_U (M, s) \models \varphi$*.*

Proof. $\prod\limits_{U}(M,\overline{w}) \Vdash \varphi$

iff $\{i \in I \mid (M,\overline{w}) \Vdash \varphi\} \in U$ by Proposition 5

iff $(M,\overline{w}) \Vdash \varphi$ because $\varnothing \notin U$ since $I \in U$ by definition and because U is a *proper* filter.

The proof for first–order logic is similar. □

Proposition 7 (An ultraproduct version of the compactness theorem)**.** *Let Σ be a set of formulas of $\mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$, all with the same number of free variables or constants, let I be the set of all finite subsets of Σ, and for each $i \in I$, let (M_i, s_i) be a model of i. Then there exists an ultrafilter U over I such that $\prod\limits_{U}(M_i, s_i)$ is a model of Σ.*

Proof. The original formulation of this result [13, Corollary 4.1.11] is for sentences of $\mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$, but the proof can easily be adapted to our setting with free variables using item (ii) of the fundamental theorem of ultraproducts [13, Theorem 4.1.9]. □

We recall the definition of ω–saturation. In that definition, the C–model is simply viewed as a structure. We state it in its general form for tuples of variables $(x_1,\ldots,x_k)$. By [13, Proposition 2.3.6] this definition is equivalent to its usual formulation for a single variable x.

Definition 37 (ω–saturated model)**.** Let C be a set of molecular connectives. A C–model $M = (W,\mathcal{R})$ is *ω–saturated* when for all finite $Y = \{w_1,\ldots,w_n\} \subseteq W$ given, if all finite subsets $\Gamma' \subseteq \Gamma(\mathsf{c}'_1,\ldots,\mathsf{c}'_n,x_1,\ldots,x_k) \subseteq \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ with new constants $\mathsf{c}'_1,\ldots,\mathsf{c}'_n$ and free variables $x_1,\ldots,x_k$ are realized in M by assignments $s : \mathcal{V} \cup \mathcal{C} \cup \{\mathsf{c}'_1,\ldots,\mathsf{c}'_n\} \to W$ such that $s(\mathsf{c}'_1) = w_1,\ldots,s(\mathsf{c}'_n) = w_n$ ($\mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ is also defined with appropriate relations and function symbols), then $\Gamma(\mathsf{c}'_1,\ldots,\mathsf{c}'_n,x_1,\ldots,x_k)$ is realized in M (by an assignment s such that $s(\mathsf{c}'_1) = w_1,\ldots,s(\mathsf{c}'_n) = w_n$ as well). ⊣

Proposition 8. *Let $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$ be countable, U a countably incomplete ultrafilter over I and M a structure. Then, the ultrapower $\prod\limits_{U} M$ is ω–saturated.*

Proof. See [13, Theorem 6.1.1]. In fact [13, Theorem 6.1.1] proves that M is ω_1–saturated, which implies that it is ω–saturated. □

6.2 A Generic van Benthem Characterization Theorem

In this section we are going to generalize the van Benthem characterization theorem for modal logic [12, Theorem 2.68] to (compound) atomic logics.

Lemma 7. *Let c be a uniform connective and let c_0 be a molecular connective appearing in the decomposition tree of c. If Æ$(c_0) = \forall$ then for all $\varphi,\psi \in \mathcal{L}$, $c_0\varphi \wedge c_0\psi$ and $c_0(\varphi \times \psi)$ are true in the same pointed models; if Æ$(c_0) = \exists$ then for all $\varphi,\psi \in \mathcal{L}$, $c_0\varphi \vee c_0\psi$ and $c_0(\varphi \times \psi)$ are true in the same pointed models, where $\times$ is either $\wedge$ or $\vee$.*

Therefore, if $c = \star(c_1,\ldots c_n)$ then

$$\begin{aligned} &\star\left(\times_1\left\{c_1(\varphi_1^1),\ldots,c_1(\varphi_1^{p_1})\right\},\ldots,\times_n\left\{c_n(\varphi_n^1),\ldots,c_n(\varphi_n^{p_n})\right\}\right) \\ \textit{and}\ &\star\left(c_1\left(\varphi_1^1 \times'_1 \ldots \times'_1 \varphi_1^{p_1}\right),\ldots,c_n\left(\varphi_n^1 \times'_n \ldots \times'_n \varphi_n^{p_n}\right)\right) \end{aligned}$$

are true in the same models, where, for all $i \in \{1,\ldots,n\}$, $\times_i \triangleq \begin{cases} \bigwedge & \text{if } \pm_i = + \\ \bigvee & \text{if } \pm_i = - \end{cases}$ and $\times'_1,\ldots,\times'_n$ are either $\wedge$ or $\vee$.

Proof. We prove it by induction on the depth of c_0. Assume first that the molecular connective c_0 is of the form $c_0 = \star_0 \in \mathbb{C}$. If $\text{Æ}(c_0) = \forall$ and the tonicity signature of $\star_0$ is + then $c_0\varphi \wedge c_0\psi$ is equivalent to $c_0(\varphi \wedge \psi)$ (that is, they are true in the same models). If $\text{Æ}(c_0) = \forall$ and the tonicity signature of $\star_0$ is $-$ then $c_0\varphi \wedge c_0\psi$ is equivalent to $c_0(\varphi \vee \psi)$. Dually, if $\text{Æ}(c_0) = \exists$ and the tonicity signature of $\star_0$ is + then $c_0\varphi \vee c_0\psi$ is equivalent to $c_0(\varphi \vee \psi)$. If $\text{Æ}(c_0) = \exists$ and the tonicity signature of $\star_0$ is $-$ then $c_0\varphi \vee c_0\psi$ is equivalent to $c_0(\varphi \wedge \psi)$. Now, we prove the induction step. By definition, c_0 is of arity 1 and is of the form $c_0 = \star_0(c'_1, \dots, c'_m)$ where $c'_1, \dots, c'_m$ are molecular connectives. Thus, only one of them is of arity 1, all the others should be of arity 0. Hence, without loss of generality, we assume that the molecular connective c_0 is of the form $c_0 = \star_0 c'_0$ with c'_0 of arity 1. We are going to apply the previous case to that case and then the induction hypothesis. If $\text{Æ}(c_0) = \text{Æ}(\star_0) = \forall$ and the tonicity signature of $\star_0$ is + then $c_0\varphi \wedge c_0\psi = \star_0 c'_0\varphi \wedge \star_0 c'_0\psi$ is equivalent to $\star_0(c'_0\varphi \wedge c'_0\psi)$. If $\text{Æ}(c_0) = \text{Æ}(\star_0) = \forall$ and the tonicity signature of $\star_0$ is $-$ then $c_0\varphi \wedge c_0\psi = \star_0 c'_0\varphi \wedge \star_0 c'_0\psi$ is equivalent to $\star_0(c'_0\varphi \vee c'_0\psi)$. Dually, if $\text{Æ}(c_0) = \text{Æ}(\star_0) = \exists$ and the tonicity signature of $\star_0$ is + then $c_0\varphi \vee c_0\psi = \star_0 c'_0\varphi \vee \star_0 c'_0\psi$ is equivalent to $\star_0(c'_0\varphi \vee c'_0\psi)$. If $\text{Æ}(c_0) = \text{Æ}(\star_0) = \exists$ and the tonicity signature of $\star_0$ is $-$ then $c_0\varphi \vee c_0\psi = \star_0 c'_0\varphi \vee \star_0 c'_0\psi$ is equivalent to $\star_0(c'_0\varphi \wedge c'_0\psi)$. Now, we use the third (or second) condition of Definition 33 of a uniform connective. If we obtain an expression of the form $c'_0\varphi \wedge c'_0\psi$ then this entails by that condition that $\text{Æ}(c'_0) = \forall$ and if we obtain an expression of the form $c'_0\varphi \vee c'_0\psi$ then this entails by that condition that $\text{Æ}(c'_0) = \exists$. So, we can apply the induction hypothesis to these expressions. Eventually, we obtain the expected result.

The second part of the lemma is proved by iterating this decomposition a finite number of times and using condition 2 of Definition 33. □

Note that the formula $\star\left(c_1\left(\varphi_1^1 \times'_1 \dots \times'_1 \varphi_1^{p_1}\right), \dots, c_n\left(\varphi_n^1 \times'_n \dots \times'_n \varphi_n^{p_n}\right)\right)$ is a formula of $\mathcal{L}_{c\mathsf{C}}$ and it is this observation that will be useful in the proof of Lemma 8.

Lemma 8. *Let* C *be a set of uniform connectives and let* $M_1 = (W_1, \mathcal{R}_1)$ *and* $M_2 = (W_2, \mathcal{R}_2)$ *be two* ω*–saturated* C*–models. For all* $c \in \mathsf{C}$*, let* V_c *be the vertices of the decomposition tree* T_c*. Let* $\mathsf{C}^+ \triangleq \bigcup_{c \in \mathsf{C}} V_c$ *be the set of vertices of all the decomposition trees associated to the connectives of* C*. For all* $c = \star(c_1, \dots, c_n) \in \mathsf{C}^+$*, we define the binary relation* $Z_c \subseteq \left(W_1^k \times W_2^k\right) \cup \left(W_2^k \times W_1^k\right)$ *(where* k *is the output type of* $\star$*) as follows: if* $\{M, M'\} = \{M_1, M_2\}$ *then for all* $\overline{w} \in \overline{w}(M, \mathsf{C})$*, all* $\overline{w'} \in \overline{w}(M', \mathsf{C})$*,* $\overline{w} Z_c \overline{w'}$ *iff* $(M, \overline{w}) \rightsquigarrow_{c\mathsf{C}} (M', \overline{w'})$*. Then, the set of binary relations* $\{Z\} \cup \left\{Z_c \mid c \in \mathsf{C}^+\right\}$ *is a* C*–bisimulation between* M_1 *and* M_2 *(we recall that* Z_{id_k} *is* Z *for all* $k \in \mathbb{N}^*$*).*

Proof. (Here we identify connectives with their skeletons.) Condition 1. of Definition 31 holds trivially. Now, we prove condition 2. Let $c = \star(c_1, \dots, c_n)$ with $\star = (\sigma, \pm, \exists, \overline{k}, (\pm_1, \dots, \pm_n))$. Let $\overline{w_1}, \dots, \overline{w_n}, \overline{w} \in \overline{w}(M, \mathsf{C})$ and $\overline{w'} \in \overline{w}(M', \mathsf{C})$ be such that $R_\star^{\pm\sigma}\overline{w_1}\dots\overline{w_n}\overline{w}$ and $\overline{w} Z_c \overline{w'}$. For all $i \in \{1, \dots, n\}$, let $\Pi_+(M, \overline{w_i}) \triangleq \left\{ST_{\overline{x_i}}(\varphi) \mid \varphi \in \mathcal{L}_{c_i\mathsf{C}} \text{ and } (M, \overline{w_i}) \Vdash \varphi\right\}$ and $\Pi_-(M, \overline{w_i}) \triangleq \left\{\neg ST_{\overline{x_i}}(\varphi) \mid \varphi \in \mathcal{L}_{c_i\mathsf{C}} \text{ and not } (M, \overline{w_i}) \Vdash \varphi\right\}$. Let $I_- \triangleq \{i \in \{1, \dots, n\} \mid \pm_i = -\}$ and $I_+ \triangleq \{i \in \{1, \dots, n\} \mid \pm_i = +\}$. Let $\Gamma(\overline{x_1}, \dots, \overline{x_n}) \triangleq \bigcup_{i \in I_+} \Pi_+(M, \overline{w_i}) \cup \bigcup_{i \in I_-} \Pi_-(M, \overline{w_i})$ and let $\Gamma \subseteq \Gamma(\overline{x_1}, \dots, \overline{x_n})$ be finite. Then, $\Gamma = \Gamma_1 \cup \dots \cup \Gamma_n$ where all $\Gamma_i \subseteq \Pi_{\pm_i}(M, \overline{w_i})$ are finite.

Let $\psi = \star(\psi_1, \dots, \psi_n)$ where for all $i \in [\![1; n]\!]$, $\psi_i \triangleq \begin{cases} \bigwedge \{\varphi \mid ST_{\overline{x}_i}(\varphi) \in \Gamma_i\} & \text{if } \pm_i = + \\ \bigvee \{\varphi \mid \neg ST_{\overline{x}_i}(\varphi) \in \Gamma_i\} & \text{if } \pm_i = - \end{cases}$.
Then, we have that $(M, \overline{w}) \Vdash \psi$. Now, by Lemma 7, there is $\chi \in \mathcal{L}_{c\mathsf{C}}$ such that χ and ψ are true in the same pointed models $(*)$. So, $(M, \overline{w}) \Vdash \chi$. Therefore, $(M', \overline{w'}) \Vdash \chi$ because $(M, \overline{w}) \rightsquigarrow_{c\mathsf{C}} (M', \overline{w'})$ by assumption (because $\overline{w} Z_c \overline{w'}$). Hence, $(M', \overline{w'}) \Vdash \psi$, again because of $(*)$. Then, by definition of $\star$, there are $\overline{w'_1}, \dots, \overline{w'_n} \in \overline{w}(M', \mathsf{C})$ such that $\overline{w'_1} ⋔_1 [\![\psi_1]\!]$ and ... and $\overline{w'_n} ⋔_n [\![\psi_n]\!]$ and $R'^{\pm\sigma}_\star \overline{w'_1} \dots \overline{w'_m}\overline{w'}$. Therefore, for all $i \in [\![1; n]\!]$, $(M', \overline{w'_i}) \Vdash \Gamma_i$ and $R'^{\pm\sigma}_\star \overline{w'}_1 \overline{w'}_2 \dots \overline{w'}_n \overline{w'}$.

