

Artificial Intelligence

A Course About Foundations



Semi-Decidability of First-Order Logic

Marco Piastra

Decidability and automation of L_{FO}

- L_{FO} is not decidable

No Turing machine can tell whether $\Gamma \models \varphi$

Are there any hopes for automating the calculus?

- L_{FO} is ***semi-decidable*** (Herbrand, 1930)

In general, Turing machine can tell (in *finite* time) that

$$\Gamma \models \varphi$$

... but not that

$$\Gamma \not\models \varphi$$

In other words, the above Turing machine, when facing the problem " $\Gamma \models \varphi$?" :

- 1) it will terminate with success if $\Gamma \models \varphi$
- 2) it might diverge if $\Gamma \not\models \varphi$

Herbrand's System

Given a universal sentence of the form:

$$\forall x_1 \forall x_2 \dots \forall x_n \varphi \quad (\text{where } \varphi \text{ does not contain quantifiers})$$

the **Herbrand's System** is the set (possibly *infinite*) of ground wffs generated by replacing the variables

$$\varphi[x_1/t_1, x_2/t_2 \dots x_n/t_n]$$

with all possible combinations of ground terms $\langle t_1, t_2 \dots t_n \rangle$ of the *signature* Σ

A term (or a wff) is ground if it does not contain *variables*

Examples:

$$H(\forall x P(x) \rightarrow Q(x)) = \{P(f(a)) \rightarrow Q(f(a)), P(g(a, b)) \rightarrow Q(g(a, b)), \dots\}$$

$$H(\forall x \forall y R(x, y)) = \{R(f(a), f(a)), R(g(a, b), f(a)), R(f(a), g(a, b)), \dots\}$$

■ **Herbrand's System** of a theory

Given a theory Φ of universal sentences, the Herbrand's system $H(\Phi)$ is the union of all Herbrand's systems for the sentences in Φ

Example:

$$\Phi = \{\varphi, \psi, \chi\}$$

$$H(\Phi) = H(\psi) \cup H(\varphi) \cup H(\chi)$$

Herbrand's Theorem

- **Herbrand's Theorem**

Given a theory of universal sentences Φ ,
 $H(\Phi)$ has a model iff Φ has a model

... but what is the usefulness of that?

*$H(\Phi)$ may well be infinite even when Φ is finite,
Furthermore, the theorem applies only to sets of universal sentences...*

Prenex normal form (PNF)

Any wff φ can be transformed into an equivalent formula of the form

$$Q_1x_1Q_2x_2 \dots Q_nx_n \psi \quad (\psi \text{ is called the } \mathbf{matrix})$$

where Q_i is either \forall or \exists and ψ does not contain quantifiers

1) Replace \rightarrow and \leftrightarrow :

$$\varphi \leftrightarrow \psi \Leftrightarrow (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$$

$$\varphi \rightarrow \psi \Leftrightarrow (\neg\varphi \vee \psi)$$

2) Push negation \neg inwards, as much as possible:

$$\neg(\varphi \wedge \psi) \Leftrightarrow (\neg\varphi \vee \neg\psi)$$

$$\neg(\varphi \vee \psi) \Leftrightarrow (\neg\varphi \wedge \neg\psi)$$

$$\neg\neg\varphi \Leftrightarrow \varphi$$

$$\neg\forall x \varphi \Leftrightarrow \exists x \neg\varphi$$

$$\neg\exists x \varphi \Leftrightarrow \forall x \neg\varphi$$

3) Move all quantifiers outwards, respecting order

CAUTION: variables *MUST* be renamed - if required - to avoid name clashes

Prenex normal form (PNF)

Any wff φ can be transformed into an equivalent formula of the form

$$Q_1x_1Q_2x_2 \dots Q_nx_n \psi \quad (\psi \text{ is called the } \mathbf{matrix})$$

where Q_i is either \forall or \exists and ψ does not contain quantifiers

Examples:

$$\begin{aligned} &\exists y (P(y) \rightarrow \forall x P(x)) \\ &\exists y \forall x (\neg P(y) \vee P(x)) \end{aligned} \quad \text{PNF}$$

$$\begin{aligned} &\exists y (\forall x P(x) \rightarrow P(y)) \\ &\exists y (\neg \forall x P(x) \vee P(y)) \\ &\exists y \exists x (\neg P(x) \vee P(y)) \end{aligned} \quad \text{PNF}$$

$$\begin{aligned} &\forall x \exists y (Q(x,y) \rightarrow P(y)) \wedge \neg \forall x P(x) \\ &\forall x \exists y (\neg Q(x,y) \vee P(y)) \wedge \exists x \neg P(x) \\ &\forall x \exists y (\neg Q(x,y) \vee P(y)) \wedge \exists z \neg P(z) \quad (\text{renaming variable}) \\ &\forall x \exists y \exists z ((\neg Q(x,y) \vee P(y)) \wedge \neg P(z)) \quad \text{PNF} \end{aligned}$$

Skolemization

In a sentence in PNF, existential quantifiers can be eliminated by extending the *signature* Σ of the *language*

Consider a sentence in PNF $Q_1x_1Q_2x_2 \dots Q_nx_n\psi$

From left to right, for each Q_ix_i of type $\exists x_i$:

- Apply to ψ the *substitution* $[x_i/k(x_1, \dots, x_j)]$ where k is a new function and x_1, \dots, x_j are the j variables of the universal quantifiers that come *before* $\exists x_i$ (k is an individual constant if $j = 0$)
- $\exists x_i$ is simply removed

Examples:

$$\exists y \forall x (\neg P(y) \vee P(x))$$

$$\forall x (\neg P(k) \vee P(x))$$

(k Skolem's constant)

$$\forall x \exists y \exists z ((\neg Q(x,y) \vee P(y)) \wedge \neg P(z))$$

$$\forall x ((\neg Q(x, k(x)) \vee P(k(x))) \wedge \neg P(m(x)))$$

($k/1$ and $m/1$ Skolem's functions)

▪ Theorem

For any sentence φ

φ has a model iff $sko(\varphi)$ (i.e. Skolemization of φ) has a model

Semi-decidability of L_{FO}

- Corollary of Herbrand's theorem

For any set of sentences Γ and sentence φ
these three statements are equivalent:

- $\Gamma \models \varphi$
- $\Gamma \cup \{\neg\varphi\}$ is not satisfiable (= it has no model)
- There exists a **finite** subset of $H(\text{sko}(\Gamma \cup \{\neg\varphi\}))$ (= Herbrand's system of the Skolemization of $\Gamma \cup \{\neg\varphi\}$) that is **inconsistent**

Therefore:

When $\Gamma \models \varphi$, a procedure that incrementally generates all the finite subsets of $H(\text{sko}(\Gamma \cup \{\neg\varphi\}))$ will certainly discover a contradiction (*in finite time*)