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

Carlos Gustavo Lopez Pombo
University of Buenos Aires, Argentina

Joint work with:
Tomás Esteban Eduardo Maibaum -- McMaster University, Canada
Nazareno Aguirre -- Universidad Nacional de Río Cuarto, Argentina
Pablo Castro -- Universidad Nacional de Río Cuarto, Argentina
Paula Chocrón -- Universidad de Buenos Aires, Argentina

Intro (Motivation)

* As proving techniques have proof calculi as formal foundations (in the context of General Logics), semantics-based techniques also require formal foundations
* Software analysis methodologies, understood as combinations of different verification techniques, also require formal foundations

Institutions and general logics

- **’84** - Goguen and Burstall formalize the model theory of a logic in category theory by introducing the concept of **institution**
- **’89** - Meseguer extends this model theoretical view of a logic to cope with the proof theoretical aspects introducing the concepts of **proof calculus**, **proof sub-calculus** and **effective proof sub-calculus**

Institution

Definition . *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:*

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

Entailment system

Definition . *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*,* $\mathbf{Sen}(\sigma)(\Gamma) \vdash^{\Sigma'} \mathbf{Sen}(\sigma)(\phi)$.

Logic

Definition . *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* $\Sigma \in |\mathsf{Sign}|$, $\phi \in \mathbf{Sen}(\Sigma)$, $\Gamma \subseteq \mathbf{Sen}(\Sigma)$: $\Gamma \vdash^{\Sigma} \phi$ *implies* $\Gamma \models^{\Sigma} \phi$.

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

General Logics

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.*

General Logics

> *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.*

This observation is entirely correct and the result was the formalization of the notion of **proof calculus**, **proof sub-calculus** and **effective proof sub-calculus** as an “implementation” or operational view of the entailment relation.

Proof calculus

Definition . *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*
- $\pi : \mathbf{proofs} \dot{\to} \mathbf{Sen}$ *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.

General Logics

> *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 <u>invariant</u> under many equivalent proof calculi that can be used for a logic.*

This observation is entirely correct and the result was the formalization of the notion of **proof calculus**, **proof sub-calculus** and **effective proof sub-calculus** as an "implementation" or operational view of the entailment relation.

<u>What about the formal aspects of the satisfiability relation?</u>

General Logics

- Many techniques for software verification are based in the semantics of the specification language: tableau techniques, sat-based techniques and many model checking approaches
- In this case what remains invariant is the satisfaction relation
- Thus, Meseguer's argument apply in exactly the same way to satisfiability relations

Satisfiability calculus

- A **satisfiability calculus** provides an operational views of the satisfiability relation of an institution
- A **satisfiability calculus** can be understood as the semantic counterpart of a proof calculus
- As such, a **satisfiability calculus**, concentrates on the "mechanics" behind the model theoretical aspects of a logic

Satisfiability calculus

Definition . **[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 category 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*
- $\mu : \mathbf{models}^{\mathsf{op}} \dot{\to} \mathbf{Mod}$ *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*.*

Satisfiability calculus

(Example: Tableau for First Order Predicate Logic)

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 (u being an ancestor of v), then the label of v is obtained from the label of u by applying one of the following rules:

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

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

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

$$\frac{X \cup \{(\forall x)P(x)\}}{X \cup \{(\forall x)P(x), P(t)\}} [\forall] \qquad \frac{X \cup \{(\exists x)P(x)\}}{X \cup \{(\exists x)P(x), P(c)\}} [\exists]$$

where, in the last rules, c is a new constant and t is a ground term.

Satisfiability calculus

(Example: Tableau for First Order Predicate Logic)

Definition . $\mathbf{M} : \mathsf{Th}_0 \to \mathsf{Struct}_{SC}$ *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* $\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle$.