So, if $\Gamma \triangleq \Gamma(\overline{x}_1, \ldots, \overline{x}_n) \cup \{\mathsf{R}'^{\pm\sigma}_{\star} \overline{x_1} \ldots \overline{x_n} \overline{x} \overline{\mathsf{c}}\}$ where the tuple of new constants $\overline{\mathsf{c}}$ is interpreted by the distinguished elements $\overline{w'}$, then M' realizes every finite subset of Γ, namely in some states related to $\overline{w'}$. Thus, by ω–saturation of M', there are $\overline{w'}_1, \ldots, \overline{w'}_n \in \overline{w}(M', \mathsf{C})$ such that for all $i \in [\![1; n]\!]$, we have that $(M', \overline{w'}_i) \Vdash \Pi_{\pm_i}(M, \overline{w}_i)$ and $R'^{\pm\sigma}_{\star} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'}$. Moreover, by definition of $\Pi_+(M, \overline{w}_i)$, we have that if $\pm_i = +$ then $(M, \overline{w_i}) \rightsquigarrow_{c_i \mathsf{C}} (M', \overline{w'}_i)$ and, by definition of $\Pi_-(M, \overline{w}_i)$, we have that if $\pm_i = -$ then $(M', \overline{w'}_i) \rightsquigarrow_{c_i \mathsf{C}} (M, \overline{w}_i)$. This entails that there are $\overline{w'}_1, \ldots \overline{w'}_n$ such that $\overline{w}_1 \bowtie_{c_1} \overline{w'}_1$ and $\ldots$ and $\overline{w}_n \bowtie_{c_n} \overline{w'}_n$ and $R'^{\pm\sigma}_{\star} \overline{w'}_1 \ldots \overline{w'}_n \overline{w'}$ where for all $i \in \{1, \ldots, n\}$, $\overline{w}_i \bowtie_{c_i} \overline{w'}_i \triangleq \begin{cases} \overline{w}_i Z_{c_i} \overline{w'}_i & \text{if } \pm_i = + \\ \overline{w'}_i Z_{c_i} \overline{w}_i & \text{if } \pm_i = - \end{cases}$. This proves condition 2 of Definition 31. Condition 3 is proved similarly. □

Theorem 4 (Characterization theorem)**.** *Let C be a set of uniform atomic connectives complete for conjunction and disjunction. Let $\varphi(\overline{x}) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ with k free variables $\overline{x} = (x_1, \ldots, x_k)$ and let $(\mathcal{L}_{\mathsf{C}}, \mathcal{E}_{\mathsf{C}}, \models)$ be a atomic logic such that all models of $\mathcal{E}_{\mathsf{C}}$ contain relations $\{R_\star \mid \star \in \mathsf{C}\}$ interpreting all the predicates occuring in $\varphi(\overline{x})$. Let $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}}) \triangleq \left\{(M, s^{\overline{w}}_{\overline{x}}) \mid (M, \overline{w}) \in \mathcal{E}_{\mathsf{C}} \text{ of type } k\right\}$ and assume that $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ is closed under ultraproducts. The two following statements are equivalent:*

1. *There exists a formula $\psi \in \mathcal{L}_{\mathsf{C}}$ such that $\varphi(\overline{x}) \leftrightarrow ST_{\overline{x}}(\psi)$ is valid on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$;*
2. *$\varphi(\overline{x})$ is invariant for C–bisimulations on $\mathcal{E}_{\mathsf{C}}$, that is, for all pointed C–models $(M, \overline{w}), (M', \overline{w'})$ of $\mathcal{E}_{\mathsf{C}}$ of type k such that $(M, \overline{w}) \rightarrow_{\mathsf{C}} \left(M', \overline{w'}\right)$, we have that $\left(M, s^{\overline{w}}_{\overline{x}}\right) \models \varphi(\overline{x})$ implies $\left(M', s^{\overline{w'}}_{\overline{x}}\right) \models \varphi(\overline{x})$.*

Proof. The direction from 1. to 2. follows from Proposition 4. Now we prove the opposite direction, from 2. to 1. Let $\varphi(\overline{x}) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ with free variables $\overline{x}$. Assume that $\varphi(\overline{x})$ is invariant for C–bisimulations on $\mathcal{E}_{\mathsf{C}}$ and consider the following set:

$$C(\varphi) \triangleq \left\{ST_{\overline{x}}(\psi) \mid \psi \in \mathcal{L}_{\mathsf{C}}, k(\psi) = k \text{ and } \varphi \models ST_{\overline{x}}(\psi) \text{ on } ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})\right\}$$

The result is a direct consequence of the following two claims:

Claim (a). $C(\varphi) \models \varphi$ on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ – that is, for all pointed C-models $(M, \overline{w}) \in \mathcal{E}_{\mathsf{C}}$ of type k, if $(M, s^{\overline{w}}_{\overline{x}}) \models C(\varphi)$, then $(M, s^{\overline{w}}_{\overline{x}}) \models \varphi$;

Claim (b). If $C(\varphi) \models \varphi$ on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$, then φ is equivalent to the translation of a formula of $\mathcal{L}_{\mathsf{C}}$.

Proof of Claim (b). Assume that $C(\varphi) \models \varphi$ on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$. Then, $C(\varphi) \cup \{\neg\varphi\}$ is not satisfiable on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$. We show that there exists a finite subset $\Gamma \subseteq C(\varphi)$ such that $\Gamma \cup \{\neg\varphi\}$ is not satisfiable on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$. Indeed, assume towards a contradiction that it is not the case, then there would be an ultraproduct of elements of $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ that would satisfy $C(\varphi) \cup \{\neg\varphi\}$ by Proposition 7. Yet, since $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ is closed under ultraproducts, this ultraproduct would belong to $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$. Thus, in that case, there would be a model of $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ that satisfies $C(\varphi) \cup \{\neg\varphi\}$. This is impossible by assumption since $C(\varphi) \models \varphi$ on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$. So, we have proved that $\Gamma \models \varphi$ on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ for some finite $\Gamma \subseteq C(\varphi)$. The converse holds by the definition of $C(\varphi)$: $\varphi \models \bigwedge \Gamma$ on $\mathcal{E}_{\mathsf{C}}$. We thus have that $\varphi \leftrightarrow \bigwedge \Gamma$ is valid on $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$, proving the claim.

Proof of Claim (a). Take any pointed C–model $(M, \overline{w})$ such that $ST_{\overline{x}}(M, \overline{w}) \models C(\varphi)$ and consider the following set of formulas translated into first–order logic:

$$\Pi(M, \overline{w}) \triangleq \left\{\neg ST_{\overline{x}}(\psi) \mid \psi \in \mathcal{L}_{\mathsf{C}}, k(\psi) = k \text{ and not } (M, \overline{w}) \Vdash \psi\right\}.$$

Now consider the set of formulas:

$$\Sigma \triangleq \Pi(M,\overline{w}) \cup \{\varphi\}.$$

We first show that:

(a.1) Σ is satisfiable in a pointed C–model of $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$.

To prove (a.1), assume towards a contradiction that Σ is not satisfiable in $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$. Then, using the same reasoning as above, in particular Proposition 7 and the closure of $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ under ultraproducts, we can prove that $\Gamma \models \neg\varphi$ holds for some finite $\Gamma \subseteq \Pi(M,\overline{w})$. That is, $\varphi \models \neg \bigwedge \Gamma$. But then, by the definition of $C(\varphi)$, we have that $\neg \bigwedge \Gamma \in C(\varphi)$, and hence $\neg \bigwedge \Gamma \in \Pi(M,\overline{w})$, which is impossible since $\Gamma \subseteq \Pi(M,\overline{w})$.

Now, claim (a) follows if we can show that

(a.2) $(M, s^{\overline{w}}_{\overline{x}}) \models \varphi$.

Here is a proof for (a.2). Σ is satisfiable in a pointed C–model of $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$, say $(M', s^{\overline{w'}}_{\overline{x}})$. Observe first that $(M',\overline{w'}) \rightsquigarrow_{\mathsf{C}} (M,\overline{w})$ because for all $\psi \in \mathcal{L}_{\mathsf{C}}$, if not $(M,\overline{w}) \Vdash \psi$ then not $(M',\overline{w'}) \Vdash \psi$, that is, if $(M',\overline{w'}) \Vdash \psi$ then $(M,\overline{w}) \Vdash \psi$. Now take two ω–saturated elementary extensions $(M_\omega,\overline{w})$ and $(M'_\omega,\overline{w'})$ of $(M,\overline{w})$ and $(M',\overline{w'})$. That such extensions exist can be proved by a standard chain construction argument [13, Proposition 3.2.6]. We also have that $(M'_\omega,\overline{w'}) \rightsquigarrow_{\mathsf{C}} (M_\omega,\overline{w})$ by properties of elementary extensions. Then, by the invariance of FOL under elementary extensions, from $(M', s^{\overline{w'}}_{\overline{x}}) \models \varphi$ (by the construction of Σ), we get that $(M'_\omega, s^{\overline{w}}_{\overline{x}}) \models \varphi$. Moreover, we also have that $(M'_\omega,\overline{w'}) \rightarrow_{\mathsf{C}} (M_\omega,\overline{w})$ because $(M'_\omega,\overline{w'}) \rightsquigarrow_{\mathsf{C}} (M_\omega,\overline{w})$ and Lemma 8. By the assumption that φ is invariant for C–bisimulations, we get $(M_\omega, s^{\overline{w}}_{\overline{x}}) \models \varphi$ – and then, by elementary extension, $(M, s^{\overline{w}}_{\overline{x}}) \models \varphi$. This completes the proof. □

Remark 4. The assumption that $ST_{\overline{x}}(\mathcal{E}_{\mathsf{C}})$ is closed under ultraproducts is not really demanding since any class of structures definable by a set of first–order sentences is closed under ultraproducts by Keisler theorem (or by our Corollaries 2 and 3 and Theorem 11 to follow). For example, the class of (modal) intuitionistic models is closed under ultraproducts since it is definable by a set of first–order sentences (we only need to impose the reflexivity and transitivity on the binary relations by means of the validity of corresponding sentences). So, our generic theorem applies to these logics. It also applies to modal logic and to many others since the class of all Kripke models is definable by an empty set of sentences.

Example 16 (Modal logic and Lambek calculus)**.** Let C be the set of modal connectives of Example 11. Proposition 3 shows that C–bisimulations are modal bisimulation. Hence, we recover van Benthem's theorem for modal logic [12, Theorem 2.68] because $(M,\overline{w}) \rightarrow_{\mathsf{C}} (M',\overline{w'})$ implies $(M',\overline{w'}) \rightarrow_{\mathsf{C}} (M,\overline{w})$: every C–bisimulation can be extended into a symmetric C–bisimulation (see Example 11). On the other hand, we obtain a new model–theoretical result for the Lambek calculus. ⊣

6.3 Two Generic Keisler Axiomatizability Theorems

In this section, we are going to adapt the Keisler theorems for FOL to molecular logics. Our generic results will then be applied back to FOL in the next section. This type of theorems provides conditions of axiomatizability of logics given semantically by their language and classes of models. More precisely, they state that a given logic is axiomatizable if its class of models is closed under a specific construction called an ultraproduct and under a specific notion of bisimulation associated to the logic.

Theorem 5. *Let C be a set of uniform connectives complete for conjunction and disjunction and let $(M,\overline{w})$ and $(M',\overline{w'})$ be pointed C–models. Then the following are equivalent:*

1. $(M,\overline{w}) \rightsquigarrow_{C} (M',\overline{w'})$*;*

2. there exists a countably incomplete ultrafilter U over $\mathbb{N}$ such that $\prod_{U}(M,\overline{w}) \to_{C} \prod_{U}(M',\overline{w'})$.

Proof. We first prove that 2. implies 1. By Proposition 6, $(M,\overline{w}) \Vdash \varphi$ iff $\prod_{U}(M,\overline{w}) \Vdash \varphi$. By assumption and Proposition 4, this implies that $\prod_{U}(M',\overline{w'}) \Vdash \varphi$ and, again by Proposition 6, the latter is equivalent to $(M',\overline{w'}) \Vdash \varphi$.

