Satisfiability calculus: the semantic counterpart of a proof calculus in general logics

Carlos G. Lopez Pombo[1,3], Pablo F. Castro[2,3], Nazareno M. Aguirre[2,3], and Thomas S.E. Maibaum[4]⋆

[1] Departmento de Computación, FCEyN, Universidad de Buenos Aires
clpombo@dc.uba.ar
[2] Departmento de Computación, FCEFQyN, Universidad Nacional de Río Cuarto
{pcastro,naguirre}@dc.exa.unrc.edu.ar
[3] CONICET
[4] Department of Computing & Software, McMaster University
tom@maibaum.org

Abstract. Since its introduction by Goguen and Burstall in 1984, the theory of Institutions has been one of the most widely accepted formalizations of abstract model theory. This work was extended by a number of researchers, José Meseguer among them, who presented *General Logics*; an abstract framework that complements the model theoretical view of Institutions by defining the categorical structures that provide a proof theory for any given logic. In this paper we intend to complete this picture by providing the notion of *Satisfiability Calculus*, which might be thought of as the semantical counterpart of the notion of proof calculus, that provides the formal foundations for those proof systems that use model construction techniques to prove or disprove a given formula, thus "implementing" the satisfiability relation of an institution.

1 Introduction

The theory of institutions, presented by Goguen and Burstall in [1], provides a formal, and generic, definition of what a logical system is from a model theoretical point of view. This work evolved in many directions: in [2], Meseguer complemented the theory of institutions by providing a categorical characterization for the notions of entailment system, also called π-institutions by other authors in [3], and the corresponding notion of proof calculi; in [4, 5] Goguen and Burstall, and Tarlecki extensively investigated the ways in which institutions can be related; in [6], Sannella and Tarlecki studied how specifications in a logical system can be structured [6]; in [7], Tarlecki presented an abstract theory of software specification and development; in [8] and [9], Mossakowski and Tarlecki, and Diaconescu respectively, proposed its use as foundation for a heterogeneous environment for software specification. It should be noted that Institutions have

⋆ The author gratefully acknowledge the support of the National Science and Engineering Research Council of Canada and McMaster University.

also been used as a very general version of abstract model theory [10], offering a suitable formal framework for addressing heterogeneity in specifications [11, 12], including applications to UML [13] and other languages related to computer science and software engineering.

The basic idea underlying Meseguer's work is the extension of entailment systems with a categorical construction expressible enough to capture the notion of proof in an abstract way. In Meseguer's words:

> A reasonable objection to the above definition of logic[5] is that *it abstracts away* the structure of proofs, since we know only that a set Γ of sentences entails another sentence φ, but no information is given about the internal structure of such a $\Gamma \vdash \varphi$ entailment. This observation, while entirely correct, may be a virtue rather than a defect, because the entailment relation is precisely what remains *invariant* under many equivalent proof calculi that can be used for a logic.

The previous remarks show that, intuitively, the notion of proof calculus (to be presented later on) provides an implementation of the entailment relation of a logic. Before Meseguer's work, there was an imbalance in the definition of a logic (in the context of institution theory) by not taking into account its deductive aspects. Meseguer concentrates exclusively on the proof theoretical aspects, of a logic, providing not only the definition of entailment system, but also complementing it with the notion of proof calculus in order to obtain what he calls a logical system. We believe he moved that imbalance in favour of models towards the syntactic aspect, ignoring the fact that the same lack of operational view he observes in the definition of entailment system now appears with respect to the notion of satisfiability (i.e., the satisfaction relation of an institution). This observation was motivated by the fact that several tools in computer science rely on model construction, either for proving properties, as with model-checkers, or for finding counterexamples, as with tableaux techniques. These techniques constitute an important stream of research in logic; in particular, these methods play an important role in automated software validation and verification.

The beginnings of these kinds of logical systems can be traced back to the works of Beth [14, 15], Herbrand [16] and Gentzen [17]; Beth's ideas were used by Smullyan to formulate the tableau method for first-order predicate logic [18]. Herbrandt's and Gentzen's work inspired the formulation of resolution systems presented by Robinson [19]. Methods like those based on resolution and tableaux are strongly related to the semantics of a logic; therefore, we can often use them to guide the construction of models; this is not possible in *pure* deductive methods, such as natural deduction or Hilbert systems, as formalized by Meseguer. Our goal is to provide an abstract characterization of this class of semantics based tools for logical systems. This is accomplished by introducing a categorical characterization of the notion of satisfiability calculus which embraces logical tools such as tableaux, resolution, Gentzen style sequents, etc. As we mentioned above,

[5] **Authors note**: He refers to the definition of logic as a structure that is constituted by an entailment system plus an institution, see Def. 6.

it can be thought of as a formalization of a semantic counterpart of Meseguer's proof calculus. We also explore the concept of mappings between satisfiability calculi and the relation between proof calculi and satisfiability calculi.

The paper is organized as follows. In Sec. 2 we present the definitions and results we will use throughout this paper. In Sec. 3 we present a categorical formalization of satisfiability calculus, prove relevant results underpinning the definitions and present examples in enough detail to illustrate the ideas. Finally in Sec. 4 we draw some conclusions and describe further lines of research.

2 Preliminaries

From now on we assume the reader has a nodding acquaintance with basic concepts from category theory. Basic definitions can be found in [20, 21]. Below we present the definitions and results we will use throughout the rest of the paper in order to define what we call satisfiability calculus. Most of these definitions were taken from [2] in order to preserve the terminology and notations chosen by Meseguer.

An *Institution* formalizes the model theory of a logic by making use of the relationships existing between signatures, the relation between the sets of formulae of two related signatures, the relation between (a) two models of the same signature and (b) the classes of models of two related signatures, and the relations between the semantic consequence relations of two related signatures. Each of these aspects is reflected by introducing the category of signatures and functors going from this category to the category Set (for the case of sets of sentences) and Cat (for the case of categories of models of a given signature).

Definition 1. [Institution]
An institution *is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *satisfying the following conditions:*

- Sign *is a category of signatures,*
- $\mathbf{Sen} : \mathsf{Sign} \to \mathsf{Set}$ *is a functor. Let* $\Sigma \in |\mathsf{Sign}|$, *then* $\mathbf{Sen}(\Sigma)$ *returns the set of* Σ*-sentences,*
- $\mathbf{Mod} : \mathsf{Sign}^{\mathsf{op}} \to \mathsf{Cat}$ *is a functor. Let* $\Sigma \in |\mathsf{Sign}|$, *then* $\mathbf{Mod}(\Sigma)$ *returns the category of* Σ*-models,*
- $\{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}$, *where* $\models^{\Sigma} \subseteq |\mathbf{Mod}(\Sigma)| \times \mathbf{Sen}(\Sigma)$, *is a family of binary relations,*

and for any signature morphism $\sigma : \Sigma \to \Sigma'$, Σ*-sentence* $\phi \in \mathbf{Sen}(\Sigma)$ *and* Σ'*-model* $\mathcal{M}' \in |\mathbf{Mod}(\Sigma)|$, *the following* $\models$*-invariance condition holds:*

$$\mathcal{M}' \models^{\Sigma'} \mathbf{Sen}(\sigma)(\phi) \quad \textit{iff} \quad \mathbf{Mod}(\sigma^{\mathsf{op}})(\mathcal{M}') \models^{\Sigma} \phi \ .$$

Let $\Sigma \in |\mathsf{Sign}|$ and $\Gamma \subseteq \mathbf{Sen}(\Sigma)$, then we define the functor $\mathbf{Mod}(\Sigma, \Gamma)$ as the full subcategory of $\mathbf{Mod}(\Sigma)$ determined by those models $\mathcal{M} \in |\mathbf{Mod}(\Sigma)|$ such that for all $\gamma \in \Gamma$, $\mathcal{M} \models^{\Sigma} \gamma$. In addition, it is possible to define a relation

$\models^{\Sigma}$ between sets of formulae and formulae in the following way: let $\alpha \in \mathbf{Sen}(\Sigma)$, then:

$$\Gamma \models^{\Sigma} \alpha \quad \text{iff} \quad \mathcal{M} \models^{\Sigma} \alpha \ \text{ for all } \mathcal{M} \in |\mathbf{Mod}(\Sigma, \Gamma)|.$$

An *entailment system* is conceived, in the same way as we did in the previous definition, by identifying a family of deductive relations, instead of a family of semantic consequence relations, where each of the elements in the family is associated to a signature. It then only remains to require these relations to satisfy the properties of reflexivity, monotonicity, transitivity, a notion of translation between two related signatures, and to reflect the properties of soundness and completeness of the deductive relation.

Definition 2. [Entailment system]
An entailment system *is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *satisfying the following conditions:*

- Sign *is a category of signatures,*
- $\mathbf{Sen} : \mathsf{Sign} \to \mathsf{Set}$ *is a functor. Let* $\Sigma \in |\mathsf{Sign}|$*; then* $\mathbf{Sen}(\Sigma)$ *returns the set of* Σ*-sentences, and*
- $\{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}$*, where* $\vdash^{\Sigma} \subseteq 2^{\mathbf{Sen}(\Sigma)} \times \mathbf{Sen}(\Sigma)$*, is a family of binary relations such that for any* $\Sigma, \Sigma' \in |\mathsf{Sign}|$*,* $\{\phi\} \cup \{\phi_i\}_{i \in \mathcal{I}} \subseteq \mathbf{Sen}(\Sigma)$*,* $\Gamma, \Gamma' \subseteq \mathbf{Sen}(\Sigma)$*, the following conditions are satisfied:*
  1. *reflexivity:* $\{\phi\} \vdash^{\Sigma} \phi$*,*
  2. *monotonicity: if* $\Gamma \vdash^{\Sigma} \phi$ *and* $\Gamma \subseteq \Gamma'$*, then* $\Gamma' \vdash^{\Sigma} \phi$*,*
  3. *transitivity: if* $\Gamma \vdash^{\Sigma} \phi_i$ *for all* $i \in \mathcal{I}$ *and* $\{\phi_i\}_{i \in \mathcal{I}} \vdash^{\Sigma} \phi$*, then* $\Gamma \vdash^{\Sigma} \phi$*, and*
  4. $\vdash$*-translation: if* $\Gamma \vdash^{\Sigma} \phi$*, then for any morphism* $\sigma : \Sigma \to \Sigma'$ *in* Sign*,* $\mathbf{Sen}(\sigma)(\Gamma) \vdash^{\Sigma'} \mathbf{Sen}(\sigma)(\phi)$*.*

Definition 3. *Let* $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *be an entailment system, then* Th*, its category of theories, is a pair* $\langle \mathcal{O}, \mathcal{A} \rangle$ *such that:*

- $\mathcal{O} = \{\, \langle \Sigma, \Gamma \rangle \mid \Sigma \in |\mathsf{Sign}| \text{ and } \Gamma \subseteq \mathbf{Sen}(\Sigma) \,\}$*, and*
- $\mathcal{A} = \left\{ \sigma : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle \;\middle|\; \begin{array}{l} \langle \Sigma, \Gamma \rangle, \langle \Sigma', \Gamma' \rangle \in \mathcal{O}, \\ \sigma : \Sigma \to \Sigma' \text{ is a morphism in } \mathsf{Sign} \text{ and} \\ \text{for all } \gamma \in \Gamma, \Gamma' \vdash^{\Sigma'} \mathbf{Sen}(\sigma)(\gamma) \end{array} \right\}.$

In addition, if a morphism $\sigma : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle$ satisfies $\mathbf{Sen}(\sigma)(\Gamma) \subseteq \Gamma'$, it is called *axiom preserving*. This defines the category Th_0 by retaining only those morphisms of Th that are axiom preserving. It is easy to see that Th_0 is a complete subcategory of Th. If we now consider the definition of $\mathbf{Mod}$ extended to signatures and sets of sentences, we get a functor $\mathbf{Mod} : \mathsf{Th}^{\mathsf{op}} \to \mathsf{Cat}$ defined as follows: let $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}|$, then $\mathbf{Mod}(T) = \mathbf{Mod}(\Sigma, \Gamma)$.