Definition 14. $\mathbf{Mods} : \mathsf{Struct}_{SC} \to \mathsf{Cat}$ *is defined as:*

$$\mathbf{Mods}(\langle Str^{\Sigma,\Gamma}, \cup, \emptyset \rangle) = \{\langle \Sigma, Cn(\widetilde{\Delta}) \rangle \mid (\exists \alpha : \Delta \to \emptyset \in ||Str^{\Sigma,\Gamma}||) \\ (\widetilde{\Delta} \to \emptyset \in \alpha \wedge (\forall \alpha' : \Delta' \to \Delta \in ||Str^{\Sigma,\Gamma}||)(\Delta' = \Delta))\}$$

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:*

$$\mathbf{Mods}(\widehat{\sigma})(\langle \Sigma, Cn(\widetilde{\Delta}) \rangle) = \langle \Sigma', Cn(\mathbf{Sen}(\sigma)(Cn(\widetilde{\Delta}))) \rangle.$$

Definition 15. *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* $\mu_\Sigma(\langle \Sigma, \Delta \rangle) = \mathbf{Mod}(\langle \Sigma, \Delta \rangle)$.

Software analysis

Proposition .
Let $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \mu \rangle$ *be a proof calculus,* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{M}, \mathbf{Mods}, \pi \rangle$ *be a satisfiability calculus,* $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$ *and* $\alpha \in \mathbf{Sen}(\Sigma)$*:*

[**Soundness**] *If there exists* $\tau \in |\mathbf{proof}(T)|$ *such that* $\pi_T(\tau) = \alpha$*, then for all* $M \in |\mathbf{models}^{\mathsf{op}}(T)|$, $\mu_T(M) \models_{\Sigma} \alpha$.

[**Completeness**] *If for all* $M \in |\mathbf{models}^{\mathsf{op}}(T)|$, $\mu_T(M) \models_{\Sigma} \alpha$*, then there exists* $\tau \in |\mathbf{proof}(T)|$ *such that* $\pi_T(\tau) = \alpha$.

Corollary .
Let $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \mu \rangle$ *be a proof calculus,* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{M}, \mathbf{Mods}, \pi \rangle$ *be a satisfiability calculus,* $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$ *and* $\alpha \in \mathbf{Sen}(\Sigma)$*, then*

If there exists $M \in |\mathbf{models}^{\mathsf{op}}(T)|$ *such that* $\mu_T(M) \not\models_{\Sigma} \alpha$*, then there is no* $\tau \in |\mathbf{proof}(T)|$ *such that* $\pi_T(\tau) = \alpha$.

Satisfiability sub-calculus

- A satisfiability sub-calculus is a restriction of a satisfiability calculus
- Provides the mechanisms for determining a sub-language (i.e. sub-category of signatures, subset of formulae and sub-category of theories) on which a particular semantics-based method can be applied

Satisfiability sub-calculus

Definition . [Satisfiability subcalculus] *A satisfiability subcalculus is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \mathsf{Sign}_0, ax, \{\models^\Sigma\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{M}, \mathbf{Mods}, \mu \rangle$ *satisfying the following conditions:*

- $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^\Sigma\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *is an institution,*
- Sign_0 *is a subcategory of* Sign *called the subcategory of admissible signatures; the restriction of the functor* $\mathbf{Sen}$ *to* Sign_0 *will be denoted by* $\mathbf{Sen}_0$*,*
- $\mathbf{ax} : \mathsf{Sign}_0 \to \mathsf{Set}$ *is a subfunctor of the functor obtained by composing* $\mathbf{Sen}_0$ *with the powerset functor, i.e., there is a natural inclusion* $\mathbf{ax}(\Sigma) \subseteq \mathcal{P}(\mathbf{Sen}(\Sigma))$ *for each* $\Sigma \in \mathsf{Sign}_0$*. Each* $\Gamma \in \mathsf{ax}(\Sigma)$ *is called a set of admissible axioms specified by* Q*. This defines a subcategory* Th_{ax} *of* Th_0 *whose objects are theories* $T = (\Sigma, \Gamma)$ *with* $\Sigma \in \mathsf{Sign}_0$ *and* $\Gamma \in \mathbf{ax}(\Sigma)$*, and whose morphisms are axiom-preserving theory morphisms* H *such that* H *is in* Sign_0*.*
- $\mathbf{M} : \mathsf{Th}_{ax} \to \mathsf{Struct}_{SC}$ *is a functor. Let* $T \in |\mathsf{Th}_{ax}|$*, 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* $T \in |\mathsf{Th}_{ax}|$*, then* $\mathbf{Mods}(\mathbf{M}(T))$ *is the set of canonical models of* T*; the composite functor* $\mathbf{Mods} \circ \mathbf{M} : \mathsf{Th}_{ax} \to \mathsf{Cat}$ *will be denoted by* **models***, and*
- $\mu : \mathbf{models}^{op} \dot{\to} \mathbf{Mod}$ *is a natural transformation such that, for each* $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_{ax}|$*, the image of* $\mu_T : \mathbf{models}^{op}(T) \to \mathbf{Mod}(T)$ *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*.*