Now, we prove that 1. implies 2. Let U be a countably incomplete ultrafilter over $\mathbb{N}$ (it exists by [12, Example 2.72]). By Proposition 8, the ultrapowers $\prod_{U} M$ and $\prod_{U} M'$ are ω–saturated. Now, for all $\varphi \in \mathcal{L}_{\mathsf{C}}$, $\prod_{U}(M,\overline{w}) \Vdash \varphi$ implies $\prod_{U}(M',\overline{w'}) \Vdash \varphi$. This claim follows from 1. and Proposition 6. Then, we apply Lemma 8 since $\prod_{U} M$ and $\prod_{U} M'$ are ω–saturated and we obtain the result. □

In the rest of that section, we are going to determine conditions for which a class of pointed models is definable by a set of formulas. Let us first be a bit more precise regarding what we mean by "definable" in our context with models and formulas of multiple types.

Definition 38 (Definability and closure under C–bisimulation and ultraproducts)**.** Let C be a set of molecular connectives, let $(M,\overline{w})$ be a pointed C–model of type k and let $T \subseteq \mathcal{L}_{\mathsf{C}}$. We write $(M,\overline{w}) \Vdash T$ when for all $\varphi \in T$ of type k, it holds that $(M,\overline{w}) \Vdash \varphi$.

Let K and K' be classes of pointed C–models. We say that K is *definable by a set of formulas (in K')* of $\mathcal{L}_{\mathsf{C}}$ when there is a set T of formulas of $\mathcal{L}_{\mathsf{C}}$ such that K is the set of pointed C–models $(M,\overline{w})$ (resp. of K') such that $(M,\overline{w}) \Vdash T$. The *types of* K, denoted $k(K)$, is the set of all types of the pointed C–models of K and we define $\overline{K} \triangleq \{(M,\overline{w}) \mid (M,\overline{w}) \text{ is a pointed } \mathsf{C}\text{–model (of } K') \text{ not in } K \text{ of type in } k(K)\}$.

We say that K is *closed under C–bisimulations* when for all $(M,\overline{w}) \in K$ and all pointed C–models $(M',\overline{w'})$, if $(M,\overline{w}) \to_{\mathsf{C}} (M',\overline{w'})$ then $(M',\overline{w'}) \in K$. We say that K is *closed under ultraproducts (ultrapowers)* when for all non-empty sets I, if for all $i \in I$ $(M_i,\overline{w}_i) \in K$ (resp. $(M,\overline{w}) \in K$) then $\prod_{U}(M_i,\overline{w}_i) \in K$ (resp. $\prod_{U}(M,\overline{w}) \in K$) for all ultraproducts U over I. ⊣

Lemma 9. *Let C be a set of molecular connectives and let $(M_i,\overline{w}_i)_{i\in I}$ be a family of pointed C–models of type k. Then, for all $\varphi \in \mathcal{L}_{\mathsf{C}}$ of type k and all tuples of free variables $\overline{x}$ of size k, $\prod_{U}(M_i,\overline{w}_i) \Vdash \varphi$ iff $\prod_{U}(M_i,s_{\overline{x}}^{\overline{w}_i}) \Vdash ST_{\overline{x}}(\varphi)$.*

Proof. We have that $\prod_{U}(M_i,\overline{w}_i) = \left(\prod_{U} M_i, \prod_{U}\overline{w}_i\right)$ where $\prod_{U}\overline{w}_i = \left(\prod_{U} w_i^1,\ldots,\prod_{U} w_i^k\right)$ if $\overline{w}_i = (w_i^1,\ldots,w_i^k)$ and $\prod_{U}(M_i,s_{\overline{x}}^{\overline{w}_i}) = \left(\prod_{U} M_i, \prod_{U} s_{\overline{x}}^{\overline{w}_i}\right)$. So, it suffices to prove that $s_{\overline{x}}^{\prod_U \overline{w}_i}$ and $\prod_{U} s_{\overline{x}}^{\overline{w}_i}$ coincide on the variables x_j of the tuple $\overline{x} = (x_1,\ldots,x_k)$ to obtain the result. And it turns out that for all $x_j \in \{x_1,\ldots,x_k\}$, $s_{\overline{x}}^{\prod_U \overline{w}_i}(x_j) = \prod_{U} w_i^j = \prod_{U} s_{\overline{x}}^{\overline{w}_i}(x_j) = \left(\prod_{U} s_{\overline{x}}^{\overline{w}_i}\right)(x_j)$. □

Theorem 6. *Let C be a set of uniform connectives complete for conjunction and disjunction and let K and K' be classes of pointed C-models such that K' is closed under ultraproducts. Then, the following are equivalent:*

1. *K is definable in K' by a set of formulas of $\mathcal{L}_{\mathsf{C}}$;*

2. *K is closed under C–bisimulations in K' and closed under ultraproducts, and $\overline{K}$ is closed under ultrapowers.*

Proof. The implication from 1. to 2. follows from Propositions 4, 5 and 6. For the converse, assume that K and $\overline{K}$ satisfy the closure conditions. Define T as the set of formulas of $\mathcal{L}_{\mathsf{C}}$ holding in K:

$$T \triangleq \bigcup_{k \in \mathbb{N}^*} \left\{ \varphi \in \mathcal{L}_{\mathsf{C}} \mid k(\varphi) = k \text{ and, for all } (M, \overline{w}) \in K \text{ of type } k,\ (M, \overline{w}) \Vdash \varphi \right\} \tag{9}$$

We will show that T defines the class K in K'. Let $(M, \overline{w})$ be a pointed C–model of K' of type k and assume that $(M, \overline{w}) \Vdash T$. To complete the proof of the theorem, we show that $(M, \overline{w})$ must be in K.

Define Σ to be the theory of FOL of $(M, \overline{w})$, with $\overline{x}$ a tuple of variables of the same size as $\overline{w}$:

$$\Sigma \triangleq \left\{ \neg ST_{\overline{x}}(\varphi) \mid \varphi \in \mathcal{L}_{\mathsf{C}} \text{ of type } k \text{ such that not } (M, \overline{w}) \Vdash \varphi \right\}.$$

Σ is finitely satisfiable in $ST_{\overline{x}}(K) \triangleq \left\{ (M, s^{\overline{w}}_{\overline{x}}) \mid (M, \overline{w}) \in K \right\}$. Indeed, assume towards a contradiction that the set $\{\neg ST_{\overline{x}}(\psi_1), \ldots, \neg ST_{\overline{x}}(\psi_n)\} \subseteq \Sigma$ is not satisfiable in $ST_{\overline{x}}(K)$. Then the formula $(\psi_1 \vee \ldots \vee \psi_n)$ of type k would be true on all pointed models in K of type k. Moreover, it belongs to $\mathcal{L}_{\mathsf{C}}$ because C is complete for disjunction. So $(\psi_1 \vee \ldots \vee \psi_n)$ would belong to T. However, it would be false in $(M, \overline{w})$, which is impossible. But then Proposition 7, adapted from [13, Corollary 4.1.11], shows that Σ is satisfiable in an ultraproduct $\prod_U (N_i, s^{\overline{v}_i}_{\overline{x}})$ of pointed models of $ST_{\overline{x}}(K)$.

Let us take $(N, \overline{v}) \triangleq \prod_U (N_i, \overline{v}_i)$, which belongs to K by closure of K under taking ultraproducts. Then, by Lemma 9, we have that for all $\varphi \in \mathcal{L}_{\mathsf{C}}$ of type k, $\prod_U (N_i, \overline{v}_i) \Vdash \varphi$ iff $\prod_U (N_i, s^{\overline{v}_i}_{\overline{x}}) \Vdash ST_{\overline{x}}(\varphi)$. So, we have that $(N, \overline{v}) \rightsquigarrow_{\mathsf{C}} (M, \overline{w})$ because for all $\varphi \in \mathcal{L}_{\mathsf{C}}$, not $(M, \overline{w}) \Vdash \varphi$ implies not $(N, \overline{v}) \Vdash \varphi$ is equivalent to, for all $\varphi \in \mathcal{L}_{\mathsf{C}}$, $(N, \overline{v}) \Vdash \varphi$ implies $(M, \overline{w}) \Vdash \varphi$. So, by Theorem 5, there exists an ultrafilter U' such that $\prod_{U'} (N, \overline{v}) \to_{\mathsf{C}} \prod_{U'} (M, \overline{w})$.

By closure under ultraproducts, the pointed model $\prod_{U'} (N, \overline{v})$ belongs to K. Moreover, $\prod_{U'} (M, \overline{w})$ is in K' by the assumption of closure of K' under ultraproduct. Hence, by closure under C–bisimulations, $\prod_{U'} (M, \overline{w})$ is in K as well. By closure of $\overline{K}$ under ultrapowers, it follows that $(M, \overline{w})$ is in K. This completes the proof. □

Theorem 7. *Let C be a set of uniform connectives complete for conjunction and disjunction and let K and K' be classes of pointed C–models of the same type such that K' is closed under ultraproducts. Then, the following are equivalent:*

1. *K is definable in K' by a single formula of $\mathcal{L}_{\mathsf{C}}$;*

2. *K is closed under C–bisimulation and ultraproducts and $\overline{K} \cap K'$ is closed under ultraproducts.*

Proof. The direction from 1. to 2. follows from Propositions 4 and 5. For the converse, we assume that K and $\overline{K}$ satisfy the stated closure conditions and that their pointed C–models are all of type k. Then, by Theorem 6, there is a set T of formulas of $\mathcal{L}_{\mathsf{C}}$ of type k defining K.

Let k be the type of the C–models of K and $\overline{K}$ and let $\overline{x}$ be a tuple of k variables. Let us define:

$$T_2 \triangleq \left\{\neg ST_{\overline{x}}(\varphi) \mid \varphi \in \mathcal{L}_{\mathsf{C}} \text{ of type } k \text{ and for all } (M,\overline{w}) \in \overline{K} \cap K',\ \text{not } (M,\overline{w}) \Vdash \varphi\right\}.$$

We are going to show that T_2 defines $ST_{\overline{x}}(\overline{K}) \triangleq \left\{ST_{\overline{x}}(M,\overline{w}) \mid (M,\overline{w}) \in \overline{K} \cap K'\right\}$. Let $(M,\overline{w}) \in \overline{K} \cap K'$ be such that $ST_{\overline{x}}(M,\overline{w}) \models T_2$ and let us define

$$\Sigma \triangleq \left\{\varphi \in \mathcal{L}_{\mathsf{C}} \mid (M,\overline{w}) \Vdash \varphi\right\}.$$

Then, Σ is finitely satisfiable in $\overline{K} \cap K'$. Indeed, assume towards a contradiction that there are $\varphi_1, \ldots, \varphi_n \in \Sigma$ which are not satisfiable in $\overline{K} \cap K'$. Then, $\neg ST_{\overline{x}}(\varphi_1 \wedge \ldots \wedge \varphi_n) \in T_2$. Therefore, $ST_{\overline{x}}(M,\overline{w}) \models \neg ST_{\overline{x}}(\varphi_1) \wedge \ldots \wedge \neg ST_{\overline{x}}(\varphi_n)$. However, $(M,\overline{w}) \Vdash (\varphi_1 \wedge \ldots \wedge \varphi_n)$. So, we reach a contradiction and Σ is finitely satisfiable in $\overline{K} \cap K'$. Thus, by compactness, we have that Σ is satisfiable in an ultraproduct $(N,\overline{v})$ of C-models of $\overline{K}$. By assumption, this ultraproduct $(N,\overline{v})$ belongs to $\overline{K} \cap K'$. Then, by definition of Σ, we have that $(M,\overline{w}) \rightsquigarrow_{\mathsf{C}} (N,\overline{v})$. Therefore, by Theorem 5, there exists a countably incomplete ultrafilter U over $\mathbb{N}$ such that $\prod_U (M,\overline{w}) \to_{\mathsf{C}} \prod_U (N,\overline{v})$. Now, if $\prod_U (M,\overline{w})$ belonged to K, then we would have by closure under C–bisimulation of K that $\prod_U (N,\overline{v})$ belongs to K as well, which is impossible. Therefore, $\prod_U (M,\overline{w})$ belongs to $\overline{K}$. Moreover, $(M,\overline{w})$ belongs to K' and since K' is closed under ultraproduct by assumption, we also have that $\prod_U (M,\overline{w})$ belongs to K'. By closure of K under ultraproducts, this also entails that $(M,\overline{w})$ belongs to $\overline{K}$. Hence, $ST_{\overline{x}}(M,\overline{w}) \in ST_{\overline{x}}(\overline{K})$ and T_2 does define $ST_{\overline{x}}(\overline{K})$.