Definition 4. *Let* $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *be an entailment system and* $\langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$*, then we define* $^{\bullet} : \mathbf{Sen}(\Sigma) \to \mathbf{Sen}(\Sigma)$ *such that* $\Gamma^{\bullet} = \left\{ \gamma \;\middle|\; \Gamma \vdash^{\Sigma} \gamma \right\}$*, and* $^{\bullet} : \mathsf{Th}_0 \to \mathsf{Th}_0$ *such that* $\langle \Sigma, \Gamma \rangle^{\bullet} = \langle \Sigma, \Gamma^{\bullet} \rangle$*.* $\Gamma^{\bullet}$ *is called the theory generated by* Γ*.*

Definition 5.
Let $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \{\vdash'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$ *be entailment systems,* $\Phi : \mathsf{Th}_0 \to \mathsf{Th}'_0$ *be a functor and* $\alpha : \mathbf{Sen} \to \mathbf{Sen}' \circ \Phi$ *a natural transformation.* Φ *is said to be* α-sensible *if and only if the following conditions are satisfied:*

1. *there is a functor* $\Phi^{\diamond} : \mathsf{Sign} \to \mathsf{Sign}'$ *such that* $\mathbf{sign}' \circ \Phi = \Phi^{\diamond} \circ \mathbf{sign}$*, where* $\mathbf{sign}$ *and* $\mathbf{sign}'$ *are the functors from the corresponding category of theories to the corresponding category of signatures, that when applied to a given theory projects its signature, and*
2. *if* $\langle \Sigma, \Gamma\rangle \in |\mathsf{Th}_0|$ *and* $\langle \Sigma', \Gamma'\rangle \in \mathsf{Th}'_0$ *such that* $\Phi(\langle \Sigma, \Gamma\rangle) = \langle \Sigma', \Gamma'\rangle$*, then* $(\Gamma')^{\bullet} = (\emptyset' \cup \alpha_{\Sigma}(\Gamma))^{\bullet}$*, where* $\emptyset' = \alpha_{\Sigma}(\emptyset)$*.*

Φ *is said to be* α-simple *if and only if* $\Gamma' = \emptyset' \cup \alpha_{\Sigma}(\Gamma)$ *is satisfied in Condition 2, instead of* $(\Gamma')^{\bullet} = (\emptyset' \cup \alpha_{\Sigma}(\Gamma))^{\bullet}$*.*

It is trivial to see, based on the monotonicity of $^{\bullet}$, that α-simplicity implies α-sensibility. An α-sensible functor has the property that its natural transformation α only depends on signatures, which is a consequence of the following lemma.

Lemma 1. *([2, Lemma 22])*
Let $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \{\vdash'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$ *be entailment systems,* $\Phi : \mathsf{Th}_0 \to \mathsf{Th}'_0$ *be a functor satisfying Cond. 1 of Def. 5; then any natural transformation* $\alpha : \mathbf{Sen} \to \mathbf{Sen}' \circ \Phi$ *can be obtained from a natural transformation* $\alpha^{\diamond} : \mathbf{Sen} \to \mathbf{Sen}' \circ \Phi^{\diamond}$ *by the horizontal composition with the functor* $\mathbf{sign} : \mathsf{Th}_0 \to \mathsf{Sign}$*.* ■

Now, from Definitions 1 and 2, it is possible to give a definition of *logic* by relating both its model-theoretic and proof-theoretic characterization.

Definition 6. [Logic]
A logic *is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *satisfying the following conditions:*

- $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *is an entailment system,*
- $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *is an institution, and*
- *the following* soundness *condition is satisfied: for any* $\Sigma \in |\mathsf{Sign}|$*,* $\phi \in \mathbf{Sen}(\Sigma)$*,* $\Gamma \subseteq \mathbf{Sen}(\Sigma)$*,*

$$\Gamma \vdash^{\Sigma} \phi \quad \textit{implies} \quad \Gamma \models^{\Sigma} \phi \ .$$

A logic is complete *if, in addition, the following condition is also satisfied: for any* $\Sigma \in |\mathsf{Sign}|$*,* $\phi \in \mathbf{Sen}(\Sigma)$*,* $\Gamma \subseteq \mathbf{Sen}(\Sigma)$*,*

$$\Gamma \models^{\Sigma} \phi \quad \textit{implies} \quad \Gamma \vdash^{\Sigma} \phi \ .$$

In Def. 2 we associated deductive relations to signatures, but we said nothing about how these relations are obtained. The next definition introduces the notion of *proof calculus*. It formalizes the possibility of associating a proof-theoretic structure to the deductive relations introduced by the definitions of entailment systems. In [2, Ex. 11, pp. 15], Meseguer presents natural deduction as a proof calculus for first-order predicate logic by resorting to *multicategories* (see [2, Definition 10]).

Definition 7. [Proof calculus]
A proof calculus *is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \pi \rangle$ *satisfying the following conditions:*

- $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *is an entailment system,*
- $\mathbf{P} : \mathsf{Th}_0 \to \mathsf{Struct}_{PC}$ *is a functor. Let* $T \in |\mathsf{Th}_0|$*, then* $\mathbf{P}(T) \in |\mathsf{Struct}_{PC}|$ *is the proof-theoretical structure of* T*,*
- $\mathbf{Pr} : \mathsf{Struct}_{PC} \to \mathsf{Set}$ *is a functor. Let* $T \in |\mathsf{Th}_0|$*, then* $\mathbf{Pr}(\mathbf{P}(T))$ *is the set of proofs of* T*; the composite functor* $\mathbf{Pr} \circ \mathbf{P} : \mathsf{Th}_0 \to \mathsf{Set}$ *will be denoted by* **proofs***, and*
- $\pi : \mathbf{proofs} \stackrel{\cdot}{\to} \mathbf{Sen}$ *is a natural transformation such that for each* $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$ *the image of* $\pi_T : \mathbf{proofs}(T) \to \mathbf{Sen}(T)$ *is the set* $\Gamma^{\bullet}$*. The map* π_T *is called the* projection from proofs to theorems *for the theory* T*.*

Finally, a *logical system* will be a logic plus a proof calculus for its proof theory.

Definition 8. *A* logical system *is a structure of the form*

$$\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \pi \rangle$$

satisfying the following conditions:

- $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *is a logic, and*
- $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \pi \rangle$ *is an proof calculus.*

3 Satisfiability calculus

In Sec. 2, we presented the relevant definitions regarding institutions and entailment systems. Additionally, we presented Meseguer's categorical formulation of a proof calculus as a means of providing structure for the abstract relation of entailment defined in an entailment system. In this section, we provide a categorical definition of a satisfiability calculus. A satisfiability calculus is the formal characterization of a method for constructing models of a given theory, thus providing the semantic counterpart of that proof calculus.

In the same way Meseguer proceeded in order to define a proof calculus, the definition of a satisfiability calculus relies on the assignment to a theory of a structure capable of expressing the construction of models of a theory.

Definition 9. [Satisfiability calculus] *A* satisfiability calculus *is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{M}, \mathbf{Mods}, \mu \rangle$ *satisfying the following conditions:*

– $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma\in|\mathsf{Sign}|}\rangle$ *is an institution,*
– $\mathbf{M} : \mathsf{Th}_0 \to \mathsf{Struct}_{SC}$ *is a functor. Let* $T \in |\mathsf{Th}_0|$*, then* $\mathbf{M}(T) \in |\mathsf{Struct}_{SC}|$ *is the model structure of* T*,*
– $\mathbf{Mods} : \mathsf{Struct}_{SC} \to \mathsf{Cat}$ *is a functor. Let* $T \in |\mathsf{Th}_0|$*, then* $\mathbf{Mods}(\mathbf{M}(T))$ *is the set of canonical models of* T*; the composite functor* $\mathbf{Mods} \circ \mathbf{M} : \mathsf{Th}_0 \to \mathsf{Cat}$ *will be denoted by* **models***, and*
– $\mu : \mathbf{models}^{\mathsf{op}} \stackrel{\cdot}{\to} \mathbf{Mod}$ *is a natural transformation such that, for each* $T = \langle \Sigma, \Gamma\rangle \in |\mathsf{Th}_0|$*, the image of* $\mu_T : \mathbf{models}^{\mathsf{op}}(T) \to \mathbf{Mod}(T)$ *is the category of models* $\mathbf{Mod}(T)$*. The map* μ_T *is called the* projection of the category of models *of the theory* T*.*

The intuition behind the previous definition is that, for any theory T, the functor $\mathbf{M}$ assigns a model structure for T in the category Struct_{SC}[6]. The functor **Mods** projects those particular structures that represent sets of conditions that can produce canonical models of a theory $T = \langle \Sigma, \Gamma\rangle$ (i.e., the structures that represent canonical models of Γ). Finally, for any theory T, the functor μ_T relates each of these sets of conditions to the corresponding canonical model.

Example 1. ***[Tableau method for first-order predicate logic]***
First we will present the tableau method for first-order logic. Let $\mathbb{I}_{FOL} = \langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma\in|\mathsf{Sign}|}\rangle$ be the institution of first-order predicate logic. Let $\Sigma \in |\mathsf{Sign}|$ and $S \subseteq \mathbf{Sen}(\Sigma)$; then a *tableau* for S is a tree such that:

1. the nodes are labeled with sets of formulae (over Σ) and the root node is labeled with S,
2. if u and v are two connected nodes in the tree (being u an ancestor of v), then the label of v is obtained from the label of u by applying one of the following rules:

$$\frac{X \cup \{A \wedge B\}}{X \cup \{A \wedge B, A, B\}}\,[\wedge] \qquad \frac{X \cup \{A \vee B\}}{X \cup \{A \vee B, A\} \qquad X \cup \{A \vee B, B\}}\,[\vee]$$

$$\frac{X \cup \{\neg\neg A\}}{X \cup \{\neg\neg A, A\}}\,[\neg_1] \qquad \frac{X \cup \{A\}}{X \cup \{A, \neg\neg A\}}\,[\neg_2] \qquad \frac{X \cup \{A, \neg A\}}{\mathbf{Sen}(\Sigma)}\,[\mathit{false}]$$

$$\frac{X \cup \{\neg(A \wedge B)\}}{X \cup \{\neg(A \wedge B), \neg A \vee \neg B\}}\,[DM_1] \qquad \frac{X \cup \{\neg(A \vee B)\}}{X \cup \{\neg(A \vee B), \neg A \wedge \neg B\}}\,[DM_2]$$

$$t \text{ is a ground term}\ \frac{X \cup \{(\forall x)P(x)\}}{X \cup \{(\forall x)P(x), P(t)\}}\,[\forall]$$

$$c \text{ is a new constant}\ \frac{X \cup \{(\exists x)P(x)\}}{X \cup \{(\exists x)P(x), P(c)\}}\,[\exists]$$

[6] Notice that the target of functor $\mathbf{M}$, when applied to a theory T, is not necessarily a model, but a structure which under certain conditions can be considered to represent the category of models of T.