Effective Satisfiability sub-calculus

- A satisfiability sub-calculus is a restriction of a satisfiability calculus
- Provides the mechanisms for determining a sub-language (i.e. sub-category of signatures, subset of formulae and sub-category of theories) on which a particular semantics-based method can be applied
- The sub-language determined must be organized in Spaces

Effective Satisfiability sub-calculus

Definition . **[Effective satisfiability sub-calculus]** *An* effective satisfiability subcalculus *is a structure of the form* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \mathsf{Sign}_0, \mathsf{Sen}_0, ax, \{\models^\Sigma\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{M}, \mathbf{Mods}, \mu \rangle$ *satisfying the following conditions:*

- $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^\Sigma\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *is an institution,*
- Sign_0 *is a subcategory of* Sign *called the subcategory of admissible signatures; the restriction of the functor* $\mathbf{Sen}$ *to* Sign_0 *will be denoted by* $\mathbf{Sen}_0$*,*
- $\mathbf{Sen}_0 : \mathsf{Space} \to \mathsf{Space}$ *is a functor such that* $\mathcal{U} \circ \mathsf{Sen}_0 = \mathsf{Sen} \circ J$
- $\mathbf{ax} : \mathsf{Sign}_0 \to \mathsf{Space}$ *is a sub-functor of the functor obtained by composing* $\mathbf{Sen}_0$ *with the functor* $\mathcal{P}_{fin} : \mathsf{Space} \to \mathsf{Space}$*, that sends each space to the space of its finite subsets. This defines a subcategory* Th_{ax} *of* Th_0 *whose objects are theories* $T = (\Sigma, \Gamma)$ *with* $\Sigma \in \mathsf{Sign}_0$ *and* $\Gamma \in \mathbf{ax}(\Sigma)$*, and whose morphisms are axiom-preserving theory morphisms* H *such that* H *is in* Sign_0.
- $\mathbf{M} : \mathsf{Th}_{ax} \to \mathsf{Struct}_{SC}$ *is a functor. Let* $T \in |\mathsf{Th}_{ax}|$*, then* $\mathbf{M}(T) \in |\mathsf{Struct}_{SC}|$ *is the model structure of* T*,*
- $\mathbf{Mods} : \mathsf{Struct}_{SC} \to \mathsf{Space}$ *is a functor. Let* $T \in |\mathsf{Th}_{ax}|$*, then* $\mathbf{Mods}(\mathbf{M}(T))$ *is the set of canonical models of* T*; the composite functor* $\mathbf{Mods} \circ \mathbf{M} : \mathsf{Th}_{ax} \to \mathsf{Space}$ *will be denoted by* $\mathbf{models}$*, and*
- $\mu : \mathbf{models}^{\mathsf{op}} \dot{\to} \mathbf{Mod}$ *is a natural transformation such that, for each* $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_{ax}|$*, the image of* $\mu_T : models^{\mathsf{op}}(T) \to \mathbf{Mod}(T)$ *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*.*
- *Denoting also by* $\mathbf{ax}, \mathbf{Pr}, \mathbf{P}$ *and* μ *the results of composing with* $\mathcal{U}$ *each of the above, the structure is a satisfiability subcalculus.*