Now, let us define

$$T_1 \triangleq \{ST_{\overline{x}}(\varphi) \mid \varphi \in T\}.$$

Then, because T defines K, we have that $T_1 \cup T_2$ is unsatisfiable. That is, there is no pointed C–model $(M,\overline{w})$ of type k such that $ST_{\overline{x}}(M,\overline{w}) \Vdash T_1 \cup T_2$. So then, by compactness, there exist $ST_{\overline{x}}(\varphi_1), \ldots, ST_{\overline{x}}(\varphi_n) \in T_1$ and $\neg ST_{\overline{x}}(\psi_1), \ldots, \neg ST_{\overline{x}}(\psi_m) \in T_2$ such that for all pointed C–models $(M,\overline{w})$, it is not the case that $ST_{\overline{x}}(M,\overline{w}) \Vdash ST_{\overline{x}}(\varphi_1) \wedge \ldots \wedge ST_{\overline{x}}(\varphi_n) \wedge \neg ST_{\overline{x}}(\psi_1) \wedge \ldots \wedge \neg ST_{\overline{x}}(\psi_m)$ $(*)$. To complete the proof, we show that K is in fact defined by the conjunction $(\varphi_1 \wedge \ldots \wedge \varphi_n)$. By definition, for any $(M,\overline{w})$ in K we have $(M,\overline{w}) \Vdash (\varphi_1 \wedge \ldots \wedge \varphi_n)$. Conversely, if $(M,\overline{w}) \Vdash (\varphi_1 \wedge \ldots \wedge \varphi_n)$ then there must be $i \in [\![1;m]\!]$ such that $ST_{\overline{x}}(M,\overline{w}) \Vdash \neg ST_{\overline{x}}(\psi_i)$ does not hold. Indeed, otherwise, we would have that $ST_{\overline{x}}(M,\overline{w}) \Vdash ST_{\overline{x}}(\varphi_1) \wedge \ldots \wedge ST_{\overline{x}}(\varphi_n) \wedge \neg ST_{\overline{x}}(\psi_1) \wedge \ldots \wedge \neg ST_{\overline{x}}(\psi_m)$, which is impossible by $(*)$. Therefore, there must be $i \in [\![1;m]\!]$ such that $(M,\overline{w}) \Vdash \psi_i$. Hence, by definition of T_2, this entails that $(M,\overline{w})$ does not belong to $\overline{K}$, whence $(M,\overline{w})$ belongs to K. □

Corollary 1. *Let $\mathsf{L} = (\mathcal{L}_{\mathsf{C}}, \mathcal{E}_{\mathsf{C}}, \Vdash)$ be a molecular logic whose set of connectives C is uniform and complete for conjunction and disjunction and whose class $\mathcal{E}_{\mathsf{C}}$ of C–models are all of the same type. Then, L is axiomatizable iff $\mathcal{E}_{\mathsf{C}}$ is closed under C–bisimulation and ultraproducts and $\overline{\mathcal{E}_{\mathsf{C}}}$ is closed under ultraproducts.*

Proof. It follows straightforwardly from the definition of axiomatizability recalled in Section 2.3 and Theorem 7. □

7 Applications to First–order Logics

The results of this section will roughly be applications and adaptations to first–order logic of the following informal corollary: if $\mathsf{L} = (\mathcal{L}, \mathcal{E}, \models)$ is a logic as expressive as a atomic logic $\mathsf{L}_\mathsf{C} = (\mathcal{L}_\mathsf{C}, \mathcal{E}_\mathsf{C}, \Vdash)$, then Theorems 5, 6 and 7 hold if we replace in their statements "pointed C–models" with "models of $\mathcal{E}$", the language $\mathcal{L}_\mathsf{C}$ with the language $\mathcal{L}$ and C–bisimulations with the appropriate notion of C–bisimulation on the models of $\mathcal{E}$. Our translations of pure predicate logic and first–order logic into atomic logics of Section 4.1 are the key results that will now allow us to find out the appropriate notions of predicate bisimulation and first–order bisimulation associated to pure predicate logic and first–order logic.

We emphasize that these bisimulation notions for classical logics are not 'inspired' by our work on atomic and molecular logics or defined *a priori*, they are automatically computed from our embedding of pure predicate logic and first-order logic into specific atomic logics that we defined in Section 4.1 and from our definitions of C-bisimulation in Definitions 29 and 31. We are just going to proceed for first-order logic as we proceeded for modal logic, the Lambek calculus, temporal logic and (modal) intuitionistic logic. The axiomatic and mathematical presentation of our results of the current section in terms of definitions followed by theorems may hide this aspect but, from a heuristic point of view, the process described in the proof of Lemma 10 was first carried out during the course of our research. The bisimulation conditions were first spelled out as they are in the proof of that lemma and the definitions of our predicate bisimulation and first-order bisimulation were then obtained from these conditions. They are presented before the proof of Lemma 10 because we have reorganized the results of our research to follow a traditional mathematical presentation. The fact that the bisimulation notions that we obtain are very close to the usual notions of (partial) isomorphism was at the same time very surprising and reassuring. Indeed, the connectives of predicate atomic logic are very specific and their corresponding bisimulation conditions are quite different from the conditions that define a partial isomorphism. The fact that their combination altogether allows us to recover a slight variant of the notion of partial isomorphism is stricking. This would have been somehow problematic if we had found out completely different invariance notions for first-order logic and pure predicate logic. Hence, we do not claim to have introduced brand new notions of invariance, they are in fact natural and intuitive variants of the usual notions. Instead, we claim to have introduced a generic notion of C-bisimulation which is somehow 'deeper' and more basic than the usual notion of bisimulation for modal logic or even the usual notion of (partial) isomorphism for first-order logic since all invariance notions introduced in the literature (including those) can all be seen as instances of our general notions of C-bisimulation of Definitions 29 and 31.

In this section, the first–order languages that we consider are assumed to be countable.

7.1 Pure Predicate Logic

Predicate atomic logic is as expressive as pure predicate logic by Theorem 1. So, we would expect that the notion of invariance of predicate atomic logic, that is C–bisimulation, is the same notion of invariance as FOL, that is isomorphism or partial isomorphism. It turns out that by applying our definitions and results we rediscover the notion of partial isomorphism, to which are added three natural conditions.

Definition 39 (Predicate bisimulation)**.** A *predicate bisimulation* between two structures M_1 and M_2 is a non–empty relation Z between finite sequences of the same length of M_1 and M_2 or M_2

and M_1 such that for all $M, M' \in \{M_1, M_2\}$ with $M = (W, \mathcal{R})$ and $M' = (W', \mathcal{R}')$, all $n \in \mathbb{N}^*$, all $w_1, \ldots, w_n \in W$, all $w'_1, \ldots, w'_n \in W'$, all $\sigma \in \mathfrak{S}_n$,

1. if $(w_1, \ldots, w_n)Z(w'_1, \ldots, w'_n)$ then for all n–ary relations $R \in \mathcal{R}$ and $R' \in \mathcal{R}'$ associated to the same predicate R, if $Rw_1 \ldots w_n$ then $R'w'_1 \ldots w'_n$;
2. if $(w_1, \ldots, w_n)Z(w'_1, \ldots, w'_n)$ then for all $w \in M$ there is $w' \in M'$ such that $(w_1, \ldots, w_n, w)Z(w'_1, \ldots, w'_n, w')$;
3. if $(w_1, \ldots, w_n)Z(w'_1, \ldots, w'_n)$ then $(w_{\sigma(1)}, \ldots, w_{\sigma(n)})Z(w'_{\sigma(1)}, \ldots, w'_{\sigma(n)})$;
4. if $(w_1, \ldots, w_n, w_{n+1})Z(w'_1, \ldots, w'_n, w'_{n+1})$ then $(w_1, \ldots, w_n)Z(w'_1, \ldots, w'_n)$;
5. if $(w_1, \ldots, w_n)Z(w'_1, \ldots, w'_n)$ then $(w'_1, \ldots, w'_n)Z(w_1, \ldots, w_n)$;
6. for all $\mathsf{c}_1, \ldots, \mathsf{c}_n \in \mathcal{C}$ it holds that $(s(\mathsf{c}_1), \ldots, s(\mathsf{c}_n))Z(s'(\mathsf{c}_1), \ldots, s'(\mathsf{c}_n))$. ⊣

Proposition 9. *A predicate bisimulation between structures without functions is a partial isomorphism satisfying moreover conditions 3, 4 and 5 of Definition 39, and vice versa. Moreover, if we remove condition 5 of a predicate bisimulation and replace it with the 'back' condition of partial isomorphisms and if we replace as well the implication of condition 1 with a bi–implication, then we obtain an equivalent definition of predicate bisimulation.*

Proof. Condition 1. of a partial isomorphism follows from conditions 1. and 6. of a predicate bisimulation. Condition 2. is the same. The "back" condition of a partial isomorphism is obtained by condition 2 and 5 of a predicate bisimulation. We have to show that condition 3 of Definition 1 of a partial isomorphism is deducible from the definition of a predicate bisimulation. Let Z be a predicate bisimulation between the pointed structures (M, s) and (M', s'). Let $\overline{w} \in M^k$ and $\overline{w'} \in {M'}^k$ with $k > 0$ and assume that $\overline{w}Z\overline{w'}$ (such a pair exists since a predicate bisimulation is non empty). For all $v \in M$, there is $v' \in M'$ such that $(\overline{w}, v)Z(\overline{w'}, v')$. Then, we have that $(v, \overline{w})Z(v', \overline{w'})$ by condition 3 and then $(v)Z(v')$ by iterative application of condition 4 of a predicate bisimulation. Hence, we have proved condition 3 of a partial isomorphism.

The second part of the proposition follows from the fact that the 'back' of condition 2 (usually formalized by the addition of the expression "vice versa") is gotten by an application of condition 2 to the symmetric pair obtained by the application of condition 5. Conversely, the symmetry of condition 5 can be obtained by the fact that the other conditions are all symmetric conditions. □

Now that we have elicited the invariance notion of pure predicate logic, that is predicate bisimulation, we are going to obtain a series of theorems as specific instances or corollaries of Theorems 5, 6 and 7. Beforehand, we generalize predicate bisimulations to the notion of $\overline{X}$–compatible predicate bisimulations.

Definition 40 (Profile and $\overline{X}$–compatible predicate bisimulation)**.** A *profile* $\overline{X}$ is a non-empty subset of $\bigcup_{k \in \mathbb{N}^*} (\mathcal{V} \cup \mathcal{C})^k$ closed under permutations on the elements of its tuples. If $\overline{X}$ is a profile then an $\overline{X}$*–formula* is a formula $\varphi(x_1, \ldots, x_k) \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}(\overline{x})$ such that $(x_1, \ldots, x_k) \in \overline{X}$.

If $\overline{X}$ is a profile then a $\overline{X}$*–compatible predicate bisimulation* Z between two pointed structures (M, s) and (M', s') is a predicate bisimulation where the condition 6. is replaced by the following condition:

6'. for all $(x_1, \ldots, x_n) \in \overline{X}$, it holds that $(s(x_1), \ldots, s(x_n))\, Z \,\big(s'(x_1), \ldots, s'(x_n)\big)$.

Therefore, a predicate bisimulation is a $\mathcal{C}^+$–compatible predicate bisimulation (we recall that $\mathcal{C}^+$ is the set of finite sequences of constants of $\mathcal{C}$). ⊣

Proposition 10. *If there is an $\overline{X}$–compatible predicate bisimulation between two pointed structures then they make true the same $\overline{X}$–formulas.*

Proof. This follows from Proposition 9 and the corresponding property for partial isomorphisms together with Condition 6' of Definition 40. □

The following lemma is really important since it connects our work about atomic logics with standard invariance notions of first–order logic such as partial bisimulations.