A sequence of nodes $s_0 \xrightarrow{\tau_0^{\alpha_0}} s_1 \xrightarrow{\tau_1^{\alpha_1}} s_2 \xrightarrow{\tau_2^{\alpha_2}} \dots$ is a *branch* if: *a*) s_0 is the root node of the tree, and *b*) for all $i \leq \omega$, $s_i \to s_{i+1}$ occurs in the tree and $\tau_i^{\alpha_i}$ is the rule applied, labeled with the formula α_i to which it was applied.

A branch $s_0 \xrightarrow{\tau_0^{\alpha_0}} s_1 \xrightarrow{\tau_1^{\alpha_1}} s_2 \xrightarrow{\tau_2^{\alpha_2}} \dots$ in a tableau is *saturated* if there exists $i \leq \omega$ such that $s_i = s_{i+1}$.

A branch $s_0 \xrightarrow{\tau_0^{\alpha_0}} s_1 \xrightarrow{\tau_1^{\alpha_1}} s_2 \xrightarrow{\tau_2^{\alpha_2}} \dots$ in a tableau is *closed* if there exists $i \leq \omega$ and $\alpha \in \mathbf{Sen}(\Sigma)$ such that $\{\alpha, \neg\alpha\} \subseteq s_i$.

Let $s_0 \xrightarrow{\tau_0^{\alpha_0}} s_1 \xrightarrow{\tau_1^{\alpha_1}} s_2 \xrightarrow{\tau_2^{\alpha_2}} \dots$ be a branch in a tableau. Examining the rules presented above, we can see that every s_i with $i < \omega$ is a set of formulae. In each step, the application of a rule decomposes one formula of the set into its constituent parts with respect to its major connective and preserving satisfiability. Thus, the limit set of the branch is a set of formulae containing all the constituent parts of the original set of formulae for which the tableau was built. As a result of this, every open branch expresses, by means of a set of formulae, the class of models satisfying it.

Now, in order to define the tableau method as a satisfiability calculus, we must provide formal definitions for $\mathbf{M}$, $\mathbf{Mods}$ and μ. To do this we must formally define Struct_{SC}, the category of legal tableaux structures. For us, a tableau will be a relation between two sets of formulae; the set of formulae for which the models are constructed, and the set of formulae from which these models are constructed (i.e., the union of the limit sets of all the branches). Then, given a signature $\Sigma \in |\mathsf{Sign}|$ and a set of axioms $\Gamma \subseteq \mathbf{Sen}(\Sigma)$, we denote by $Str^{\Sigma,\Gamma}$ the category of tableaux for sets of formulae over signature Σ and assuming the set of axioms Γ. In $Str^{\Sigma,\Gamma}$, objects are sets of formulae over signature Σ, and morphisms represent tableaux for the set occurring in their source and having as sets of formulae at the end of open branches, subsets of the set of formulae occurring as their target. Struct_{SC} is then defined to be the category in which objects are all possible structures $Str^{\Sigma,\Gamma}$, and morphisms are the homomorphic extension of the morphisms in $||\mathsf{Th}_0||$ to the structure of the tableaux presented above.

The functor $\mathbf{M}$ must be understood as the relation between a theory in $|\mathsf{Th}_0|$ and its category of legal structures representing tableaux, so to every theory T, $\mathbf{M}$ associates the strict monoidal category [20] $\langle Str^{\Sigma,\Gamma}, \cup, \emptyset\rangle$, and for every theory morphism $\sigma : \langle\Sigma, \Gamma\rangle \to \langle\Sigma', \Gamma'\rangle$, $\mathbf{M}$ associates a morphism $\widehat{\sigma} : Str^{\Sigma,\Gamma} \to Str^{\Sigma',\Gamma'}$ which is the homomorphic extension of σ to the structure of the tableaux.

Given $\langle\Sigma, \Gamma\rangle \in |\mathsf{Th}_0|$, the functor $\mathbf{Mods}$ provide the means for obtaining the category containing the closure of those structures in $Str^{\Sigma,\Gamma}$ that represent the closure of the branches in saturated tableaux and, finally, the natural transformation μ relates the structures representing saturated tableaux with the model satisfying the set of formulae denoted by the source of the morphism.

A more formal presentation, accompanied by the results supporting this example, can be found in App. A.1.

Now we show how resolution methods can also be defined as a satisfiability calculus.

Example 2. ***[Resolution method for first-order predicate logic]***
Let us describe resolution for first-order logic; we describe the one introduced in [22]. We use the notation $[A_0, \ldots, A_n]$ to denote a list of formulae; resolution builds a list of lists representing a disjunction of conjunctions. The rules are as follows:

$$\frac{[A_0, \ldots, \neg\neg A]}{[A_0, \ldots, A]}\,[\neg\neg] \quad \frac{\begin{array}{c}[A_0, \ldots, A_n, \neg A]\\ [A'_0, \ldots, A'_n, A]\end{array}}{[A_0, \ldots, A_n, A'_0, \ldots, A'_n]}\,[\neg]$$

$$\frac{[A_0, \ldots, A \wedge A']}{[A_0, \ldots, A, A']}\,[\wedge] \quad \frac{[A_0, \ldots, \neg(A \vee A')]}{[A_0, \ldots, \neg A, \neg A']}\,[\neg\wedge]$$

$$\frac{[A_0, \ldots, A \vee A']}{\begin{array}{c}[A_0, \ldots, A]\\ [A_0, \ldots, A']\end{array}}\,[\vee] \quad \frac{[A_0, \ldots, \neg(A \wedge A')]}{\begin{array}{c}[A_0, \ldots, \neg A]\\ [A_0, \ldots, \neg A']\end{array}}\,[\neg\wedge]$$

$$\text{for any closed term } t\ \frac{[A_0, \ldots, A_n, \forall x : A(x)]}{[A_0, \ldots, A_n, A[x/t]}\,[\forall]$$

$$\text{for a new constant } c\ \frac{[A_0, \ldots, A_n, \exists x : A(x)]}{[A_0, \ldots, A_n, A[x/c]]}\,[\exists]$$

Here we use $A[x]$ to denote a formula with free variable x, and $A[x/t]$ to denote the formula resulting from replacing variable x by term t everywhere in A. For the sake of simplicity, we assume that lists of formulae do not have repeated elements. A resolution is a sequence of lists of formulae. If a resolution contains an empty list (i.e., $[]$) we say that the resolution is closed; otherwise it is an *open* resolution.

For every signature $\Sigma \in |\mathsf{Sign}|$ and each $\Gamma \subset \mathbf{Sen}(\Sigma)$, we denote by $Str^{\Sigma,\Gamma}$ the category whose objects are lists of formulae, and a morphism $\sigma : [A_0, \ldots, A_n] \to [A'_0, \ldots, A'_m]$ represents a sequence of application of resolution rules for $[A'_0, \ldots, A'_m]$.

Then, Struct_{SC} is a category whose objects are $Str^{\Sigma,\Gamma}$, for each signature $\Sigma \in |\mathsf{Sign}|$ and each set of formulae $\Gamma \in \mathbf{Sen}(\Sigma)$, and whose morphisms are of the form $\widehat{\sigma} : Str^{\Sigma,\Gamma} \to Str^{\Sigma',\Gamma'}$, obtained by extending $\sigma : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle$ in $||\mathsf{Th}_0||$ homomorphically.

In a similar way to Ex. 1, the functor $\mathbf{M} : \mathbf{Th}_0 \to \mathsf{Struct}_{SC}$ is defined as $\mathbf{M}(\langle \Sigma, \Gamma \rangle) = \langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle$.

$\mathbf{Mods} : \mathsf{Struct}_{SC} \to \mathsf{Set}$ is defined as in the example above.

In App. A.1 the reader can find the formal details for Ex. 1; it is straightforward to rephrase the definitions and results for the resolution method.

A typical use for these methods is the search for counterexamples for a given formula. To do that, we start applying rules to the negation of this formula; once a saturated tableau is obtained, if all the branches are closed, then there is no model of the axioms and the negation of the formula, thus the formula is a theorem. On the other hand, if there exists an open branch, the limit set of that branch characterizes a class of counterexamples for the formula. This is in contrast to Hilbert systems, where we start from the axioms, and then we apply deduction rules until we get the desired formula.

3.1 Mapping satisfiability calculi

In [5], Tarlecki discussed extensively the ways in which different institutions can be related, and the ways in which they should be interpreted. As in previous work, [23], we will concentrate only on institution representations because they fit our needs better.

The following definition was taken from [5], and formalizes the notion of institution representation.

