힐베르트식의 일차논리 형식화

임기정

1 구문론(syntax)

1.1 시그니처(signature)

$\mathcal{F}, \mathcal{R}, \mathcal{C}$를 각각 함수 기호들의 집합, 관계 기호들의 집합, 상수 기호들의 집합이라고 하자.

$$n_{\mathcal{F}} : \mathcal{F} \to \mathbb{N} \quad \text{and} \quad n_{\mathcal{R}} : \mathcal{R} \to \mathbb{N}$$

이 각각 함수 기호들의 arity와 관계 기호들의 arity에 대한 함수일 때, 순서쌍

$$\Sigma := \langle \mathcal{F}, \mathcal{R}, \mathcal{C}, n_{\mathcal{F}}, n_{\mathcal{R}} \rangle$$

를 <u>시그니처</u>라고 한다. Coq으로는 다음과 같이 형식화할 수 있다.

```
Section FOL_SYNTAX.

Let arity := nat.

#[projections(primitive)]
Record folSignature : Type :=
  { function_symbols : Set
  ; relation_symbols : Set
  ; constant_symbols : Set
  ; function_arities : function_symbols -> arity
  ; relation_arities : relation_symbols -> arity
  }.
```

1.2 논리항(term)과 논리식(formula)

시그니처

$$\Sigma = \langle \mathcal{F}, \mathcal{R}, \mathcal{C}, n_{\mathcal{F}}, n_{\mathcal{R}} \rangle$$

가 주어졌다고 하자. 논리항은 다음과 같이 정의된다.

(1) (개체변수) $x \in \mathbb{N}$이면 x는 논리항이다.

(2) (상수) $c \in \mathcal{C}$이면 c는 논리항이다.

(3) (함수 적용) $f \in \mathcal{F}$일 때 $n := n_{\mathcal{F}}(f)$라 하면, 논리항 $t_1, \cdots, t_n$에 대하여 $f(t_1, \cdots, t_n)$ 또한 논리항이다.

($*$) 단, 규칙 (1), (2), (3)만을 유한 번만 적용하여 얻을 수 있어야 한다.

한편, 논리식은 다음과 같이 정의된다.

(1) (원자논리식) $R \in \mathcal{R}$일 때 $n := n_{\mathcal{R}}(R)$라 하면, 논리항 $t_1, \cdots, t_n$에 대하여 $R(t_1, \cdots, t_n)$는 논리식이다.

(2) (등식) 논리항 t_1, t_2에 대하여 $t_1 \doteq t_2$는 논리식이다.

(3) (부정) 논리식 φ_1에 대하여 $\dot{\neg}\varphi_1$도 논리식이다.

(4) (함의) 논리식 φ_1, φ_2에 대하여 $\varphi_1 \dot{\rightarrow} \varphi_2$ 또한 논리식이다.

(5) (보편 양화) $y \in \mathbb{N}$일 때 논리식 φ_1에 대하여 $\dot{\forall} y \varphi_1$도 논리식이다.

($*$) 단, 규칙 (1), (2), (3), (4), (5)만을 유한 번만 적용하여 얻을 수 있어야 한다.

Coq으로는 다음과 같이 형식화할 수 있다.

```
Definition ivar : Set := nat. (* The set of individual variables *)

Context {Sigma : folSignature}.

Inductive trm : Set := (* The set of first-order terms *)
  | IVar_trm (x : ivar) : trm
  | Cnst_trm (c : Sigma.(constant_symbols)) : trm
  | Fapp_trm (f : Sigma.(function_symbols)) (ts : trms (L.(function_arities)
     f)) : trm
with trms : arity -> Set :=
  | O_trms : trms O
  | S_trms (n : arity) (t : trm) (ts : trms n) : trms (S n).

Inductive frm : Set := (* The set of first-order formulae *)
  | Atm_frm (R : Sigma.(relation_symbols)) (ts : trms (L.(relation_arities)
    R)) : frm
  | Eqn_frm (t1 : trm) (t2 : trm) : frm
  | Neg_frm (p1 : frm) : frm
  | Imp_frm (p1 : frm) (p2 : frm) : frm
  | All_frm (y : ivar) (p1 : frm) : frm.
```

Unique Readability Theorem. 임의의 well-formed formula는 그 구조가 유일하게 결정된다.

Proof. Inductive type의 data constructor로 형식화했기 때문에 성립한다. □

Abstract syntax tree로 정의하면 유일읽음성 정리는 사람이 숨을 쉬듯이 당연하게 성립한다. 따라서 형식화하지 않는다. ⊣

Depth. 논리항과 논리식에 대한 깊이(*depth*)를 정의해 두면, 써먹을 데가 꽤 있다. ⊣

Enumerability Theorem.

1. $\mathcal{F}$와 $\mathcal{C}$가 열거가능한 집합이면, 모든 논리항을 열거할 수 있다.
2. $\mathcal{F}$, $\mathcal{C}$와 $\mathcal{R}$이 열거가능한 집합이면, 모든 논리식을 열거할 수 있다.

⊣