Lemma 10. *Let $\mathsf{C}^{\mathcal{P}}$ be the set of predicate atomic connectives (introduced in Definition 19). A $\mathsf{C}^{\mathcal{P}}$–bisimulation between two $\mathsf{C}^{\mathcal{P}}$–models is a predicate bisimulation between their associated structures, but which does not necessarily fulfill condition 6.*

Proof. Let us consider the predicate atomic connectives $\mathsf{C}^{\mathcal{P}} = \mathbb{B} \cup \left\{\mathsf{R}_{f_l^k} \mid \mathsf{R} \in \mathcal{P} \text{ of arity } k \text{ and } f_l^k : [\![1;k]\!] \to [\![1;l]\!] \text{ surjective with } l \leq k\right\} \cup \{\bot\} \cup \{[\sigma_k] \mid k \in \mathbb{N}^*, \sigma_k \in \mathfrak{S}_k\} \cup \{\|_{k_1,k_2} \mid k_1, k_2 \in \mathbb{N}^*\} \cup \{\supset_k, \forall_k \mid k \in \mathbb{N}^*\} \cup \{\forall_0\}$ (defined in Definition 19). Let M_1 and M_2 be two $\mathsf{C}^{\mathcal{P}}$–models and let Z be a non-empty binary relation between finite sequences of M_1 and M_2 of the same length. Then, by Definition 29, Z is a $\mathsf{C}^{\mathcal{P}}$–bisimulation between M_1 and M_2 when for all $M, M' \in \{M_1, M_2\}$ with $M = (W, \mathcal{R})$ and $M' = (W', \mathcal{R}')$, all $\overline{w}, \overline{v}, \overline{u} \in \overline{w}(M, \mathsf{C})$, all $w, v \in W$ and all $\overline{w'}, \overline{v'}, \overline{u'} \in \overline{w}(M', \mathsf{C})$, all $w', v' \in W'$, all $\mathsf{R} \in \mathcal{P}$, it holds that

- condition for $\mathsf{R}_{f_l^k}$: if $\overline{w}Z\overline{w'}$ and $R\overline{w}$ then $R'\overline{w'}$, for all relations R and R' both associated to R (in the associated structure);
- condition for $\bot$: if wZw' and $R'_\bot(w')$ then $R_\bot(w)$, this condition always holds;
- condition for $[\sigma_k]$: for all $k \in \mathbb{N}^*$ and all $\sigma_k \in \mathfrak{S}_k$, if $\overline{w}Z\overline{w'}$ and $R'^{\sigma_k}\overline{w'v'}$ then there is $\overline{v} \in \overline{w}(M, \mathsf{C})$ such that $\overline{v}Z\overline{v'}$ and $R^{\sigma_k}\overline{wv}$,

  that is, if $\overline{w}Z\overline{w'}$ then $\sigma_k(\overline{w})Z\sigma_k(\overline{w'})$, where for all $\overline{w} = (w_1, \ldots, w_k)$, $\sigma_k(\overline{w}) \triangleq (w_{\sigma(1)}, \ldots, w_{\sigma(k)})$;
- condition for $\|_{k_1,k_2}$: for all $k_1, k_2 \in \mathbb{N}^*$, if $\overline{w}Z\overline{w'}$ and $R'_{k_1,k_2}\overline{v'u'w'}$ then there are $\overline{v}, \overline{u} \in \overline{w}(M, \mathsf{C})$ such that $\overline{v}Z\overline{v'}$, $\overline{u}Z\overline{u'}$ and $R_{k_1,k_2}\overline{vuw}$,

  that is, for all $n > 1$, if $(w_1, \ldots, w_n)Z(w'_1, \ldots, w'_n)$ then for all $1 \leq k < n$, $(w_1, \ldots, w_k)Z(w'_1, \ldots, w'_k)$ and $(w_{k+1}, \ldots, w_n)Z(w'_{k+1}, \ldots, w'_n)$;
- condition for $\supset_k$: for all $k \in \mathbb{N}^*$, if $\overline{w}Z\overline{w'}$ and $R_k\overline{v'u'w'}$ then there are $\overline{v}, \overline{u} \in \overline{w}(M, \mathsf{C})$ such that $\overline{v'}Z\overline{v}$, $\overline{u}Z\overline{u'}$ and $R_k\overline{vuw}$,

  that is, if $\overline{w}Z\overline{w'}$ then $\overline{w'}Z\overline{w}$;
- condition for $\forall_k$: for all $k \in \mathbb{N}^*$, if $\overline{w}Z\overline{w'}$ and $R'_{k,1}\overline{w'}v'\overline{u'}$ then there are $v, \overline{u} \in \overline{w}(M, \mathsf{C})$ such that vZv' and $\overline{u}Z\overline{u'}$ and $R_{k,1}\overline{w}v\overline{u}$,

  that is, if $\overline{w}Z\overline{w'}$ then for all $v' \in W'$ there is $v \in W$ such that vZv' and $(\overline{w}, v)Z(\overline{w'}, v')$;
- condition for $\forall_0$: if wZw' and $v' \in W'$ then there is $v \in W$ such that vZv',

  that is, for all $w' \in W'$ there is $w \in W$ such that wZw'.

Now, if we combine the conditions above, we obtain the definition of a predicate bisimulation on the structures associated to M_1 and M_2 without condition 6 (we recall that associated structures are defined in Definition 22). □

Lemma 11. *Let M be a structure without functions. If M is ω–saturated then the $\mathsf{C}^{\mathcal{P}}$–model $T(M)$ (defined in Definition 21) is also ω–saturated (for a first–order language with predicate symbols associated only to the relations $R_{f_l^k}$ of $T(M)$).*

Proof. We introduce the first–order language $\mathcal{L}_{\mathsf{FOL}}^T$ for $T(M)$ based on the l–ary predicates $\mathsf{R}_{f_l^k}$ associated to each relation $R_{f_l^k}$ such that for all assignments s over $T(M)$, we have that $(T(M), s) \models \mathsf{R}_{f_l^k} x_1 \dots x_l$ iff $R_{f_l^k} s(x_1) \dots s(x_l)$ holds. We define also the first–order language $\mathcal{L}_{\mathsf{FOL}}$ for M based on the k–ary predicates R associated to each relation R (which itself defines the relations $R_{f_l^k}$ of $T(M)$) such that for all assignments s over M, we have that $(M, s) \models \mathsf{R} x_1 \dots x_k$ iff $R s(x_1) \dots s(x_k)$ holds. Then we define the translation $T^- : \mathcal{L}_{\mathsf{FOL}}^T \to \mathcal{L}_{\mathsf{FOL}}$ by stating $T^-(\mathsf{R}_{f_l^k} x_1 \dots x_l) \triangleq \mathsf{R} x_{f_l^k(1)} \dots x_{f_l^k(k)}$ and the induction steps being as usual. Thus, for all $\varphi \in \mathcal{L}_{\mathsf{FOL}}^T$, one can easily prove that $(T(M), s) \models \varphi$ iff $(M, s) \models T^-(\varphi)$. The ω–saturation of $T(M)$ for $\mathcal{L}_{\mathsf{FOL}}^T$ then follows from the ω–saturation of M for $\mathcal{L}_{\mathsf{FOL}}$ since every set of formulas of $\mathcal{L}_{\mathsf{FOL}}^T$ can be translated into a set of formulas of $\mathcal{L}_{\mathsf{FOL}}$ by means of T^-. □

The following Theorem 8 is a generalized version of the Keisler-Shelah isomorphism theorem [13, Theorem 6.1.15] of model theory for the pure predicate language with free variables and constants. With our approach, isomorphisms are replaced by predicate bisimulations.

Theorem 8. *Assume that $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$ is countable, let (M, s) and (M', s') be two pointed structures and let $\overline{X}$ be a profile. Then the following are equivalent:*

1. *(M, s) and (M', s') make true the same $\overline{X}$–formulas of $\mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$;*
2. *there exists an ultrafilter U and a $\overline{X}$–compatible predicate bisimulation between $\prod_U (M, s)$ and $\prod_U (M', s')$.*

Proof. The proof follows the same reasoning as the proof of Theorem 5, using Lemma 8 and Propositions 6 and 8 as well as the 'translation' Lemmas 1 and 2.

First, we prove that 2. implies 1. By Proposition 6, for all $\overline{X}$–formulas $\varphi \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$, we have that $(M, s) \Vdash \varphi$ iff $\prod_U (M, s) \Vdash \varphi$. By assumption and Proposition 10, this implies that $\prod_U (M', s') \Vdash \varphi$ and, again by Proposition 6, the latter is equivalent to $(M', s') \Vdash \varphi$.

Now, we prove that 1. implies 2. Let U be a countably incomplete ultrafilter over $\mathbb{N}$ (it exists by [12, Example 2.72]). By Proposition 8, the ultrapowers $\prod_U M$ and $\prod_U M'$ are ω–saturated and therefore $T\left(\prod_U M\right)$ and $T\left(\prod_U M'\right)$ are ω–saturated as well by Lemma 11. Now, for all $\overline{X}$–formula $\varphi \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$, $\prod_U (M, s) \Vdash \varphi$ implies $\prod_U (M', s') \Vdash \varphi$ because of 1. and Proposition 6. That is, for all $\overline{X}$–formula $\varphi \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$, $\left(\prod_U M, \prod_U s\right) \Vdash \varphi$ implies $\left(\prod_U M', \prod_U s'\right) \Vdash \varphi$. Therefore, by Lemma 1, for all $\overline{X}$–formula $\varphi \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$, we have that $T\left(\prod_U M, \prod_U s\right) \Vdash T(\varphi)$ implies $T\left(\prod_U M', \prod_U s'\right) \Vdash T(\varphi)$. That is, for all $\overline{X}$–formula $\varphi(x_1, \dots, x_k) \in \mathcal{L}_{\mathsf{FOL}}^{\mathcal{P}}(\overline{x})$, we have that $\left(T\left(\prod_U M\right), \left(\prod_U s(x_1), \dots, \prod_U s(x_k)\right)\right) \Vdash T(\varphi)$

implies $\left(T\left(\prod_U M'\right), \left(\prod_U s'(x_1), \ldots, \prod_U s'(x_k)\right)\right) \Vdash T(\varphi)$ $(*)$. Then, using Lemmas 1 and 2, we can prove the following:

Claim. For all $(x_1, \ldots, x_k) \in \overline{X}$,

$$\left(T\left(\prod_U M\right), \left(\prod_U s(x_1), \ldots, \prod_U s(x_k)\right)\right) \rightsquigarrow_{\mathsf{C}^{\mathcal{P}}} \left(T\left(\prod_U M'\right), \left(\prod_U s'(x_1), \ldots, \prod_U s'(x_k)\right)\right).$$

Proof of the Claim. Let $\varphi \in \mathcal{L}_{\mathsf{C}^{\mathcal{P}}}$ and let $(x_1, \ldots, x_k) \in \overline{X}$. Assume that

$$\left(T\left(\prod_U M\right), \left(\prod_U s(x_1), \ldots, \prod_U s(x_k)\right)\right) \Vdash \varphi.$$

Then, by Lemma 2,

$$ST_{\overline{x}}\left(T\left(\prod_U M\right), \left(\prod_U s(x_1), \ldots, \prod_U s(x_k)\right)\right) \Vdash ST_{\overline{x}}(\varphi).$$

That is,

$$\left(ST\left(T\left(\prod_U M\right)\right), s_{\overline{x}}^{(\prod_U s(x_1), \ldots, \prod_U s(x_k))}\right) \Vdash ST_{\overline{x}}(\varphi).$$

So, by Lemma 1,

$$\left(T\left(ST\left(T\left(\prod_U M\right)\right)\right), \left(\prod_U s(x_1), \ldots, \prod_U s(x_k)\right)\right) \Vdash T\left(ST_{\overline{x}}(\varphi)\right).$$

That is, again by Lemma 2,

$$\left(T\left(\prod_U M\right), \left(\prod_U s(x_1), \ldots, \prod_U s(x_k)\right)\right) \Vdash T\left(ST_{\overline{x}}(\varphi)\right).$$

So, by $(*)$, we have that

$$\left(T\left(\prod_U M'\right), \left(\prod_U s'(x_1), \ldots, \prod_U s'(x_k)\right)\right) \Vdash T\left(ST_{\overline{x}}(\varphi)\right).$$

That is, again by Lemma 2,

$$\left(T\left(ST\left(T\left(\prod_U M'\right)\right)\right), \left(\prod_U s'(x_1), \ldots, \prod_U s'(x_k)\right)\right) \Vdash T\left(ST_{\overline{x}}(\varphi)\right).$$

Then, by Lemma 1,

$$\left(ST\left(T\left(\prod_U M'\right)\right), s_{\overline{x}}^{(\prod_U s'(x_1), \ldots, \prod_U s'(x_k))}\right) \Vdash ST_{\overline{x}}(\varphi).$$

Thus, by Lemma 2,

$$\left(T\left(\prod_U M'\right), \left(\prod_U s'(x_1), \ldots, \prod_U s'(x_k)\right)\right) \Vdash \varphi.$$

and we have proved the claim. Hence, by Lemmas 8 and 10, there is a $\overline{X}$–compatible predicate bisimulation between the structures associated to $\left(T\left(\prod_U M\right), \prod_U s\right)$ and $\left(T\left(\prod_U M'\right), \prod_U s\right)$

and therefore, because of Lemma 2, between $\left(\prod_U M, \prod_U s\right)$ and $\left(\prod_U M', \prod_U s\right)$, that is, between $\prod_U (M,s)$ and $\prod_U (M',s')$. □

The following corollary with sentences in place of $\overline{X}$–formulas is an exact counterpart of the Keisler-Shelah isomorphism theorem [13, Theorem 6.1.15] in which isomorphisms are replaced by predicate bisimulations.