Definition 10. [Institution representation]
Let $\mathbb{I} = \langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models_\Sigma\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\mathbb{I}' = \langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'_\Sigma\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$ *be institutions. Then,* $\langle \gamma^{Sign}, \gamma^{Sen}, \gamma^{Mod}\rangle : I \to I'$ *is a* representation map *of institutions if and only if:*

- $\gamma^{Sign} : \mathsf{Sign} \to \mathsf{Sign}'$ *is a functor,*
- $\gamma^{Sen} : \mathbf{Sen} \stackrel{\cdot}{\to} \gamma^{Sign} \circ \mathbf{Sen}'$*, is a natural transformation (i.e., a natural family of functions* $\gamma^{Sen}_\Sigma : \mathbf{Sen}(\Sigma) \to \mathbf{Sen}'(\gamma^{Sign}(\Sigma))$*), such that for each* $\Sigma_1, \Sigma_2 \in |\mathsf{Sign}|$ *and* $\sigma : \Sigma_1 \to \Sigma_2$ *morphism in* Sign*,*

$$\begin{tikzcd}
\mathbf{Sen}(\Sigma_2) \arrow[r, "\gamma^{Sen}_{\Sigma_2}"] & \mathbf{Sen}'(\gamma^{Sign}(\Sigma_2)) & \Sigma_2 \\
\mathbf{Sen}(\Sigma_1) \arrow[u, "\mathbf{Sen}(\sigma)"] \arrow[r, "\gamma^{Sen}_{\Sigma_1}"] \arrow[ur, phantom, "\odot"] & \mathbf{Sen}'(\gamma^{Sign}(\Sigma_1)) \arrow[u, "\mathbf{Sen}'(\gamma^{Sign}(\sigma))"'] & \Sigma_1 \arrow[u, "\sigma"']
\end{tikzcd}$$

- $\gamma^{Mod} : (\gamma^{Sign})^{\mathsf{op}} \circ \mathbf{Mod}' \stackrel{\cdot}{\to} \mathbf{Mod}$*, is a natural transformation (i.e., the family of functors* $\gamma^{Mod}_\Sigma : \mathbf{Mod}'((\gamma^{Sign})^{\mathsf{op}}(\Sigma)) \to \mathbf{Mod}(\Sigma)$ *is natural), such that for each* $\Sigma_1, \Sigma_2 \in |\mathsf{Sign}|$ *and* $\sigma : \Sigma_1 \to \Sigma_2$ *a morphism in* Sign*,*

$$\begin{tikzcd}
\mathbf{Mod}'((\gamma^{Sign})^{\mathsf{op}}(\Sigma_2)) \arrow[r, "\gamma^{Mod}_{\Sigma_2}"] \arrow[d, "\mathbf{Mod}'((\gamma^{Sign})^{\mathsf{op}}(\sigma^{\mathsf{op}}))"'] \arrow[dr, phantom, "\odot"] & \mathbf{Mod}(\Sigma_2) \arrow[d, "\mathbf{Mod}(\sigma^{\mathsf{op}})"] & \Sigma_2 \\
\mathbf{Mod}'((\gamma^{Sign})^{\mathsf{op}}(\Sigma_1)) \arrow[r, "\gamma^{Mod}_{\Sigma_1}"] & \mathbf{Mod}(\Sigma_1) & \Sigma_1 \arrow[u, "\sigma"']
\end{tikzcd}$$

such that for any $\Sigma \in |\mathsf{Sign}|$*, the function* $\gamma^{Sen}_{\Sigma} : \mathbf{Sen}(\Sigma) \to \mathbf{Sen}'(\gamma^{Sign}(\Sigma))$ *and the functor* $\gamma^{Mod}_{\Sigma} : \mathbf{Mod}'(\gamma^{Sign}(\Sigma)) \to \mathbf{Mod}(\Sigma)$ *preserves the following satisfaction condition: for any* $\alpha \in \mathbf{Sen}(\Sigma)$ *and* $\mathcal{M}' \in |\mathbf{Mod}(\gamma^{Sign}(\Sigma))|$*,*

$$\mathcal{M}' \models^{\gamma^{Sign}(\Sigma)} \gamma^{Sen}_{\Sigma}(\alpha) \ \ \mathit{iff} \ \ \gamma^{Mod}_{\Sigma}(\mathcal{M}') \models^{\Sigma} \alpha \ .$$

An institution representation $\gamma : I \to I'$ expresses how the "poorer" set of sentences (respectively, category of models) associated to I is encoded in the "richer" one associated to I', and this is done by:

- constructing, for a given I-signature Σ, an I'-signature into which Σ can be interpreted,
- translating, for a given I-signature Σ, the set of Σ-sentences to the corresponding I'-sentences,
- obtaining, for a given I-signature Σ, the category of Σ-models from the corresponding category of Σ'-models.

The direction of the arrows shows how the whole of I is represented by some parts of I'. The following two results, presented in [5], provide the relation between I and I'.

Proposition 1. *Let* $\mathbb{I}$ *and* $\mathbb{I}'$ *be the institutions* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$*, respectively. Let* $\rho : \mathbb{I} \to \mathbb{I}'$ *be an institution representation. Then, for all* $\Sigma \in |\mathsf{Sign}|$*,* $\Gamma \subseteq \mathbf{Sen}(\Sigma)$ *and* $\varphi \in \mathbf{Sen}(\Sigma)$*, if* $\Gamma \models^{\Sigma} \varphi$*, then* $\rho^{Sen}(\Gamma) \models^{\rho^{Sign}(\Sigma)} \rho^{Sen}(\varphi)$*.*

Definition 11. *Let* $\mathbb{I}$ *and* $\mathbb{I}'$ *be the institutions* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$*, respectively. Let* $\rho : \mathbb{I} \to \mathbb{I}'$ *be an institution representation. Then,* $\mathbb{I}$ *has the* ρ-expansion property *if for all* $\langle \Sigma, \Gamma\rangle \in |\mathsf{Th}^{\mathbb{I}}_0|$*,* $\mathcal{M} \in |\mathbf{Mod}^{\mathbb{I}}(\langle \Sigma, \Gamma\rangle)|$*, there exists* $\mathcal{M}' \in |\mathbf{Mod}^{\mathbb{I}'}(\langle \rho^{Sign}(\Sigma), \rho^{Sen}(\Gamma)\rangle)|$ *such that* $\mathcal{M} = \rho^{Mod}(\mathcal{M}')$*.*

Theorem 1. *Let* $\mathbb{I}$ *and* $\mathbb{I}'$ *be the institutions* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$*, respectively. Let* $\rho : \mathbb{I} \to \mathbb{I}'$ *be an institution representation. Then, for all* $\Sigma \in |\mathsf{Sign}|$*,* $\Gamma \subseteq \mathbf{Sen}(\Sigma)$ *and* $\varphi \in \mathbf{Sen}(\Sigma)$*, if every* $\mathcal{M} \in \mathbf{Mod}(\langle \Sigma, \Gamma\rangle)$ *has the* ρ*-expansion property, then* $\Gamma \models^{\Sigma} \varphi$ *if and only if* $\rho^{Sen}(\Gamma) \models^{\rho^{Sign}(\Sigma)} \rho^{Sen}(\varphi)$*.*

In many cases (those in which the class of models of a signature in the source institution is completely axiomatizable in the language of the target one), Def. 10 can easily be extended to map signatures of one institution to theories of another. This is done so the class of models of the richer one can be restricted, by means of the addition of axioms (thus the need for theories in the image of the functor γ^{Sign}), in order to be exactly the class of models obtained by translating to it the class of models of the corresponding signature of the poorer one. This new definition of institution representation guaranties that the ρ-expansion property holds, and consequently Thm. 1 is satisfied.

In the same way, when the previously described extension is possible, we can obtain what Meseguer calls a *map of institutions* by reformulating the definition so the functor between signatures of one institution and theories of the other is $\gamma^{Th} : \mathsf{Th}_0 \to \mathsf{Th}'_0$, which has to be γ^{Sen}-sensible with respect to the entailment systems induced by the institutions I and I'. Now, if $\langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$, then γ^{Th_0} can be defined as follows:

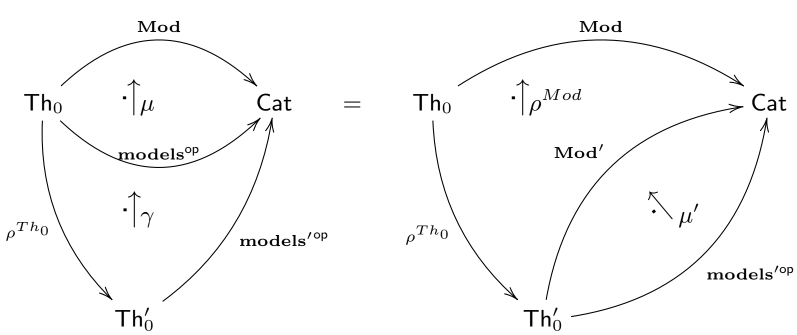
$$\gamma^{Th_0}(\langle \Sigma, \Gamma \rangle) = \langle \gamma^{Sign}(\Sigma), \Delta \cup \gamma^{Sen}_{\Sigma}(\Gamma) \rangle \ ,$$

where $\Delta \subseteq \mathbf{Sen}(\rho^{Sign}(\Sigma))$. Then, it is easy to prove that γ^{Th_0} is γ^{Sen}-simple because it is the γ^{Sen}-extension of γ^{Th_0} to theories, thus being γ^{Sen}-sensible.

The notion of a *map of satisfiability calculi* is the natural extension of a map of institutions in order to consider the more material version of the satisfiability relation. In some sense, if a map of institutions provides a means for representing one satisfiability relation in terms of another in a semantics preserving way, the map of satisfiability calculi provides a means for representing a model construction technique in terms of another. This is done by showing how model construction techniques for richer logics express techniques associated with poorer ones.

Definition 12. *Let* $\mathbb{S} = \langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{M}, \mathbf{Mods}, \mu \rangle$ *and* $\mathbb{S}' = \langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}, \mathbf{M}', \mathbf{Mods}', \mu' \rangle$ *be satisfiability calculi. Then,* $\langle \rho^{Sign}, \rho^{Sen}, \rho^{Mod}, \gamma \rangle : \mathbb{S} \to \mathbb{S}'$ *is a* map of satisfiability calculi *if and only if:*

1. $\langle \rho^{Sign}, \rho^{Sen}, \rho^{Mod} \rangle : \mathbb{I} \to \mathbb{I}'$ *is a map of institutions, and*
2. $\gamma : \rho^{Th_0} \circ \mathbf{models}'^{\mathsf{op}} \stackrel{\cdot}{\to} \mathbf{models}^{\mathsf{op}}$ *is a natural transformation such that the following equality holds:*



*Example 3. **[Mapping modal logic to first-order logic]***
A trivial example of a mapping between satisfiability calculi is the mapping between the tableau method for propositional logic, and the one for first-order logic. It is straightforward since the tableau method for first-order logic is an extension of that of propositional logic.

Let us introduce a more interesting example; we will map the tableau method for modal logic (as presented by Fitting [22]) to the first-order predicate logic tableau method. The mapping between the institutions is given by the standard translation of modal logic to first-order logic. Let us recast here the tableau method for the system K. In [22] formulae are prefixed by labels denoting semantic states. Labeled formulae are then terms of the form: $\ell : \varphi$, where φ is a modal formula and ℓ is a sequence of natural numbers $n_0, \ldots, n_k$. The relationship R between these labels, is then defined in the following way: $\ell R \ell' \equiv \exists n : \ell, n = \ell'$. The new rules are the following:

$$\text{For all } \ell' \text{ such that } \ell R \ell' \text{ and such that } \ell' \text{ appears in } X \; \frac{X \cup \{\ell : \Box\varphi\}}{X \cup \{\ell : \Box\varphi, \ell' : \varphi\}} \, [\Box]$$

$$\text{For } \ell' \text{ such that } \ell R \ell' \; \frac{X \cup \{\ell : \Diamond\varphi\}}{X \cup \{\ell : \Diamond\varphi, \ell' : \varphi\}} \, [\Diamond]$$

The rules for the propositional connectives are the usual ones obtained by labeling the formulae with a given label. Notice that labels denote states of a Kripke frame; this is related in some way with the tableau method used for first-order predicate logic. Branches, saturated branches and closed branched are defined in the same way as in Ex. 1, but considering the relations between sets to be also indexed by the relation used at that point. Thus, $s_i \xrightarrow[R_i]{\tau_{\alpha_i}} s_{i+1}$ must be understood as: the set s_{i+1} is obtained from s_i by applying rule τ_i to formula $\alpha_i \in s_i$ under the accessibility relation R_i.