Software analysis

- Software verification techniques are usually divided into two categories, **lightweight** (related to model construction or counterexample searching and usually unassisted) and **heavyweight** (related to theorem proving, usually assisted)
- Software analysis has departed long ago from having to choose one among all the tools available, specially, one of these two categories of tools
- Thus, the key is how to relate them through a methodology, with formal foundations, for software analysis

Software analysis

* A well known method for software analysis based on the time consumption and expertise required for the technique involved is:
  * a **lightweight** technique for model searching is used in order to construct a counterexample of the property:
    * if it exists then the specification and the property are refined,
    * if not we gained confidence in the specification and the property and then,
  * a **heavyweight** technique is applied in order to prove the property

Institution representation

Definition . **[Institution representation]**
Let $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models_\Sigma\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'_\Sigma\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$ *be institutions* I *and* I', *respectively. 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,*
- $\gamma^{Sen} : \mathbf{Sen} \dot{\to} \gamma^{Sign} \circ \mathbf{Sen}'$, *is a natural transformation,*
- $\gamma^{Mod} : (\gamma^{Sign})^{\mathsf{op}} \circ \mathbf{Mod}' \dot{\to} \mathbf{Mod}$, *is a natural transformation,*

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* $\mathcal{M}' \in |\mathbf{Mod}(\gamma^{Sign}(\Sigma))|$,

$$\mathcal{M}' \models_{\gamma^{Sign}(\Sigma)} \gamma^{Sen}_\Sigma(\alpha) \quad \text{iff} \quad \gamma^{Mod}_\Sigma(\mathcal{M}') \models_\Sigma \alpha \; .$$

Definition . **[Map of Institution]** *Let* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models_\Sigma\}_{\Sigma \in |\mathsf{Sign}|}\rangle$ *and* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'_\Sigma\}_{\Sigma \in |\mathsf{Sign}'|}\rangle$ *be institutions* I *and* I', *respectively, and* $\gamma = \langle \gamma^{Sign}, \gamma^{Sen}, \gamma^{Mod}\rangle : I \to I'$ *an institution representation. Then, if a functor* $\gamma^{Th} : \mathsf{Th}_0 \to \mathsf{Th}'_0$ *is* γ^{Sign}*-sensible (see [2, pp. 21]), then* $\langle \gamma^{Th}, \gamma^{Sen}, \gamma^{Mod}\rangle : I \to I'$ *is said to be a* map of institutions.

Software analysis

Theorem . *Let* $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \mu \rangle$ *be a proof calculus for the logic* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\vdash^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$, $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$ *and* $\alpha \in \mathbf{Sen}(\Sigma)$. *Let* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \mathsf{Sign}'_0, \mathsf{Sen}'_0, ax', \{\models'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}'|}, \mathbf{M}', \mathbf{Mods}', \mu' \rangle$ *be an effective satisfiability sub-calculus,* $\rho^{Th} : \mathsf{Th}_0 \to \mathsf{Th}'_{ax}$ *a functor and* $\langle \rho^{Th}, \rho^{Sen}, \rho^{Mod} \rangle$ *a map of institutions from* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *to* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'^{\Sigma}\}_{\Sigma \in |\mathsf{Sign}|} \rangle$, *then*

If there exists $M \in |\mathbf{models}^{\mathsf{op}}(T)|$, *and* $\mathcal{M} \in |\mu_{\rho^{Th}(T)}(M)|$ *such that* $\mathcal{M} \not\models_{\mathbf{Sign} \circ \rho^{Th}(T)} \rho^{Th}(\alpha)$, *then there is no* $\tau \in |\mathbf{proof}(T)|$ *such that* $\pi_T(\tau) = \alpha$.

Software analysis

Definition . **[Negation]** *An* institution $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models_\Sigma\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *is said to have* negation *if for all* $\Sigma \in |\mathsf{Sign}|$, $\varphi \in \mathbf{Sen}(\Sigma)$, $\mathcal{M} \in |\mathbf{Mod}(\Sigma)|$ *there exists a formula* $\psi \in \mathbf{Sen}(\Sigma)$ *(usually denoted as* $\neg\varphi$*) such that* $\mathcal{M} \models_\Sigma \psi$ *if and only if it is not true that* $\mathcal{M} \models_\Sigma \varphi$.