Corollary 2. *Assume that $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$ is countable and let (M,s) and (M',s') be two pointed structures. Then the following are equivalent:*

1. *(M,s) and (M',s') make true the same sentences of $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$;*
2. *there exists an ultrafilter U and a predicate bisimulation between $\prod_U (M,s)$ and $\prod_U (M',s')$.*

Proof. Every sentence $\varphi \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$ is equivalent to the sentence $\varphi \wedge (\bot \mathtt{c} \to \bot \mathtt{c})$ where $\mathtt{c}$ is an arbitrary constant of $\mathcal{C}$ (if $\mathcal{C}$ is empty we consider the set of constants $\mathcal{C} = \{\mathtt{c}\}$ instead). Thus, (M,s) and (M',s') make true the same sentences iff they make true the same $\mathcal{C}^+$–formulas, because the set of sentences with at least one constant is the set of $\mathcal{C}^+$–formulas. The result then follows by a direct application of Theorem 8. □

Theorem 9. *Assume that $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$ is countable, let K be a class of pointed structures and let $\overline{X}$ be a profile. Then, the following are equivalent:*

1. *K is definable by a set of $\overline{X}$–formulas of $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$;*
2. *K is closed under $\overline{X}$–compatible predicate bisimulations and ultraproducts, and $\overline{K}$ is closed under ultrapowers.*

Proof. The proof follows the same reasoning as the proof of Theorem 6. The notion of $\overline{X}$–compatibility is used in the direction from 1. to 2. As for the direction from 2. to 1., we consider the following set of pure predicate formulas with free variables or constants:

$$T \triangleq \left\{\varphi \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}) \mid \text{ for all } (M,s) \in K, (M,s) \models \varphi\right\}$$

Let (M_0,s_0) be any pointed structure such that $(M_0,s_0) \Vdash T$. We are going to prove that $(M_0,s_0) \in K$. Let us consider the following set of formulas:

$$\Sigma \triangleq \left\{\varphi(x_1,\dots,x_k) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x}) \mid (M_0,s_0) \models \varphi \text{ and } (x_1,\dots,x_k) \in \overline{X}\right\}$$

Σ is finitely satisfiable in K. Indeed, assume that the finite set $\{\psi_1,\dots,\psi_n\} \subseteq \Sigma$ is not satisfiable in K. Then, $\neg\psi_1 \vee \dots \vee \neg\psi_n$ is true on all pointed structures of K. So, $\neg\psi_1 \vee \dots \vee \neg\psi_n \in T$. However, it would be false on (M_0,s_0), which is impossible. But then, Proposition 7 shows that Σ is satisfiable in an ultraproduct $\prod_U (N_i,s_i)$ of pointed structures $(N_i,s_i) \in K$. Let us take $(N,s) = \prod_U (N_i,s_i)$. Then, $(N,s) \in K$ by closure of K under ultraproduct. Moreover, $(N,s) \models \Sigma$. So, for all $\varphi(x_1,\dots,x_k) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ such that $(x_1,\dots,x_k) \in \overline{X}$, $(M_0,s_0) \models \varphi$ implies $(N,s) \models \varphi$. Thus, by closure of $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ under Boolean negation, for all $\varphi(x_1,\dots,x_k) \in \mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ such that $(x_1,\dots,x_k) \in \overline{X}$, $(M_0,s_0) \models \varphi$ iff $(N,s) \models \varphi$. Thus, by Theorem 8, there exists an ultrafilter U over a non-empty set I and a $\overline{X}$–compatible predicate bisimulation between

$\prod_U(M_0, s_0)$ and $\prod_U(N, s)$. Now, $\prod_U(N, s) \in K$ because $(N, s) \in K$ and closure of K under ultraproduct. So, $\prod_U(M_0, s_0) \in K$ by closure of K under $\overline{X}$–compatible bisimulation. Finally, since $\overline{K}$ is closed under ultrapower, (M_0, s_0) must belong to K, since otherwise $\prod_U(M_0, s_0)$ would not be in K. This completes the proof of the theorem within brackets. □

Corollary 3. *Let K be a class of pointed structures. Then, the following are equivalent:*

1. *K is definable by a set of sentences of $\mathcal{L}^{\mathcal{P}}_{FOL}$;*
2. *K is closed under predicate bisimulations and ultraproducts, and $\overline{K}$ is closed under ultrapowers.*

Proof. Every set of sentences is equivalent to a set of sentences which all contain at least one constant: it suffices to replace each sentence φ of the set which does not contain a constant with the sentence $\varphi \wedge (\bot \mathsf{c} \rightarrow \bot \mathsf{c})$ (if $\mathcal{C}$ is empty, we consider the set of constants $\mathcal{C} = \{\mathsf{c}\}$ instead). Then, it suffices to apply Theorem 9 to the set of sentences with at least one constant, which is the same as the set of $\mathcal{C}^+$–formulas. □

Theorem 10. *Assume that $\mathcal{L}^{\mathcal{P}}_{FOL}$ is countable, let K be a class of pointed structures and let $\overline{x}$ be a tuple of variables or constants. Then, the following are equivalent:*

1. *K is definable by means of a single $\{\overline{x}\}$–formula $\varphi \in \mathcal{L}^{\mathcal{P}}_{FOL}$;*
2. *Both K and $\overline{K}$ are closed under $\{\overline{x}\}$–compatible predicate bisimulations and ultraproducts.*

Proof. It is similar to the proof of Theorem 7. The direction from 1. to 2. is easy. For the converse, we assume that K and $\overline{K}$ satisfy the stated closure conditions. Then, both are closed under ultraproducts, hence by Theorem 9, there are sets of $\{\overline{x}\}$–formulas of $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$ T_1 and T_2 defining K and $\overline{K}$ respectively. Obviously, their union is inconsistent in the sense that there is no pointed structure (M, s) such that $(M, s) \Vdash T_1 \cup T_2$. So then, by Proposition 7, there exist $\varphi_1, \ldots \varphi_n \in T_1$ and $\psi_1, \ldots, \psi_m \in T_2$ such that for all pointed structures (M, s), it is not the case that $(M, s) \Vdash \varphi_1 \wedge \ldots \wedge \varphi_n \wedge \psi_1 \wedge \ldots \wedge \psi_m$ $(*)$. To complete the proof, we show that K is in fact defined by the conjunction $\varphi_1 \wedge \ldots \wedge \varphi_n$. By definition, for any (M, s) in K we have $(M, s) \Vdash \varphi_1 \wedge \ldots \wedge \varphi_n$. Conversely, if $(M, s) \Vdash \varphi_1 \wedge \ldots \wedge \varphi_n$ then there must be $i \in [\![1; m]\!]$ such that $(M, s) \Vdash \psi_i$ does not hold. Indeed, otherwise, we would have that $(M, s) \Vdash \varphi_1 \wedge \ldots \wedge \varphi_n \wedge \psi_1 \wedge \ldots \wedge \psi_m$, which is impossible by $(*)$. Hence, it is not the case that $(M, s) \Vdash T_2$. Therefore, (M, s) does not belong to $\overline{K}$, whence (M, s) belongs to K. □

7.2 First–order Logic

The results of the previous section hold for pure predicate logic, that is first–order logic without functions. In this section, we are going to extend them to first–order logic (with functions).

Definition 41 (First–order bisimulation). A *first–order bisimulation* between two structures M_1 and M_2 is a pair of non–empty relations (Z, Z_0) between finite sequences of M_1 and M_2 of the same length such that for all $M, M' \in \{M_1, M_2\}$ with $M = (W, \mathcal{R})$ and $M' = (W', \mathcal{R}')$, all $n \in \mathbb{N}^*$, all $w_1, \ldots, w_n \in W$, all $w'_1, \ldots, w'_n \in W'$, all $\sigma \in \mathfrak{S}_n$, all functions $f_1, \ldots, f_n$ of M and corresponding functions $f'_1, \ldots, f'_n$ of M', all tuples $\overline{v}_1, \ldots, \overline{v}_n$ of W and $\overline{v'}_1, \ldots, \overline{v'}_n$ of W',

1. if $(w_1, \ldots, w_n) Z_0 (w'_1, \ldots, w'_n)$ then for all n–ary relations $R \in \mathcal{R}$ and $R' \in \mathcal{R}'$ associated to the same predicate R, if $R w_1 \ldots w_n$ then $R' w'_1 \ldots w'_n$;

2. if $(w_1, \ldots, w_n) Z_0 (w'_1, \ldots, w'_n)$ and $f_1(\overline{v'}_1) = w'_1, \ldots, f_n(\overline{v'}_n) = w'_n$ then there are finite sequences $\overline{v}_1, \ldots, \overline{v}_n$ of M such that $\overline{v}_1 Z_0 \overline{v'}_1, \ldots, \overline{v}_n Z_0 \overline{v'}_n$ and $f_1(\overline{v}_1) = w_1, \ldots, f_n(\overline{v}_n) = w_n$;

3. if $(w_1, \ldots, w_n) Z (w'_1, \ldots, w'_n)$ then $(w_1, \ldots, w_n) Z_0 (w'_1, \ldots, w'_n)$;

4. if $(w_1, \ldots, w_n) Z (w'_1, \ldots, w'_n)$ then for all $w \in M$ there is $w' \in M'$ such that $(w_1, \ldots, w_n, w) Z (w'_1, \ldots, w'_n, w')$;

5. if $(w_1, \ldots, w_n) Z (w'_1, \ldots, w'_n)$ then $(w_{\sigma(1)}, \ldots, w_{\sigma(n)}) Z (w'_{\sigma(1)}, \ldots, w'_{\sigma(n)})$;

6. if $(w_1, \ldots, w_n, w_{n+1}) Z (w'_1, \ldots, w'_n, w'_{n+1})$ then $(w_1, \ldots, w_n) Z (w'_1, \ldots, w'_n)$;

7. if $(w_1, \ldots, w_n) Z (w'_1, \ldots, w'_n)$ then $(w'_1, \ldots, w'_n) Z (w_1, \ldots, w_n)$;

8. for all $\mathsf{c}_1, \ldots, \mathsf{c}_n \in \mathcal{C}$ it holds that $(s(\mathsf{c}_1), \ldots, s(\mathsf{c}_n)) Z (s'(\mathsf{c}_1), \ldots, s'(\mathsf{c}_n))$.

If $\overline{X}$ is a profile, an *$\overline{X}$–compatible first–order bisimulation* is a first–order bisimulation where the condition 8. is replaced by the following condition:

8'. for all $(x_1, \ldots, x_n) \in \overline{X}$, it holds that $(s(x_1), \ldots, s(x_n)) \, Z \, \big(s'(x_1), \ldots, s'(x_n)\big)$. $\dashv$

Lemma 12. *Let $\mathsf{C}^{\mathcal{P},\mathcal{F}}$ be the set of first–order atomic connectives and let $M_1 = (W_1, \mathcal{R}_1)$ and $M_2 = (W_2, \mathcal{R}_2)$ be two ω–saturated $\mathsf{C}^{\mathcal{P},\mathcal{F}}$–models. We define the binary relation $Z_0 \subseteq (W_1 \times W_2) \cup (W_2 \times W_1)$ as follows: if $\{M, M'\} = \{M_1, M_2\}$ then for all $\overline{w} \in \overline{w}(M, \mathsf{C})$, all $\overline{w'} \in \overline{w}(M', \mathsf{C})$, $\overline{w} Z_0 \overline{w'}$ iff $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}^{\mathcal{F}}} (M', \overline{w'})$. We also define the binary relation Z between finite sequences of M_1 and M_2 of the same length by $\overline{w} Z \overline{w'}$ iff $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}^{\mathcal{P},\mathcal{F}}} (M', \overline{w'})$. Then, the pair of binary relations (Z, Z_0) is a first–order bisimulation between M_1 and M_2.*

Proof. Applying Lemma 8 to $\mathsf{C}^{\mathcal{F}}$, from the definition of a $\mathsf{C}^{\mathcal{F}}$–bisimulation, we obtain conditions 1. – 2. of a first–order bisimulation. Likewise, by Lemmas 8 and 10, Z is a predicate bisimulation. Hence, conditions 4. – 7. of a first–order bisimulation are fulfilled. Moreover, we also have the condition that if $(w_1, \ldots, w_n) Z (w'_1, \ldots, w'_n)$ then for all n–ary relations $R_{p_\chi} \in \mathcal{R}$ and $R'_{p_\chi} \in \mathcal{R}'$ associated to the same propositional letter p_χ, if $R_{p_\chi} w_1 \ldots w_n$ then $R'_{p_\chi} w'_1 \ldots w'_n$. This last condition reformulates as if $\overline{w} Z \overline{w'}$ then $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}^{\mathcal{F}}} (M', \overline{w'})$. That is, if $\overline{w} Z \overline{w'}$ then $\overline{w} Z_0 \overline{w'}$, which is condition 3. Hence, (Z, Z_0) is a first–order bisimulation between M_1 and M_2. □

Theorem 11. *Theorems 8, 9 and 10 and Corollaries 2 and 3 hold if we replace at the same time the term "predicate bisimulation" with "first–order bisimulation" and the languages $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}$ and $\mathcal{L}^{\mathcal{P}}_{\mathsf{FOL}}(\overline{x})$ with the languages with function symbols $\mathcal{L}^{\mathcal{P}\mathcal{F}}_{\mathsf{FOL}}$ and $\mathcal{L}^{\mathcal{P}\mathcal{F}}_{\mathsf{FOL}}(\overline{x})$ respectively.*

Proof. It is the same as the proofs of Theorems 8, 9 and 10 and Corollaries 2 and 3. Only the 'translation' Lemmas 1 and 2 have to be replaced by the 'translation' Lemmas 3 and 4 and the Lemmas 8 and 10 have to be replaced by Lemma 12. □

The invariance notions of pure predicate logic and first–order logic *with equality* are also predicate bisimulation and first–order bisimulation because the equality predicate is considered on a par with the other predicates, it is just like any other predicate. In that case, the corresponding condition 1. of Definitions 39 and 41 for the equality predicate is the following: if $(w, w) Z_0 (w'_1, w'_2)$ then $w'_1 = w'_2$. With the other conditions, this last condition entails that if $(w, \ldots, w) Z_0 (w'_1, \ldots, w'_n)$ then $w'_1 = \ldots = w'_n$.