Consider now the standard translation from modal logic to first-order logic. Therefore, the tuple $\langle \rho^{Sign}, \rho^{Sen}, \rho^{Mod} \rangle$ is defined as follows:

- ρ^{Sign} is the function that translates modal vocabularies to first-order signatures as follows: $\{p_0, p_1, \ldots\} \mapsto \langle R, p_0, p_1, \ldots \rangle$, where R is a binary relation symbol, and each p_i is an unary relation symbol.
- ρ^{Sen}, is the standard translation from modal formulae to first-order formulae, see [24] for the details.
- ρ^{Mod}, is a natural transformation that, given a signature and a first-order model of a translation of a modal language, constructs the corresponding modal model, using the interpretation of the relation R.

The proof that this is a mapping between institutions relies on the correctness of the translation [24]. Using this map we can define a mapping between the corresponding satisfiability calculi. The natural transformation: $\gamma : \rho^{Th_0} \circ \mathbf{models}'^{\mathsf{op}} \dot{\to} \mathbf{models}^{\mathsf{op}}$ is defined as follows:

- Let us define the functors $\gamma_T : \rho^{Th_0} \circ \mathbf{models}'^{\mathsf{op}}(T) \dot{\to} \mathbf{models}^{\mathsf{op}}(T)$ for any theory $T \in |\mathsf{Th}_0|$ as a map that take each canonical model for a first-order theory, which is a translation of a modal theory, to a canonical model of the corresponding modal theory. The mapping is standard in the sense that the relation R defines the canonical Kripke structure, and each unary predicate defines the truth value of the propositions in each state.

To be a mapping between satisfiability calculi the following equality must hold for any theory $\langle \Sigma, \Gamma \rangle \in ||\mathsf{Th}_0||$:

$$\mu_{\langle \Sigma, \Gamma \rangle} \circ \gamma_{\langle \Sigma, \Gamma \rangle} = \rho^{Mod}_{\rho^{Sign}(\Sigma)} \circ \mu'_{\rho^{Sign}(\Sigma)} \; .$$

This means that building a tableau using the first-order rules for the translation of a modal theory, then obtaining the corresponding canonical model in modal logic using γ, and therefore obtaining the class of models by using μ, is exactly the same as obtaining the first-order models by μ' and then the corresponding modal models by using ρ^{Mod}. Roughly speaking, this implies that the translation of saturated tableaus is coherent with respect to the mapping of institutions.

A more formal presentation, accompanied by the results supporting this example, can be found in App. A.2.

4 Conclusions and Further work

Meseguer [2] introduced the notion of proof calculus, which in some sense *implements* the deduction relation of an entailment system. In this paper we made an attempt to complete the picture by providing the notion of Satisfiability Calculus, which might be thought of as the semantical counterpart of the notion of proof calculus, and provides the formal foundations for those proof systems that use model construction techniques to prove or disprove a given formula, thus *implementing* the satisfiability relation of an institution. These techniques constitute an important stream of research in logic; in particular, these methods play an important role in automatic software validation and verification.

Methods like resolution and tableaux are strongly related to the semantics of a logic; and, therefore, we can often use them to construct models; this is not possible in pure deductive methods, such as natural deduction or Hilbert systems, as formalized by Meseguer. Our goal was to provide an abstract characterization of this class of semantics based tools for logical systems. This was accomplished by introducing a categorical characterization of the notion of a satisfiability calculus, which embraces logical tools such as tableaux and resolution, Gentzen style sequents, etc. As we mentioned above, it can be thought of as a formalization of a semantic counterpart of Meseguer's proof calculus.

Given a logical system for which, following our new definition that includes the notion of a satisfiability calculus, we can provide both a proof calculus and a satisfiability calculus implementing the entailment relation and the satisfaction relation, respectively. There clearly exist connections between them that can be explored; this is especially true when the underlying structure used in both definitions is the same (for example, the case for the tableau method for first-order predicate logic, see Ex. 1).

If we examine the definitions of proof calculus and satisfiability calculus, it is easy to see that the restrictions over the natural family of functors $\pi_{\langle \Sigma, \Gamma \rangle} : \mathbf{proofs}(\langle \Sigma, \Gamma \rangle) \to \mathbf{Sen}(\langle \Sigma, \Gamma \rangle)$ and $\mu_{\langle \Sigma, \Gamma \rangle} : \mathbf{models}^{\mathsf{op}}(\langle \Sigma, \Gamma \rangle) \to \mathbf{Mod}(\langle \Sigma, \Gamma \rangle)$ to yield $\Gamma^{\bullet}$ and $\mathbf{Mod}(\langle \Sigma, \Gamma \rangle)$, respectively, maybe too restrictive. Partial implementations of both the entailment relation and the satisfiability relation are gaining visibility in the software engineering community. Examples on the syntactic side are the implementation of less expressive calculi, either for the sake of simplicity, as in the case of the finitary definition of the reflexive and transitive closure in the Kleene algebras with tests [25], the case of the implementation of rewriting tools like Maude [26] as a partial implementation of equational logic, etc. Examples on the semantic side are the many bounded model checkers for undecidable languages that are being implemented, such as Alloy [27] for relational logic, the growing family of SMT-solvers [28], etc. Removing this "restriction" implies allowing these partial implementations, in so far as they comply with behaving as a natural family of methods, which in this case implies that the monotonicity of deduction (respectively satisfaction) under change of notation. Additionally, mappings between partial proof calculi (respectively, satisfiability calculi) can provide an ordering for how good a method is as an approximation of the ideal entailment relation (respectively, satisfaction relation).

We also explored the concept of mappings between satisfiability calculi and the relation between proof calculi and satisfiability calculi.

Finally, extending the definition of satisfiability calculus to a structure capable of managing the concept of validity does not present any difficulty.

References

1. Goguen, J.A., Burstall, R.M.: Introducing institutions. In Clarke, E.M., Kozen, D., eds.: Proceedings of the Carnegie Mellon Workshop on Logic of Programs. Volume 184 of Lecture Notes in Computer Science., Springer-Verlag (1984) 221–256. Cited on page: 1
2. Meseguer, J.: General logics. In Ebbinghaus, H.D., Fernandez-Prida, J., Garrido, M., Lascar, D., Artalejo, M.R., eds.: Proceedings of the Logic Colloquium '87. Volume 129., Granada, Spain, North Holland (1989) 275–329. Cited on page: 1, 3, 5, 6, 14
3. Fiadeiro, J.L., Maibaum, T.S.E.: Generalising interpretations between theories in the context of π-institutions. In Burn, G., Gay, D., Ryan, M., eds.: Proceedings of the First Imperial College Department of Computing Workshop on Theory and Formal Methods, London, UK, Springer-Verlag (1993) 126–147. Cited on page: 1
4. Goguen, J.A., Burstall, R.M.: Institutions: abstract model theory for specification and programming. Journal of the ACM **39**(1) (1992) 95–146. Cited on page: 1
5. Tarlecki, A.: Moving between logical systems. In Haveraaen, M., Owe, O., Dahl, O.J., eds.: Selected papers from the 11th Workshop on Specification of Abstract Data Types Joint with the 8th COMPASS Workshop on Recent Trends in Data Type Specification. Volume 1130 of Lecture Notes in Computer Science., Springer-Verlag (1996) 478–502. Cited on page: 1, 10, 11
6. Sannella, D., Tarlecki, A.: Specifications in an arbitrary institution. Information and computation **76**(2–3) (1988) 165–210. Cited on page: 1
7. Tarlecki, A.: Abstract specification theory: an overview. In Broy, M., Pizka, M., eds.: Proceedings of the NATO Advanced Study Institute on Models, Algebras and Logic of Engineering Software. NATO Science Series, Marktoberdorf, Germany, IOS Press (2003) 43–79. Cited on page: 1
8. Mossakowski, T., Tarlecki, A.: Heterogeneous logical environments for distributed specifications. In Corradini, A., Montanari, U., eds.: Proceedings of 19th International Workshop in Algebraic Development Techniques. Volume 5486 of Lecture Notes in Computer Science., Pisa, Italy, Springer-Verlag (2009) 266–289. Cited on page: 1

9. Diaconescu, R., Futatsugi, K.: Logical foundations of CafeOBJ. Theoretical Computer Science **285**(2) (2002) 289–318. Cited on page: 1
10. Diaconescu, R., ed.: Institution-independent Model Theory. Volume 2 of Studies in Universal Logic. Birkhäuser (2008). Cited on page: 2
11. Mossakowski, T., Maeder, C., Luttich, K.: The heterogeneous tool set, Hets. In Grumberg, O., Huth, M., eds.: Proceedings of the 13th. International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2007). Volume 4424 of Lecture Notes in Computer Science., Braga, Portugal, Springer-Verlag (2007) 519–522. Cited on page: 2
12. Tarlecki, A.: Towards heterogeneous specifications. In Gabbay, D., de Rijke, M., eds.: Frontiers of Combining Systems. Volume 2 of Studies in Logic and Computation. Research Studies Press (2000) 337–360. Cited on page: 2
13. Cengarle, M.V., Knapp, A., Tarlecki, A., Wirsing, M.: A heterogeneous approach to UML semantics. In Degano, P., DeNicola, R., Meseguer, J., eds.: Proceedings of Concurrency, graphs and models (Essays dedicated to Ugo Montanari on the occasion of his 65th. birthday). Number 5065 in Lecture Notes in Computer Science, Edinburgh, Scotland, Springer-Verlag (2008) 383–402. Cited on page: 2
14. Beth, E.W.: The Foundations of Mathematics. North Holland (1959) . Cited on page: 2
15. Beth, E.W.: Semantic entailment and formal derivability. In Hintikka, J., ed.: The Philosophy of Mathematics. Oxford University Press (1969) 9–41 Reprinted from [29].. Cited on page: 2, 17
16. Herbrand, J.: Investigation in proof theory. In Goldfarb, W.D., ed.: Logical Writings. Harvard University Press (1969) 44–202 Translated to english from [30]. . Cited on page: 2, 17
17. Gentzen, G.: Investigation into logical deduction. In Szabo, M.E., ed.: The Collected Papers of Gerhard Gentzen. North Holland (1969) 68–131 Translated to english from [31].. Cited on page: 2, 17
18. Smullyan, R.M.: First-order Logic. Dover Publishing (1995). Cited on page: 2
19. Robinson, J.A.: A machine-oriented logic based on the resolution principle. Journal of the ACM **12**(1) (1965) 23–41. Cited on page: 2
20. McLane, S.: Categories for working mathematician. Graduate Texts in Mathematics. Springer-Verlag, Berlin, Germany (1971). Cited on page: 3, 8
21. Fiadeiro, J.L.: Categories for software engineering. Springer-Verlag (2005). Cited on page: 3
22. Fitting, M.: Tableau methods of proof for modal logics. Notre Dame Journal of Formal Logic **13**(2) (1972) 237–247 Lehman College.. Cited on page: 9, 12, 13
23. Lopez Pombo, C.G., Frias, M.F.: Fork algebras as a sufficiently rich universal institution. In Johnson, M., Vene, V., eds.: Proceedings of the 11th. International Conference on Algebraic Methodology and Software Technology, AMAST 2006. Volume 4019 of Lecture Notes in Computer Science., Kuressaare, Estonia, Springer-Verlag (2006) 235–247. Cited on page: 10
24. Blackburn, P., de Rijke, M., Venema, Y.: Modal logic. Number 53 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press (2001). Cited on page: 13
25. Kozen, D.: Kleene algebra with tests. ACM Transactions on Programming Languages and Systems **19**(3) (1997) 427–443. Cited on page: 15
26. Clavel, M., Eker, S., Lincoln, P., Meseguer, J.: Principles of Maude. In Meseguer, J., ed.: Proceedings of the 1st. International Workshop on Rewriting Logic and its Applications (WRLA'96). Volume 4 of Electronic Notes in Theoretical Computer Science., Elsevier (1996) 65–89. Cited on page: 15