Corollary . *Let* $\langle \mathsf{Sign}, \mathbf{Sen}, \{\vdash^\Sigma\}_{\Sigma \in |\mathsf{Sign}|}, \mathbf{P}, \mathbf{Pr}, \mu \rangle$ *be a proof calculus for the logic* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\vdash^\Sigma\}_{\Sigma \in |\mathsf{Sign}|}, \{\models^\Sigma\}_{\Sigma \in |\mathsf{Sign}|} \rangle$, $T = \langle \Sigma, \Gamma \rangle \in |\mathsf{Th}_0|$ *and* $\alpha \in \mathbf{Sen}(\Sigma)$. *Let* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \mathsf{Sign}'_0, \mathsf{Sen}'_0, ax', \{\models'^\Sigma\}_{\Sigma \in |\mathsf{Sign}'|}, \mathbf{M}', \mathbf{Mods}', \mu' \rangle$ *be an effective satisfiability sub-calculus,* $\rho^{Th} : \mathsf{Th}_0 \to \mathsf{Th}'_{ax}$ *a functor and* $\langle \rho^{Th}, \rho^{Sen}, \rho^{Mod} \rangle$ *a map of institutions from* $\langle \mathsf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \{\models^\Sigma\}_{\Sigma \in |\mathsf{Sign}|} \rangle$ *to* $\langle \mathsf{Sign}', \mathbf{Sen}', \mathbf{Mod}', \{\models'^\Sigma\}_{\Sigma \in |\mathsf{Sign}|} \rangle$, *then*

If there exists $M \in |\mathbf{models}^{\mathsf{op}}(T \cup \neg\alpha)|$, *then there is no* $\tau \in |\mathbf{proof}(T)|$ *such that* $\pi_T(\tau) = \alpha$.

Software analysis

(Example: First Order Predicate Logic - Propositional Logic)

Definition 14. $\gamma^{Sign} : \mathsf{Sign}_{FOL} \to 2^{\mathcal{V}}$ *is defined as the functor such that: for all* $\Sigma \in |\mathsf{Sign}_{FOL}|$, $\gamma^{Sign}(\Sigma) = \{v_p \mid p$ *is a ground atomic formula in* $\mathbf{Sen}_{FOL}(\Sigma)\}$.

Definition 15. *Let* $n \in \mathbb{N}$ *and* $\Sigma = \langle \{P_j\}_{j \in \mathcal{J}}, \{f_k\}_{k \in \mathcal{K}} \rangle \in |\mathsf{Sign}_{FOL}|$. *Then* $\gamma^{Sen}_{\Sigma} : \mathbf{Sen}_{FOL}(\Sigma) \to \gamma^{Sign} \circ \mathbf{Sen}_{PL}(\Sigma)$ *is defined as:*

$\gamma^{Sen}_{\Sigma}(\alpha) = Tr^n_{PL}(Tr^n(\alpha))$, *where:*

- $Tr^n((\exists x)A) = \bigvee_{i=1}^{n} Tr^n(A|^{c_i}_{x})$,
- $Tr^n(A \vee B) = Tr^n(A) \vee Tr^n(B)$,
- $Tr^n(\neg A) = \neg Tr^n(A)$, *and*
- $Tr^n(P(t_1, \ldots, t_k)) = (fix(P(t_1, \ldots, t_k)))$, *for all* $P \in \{P_j\}_{j \in \mathcal{J}}$

$$, \text{ where } fix(A) = \mu_X \begin{bmatrix} \bigvee_{c \in \{c_1, \ldots, c_n\}} X|^{c}_{f(t_1, \ldots, t_k)} \wedge c = f(t_1, \ldots, t_k) \\ ;\text{ for all } f(t_1, \ldots, t_k) \in X \text{ such that} \\ t_1, \ldots, t_k \in \{c_1, \ldots, c_n\}. \end{bmatrix} (A).$$

- $Tr^n_{PL}(P(t_1, \ldots, t_k)) = v_{\text{“}P(t_1, \ldots, t_k)\text{”}}$,
- $Tr^n_{PL}(t = t') = v_{\text{“}t = t'\text{”}}$,
- $Tr^n_{PL}(A \vee B) = Tr^n_{PL}(A) \vee Tr^n_{PL}(B)$, *and*
- $Tr^n_{PL}(\neg A) = \neg Tr^n_{PL}(A)$.