Proposition 11. *If there is a first–order bisimulation between two structures then there is a predicate bisimulation and a partial isomorphism between them. Any two finite or countable structures such that there is a predicate or (if they contain functions) a first–order bisimulation between them are isomorphic.*

Proof. The first part follows from the fact that conditions 1.-3. of first-order bisimulations imply conditions 1. of predicate bisimulations and partial isomorphisms (see Definition 1): one can easily prove by induction on propositional letteric formulas φ that if $(w_1, \ldots, w_n)Z_0(v_1, \ldots, v_n)$ then (M_1, s_1) and (M_2, s_2) make true the same propositional letteric formulas $\varphi(x_1, \ldots, x_n) \in \mathcal{L}^{\mathcal{PF}}_{\mathsf{FOL}}$, where (M_1, s_1) and (M_2, s_2) are defined like in Definition 1. The other conditions of a partial isomorphism are fulfilled by the remaining conditions of a predicate or first–order bisimulation. So, because we proved that a predicate bisimulation is also a partial isomorphism (if there is no function), the second part of the proposition follows from [13, Proposition 2.4.4]. □

8 The Role of Boolean Negation

Many non–classical logics come with Boolean negation, often denoted $\neg$. Yet, Boolean negation is not a (atomic) connective considered as primitive in this article (see Section 3.4) and we cannot directly apply the results of this article to a non–classical logic if it contains the Boolean negation. In order to apply them, the set of connectives of a given non–classical logic has first to be redefined so that Boolean negation does not occur as a primitive connective anymore. Roughly, there are two ways we can do without it: either we consider the negative normal forms of formulas or we consider the combination of the material implication $\rightarrow$ with the falsum constant $\bot$ since $\neg\varphi = \varphi \rightarrow \bot$. As for the negative normal form, that is how we proceeded for modal logic and as for the combination of material implication and falsum that is how we proceeded for first–order logic. The standard set of connectives for modal logic is $\mathsf{C} = \{p, \neg, \wedge, \Box\}$. To apply our results, we have first transformed this set of connectives into the equivalent set of atomic connectives $\{p, \neg p, \wedge, \vee, \Box, \Diamond\}$ by removing the Boolean negation $\neg$ from C and adding the duals of the connectives p, $\wedge$ and $\Box$, that is $\neg p$, $\vee$ and $\Diamond$ respectively. We could have proceeded differently and considered the set of connectives $\{p, \bot, \wedge, \rightarrow, \Box\}$. In doing so, we would have obtained the same notion of invariance in both cases. This transformation can be applied to any set of connectives containing the Boolean negation $\neg$ in order to remove it. Alternatively, one can introduce the material implication $\rightarrow$ as a primitive connective together with the falsum constant $\bot$. Likewise for first–order logic. We could have considered the set of connectives $\{\mathsf{R}, \neg\mathsf{R}, \wedge, \vee, \forall, \exists\}$ and in that case we would have obtained an equivalent definition of predicate or first–order bisimulations, detailed in Proposition 9. Basically, the inclusion of the Boolean negation in a logic, in one way or another, entails that its associated notion of bisimulation is a symmetric relation. This is the case for example for modal logic but not for the Lambek calculus since the latter cannot express Boolean negation. That is why we have a notion of "directed" bisimulation for the Lambek calculus. We can formalize these considerations by the following definition and proposition.

Definition 42. Let C be a set of molecular connectives. We say that C is *closed under Boolean negation* when for all $\varphi \in \mathcal{L}_{\mathsf{C}}$ there is a formula, denoted $-\varphi$, which belongs to $\mathcal{L}_{\mathsf{C}}$ such that for all pointed C–model $(M, \overline{w})$, it holds that $(M, \overline{w}) \Vdash -\varphi$ iff $(M, \overline{w}) \Vdash \varphi$ does not hold. ⊣

Proposition 12. *Let C be a set of atomic connectives. If C is closed under Boolean negation then Theorems 4, 5, 6 and 7 hold for C–bisimulations which are symmetric.*

Proof. If C is closed under Boolean negation then for any two pointed C–models $(M, \overline{w})$ and $(M', \overline{w'})$, if $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}} (M', \overline{w'})$ then $(M', \overline{w'}) \rightsquigarrow_{\mathsf{C}} (M, \overline{w})$. Indeed, assuming that $(M, \overline{w}) \rightsquigarrow_{\mathsf{C}}$

$(M',\overline{w'})$, we have that for all $\varphi \in \mathcal{L}_{\mathsf{C}}$, if not $(M,\overline{w}) \Vdash \varphi$ then $(M,\overline{w}) \Vdash -\varphi$ with $-\varphi \in \mathcal{L}_{\mathsf{C}}$, so $(M',\overline{w'}) \Vdash -\varphi$ by assumption and thus not $(M',\overline{w'}) \Vdash \varphi$. That is, for all $\varphi \in \mathcal{L}_{\mathsf{C}}$, if $(M',\overline{w'}) \Vdash \varphi$ then $(M,\overline{w}) \Vdash \varphi$, *i.e.* $(M',\overline{w'}) \rightsquigarrow_{\mathsf{C}} (M,\overline{w})$. Hence, the C–bisimulation defined in Lemma 8 is in fact symmetric. Since it is this C–bisimulation that we use in the proofs of Theorems 4, 5, 6 and 7 via this lemma, we can assume that the C–bisimulations mentionned in these theorems are in fact symmetric. □

9 Related Work

9.1 Comparison with Olkhovikov's Work

The closest work to ours is by Olkhovikov [32, 33, 34] who investigates generalizations of the van Benthem characerization theorm. The publications [32] and [33] deal in particular with (modal) intuitionistic (predicate) logic. Following our methodology, we have rediscovered Olkhovikov's definitions in Examples 13, 8 and 14. Yet, it remained to prove a result, namely the following, whose proof is interesting for its own sake:

Fact 1. *Let $M = (W, \{R, P\})$ and $M' = (W', \{R', P'\})$ be two ω–saturated intuitionistic models and let Z be the maximal C–bisimulation between M and M' for set inclusion (C is defined in Example 13). Then, the following two conditions are equivalent:*

1. *Condition $(**)$ of Example 13: for all $v \in W$ and all $w', v', u' \in W'$, if vZv' and $R'v'w'$ and $R'u'w'$ then there are $u, w \in W$ such that $u'Zu$, wZw' and Rvw and Ruw;*
2. *Condition "step" of [33, Definition 1]: for all $v \in W$ and all $w', v' \in W'$, if vZv' and $R'v'w'$ then there is $w \in W$ such that wZw' and $w'Zw$ and Rvw.*

Proof. Because we deal with ω–saturated models, the *maximal* (for set inclusion) C–bisimulation between M and M' is the relation $\rightsquigarrow_{\mathsf{C}}$ of Definition 32 by Propositions 2 and 4. The key observation to make is that if Rvw then $(M,v) \rightsquigarrow_{\mathsf{C}} (M,w)$, and likewise if $R'v'w'$, which is a classical result of intuitionistic logic. Now, for the direction 1. to 2., assume that vZv' and $R'v'w'$. Then, because R' is reflexive, we have that $R'w'w'$. So, applying condition $(**)$, we obtain that there are $u, w \in W$ such that $w'Zu$, wZw' and Rvw and Ruw (u' is just replaced by w'). Now, because Ruw, we have by the key observation that $(M,u) \rightsquigarrow_{\mathsf{C}} (M,w)$. Therefore, we also have by transitivity, because $w'Zu$ and thus $(M',w') \rightsquigarrow_{\mathsf{C}} (M,u)$, that $(M',w') \rightsquigarrow_{\mathsf{C}} (M,w)$. That is, $w'Zw$. So, we have proved that there is $w \in W$ such that wZw' and $w'Zw$ and Rvw. Conversely, for the direction 2. to 1., assume that vZv' and $R'v'w'$ and $R'u'w'$. So, applying condition "step", we obtain that there is $w \in W$ such that wZw' and $w'Zw$ and Rvw. So, we have wZw' and Rvw; it remains to find u. We take u to be w. Then, we do have that Ruw because R is reflexive. Moreover, we have that $(M',u') \rightsquigarrow_{\mathsf{C}} (M',w')$ and $w'Zu$. Thus, $(M',u') \rightsquigarrow_{\mathsf{C}} (M',w')$ and $(M',w') \rightsquigarrow_{\mathsf{C}} (M,u)$. Then, by transitivity, $(M',u') \rightsquigarrow_{\mathsf{C}} (M,u)$. That is, because we deal with ω–saturated models, $u'Zu$. □

In [33], Olkhovikov also provides a generic van Benthem style characterization theorem for a number of logics defined by specific kinds of connectives. He introduces a normal form for connectives in terms of formulas of FOL that he calls *k–ary guarded x–connectives*. A *k–ary guarded x–connective of degree 0* $\mu = \psi(\mathsf{P}_1(x), \ldots, \mathsf{P}_k(x))$ is a Boolean combination of the unary predicates $\mathsf{P}_1(x), \ldots, \mathsf{P}_k(x)$. A *$k$–ary $\forall$–guarded x_1–connective of degree* $n+1$ is a formula of the form $\forall x_2 \ldots x_{m+1} \left(\bigwedge_{i=1}^{m} S_i(x_i, x_{i+1}) \to \mu^- \right)$ where $S_1, \ldots, S_m$ are binary predicates and μ^- is a k–ary guarded x_{m+1}–connective of degree n (provided that formula is not equivalent to k–ary guarded x_1–connective of a smaller degree). *k–ary $\exists$–guarded x_1–connectives* are defined similarly. If one sets $\mathsf{R}x_1 \dots x_m x_{m+1}$ for $\bigwedge_{i=1}^{m} S_i(x_i, x_{i+1})$ then $\forall x_2 \dots x_{m+1} \left(\bigwedge_{i=1}^{m} S_i(x_i, x_{i+1}) \to \mu^- \right)$ can be viewed as the first–order formula with free variable x_1 defining a atomic connective: $\forall x_2 \dots x_{m+1} \left(\bot(x_2) \lor \dots \lor \bot(x_m) \lor \mu^-(x_{m+1}) \lor -\mathsf{R}x_1 \dots x_m x_{m+1} \right)$. Hence, guarded connectives of degree not exceeding 1 are captured by specific plain atomic connectives. It is unclear whether Olkhovikov's regular connectives of degree 2 are also captured by uniform connectives.

In any case, our results strictly extend those of Olkhovikov because we are able to provide a van Benthem characterization for connectives defined by formulas of FOL with *multiple* free variables. It is this feature that plays a key role for the cases of temporal logic and classical logic. In that respect, the results we obtain for FOL in Section 7 are not derivable by any means from Olkhovikov's results and methods. Moreover, our approach is more general than Olkhovikov's since we obtain in a generic fashion not only a van Benthem style characterization for the logics considered but also a number of results regarding the definability of classes of pointed models (Theorems 6 and 7). It is this second part which allowed us to recover Keisler theorems for FOL. It is also made more clear and explicit than in Olkhovikov's publication how the suitable notions of bisimulation are defined from logics given by their set of connectives. Finally, we showed in Examples 8 and 14 how his results about (modal) intuitionistic logic [32, 33] can be recovered in our setting as specific instances of our general results.

9.2 Other Related Work

Van Benthem characerization theorems have been proved for many non–classical logics, such as (modal) intuitionistic logic [33], intuitionistic predicate logic [32], temporal logic [29], sabotage modal logic [8], graded modal logic [14], fuzzy modal logic [41], coalgebraic modal logics [39], neighbourhoud semantics of modal logic [25], the modal mu–calculus [26], hybrid logic [2]. We showed that our generic Theorem 4 subsumes some of them [33, 32, 29]. However, some others are not in the scope of our theorem because the correspondence language to which they refer extends first–order logic. For example, the van Benthem style characterization theorem for coalgebric modal logic [39] is with respect to coalgebric predicate logic, the one for fuzzy modal logic [41] is with respect to first–order fuzzy predicate logic and the one for the modal mu–calculus [26] is with respect to monadic second–order logic.