27. Jackson, D.: Alloy: a lightweight object modelling notation. ACM Transactions on Software Engineering and Methodology **11**(2) (2002) 256–290. Cited on page: 15
28. Moura, L.D., Bjørner, N.: Satisfiability modulo theories: introduction and applications. Communications of the ACM **54**(9) (2011) 69–77. Cited on page: 15
29. Beth, E.W.: Semantic entailment and formal derivability. Mededlingen van de Koninklijke Nederlandse Akademie van Wetenschappen, Afdeling Letterkunde **18**(13) (1955) 309–342 Reprinted in [15].. Cited on page: 16
30. Herbrand, J.: Recherches sur la theorie de la demonstration. PhD thesis, Université de Paris (1930) English translation in [16].. Cited on page: 16
31. Gentzen, G.: Untersuchungen tiber das logische schliessen. Mathematische Zeitschrijt **39** (1935) 176–210 and 405–431 English translation in [17]. . Cited on page: 16

A Formal definitions and proofs

In this section we will present detailed explanations, definitions and proofs of the results supporting the examples we presented in Sec. 3.

A.1 Tableau method for first-order predicate logic

In Ex. 1 we presented the tableau method for first-order predicate logic and the intuitions for how it fits in to the definition of a satisfiability calculus. In this section we will provide the formal definitions and the results proving it.

From now on we will work with $\mathbb{I}_{FOL} = \langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$, the institution of first-order predicate logic.

Definition 13. *Let $\Sigma \in |\mathsf{Sign}|$ and $\Gamma \subseteq \mathbf{Sen}(\Sigma)$, then we define $Str^{\Sigma,\Gamma} = \langle \mathcal{O}, \mathcal{A} \rangle$ such that $\mathcal{O} = 2^{\mathbf{Sen}(\Sigma)}$ and $\mathcal{A} = \{\alpha : \{A_i\}_{i \in \mathcal{I}} \to \{B_j\}_{j \in \mathcal{J}} \mid \alpha = \{\alpha_j\}_{j \in \mathcal{J}}\}$ where for all $j \in \mathcal{J}$, α_j is a branch in a tableau for $\Gamma \cup \{B_j\}$ with leaves $\Delta \subseteq \{A_i\}_{i \in \mathcal{I}}$. It should be noted that $\Delta \models_{\Sigma} \Gamma \cup \{B_j\}$.*

Lemma 2. *Let $\Sigma \in |\mathsf{Sign}|$ and $\Gamma \subseteq \mathbf{Sen}(\Sigma)$; then $\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle$, where $\cup : Str^{\Sigma,\Gamma} \times Str^{\Sigma,\Gamma} \to Str^{\Sigma,\Gamma}$ is the typical bi-functor on sets and functions, and $\emptyset$ is the neutral element for $\cup$, is a strict monoidal category.*

Definition 14. Struct_{SC} *is defined as $\langle \mathcal{O}, \mathcal{A} \rangle$ where $\mathcal{O} = \{Str^{\Sigma,\Gamma} \mid \Sigma \in |\mathsf{Sign}| \wedge \Gamma \subseteq \mathbf{Sen}(\Sigma)\}$, and $\mathcal{A} = \{\widehat{\sigma} : Str^{\Sigma,\Gamma} \to Str^{\Sigma',\Gamma'} \mid \sigma : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle \in ||\mathsf{Th}_0||\}$, the homomorphic extension of the morphisms in $||\mathsf{Th}_0||$.*

Lemma 3. Struct_{SC} *is a category.*

Proof. Morphisms $\widehat{\sigma} \in \mathcal{A}$ are the homomorphic extension of the morphisms $\sigma \in ||\mathsf{Th}_0||$ to the structure of the tableaux, translating sets of formulae and preserving the application of the rules. Following this, the composition of $\widehat{\sigma_1}, \widehat{\sigma_2} \in \mathcal{A}$, the homomorphic extensions $\sigma_1, \sigma_2 \in ||\mathsf{Th}_0||$, not only exists, but it is the homomorphic extension of the morphism $\sigma_1 \circ \sigma_2 \in ||\mathsf{Th}_0||$. The associativity of the composition is also trivial to prove by considering that the morphisms are homomorphic extensions and by the associativity of the composition of morphisms in Th_0. The identity morphism is the homomorphic extension of the identity morphism for the corresponding signature.

Definition 15. $\mathbf{M} : \mathsf{Th}_0 \to \mathsf{Struct}_{SC}$ *is defined as* $\mathbf{M}(\langle \Sigma, \Gamma \rangle) = \langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle$ *and* $\mathbf{M}(\sigma : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle) = \widehat{\sigma} : \langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle \to \langle Str^{\Sigma',\Gamma'}, \cup, \emptyset \rangle$*, the homomorphic extension of* σ *to the structures in* $\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle$.

Lemma 4. $\mathbf{M}$ *is a functor.*

Proof. Let $id_{\langle \Sigma,\Gamma \rangle} : \langle \Sigma, \Gamma \rangle \to \langle \Sigma, \Gamma \rangle \in ||\mathsf{Th}_0||$ be the identity morphism for $\langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$. $\mathbf{M}(id_{\langle \Sigma,\Gamma \rangle}) = id_{\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle}$ because, by Def. 14, it is the homomorphic extension of $id_{\langle \Sigma,\Gamma \rangle}$ to the structures in $Str^{\Sigma,\Gamma}$.

Let $\sigma_1 : \langle \Sigma_1, \Gamma_1 \rangle \to \langle \Sigma_2, \Gamma_2 \rangle, \sigma_2 : \langle \Sigma_2, \Gamma_2 \rangle \to \langle \Sigma_3, \Gamma_3 \rangle \in ||\mathsf{Th}_0||$; now, as composition of homomorphisms is a homomorphism, then $\mathbf{M}(\sigma_1 \circ \sigma_2)$ is the composition $\mathbf{M}(\sigma_1) \circ \mathbf{M}(\sigma_2)$.

Definition 16. *Let* $\Sigma \in |\mathsf{Sign}|$, $\Delta \subseteq \mathbf{Sen}(\Sigma)$*, and consider* $\{F_i\}_{i<\omega}$ *an enumeration of* $\mathbf{Sen}(\Sigma)$ *such that for every formula* α*, its sub-formuli are enumerated before* α*. Then* $Cn(\Delta)$ *is defined as follows:*

- $Cn(\Delta) = \bigcup_{i<\omega} Cn^i(\Delta)$
- $Cn^0(\Delta) = \Delta$, $Cn^{i+1}(\Delta) = \begin{cases} Cn^i(\Delta) \cup \{F_i\} & \text{, if } Cn^i(\Delta) \cup \{F_i\} \text{ is consistent.} \\ Cn^i(\Delta) \cup \{\neg F_i\} & \text{, otherwise.} \end{cases}$

Definition 17. $\mathbf{Mods} : \mathsf{Struct}_{SC} \to \mathsf{Cat}$ *is defined as* $\mathbf{Mods}(\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle) = \{\langle Cn(\widetilde{\Delta}), \Sigma \rangle \mid (\exists \alpha : \Delta \to \emptyset \in |Str^{\Sigma,\Gamma}|)(\widetilde{\Delta} \to \emptyset \in \alpha \wedge (\forall \alpha' : \Delta' \to \Delta \in ||Str^{\Sigma,\Gamma}||)(\Delta' = \Delta))\}$ *and for all* $\sigma : \Sigma \to \Sigma' \in |\mathsf{Sign}|$ *(and* $\widehat{\sigma} : \langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle \to \langle Str^{\Sigma',\Gamma'}, \cup, \emptyset \rangle \in ||\mathsf{Struct}_{SC}||$*), the following holds:* $\mathbf{Mods}(\widehat{\sigma})(\langle Cn(\widetilde{\Delta}), \Sigma \rangle) = \langle Cn(\mathbf{Sen}(\sigma)(Cn(\widetilde{\Delta}))), \Sigma' \rangle$.

Lemma 5. $\mathbf{Mods}$ *is a functor.*

Proof. As for each theory $\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle \in |\mathsf{Struct}_{SC}|$, $\mathbf{Mods}(\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle)$ is a discrete category containing the canonical models for $\langle \Sigma, \Gamma \rangle$, the only property that must be proved is that for all $\widehat{\sigma} : \langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle \to \langle Str^{\Sigma',\Gamma'}, \cup, \emptyset \rangle \in ||\mathsf{Struct}_{SC}||$, $o \in |\mathbf{Mods}(\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle)|$, $\mathbf{Mods}(\widehat{\sigma})(o) \in |\mathbf{Mods}(\langle Str^{\Sigma',\Gamma'}, \cup, \emptyset \rangle)|$. Let $o = \langle Cn(\widetilde{\Delta}), \Sigma \rangle$, then, by definition, $\mathbf{Mods}(\widehat{\sigma})(o) = \langle Cn(\mathbf{Sen}(\sigma)(Cn(\widetilde{\Delta})), \Sigma' \rangle$. Observe that, as a consequence of the fact that $\widehat{\sigma}$ is the homomorphic extension of $\mathbf{Sen}(\sigma)$ to the structure of tableaux, the canonical model obtained by applying $\mathbf{Mods}(\widehat{\sigma})$ to a particular element of $\mathbf{Mods}(\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle)$ is a canonical model obtained from a branch of a tableau in $\langle Str^{\Sigma',\Gamma'}, \cup, \emptyset \rangle$.

Definition 18. *Let* $\langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$*, then we define* $\mu_\Sigma : \mathbf{models}^{\mathsf{op}}(\langle \Sigma, \Gamma \rangle) \to \mathbf{Mod}_{FOL}(\langle \Sigma, \Gamma \rangle)$ *as* $\mu_\Sigma(\langle \Delta, \Sigma \rangle) = \mathbf{Mod}(\langle \Sigma, \Delta \rangle)$.