Software analysis

(Example: First Order Predicate Logic - Propositional Logic)

Definition . *Let* $n \in \mathbb{N}$, $\gamma^{Sign} : \mathsf{Sign}_{FOL} \to 2^{\mathcal{V}}$ *be the functor of Def. 14 and* $\gamma^{Sen} : \mathbf{Sen}_{FOL} \to \gamma^{Sign} \circ \mathbf{Sen}_{PL}$ *be the natural family of functions of Def. 15. Then, we define* $\gamma^{\mathsf{Th}_0} : \mathsf{Th}_{FOL0} \to \mathsf{Th}_{PL0}$ *as:*

$$\gamma^{\mathsf{Th}_0}(\langle \Sigma, \Gamma \rangle) = \langle \gamma^{Sign}(\Sigma), \{\gamma_{\Sigma}^{Sen}(\alpha) | \alpha \in \Gamma\} \rangle \ .$$

Remark . The functor $\gamma^{\mathsf{Th}_0} : \mathsf{Th}_{FOL0} \to \mathsf{Th}_{PL0}$ is γ^{Sign}-sensible.

Definition . *Let* $n \in \mathbb{N}$ *and* $\Sigma = \langle \{P_j\}_{j \in \mathcal{J}}, \{f_k\}_{k \in \mathcal{K}} \rangle \in |\mathsf{Sign}_{FOL}|$. *Then we define* $\gamma_{\Sigma}^{Mod} : \gamma^{Sign} \circ \mathbf{Mod}_{PL}(\Sigma) \to \mathbf{Mod}_{FOL}(\Sigma)$ *as follows: for all* $val : \gamma^{Sign}(\Sigma) \to \{0,1\} \in |\mathbf{Mod}_{PL}(\gamma^{Sign}(\Sigma))|$, $\gamma_{\Sigma}^{Mod}(val) = \langle \mathcal{S}, \mathcal{C}, \mathcal{P}, \mathcal{F} \rangle$ *such that:*

- $\mathcal{S} = \{c_1, \ldots, c_n\}$,
- $\mathcal{P} = \{\overline{P} | P \in \{P_j\}_{j \in \mathcal{J}}\}$, *where*
  $\overline{P} = \{\langle c_1, \ldots, c_k \rangle | c_1, \ldots, c_k \in \mathcal{S}, val\left(v_{P(c_1,\ldots,c_k)}\right) = 1\}$, *and*
- $\mathcal{F} = \{\overline{f} | f \in \{f_k\}_{k \in \mathcal{K}}\}$, *where*
  $\overline{f} = \{\langle c_1, \ldots, c_k \rangle \mapsto c | c_1, \ldots, c_k, c \in \mathcal{S}, val\left(v_{c=f(c_1,\ldots,c_k)}\right) = 1\}$.

Outro (Conclusions)

- We provided formal foundations for semantics-based methods for software verification like tableau techniques, sat-based methods and many model-checkers for logical languages
- We provided formal foundations for a well-known methodology for software analysis based on the concept of proof calculus and effective satisfiability sub-calculus

Outro (Further work)

- Explore conditions under which *map of institutions* reducing the expressive power still allow modular analysis (requires not to loose morphisms)
- Study relations between structures representing proofs and canonical models in proof calculi and satisfiability calculi, respectively.

¿? & ¡!

dedicated to the memory of Comandante Hugo Chavez Frías
¡Hasta la victoria siempre, patria o muerte!