FUNDAMENTALS OF LOGIC
NO.11 HERBRAND THEOREM

Tatsuya Hagino
hagino@sfc.keio.ac.jp

lecture URL

https://vu5.sfc.keio.ac.jp/slide/

So Far

- Propositional Logic
  - Logical connectives ($\wedge$, $\vee$, $\rightarrow$, $\neg$)
  - Truth table
  - Tautology
  - Normal form
  - Axiom and theorem
  - LK framework
  - Soundness and completeness

- Predicate Logic
  - Logical Formulas (language, term)
  - Quantifiers ($\forall x\ P(x)$, $\exists x\ P(x)$)
  - Closed formulae (bound and free variables)
  - Semantics of predicate logic (domain, interpretation, structure)
  - Valid formulae
  - Prenex formulae
  - LK framework for predicate logic
  - Soundness and completeness

Exercise: Write in Predicate Logic

- Let N, P, D be the following predicates:
  - $N(x)$ = " x is a natural number $(1, 2, 3, 4, \ldots)$. "
  - $P(x)$ = " x is a prime number"
  - $D(x, y)$ = "x is divisible by y " = "y is a divisor of x "
  - $x < y$ = " x is smaller than y ."
- Let g, l be the following functions:
  - $g(x, y)$ = "the greatest common divisor of x and y"
  - $l(x, y)$ = "the least common multiple of x and y"

- Please write the following sentences in predicate logic.
  1. A prime number is a natural number.
  2. A prime number can only be divisible by 1 and itself.
  3. There are infinitely many prime numbers. (i.e. Given a natural number, there is always a prime number which is bigger than the given one.)
  4. A prime number bigger than 2 is odd.

(cont.)

5. $g(x, y)$ is a divisor of x and y .

6. $g(x, y)$ is greater and any common divisor of x and y .

7. $l(x, y)$ is a multiple of x and y .

8. $l(x, y)$ is smaller than any multiple of x and y .

Proof in Predicate Logic

- Proof in Propositional Logic
  - There is an algorithm to determine whether a give formula is provable or not.
  - The algorithm is a finite method.
- Proof in Predicate Logic
  - There is no algorithm to determine whether a given formula is provable or not.
- Partial Algorithm
  - If a give formula is provable, the partial algorithm can show it.
  - If it is not provable, the algorithm may not show anything.
  - The algorithm may not terminate (i.e. not finite method).

Skolemization

- Prenex Normal Form
  - Any logical formula can be transformed to a formula of the form $Q_1 x_1 \cdots Q_n x_n\ A$.
  - Q_i is either ∀ or ∃.
  - A does not contain any quantifiers.

- ∀ themselves or ∃ themselves can be exchanged without changing the meaning, but ∀ and ∃ cannot be exchanged in general.
  - $\forall x \exists y\ A \not\equiv\ \exists y \forall x\ A$

$$\forall x_1 \cdots \forall x_n \exists y\ A$$

For x_1 to x_n, there exists y

- *Skolemization*
  - $\forall x_1 \cdots \forall x_n \exists y\ A$
  - y is determined by $x_1, \ldots, x_n$.
  - Write the relation as a new function f (*Skolem function*)
  - $\forall x_1 \cdots \forall x_n\ A[f(x_1, \ldots, x_n)/y]$

- **Theorem**: The satisfiability of $\forall x_1 \cdots \forall x_n \exists y\ A$ and $\forall x_1 \cdots \forall x_n\ A[f(x_1, \ldots, x_n)/y]$ is the same.

  - Note: $\forall x_1 \cdots \forall x_n \exists y\ A \not\equiv \forall x_1 \cdots \forall x_n\ A[f(x_1, \ldots, x_n)/y]$

Example of Skolemization

- Let $L(x,y) =$"x likes y" and $S(x) =$"x is an SFC student". Skolemize the following formulae:

1. $\forall x \exists y\, L(x,y)$
2. $\exists x \forall y\, L(x,y)$
3. $\exists x \exists y\, L(x,y)$
4. $\forall x \big(\forall y\, L(x,y) \rightarrow S(x)\big)$

Universal Prenex Normal Form

- By repeating Skolemization, a formula is transformed into a Prenex normal form with only universal quantifiers.
  - $\forall x_1 \cdots \forall x_n \ A$
  - A does not contain any quantifiers.
  - The satisfiability is the same as the original formula.
  - Called *universal prenex normal form*

- Furthermore, A can be converted into a conjunctive normal form.
  - $\forall x_1 \cdots \forall x_n \left(\left(L_{11} \vee \cdots \vee L_{1k_1} \right) \wedge \cdots \wedge \left(L_{m1} \vee \cdots \vee L_{mk_m} \right) \right)$
  - where L_{ij} is a literal (i.e. predicate or its negation)

- From duality, any formula can be transformed into the following form:
  - $\exists x_1 \cdots \exists x_n \left(\left(L_{11} \wedge \cdots \wedge L_{1k_1} \right) \vee \cdots \vee \left(L_{m1} \wedge \cdots \wedge L_{mk_m} \right) \right)$

Clause

- *Clause*
  - Disjunction of literals (predicate or its negation)
  - $L_1 \vee \cdots \vee L_n$
  - L_i is a predicate P or $\neg P$

- Converting a logical formula to clauses:
  1. Convert to prenex normal form
  2. Skolemize to replace existential quantifiers with functions
  3. Convert to conjunctive normal form
  4. Divide conjunctions

- The satisfiability of the original logical formula is equivalent to the satisfiability of the converted clauses.

Example

- Convert $\forall x \left(\forall y \left(P(x, y) \vee Q(y) \right) \rightarrow R(x) \right)$ to an equivalent set of clauses:

Herbrand Interpretation

- *Herbrand universe* H_L of language L
  - The set of terms of L which do not contain any variables.
  - In case L does not contain any constants, H_L is empty. To avoid this, add a constant to L before constructing H_L.