Fact 1 *Let* $\Sigma \in |\mathsf{Sign}_{FOL}|$ *and* $\Gamma \subseteq \mathbf{Sen}_{FOL}(\Sigma)$*, then* $\mu_{\langle \Sigma,\Gamma \rangle}$ *is a functor.*

Lemma 6. μ *is a natural family of functors.*

Proof. Let $\langle \Sigma, \Gamma \rangle, \langle \Sigma', \Gamma' \rangle \in |\mathsf{Th}_0|$ and $\sigma : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle \in |\mathsf{Th}_0|$. Then, the naturality condition for μ can be expressed in the following way:

$$\begin{tikzcd}
\langle \Delta', \Sigma' \rangle \arrow[r, "\mathbf{models}^{\mathsf{op}}(\sigma)"] \arrow[d, "\mu_{\Sigma'}"] & \langle \Delta, \Sigma \rangle \arrow[d, "\mu_{\Sigma}"] \\
\mathbf{Mod}(\langle \Delta', \Sigma' \rangle) \arrow[r, "\mathbf{Mod}(\sigma)"] & \mathbf{Mod}(\langle \Delta, \Sigma \rangle)
\end{tikzcd}$$

It is trivial to check that this condition holds by observing that canonical models are closed theories, thus behaving as theory presentations in Th_0.

Now, from Lemmas 4, 5, and 6, and considering the hypothesis that $\mathbb{I}_{FOL}$ is an institution, the following corollary follows.

Corollary 1. $\langle \mathsf{Sign}_{FOL}, \mathbf{Sen}_{FOL}, \mathbf{Mod}_{FOL}, \{\models^{\Sigma}_{FOL}\}_{\Sigma \in |\mathsf{Sign}_{FOL}|}, \mathbf{M}, \mathbf{Mods}, \mu \rangle$ *is a satisfiability calculus.*

A.2 Mapping modal logic to first-order logic

In Ex. 3 we presented the intuitions behind the mapping from the tableau method for modal logic to the tableau method for first-order predicate logic. In this section we will provide the formal definitions and the results proving its existence.

Assume $\langle \mathsf{Sign}_{FOL}, \mathbf{Sen}_{FOL}, \mathbf{M}_{FOL}, \mathbf{Mods}_{FOL}, \{\models^{\Sigma}_{FOL}\}_{\Sigma \in |\mathsf{Sign}_{FOL}|}, \mu_{FOL} \rangle$ is the satisfiability calculus for first-order predicate logic, denoted by $\mathbb{SC}_{FOL}$, and $\langle \mathsf{Sign}_K, \mathbf{Sen}_K, \mathbf{M}_K, \mathbf{Mods}_K, \{\models^{\Sigma}_{K}\}_{\Sigma \in |\mathsf{Sign}_K|, \mu_K} \rangle$ is the satisfiability calculus for modal logic, denoted by $\mathbb{SC}_K$.

Definition 19. $\rho^{Sign} : \mathsf{Sign}_K \to \mathsf{Sign}_{FOL}$ *is defined as* $\rho^{Sign}(\langle \{p_i\}_{i \in \mathcal{I}} \rangle) = \langle R, \{p_i\}_{i \in \mathcal{I}} \rangle$ *by mapping each propositional variable* p_i*, for all* $i \in \mathcal{I}$*, to a first-order predicate logic predicate* p_i*, and adding a binary predicate* R*, and* $\rho^{Sign}(\sigma : \langle \{p_i\}_{i \in \mathcal{I}} \rangle \to \langle \{p'_{i'}\}_{i' \in \mathcal{I}'} \rangle) = \sigma' : \langle R, \{p_i\}_{i \in \mathcal{I}} \rangle \to \langle R', \{p'_{i'}\}_{i' \in \mathcal{I}'} \rangle$ *mapping* R *to* R'*, and* p_i *to* p'_i *for all* $i \in \mathcal{I}$.

Lemma 7. ρ^{Sign} *is a functor.*

Proof. To show that ρ^{Sign} is a functor we have to prove that it preserves identity and composition. Consider a signature $\Sigma = \langle \{p_i\}_{i \in \mathcal{I}} \rangle$; the identity is just the mapping $\{p_i \mapsto p_i\}_{i \in \mathcal{I}}$. By Def. 19 we obtain that $\rho^{\Sigma}(\{p_i \mapsto p_i\}_{i \in \mathcal{I}}) = \{R \mapsto R\} \cup \{p_i \mapsto p_i\}_{i \in \mathcal{I}} \rangle$, thus yielding the identity for signature $\langle R, \{p_i\}_{i \in \mathcal{I}} \rangle$.

Let $\Sigma, \Sigma', \Sigma'' \in |\mathsf{Sign}|$ and assume there are two morphisms $\sigma : \Sigma \to \Sigma', \sigma' : \Sigma' \to \Sigma'' \in ||\mathsf{Th}_0||$. Then $\rho^{Sign}(\sigma \circ \sigma') = \rho^{Sign}(\{p_i \mapsto p''_i\})$, and therefore $\rho^{Sign}(\sigma \circ \sigma') = \{R \mapsto R''\} \cup \{p_i \mapsto p''_i\}$. By definition of composition of functions $\{R \mapsto R''\} \cup \{p_i \mapsto p''_i\} = \{R \mapsto R'\} \cup \{p_i \mapsto p'_i\} \circ \{R' \mapsto R''\} \cup \{p'_i \mapsto p''_i\}$, and consequently $\{R \mapsto R''\} \cup \{p_i \mapsto p''_i\} = \rho^{Sign}(\sigma) \circ \rho^{Sign}(\sigma')$.

Fact 2 ρ^{Th_0}*, the extension of* ρ^{Sign} *defined as in Sec. 3.1, is* ρ^{Sen}*-sensible.*

Definition 20. *Let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$*, then* $\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle} : \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \to \rho^{Sign}\circ\mathbf{Sen}_{FOL}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)$ *is defined recursively as* $\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) = T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\alpha)$ *where:*

$$\begin{aligned}
T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(p_i) &= p'_i(x) \text{ , for all } i \in \mathcal{I}.\\
T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\neg\alpha) &= \neg T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\alpha)\\
T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\alpha\vee\beta) &= T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\alpha) \vee T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\beta)\\
T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,x}(\Diamond\alpha) &= (\exists y)(R(x,y) \wedge T_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle,y}(\alpha))
\end{aligned}$$

Fact 3 *Let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$*,* $\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ *is a function.*

Lemma 8. ρ^{Sen} *is a natural family of functions.*

Proof. To prove this lemma we must prove that the equality $\mathbf{Sen}_K(\sigma)\circ\rho^{Sen}_{\Sigma'} = \rho^{Sen}_{\Sigma}\circ\mathbf{Sen}_{FOL}(\rho^{Sign}(\sigma))$ holds for every formula $\alpha \in |\mathsf{Th}^K_0|$. Notice that $\mathbf{Sen}_K(\sigma)$ and $\mathbf{Sen}_{FOL}(\rho^{Sign}(\sigma))$ only translate extra-logical symbols preserving the logical structure of the formulae because they are homomorphic extensions of the morphisms to the structure of the formulae induced by $\mathbf{Sen}_K$ and $\mathbf{Sen}_{FOL}$. On the other hand, the reader can see that two formulae that are α-convertible yield, after the application of ρ^{Sen}_{Σ} and $\rho^{Sen}_{\Sigma'}$, α-convertible formulae in the target category of sentences such that, by Def. 19, preserving the mapping of extra-logical symbols.

Definition 21. *Let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$*, then we define* $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle} : \rho^{Sign}\circ\mathbf{Mod}_{FOL}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \to \mathbf{Mod}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)$ *as follows:*

- *for all* $\mathcal{M} = \langle S, \overline{R}, \{\overline{p_i}\}_{i\in\mathcal{I}}\rangle \in |\mathbf{Mod}_{FOL}(\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle)|$*,* $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\mathcal{M}) = \langle S, \overline{R}, \ell\rangle$*, with* $\ell(p_i) = \{s \in S | \overline{p_i}(s)\}$*.*[7]
- *let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$*, then for all homomorphism* $h : \langle S_1, \overline{R_1}, \{\overline{p_{1i}}\}_{i\in\mathcal{I}}\rangle \to \langle S_2, \overline{R_2}, \{\overline{p_{2i}}\}_{i\in\mathcal{I}}\rangle \in ||\mathbf{Mod}_{FOL}(\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle)||$*, then we define* $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(h)$ *to be* $\widehat{h}$*, where* $\widehat{h}(s_1) = s_2$ *if and only if* $h(s_1) = s_2$ *for all* $s_1 \in S_1$*.*

Lemma 9. *Let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$*, then* $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ *is a functor.*

Proof. It is trivial to prove that $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ preserves identities by noting the definition of $\widehat{h}$ in terms of h.

The preservation of compositions follows by observing that, as predicates are mapped positionally, if we consider a pair of homomorphisms $h_1, h_2 \in ||\mathbf{Mod}_{FOL}(\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle)||$, the resulting homomorphism $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(h_1\circ h_2) \in ||\mathbf{Mod}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)||$ is exactly the homomorphism $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(h_1)\circ\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(h_2) \in ||\mathbf{Mod}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)||$.

Lemma 10. ρ^{Mod} *is a natural family of functors (i.e., a natural transformation).*

[7] Notice that $\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle = \rho^{Sign}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)$ where $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$.

Proof. Predicate symbols are mapped by resorting to the injective function $\sigma : \langle\{p_i\}_{i\in\mathcal{I}}\rangle \to \langle\{p'_i\}_{i\in\mathcal{I}'}\rangle \in ||\mathsf{Sign}_K||$ and, as $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ and $\rho^{Mod}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}$ maps predicates interpreting the symbols in the ρ^{Sign}-translation of the signature positionally, the reduct operations $\mathbf{Mod}_K(\sigma)$ and $\mathbf{Mod}_{FOL}(\rho^{Sign}(\sigma))$ commute with $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ and $\rho^{Mod}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}$, thus proving the naturality condition.

The next corollary follows from Lemmas 7, 8, and 10.

Corollary 2. *$\langle\rho^{Sign}, \rho^{Sen}, \rho^{Mod}\rangle$ is a map of institutions.*

Now, we have to prove that structures representing the tableaux for first-order predicate logic for properties resulting from the translation of modal logic properties can indeed be translated to modal logic tableaux for the original modal logic properties.

Fact 4 *Let $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$; for all $\alpha \in |\mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)|$, if $\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) = \beta$, then any quantified sub-formuli in β is either of the form: a) $(\forall x)(R(y,x) \implies \varphi(x))$, or b) $(\exists x)(R(y,x) \wedge \varphi(x))$.*

Then last fact above shows that whenever we are dealing with a set of formulae coming from the application of function $\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$, the resulting first-order predicate logic tableaux will have a very particular shape because the application of rule $[\forall]$ (respectively $[\exists]$) is restricted to the formulae coming from the translation.

