

## 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)

A 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

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

A *ground* term or wff  
does not contain variables

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

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

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  $\Phi$ ,  
 $H(\Phi)$  has a model iff  $\Phi$  has a model

... but what is the utility of that?

$H(\Phi)$  may well be infinite even when  $\Phi$  is finite,

Furthermore, the theorem applies only to sets of universal sentences...

# Prenex normal form (PNF)

Any wff  $\varphi$  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  $\psi$  does not contain quantifiers

Equivalences:

$$\begin{aligned} \models (\neg\forall x \varphi) &\leftrightarrow (\exists x \neg\varphi) & \models (\neg\exists x \varphi) &\leftrightarrow (\forall x \neg\varphi) \\ \models ((\forall x \varphi) \wedge \psi) &\leftrightarrow (\forall x (\varphi \wedge \psi)) & \models ((\exists x \varphi) \wedge \psi) &\leftrightarrow (\exists x (\varphi \wedge \psi)) \\ \models ((\forall x \varphi) \vee \psi) &\leftrightarrow (\forall x (\varphi \vee \psi)) & \models ((\exists x \varphi) \vee \psi) &\leftrightarrow (\exists x (\varphi \vee \psi)) \\ \models (\varphi \rightarrow (\forall x \psi)) &\leftrightarrow (\forall x (\varphi \rightarrow \psi)) & \models (\varphi \rightarrow (\exists x \psi)) &\leftrightarrow (\exists x (\varphi \rightarrow \psi)) \end{aligned}$$

However:

$$\models ((\forall x \varphi) \rightarrow \psi) \leftrightarrow (\exists x (\varphi \rightarrow \psi)) \quad \models ((\exists x \varphi) \rightarrow \psi) \leftrightarrow (\forall x (\varphi \rightarrow \psi))$$

Caution: *variables MUST be renamed, when required, in order to avoid clashes*

Examples:  $\exists y (P(y) \rightarrow \forall x P(x))$

$$\exists y \forall x (P(y) \rightarrow P(x))$$

(PNF, using  $(\varphi \rightarrow (\forall x \psi)) \leftrightarrow (\forall x (\varphi \rightarrow \psi))$ )

$$\exists y (\forall x P(x) \rightarrow P(y))$$

$$\exists y \exists x (P(x) \rightarrow P(y))$$

(PNF, using  $((\forall x \varphi) \rightarrow \psi) \leftrightarrow (\exists x (\varphi \rightarrow \psi))$ )

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

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

(Using  $(\neg\forall x \varphi) \leftrightarrow (\exists x \neg\varphi)$ )

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

(substitution  $[x/z]$ )

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

(PNF)

# Skolemization

In a sentence in PNF, existential quantifiers can be eliminated by extending the *signature*  $\Sigma$  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  $\psi$  the *substitution*  $[x_i/k(x_1, \dots, x_j)]$  where  $k$  is a new function and  $x_1, \dots, x_j$  are the variables of  $j$  the universal quantifiers that come *before*  $\exists x_i$  ( $k$  is an individual constant if  $j = 0$ )
- $\exists x_i$  is simply removed

Examples:

$$\exists y \forall x (P(y) \rightarrow P(x))$$

$$\forall x (P(k) \rightarrow P(x))$$

( $k$  Skolem's constant)

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

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

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

## ▪ Theorem

For any sentence  $\varphi$ ,

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

# Semi-decidability of $L_{PO}$

- Corollary of Herbrand's theorem

These three statements are equivalent:

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

Therefore:

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