- Formal definition of *Herbrand universe*
  - $H_0 = \{c \mid c \text{ is a constant of } L\}$
  - $H_{k+1} = H_k \cup \{f(t_1, \dots, t_n) \mid f \text{ is an } n \text{ ary function in } L,\ t_1, \dots, t_n \in H_k\}$
  - $H_L = H_\infty$

- *Herbrand basis*
  - Atomic formulae with Herbrand Universe elements.
  - $\{P(t_1, \dots, t_n) \mid P \text{ is an } n \text{ ary predicate in } L,\ t_1, \dots, t_n \in H_L\}$

- *Herbrand interpretation*
  - A subset of Herbrand basis J
  - Atomic formulae in J are regarded as valid.

Herbrand Theorem

- *Herbrand structure*: $\mu = \langle H_L, J \rangle$
  - For each constant c: $c^J = c$
  - For each function symbol f: $f^J(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$
  - For each predicate symbol P: $\mu \vDash P(t_1, \cdots, t_n) \iff (t_1, \ldots, t_n) \in J$

- *Herbrand Theorem*
  - For a universal prenex normal form $\forall x_1 \cdots \forall x_n \ A$
    - A does not contain any quantifiers.
  - The followings are equivalent:
    - $\forall x_1 \cdots \forall x_n \ A$ is unsatisfiable.
    - There exists a natural number m and H_L terms $t_{i1}, \ldots, t_{in}$ $(i = 1, \ldots, m)$,

$$A[t_{11}/x_1, \ldots, t_{1n}/x_n] \land \cdots \land A[t_{m1}/x_1, \ldots, t_{mn}/x_n]$$

is unsatisfiable in any Herbrand structure $\langle H_L, J \rangle$.

Meaning of Herbrand Structure

- Herbrand Structure
  - Do not interpret the meaning of constants or function symbols, but treat them as symbols.
  - Give interpretation of predicate symbols only.

- Property
  - The interpretation of constants and function symbols are left to the interpretation of predicates.
  - For any interpretation (including interpreting constants and function symbols), we can create an interpretation in Herbrand structure.
  - In order to check the satisfiability of a logical formula, the structures can be restricted to Herbrand structures.

Applying Herbrand Theorem

- Show that $\forall x \exists y \big(P(x) \land \neg P(y)\big)$ is unsatisfiable using Herbrand theorem:
  1. Convert to universal prenex normal form: $\forall x \left(P(x) \land \neg P\big(f(x)\big)\right)$
  2. Herbrand universe: $H = \left\{c, f(c), f\big(f(c)\big), f\left(f\big(f(c)\big)\right), \ldots\right\}$
  3. First, $P(x) \land \neg P(f(x))$ with assignment of x to c is $P(c) \land \neg P(f(c))$, and it is satisfiable.
  4. Next, combine the above formula with $P(x) \land \neg P(f(x))$ with assignment of x to $f(c)$.
     $$\Big(P(c) \land \neg P\big(f(c)\big)\Big) \land \left(P\big(f(c)\big) \land \neg P\left(f\big(f(c)\big)\right)\right)$$
     There is no Herbrand interpretation which make both $\neg P\big(f(c)\big)$ and $P\big(f(c)\big)$ valid. Therefore, it is unsatisfiable.
  5. Using Herbrand theorem, $\forall x \exists y \big(P(x) \land \neg P(y)\big)$ is unsatisfiable.

- Therefore the negation of the formula is valid.
  - $\neg \forall x \exists y \big(P(x) \land \neg P(y)\big)$ is valid, i.e.
  - $\exists x \forall y \big(P(x) \to P(y)\big)$ is valid.

Herbrand Theorem for Clauses

- *Ground Instance*
  - A formula or clause without variables.

- *Herbrand Theorem for Clauses*
  - Let S be a set of clauses, the followings are equivalent:
    - S is unsatisfiable.
    - There is a finite set of ground instances of S which is unsatisfiable.

- A partial algorithm of showing A is valid:
  1. Convert $\neg A$ to a set of clauses S.
  2. Assign elements of H_0 (constants) and get ground instances S_0 and check its unsatisfiability (S_0 is a finite set).
  3. Assign elements of H_1 and get ground instances S_1 and check its unsatisfiability.
  4. Assign elements of H_2 and get ground instances S_2 and check its unsatisfiability.
  5. ...
  6. Repeat until finding H_k of which ground instances S_k is unsatisfiable.

Dual Form of Herbrand Theorem

- Since the validity of A and the satisfiability of $\neg A$ is equivalent, there is a dual form of Herbrand Theorem.

- *Herbrand Theorem (dual form)*
  - If $\exists x_1 \cdots \exists x_n\ A$ is an existential prenex normal form,
    - A does not contain any quantifiers.
  - The followings are equivalent:
    - $\exists x_1 \cdots \exists x_n\ A$ is valid.
    - There exists a natural number m and H_L terms $t_{i1}, \ldots, t_{in}$ ($i = 1, \ldots, m$), and

$$A[t_{11}/x_1, \ldots, t_{1n}/x_n] \vee \cdots \vee A[t_{m1}/x_1, \ldots, t_{mn}/x_n]$$

is valid in any Herbrand structure $\langle H_L, J \rangle$.

Summary

- Proof
  - Propositional logic has an algorithm of proving formulae, but
  - Predicate logic does not have.
- Skolemization
  - Universal prenex normal form
  - Conversion to clauses
- Herbrand Theorem
  - Herbrand universe, interpretation and structure
  - There is an partial algorithm for showing universal prenex normal form is unsatisfiable or not.