The next definition provides the means for obtaining modal logic tableaux from first-order predicate logic tableaux. In order to simplify the following definition, we will restrict ourselves to those first-order predicate logic tableaux in which, when the rule $[\forall]$ (respectively $[\exists]$) is applied, the rules $[\vee]$ and $[\neg]$ (respectively $[\wedge]$) are applied. Notice that this assumption does not limit the definitions and results in any way because any other legal tableau for the same set of formulae that does not satisfy this property can be reordered to satisfy it.

Definition 22. *We define T, a function translating first-order logic tableaux to modal logic tableaux, as follows*[8]*:*

$$\dfrac{\overline{x}\ \textit{is a new constant}\ \dfrac{X \cup \{(\exists x)(R(\overline{y},x) \wedge P(x))\}}{X \cup \{(\exists x)(R(\overline{y},x) \wedge P(x)), R(\overline{y},\overline{x}) \wedge P(\overline{x})\}}\ [\exists]}{X \cup \{(\exists x)(R(\overline{y},x) \wedge P(x)), R(\overline{y},\overline{x}) \wedge P(\overline{x}), R(\overline{y},\overline{x}), P(\overline{x})\}}\ [\wedge]$$

$$\downarrow T$$

$$\overline{x}\ \textit{is a new label such that}\ R(\overline{y},\overline{x})\ \dfrac{X \cup \{\ell : \Diamond P\}}{X \cup \{\ell : \Diamond P, \overline{x} : P\}}\ [\Diamond]$$

$$\dfrac{\overline{x}\ \textit{is a ground term}\ \dfrac{X \cup \{(\forall x)(\neg R(\overline{y},x) \vee P(x))\}}{X \cup \{(\forall x)(\neg R(\overline{y},x) \vee P(x)), \neg R(\overline{y},\overline{x}) \vee P(\overline{x})\}}\ [\forall]}{\begin{array}{l|l} X \cup \{(\forall x)(\neg R(\overline{y},x) \vee P(x)), & X \cup \{(\forall x)(\neg R(\overline{y},x) \vee P(x)), \\ \neg R(\overline{y},\overline{x}) \vee P(\overline{x}), & \neg R(\overline{y},\overline{x}) \vee P(\overline{x}), \\ \neg R(\overline{y},\overline{x})\} & P(\overline{x})\} \end{array}}\ [\vee]$$

$$\downarrow T$$

[8] Notice that translation rules for the rules for the propositional operators take care of the labeling by just preserving them.

$$\overline{x} \text{ is a label ocurring in } X \cup \{\ell\} \text{ such that } R(\ell, \overline{x}) \; \frac{X \cup \{\ell : \Box P\}}{X \cup \{\ell : \Box P, \overline{x} : P\}} \; [\Box]$$

Definition 23.
Let $\langle\langle\{p_i\}_{i\in\mathcal{I}}\rangle, \Gamma\rangle \in |\mathsf{Th}_0^K|$, *then* $\gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle} : \rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL}(\langle\langle\{p_i\}_{i\in\mathcal{I}}\rangle, \Gamma\rangle) \to \mathbf{models}^{\mathsf{op}}_{K}(\langle\langle\{p_i\}_{i\in\mathcal{I}}\rangle, \Gamma\rangle)$ *is defined as* $\gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle) = \langle\{\alpha \in |\mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)| \mid \rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) \in \Delta\}, \langle\{p_i\}_{i\in\mathcal{I}}\rangle\rangle$.

Lemma 11. *Let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$, *then* $\gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ *is a functor.*

Proof. This lemma follows trivially by observing that the categories obtained by applying $\mathbf{models}_{FOL}$ and $\mathbf{models}_K$ are discrete, and that given $\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle$, $\langle\{\alpha \in |\mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)| \mid \rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) \in \Delta\}, \langle\{p_i\}_{i\in\mathcal{I}}\rangle\rangle$ is the canonical model obtained from a branch of the modal logic tableau resulting from the application of T to the tableau from which $\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle$ was obtained.

Lemma 12. $\gamma : \rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL} \to \mathbf{models}^{\mathsf{op}}_{K}$ *is a natural family of functors (i.e. a natural transformation).*

Proof. Let $\sigma : \langle\{p_i\}_{i\in\mathcal{I}}\rangle \to \langle\{p'_i\}_{i\in\mathcal{I}'}\rangle \in ||\mathsf{Sign}_K||$ and $\varphi : \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle \to \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle \in ||\mathsf{Sign}_{FOL}||$ such that $\rho^{Sign}(\sigma) = \varphi$ then, the naturality condition for γ can be drawn as follows:

$$\begin{tikzcd}
\rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL}(\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle) \arrow[r, "\gamma_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}"] \arrow[d, "\rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL}(\sigma^{\mathsf{op}})"] & \mathbf{models}^{\mathsf{op}}_{K}(\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle) \arrow[d, "\mathbf{models}^{\mathsf{op}}_{K}(\sigma^{\mathsf{op}})"] \\
\rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \arrow[r, "\gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}"] & \mathbf{models}^{\mathsf{op}}_{K}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)
\end{tikzcd}$$

Let $\langle\Delta, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle \in |\rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL}(\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle)|$, then, by Def. 23, we get that $\gamma_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}(\langle\Delta, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle) = \langle\{\alpha \in |\mathbf{Sen}_K(\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle)| \mid \rho^{Sen}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}(\alpha) \in \Delta\}, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle$ holds. Thus,

$$\begin{aligned}
&\mathbf{models}^{\mathsf{op}}_{K}(\sigma^{\mathsf{op}})(\langle\{\alpha \in \mathbf{Sen}_K(\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle) \mid \rho^{Sen}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}(\alpha) \in \Delta\}, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle) \\
&= \langle\{\beta \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \mathbf{Sen}_K(\sigma)(\beta) \in \\
&\qquad \{\alpha \in \mathbf{Sen}_K(\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle) \mid \rho^{Sen}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}(\alpha) \in \Delta\}\}, \langle\{p_i\}_{i\in\mathcal{I}}\rangle\rangle \\
&= \langle\{\beta \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \rho^{Sen}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}(\mathbf{Sen}_K(\sigma)(\beta)) \in \Delta\}, \langle\{p_i\}_{i\in\mathcal{I}}\rangle\rangle
\end{aligned}$$

On the other hand,

$$\begin{aligned}
&\rho^{Th_0} \circ \mathbf{models}^{\mathsf{op}}_{FOL}(\sigma^{\mathsf{op}})(\langle\Delta, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle) \\
&= \mathbf{models}^{\mathsf{op}}_{FOL}(\rho^{Th_0}(\sigma)^{\mathsf{op}})(\langle\Delta, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle) \\
&= \mathbf{models}^{\mathsf{op}}_{FOL}(\varphi^{\mathsf{op}})(\langle\Delta, \langle R', \{p'_i\}_{i\in\mathcal{I}'}\rangle\rangle) \\
&= \langle\{\alpha \in \mathbf{Sen}_{FOL}(\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle) \mid \mathbf{Sen}(\varphi)(\alpha) \in \Delta\}, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle
\end{aligned}$$

Then,

$$\begin{aligned}
&\gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\langle\{\alpha \in \mathbf{Sen}_{FOL}(\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle) \mid \mathbf{Sen}(\varphi)(\alpha) \in \Delta\}, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle) \\
&= \langle\{\beta \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\beta) \in \\
&\qquad \{\alpha \in \mathbf{Sen}_{FOL}(\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle) \mid \mathbf{Sen}_{FOL}(\varphi)(\alpha) \in \Delta\}\}, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle \\
&= \langle\{\beta \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \mathbf{Sen}_{FOL}(\varphi)(\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\beta)) \in \Delta\}, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle
\end{aligned}$$

The only property that remains to be proved is that $\rho^{Sen}_{\langle\{p'_i\}_{i\in\mathcal{I}'}\rangle}(\mathbf{Sen}_K(\sigma)(\beta)) = \mathbf{Sen}_{FOL}(\rho^{Sign}(\sigma))(\rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\beta))$. This property follows trivially by definition of ρ^{Sign}, ρ^{Sen} and, both $\mathbf{Sen}_K$ and $\mathbf{Sen}_{FOL}$.

Finally, the following lemma prove the equivalence of the two cells shown in Def. 12.

Lemma 13. *Let* $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$, *then*

$$\mu_{K\langle\{p_i\}_{i\in\mathcal{I}}\rangle} \circ \gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle} = \rho^{Mod}_{\rho^{Sign}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)} \circ \mu_{FOL\,\rho^{Sign}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)} \ .$$

Proof. To prove this property we must prove that $\mu_{K\langle\{p_i\}_{i\in\mathcal{I}}\rangle} \circ \gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}$ and $\rho^{Mod}_{\rho^{Sign}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)} \circ \mu_{FOL\,\rho^{Sign}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)}$, are the same functors.

Let $\langle\{p_i\}_{i\in\mathcal{I}}\rangle \in |\mathsf{Sign}_K|$, and $\Gamma \subseteq \mathbf{Sen}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)$. Let $\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle \in |\mathbf{models}^{op}_{FOL}(\rho^{Th_0}(\langle\langle\{p_i\}_{i\in\mathcal{I}}\, \Gamma\rangle))|$; then, $\gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle) = \langle\{\alpha \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) \in \Delta\}, \langle\{p_i\}_{i\in\mathcal{I}}\rangle\rangle$.

Thus, we obtain that $\mu_{K\langle\{p_i\}_{i\in\mathcal{I}}\rangle} \circ \gamma_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle)$ if the class of models $\mathbf{Mod}_K(\langle\langle\{p_i\}_{i\in\mathcal{I}}\rangle, \{\alpha \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) \in \Delta\}\rangle)$.

Now, as $\mu_{FOL\,\rho^{Sign}(\langle\{p_i\}_{i\in\mathcal{I}}\rangle)}(\langle\Delta, \langle R, \{p_i\}_{i\in\mathcal{I}}\rangle\rangle) = \mathbf{Mod}_{FOL}(\langle\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle, \Delta\rangle)$, it only remains to be proved that $\rho^{Mod}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\mathbf{Mod}_{FOL}(\langle\langle R, \{p_i\}_{i\in\mathcal{I}}\rangle, \Delta\rangle)) = \mathbf{Mod}_K(\langle\langle\{p_i\}_{i\in\mathcal{I}}\rangle, \{\alpha \in \mathbf{Sen}_K(\langle\{p_i\}_{i\in\mathcal{I}}\rangle) \mid \rho^{Sen}_{\langle\{p_i\}_{i\in\mathcal{I}}\rangle}(\alpha) \in \Delta\}\rangle)$.

It is easy to see that this equality holds because, on the one hand, we have the modal logic reducts of the first-order predicate logic models of the formulas in Δ and, on the other hand, we have the modal logic models of the reverse translation of the formulae in Δ. Notice that by the way in which tableaux are related through T, the formulae in Δ can be reverse translated.

The next corollary follows from Coro. 2, and Lemmas 12 and 13.

Corollary 3. $\langle\rho^{Sign}, \rho^{Sen}, \rho^{Mod}, \gamma\rangle$ *is a map of satisfiability calculi.*