Results similar to Theorems 9 and 10 (and 11) already exist in the literature of model theory, namely [13, Theorem 4.1.12] and Keisler theorem [13, Corollary 6.1.16]. The first difference is that we only deal here with countable langauges. The second difference consists in replacing the closure under predicate bisimulation by a closure under elementary equivalence for [13, Theorem 4.1.12] or by a closure under isomorphism for [13, Corollary 6.1.16]. The third difference consists in considering arbitrary sets of formulas and not only the set of all sentences.

Our Corollaries 2 and 3 are exact counterparts of the Keisler-Shelah isomorphism theorem [13, Theorem 6.1.15] and the Keisler Theorem [13, Corollary 6.1.16], the only difference being that the notion of isomorphism is replaced by our notions of predicate bisimulation and first–order bisimulation. Similar results have also been proved for modal logic [12, Theorems 2.75; 2.76] and temporal logic [29]. The difference once again lies in the fact that isomorphisms are replaced by modal bisimulation and temporal bisimulation. A result similar to our Corollary 3 was already proved by van Benthem & Doets [40]. In their result, predicate bisimulations and isomorphisms are replaced by partial isomorphisms.

10 Conclusion

We have introduced a generic method that allows us to find out an appropriate notion of bisimulation for an arbitrary logic whose truth conditions are defined by first–order formulas. This bisimulation notion comes as well with a number of associated model–theoretical results of the logic considered (Theorems 4, 5, 6 and 7). We have applied this method to modal logic, (modal) intuitionistic logic, temporal logic, the Lambek calculus and first–order logic. In doing so, we have rediscovered the definitions of bisimulation of the literature for modal logic, the Lambek calculus, (modal) intuitionistic logic and temporal logic and discovered new invariance notions for first–order logic. We have also rediscovered some of the associated model–theoretical results of modal logic, temporal logic, (modal) intuitionistic logic as well as novel results regarding the Lambek calculus and first–order logic (in particular Theorems 8, 9, 10, 11 and Corollaries 2 and 3). As for first–order logic, the main novelty with respect to the literature lies in the fact that isomorphisms are replaced by predicate bisimulations and first–order bisimulations and that we generalize existing results and consider arbitrary sets of first–order formulas, and not only the set of all sentences. On countable structures, the notions of isomorphism, partial isomorphism, predicate bisimulation and first–order bisimulation coincide. We expect that our notions of predicate and first–order bisimulation differ from isomorphisms on uncountable structures. By the (upward) Löwenheim–Skolem theorem and because our results hold for countable languages, they open new perspectives for the study of uncountable structures such as the non–standard models of arithmetic.

These generalizations and new versions of existing theorems are surprising, but they confirm, together with the rediscovery of numerous existing results, the soundness and generic character of our overall approach. Our method is applicable to a much wider class of logics than the examples of logics that we have dealt with in the article, in fact an infinite number of logics. Other examples include obviously all the atomic logics listed in Figures 2 and 3 as well as all gaggle logics [6, 7], some of them having been already well-studied in that respect. We do not claim to have introduced brand new notions of invariance for first-order logic, they are in fact natural and intuitive variants of the usual notions of (partial) isomorphism. Instead, we claim to have introduced a generic notion of C-bisimulation which is somehow 'deeper' and more basic than the usual notion of bisimulation for modal logic or even the usual notion of (partial) isomorphism for first-order logic. Indeed, all invariance notions introduced in the literature (including those) can all be seen as instances of our general notions of C-bisimulation of Definitions 29 and 31.

From a pragmatic point of view, if we want to apply our generic results to an arbitrary logic, the difficult part is to translate it into an atomic logic. Yet this is really rewarding, since once this translation has been realized we obtain automatically its appropriate notion of bisimulation as well as the associated model–theoretical results of Theorems 4, 5, 6 and 7. This translation into molecular logics is always possible for logics for which the truth conditions of connectives are expressible in terms of formulas of first–order logic. This is guaranteed by our Theorem 3. In that case, we can always find out a notion of invariance which preserves the truth of formulas of the logic considered. However, it might not completely characterize the logic as a fragment of first–order logic in the sense that the van Benthem characterization theorem (our Theorem 4) might not hold anymore, because the translation into atomic logic might not yield *uniform* atomic connectives. We believe that this is not a defect of our approach but something inherent to the structure and nature of non–classical logics. This said, our characterization theorem might be extended to a larger class of atomic logics than those defined in terms of uniform atomic connectives.

This article contributes to the systematic exploration of non–classical logics and to the development of a generic model theory for non–classical logics. It illustrates in particular the central role played by atomic and molecular logics. They behave as 'paradigmatic logics'. We focused here on the model theory of atomic logics, their proof theory is dealt with in [6, 7], also in a systematic and generic manner.[1] The advantage of our overall approach is that it provides a uniform and generic methodology to explore and classify non–classical logics.

We have restricted our investigations in this article to non–classical logics which are fragments of first–order logic. Yet, the same methodology could be applied if we replaced first–order logic with second–order logic and in fact with any higher–order logic. Indeed, we could for example consider logics whose connectives are defined by formulas of *second–order* logic instead of first–order logic. Many of our results may still hold or may need to be adapted to that extended setting. However, we may also lose some results, like our van Benthem style characterization theorem (Theorem 4) or the Keisler style theorems (Theorems 6 and 7). Indeed, their proofs rely greatly on the compactness of first–order logic, which is lost if we move from first–order logic to second–order logic. In fact the automata–based proof techniques used to prove the van Benthem characterization theorem for the modal mu–calculus for example, which is shown to be the fragment of monadic *second–order* logic invariant under bisimulation, are quite different [26]. In any case, we believe that our overall approach and methodology is the right track to follow if we want to explore and study non–classical logics in a systematic and comprehensive way, in particular because it relies on a class and hierarchy of logics which are naturally well defined and articulated and which have already been well studied.

[1] [7] differs very slightly from [6]. It essentially corrects minor mistakes and typos and proves that the rule of associativity is derivable in $\mathsf{GGL_C}$.

References

[1] Gerard Allwein and J. Michael Dunn. Kripke models for linear logic. *The Journal of Symbolic Logic*, 58(2):514–545, June 1993.

[2] Carlos Areces, Patrick Blackburn, and Maarten Marx. Hybrid logics: Characterization, interpolation and complexity. *J. Symb. Log.*, 66(3):977–1010, 2001.

[3] Carlos Areces, Raul Fervari, and Guillaume Hoffmann. Relation-changing modal operators. *Logic Journal of the IGPL*, 23(4):601–627, 2015.

[4] Peter Arndt, Rodrigo de Alvarenga Freire, Odilon Otavio Luciano, and Hugo Luiz Mariano. A global glance on categories in logic. *Logica Universalis*, 1(1):3–39, 2007.

[5] Guillaume Aucher. Displaying Updates in Logic. *Journal of Logic and Computation*, 26(6):1865–1912, March 2016.

[6] Guillaume Aucher. Towards universal logic: gaggle logics. *Journal of Applied Logics – IfCoLog Journal of Logics and their Applications*, 7(6):875–945, 2020.

[7] Guillaume Aucher. *Selected Topics from Contemporary Logics*, chapter Towards Universal Logic: Gaggle Logics, pages 5–73. Landscapes in Logic. College Publications, October 2021.

[8] Guillaume Aucher, Johan van Benthem, and Davide Grossi. Modal logics of sabotage revisited. *J. Log. Comput.*, 28(2):269–303, 2018.

[9] Jean-Yves Béziau. *Logica Universalis*, chapter From Consequence Operator to Universal Logic: A Survey of General Abstract Logic. Birkhäuser Basel, 2007.

[10] Jean-Yves Béziau. Editorial: Introduction to the universal logic corner. *J. Log. Comput.*, 19(6):1111, 2009.

[11] Katalin Bimbó and J. Michael Dunn. *Generalized Galois Logics: Relational Semantics of Nonclassical Logical Calculi*. Number 188. Center for the Study of Language and Information, 2008.

[12] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Computer Science*. Cambridge University Press, 2001.

[13] C. C. Chang and H. J. Keisler. *Model Theory*. Studies in Logic and the Foundations of Mathematics. Elsevier, 1998.

[14] Maarten de Rijke. A note on graded modal logic. *Studia Logica*, 64(2):271–283, 2000.

[15] J Michael Dunn. Gaggle theory: an abstraction of galois connections and residuation, with applications to negation, implication, and various logical operators. In *European Workshop on Logics in Artificial Intelligence*, pages 31–51. Springer Berlin Heidelberg, 1990.

[16] J. Michael Dunn. *Philosophy of Language and Logic*, volume 7 of *Philosophical Perspectives*, chapter Perp and star: Two treatments of negation, pages 331–357. Ridgeview Publishing Company, Atascaredo, California, USA, 1993.

[17] J. Michael Dunn and Gary M. Hardegree. *Algebraic Methods in Philosophical Logic*. Number 41 in Oxford Logic Guides. Clarendon Press: Oxford, 2001.

[18] J. Michael Dunn and Chunlai Zhou. Negation in the context of gaggle theory. *Studia Logica*, 80(2-3):235–264, 2005.

[19] Herbert B. Enderton. *An Introduction to Mathematical Logic*. Academic Press, 2001.

[20] Raul Fervari, Francisco Trucco, and Beta Ziliani. Mechanizing bisimulation theorems for relation-changing logics in coq. In Luís Soares Barbosa and Alexandru Baltag, editors, *Dynamic Logic. New Trends and Applications, DaLí 2019*, volume 12005 of *Lecture Notes in Computer Science*, pages 3–18. Springer, 2019.

[21] Marta García-Matos and Jouko Väänänen. Abstract model theory as a framework for universal logic. In Jean-Yves Béziau, editor, *Logica Universalis*, pages 19–33, Basel, 2007. Birkhäuser Basel.

[22] Robert Goldblatt. Semantic analysis of orthologic. *Journal of Philosophical Logic*, 3:19–35, 1974.

[23] Rajeev Goré. Substructural logics on display. *Logic Journal of IGPL*, 6(3):451–504, 1998.

[24] Viktor Grishin. On a generalization of the Ajdukiewicz-Lambek system. In A. I. Mikhailov, editor, *Studies in Nonclassical Logics and Formal Systems*, pages 315–334. Nauka, Moscow, 1983.

[25] Helle Hvid Hansen, Clemens Kupke, and Eric Pacuit. Neighbourhood structures: Bisimilarity and basic model theory. *Log. Methods Comput. Sci.*, 5(2), 2009.

[26] David Janin and Igor Walukiewicz. On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic. In Ugo Montanari and Vladimiro Sassone, editors, *CONCUR '96, Concurrency Theory, 7th International Conference, Pisa, Italy, August 26-29, 1996, Proceedings*, volume 1119 of *Lecture Notes in Computer Science*, pages 263–277. Springer, 1996.

[27] Saul A. Kripke. Semantical analysis of modal logic, i: Normal propositional calculi. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 8:113–116, 1963.

[28] Saul A. Kripke. *Formal Systems and Recursive Functions*, chapter Semantical Analysis of Intuitionistic Logic, I, pages 92–130. North Holland, Amsterdam, 1965.

[29] Natasha Kurtonina and Maarten de Rijke. Bisimulations for temporal logic. *Journal of Logic, Language and Information*, 6(4):403–425, 1997.

[30] Joachim Lambek. The mathematics of sentence structure. *American mathematical monthly*, 65:154–170, 1958.

[31] Michael Moortgat. Symmetries in natural language syntax and semantics: the lambek-grishin calculus. In *Logic, Language, Information and Computation*, pages 264–284. Springer, 2007.

[32] Grigory K. Olkhovikov. Model-theoretic characterization of intuitionistic predicate formulas. *J. Log. Comput.*, 24(4):809–829, 2014.

[33] Grigory K. Olkhovikov. On expressive power of basic modal intuitionistic logic as a fragment of classical FOL. *J. Appl. Log.*, 21:57–90, 2017.

[34] Grigory K. Olkhovikov. On generalized van Benthem-type characterizations. *Ann. Pure Appl. Log.*, 168(9):1643–1691, 2017.

[35] Graham Priest. *An introduction to non-classical logic*. Cambridge University Press, 2011.

[36] Arthur Prior. *Past, Present and Future*. Oxford: Clarendon Press, 1967.

[37] Greg Restall. *An Introduction to Substructural Logics*. Routledge, 2000.

[38] Joseph J. Rotman. *An Introduction to the Theory of Groups*, volume 148 of *Graduate texts in mathematics*. Springer, 1995.

[39] Lutz Schröder, Dirk Pattinson, and Tadeusz Litak. A van Benthem/Rosen theorem for coalgebraic predicate logic. *J. Log. Comput.*, 27(3):749–773, 2017.

[40] Johan van Benthem and Kees Doets. Higher-order logic. In Dov Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 1, pages 275–329. Reidel Publishing Company, 1983.

[41] Paul Wild, Lutz Schröder, Dirk Pattinson, and Barbara König. A van Benthem theorem for fuzzy modal logic. In Anuj Dawar and Erich Grädel, editors, *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, pages 909–918. ACM, 2018.