L.14 Decidability of Predicate Logic

Predicate logic can be viewed abstractly as dealing with:

- predicate symbols (like $(P(x, y)$ etc)
- operation/function symbols (like $+$ etc)
- possibly some "constants" (like "2", "Socrates" etc)

A signature is a particular choice of some of each of these. We always have propositional logic available too.

For example, we may have the following symbols in our signature:

$$man(x), mortal(x), Socrates.$$

Here, the first two are one-variable predicates, and there is one constant: "Socrates".

Then an argument of predicate logic might be:

$$\begin{array}{c} \forall x : man(x) \to mortal(x) \\ man(Socrates) \\ \hline \therefore mortal(Socrates) \end{array}$$

This seems to be a valid argument! It says "If x is a man then x is mortal. Socrates is a man. Therefore Socrates is mortal." (Previously we formulated it in terms of propositional logic.)

We could write it as a single line as follows:

$$(\forall x : man(x) \to mortal(x)) \wedge man(Socrates) - > mortal(Socrates).$$

It seems that regardless of how we interpret the symbols, this is always true!

Another example is

$$(\neg \exists x : P(x)) \to (\forall x : \neg P(x)).$$

Again, it doesn't matter what $P(x)$ means, this is true regardless. (This time the signature consists of $P(x)$ only.)

We can give examples from mathematics too. Suppose $nat(x)$ is the predicate "x is a natural number". and $G(y, x)$ says "$y > x$". So the signature consists of $nat(x), G(x)$. What does the following mean?

$$\forall x : (nat(x) \to (\exists y : nat(y) \wedge G(y, x))).$$

It says: "for every natural number, there is a natural number bigger than it". We know this is true, but it doesn't follow from predicate logic alone (unlike the earlier examples): if we interpreted the symbols other than in the "standard" way, it might not be true any more. So we seem to have two notions of truth in predicate logic: one absolute, one relative to some particular interpretation.

We can formulate the notion of a WFF (well-formed formula) in predicate logic, just like in propositional logic. Then we can ask: which WFFs are always true, regardless of how we interpret the symbols appearing in them? Let's call such a WFF *valid*.

Above we saw two examples of valid formulas of predicate logic:

1. $(\forall x : man(x) \to mortal(x)) \wedge man(Socrates) - > mortal(Socrates)$, and
2. $(\neg \exists x : P(x)) \to (\forall x : \neg P(x))$

However the following formula is not valid:

- $\forall x : (nat(x) \to (\exists y : nat(y) \wedge G(y, x)))$.

This is because there are ways to interpret the symbols in which the above WFF becomes false. We could ask a different question: is there a way to decide if a formula involving only natural numbers with everything interpreted as normal is valid? More on this shortly.

In propositional logic, we know how to answer the validity question: a WFF is valid if and only if its truth table always gives "true". But in the richer language of predicate logic, the problem is much harder! In predicate logic, we can't use truth tables, since the universes of predicates may be too large!

But is there any algorithmic way to decide if a WFF of predicate logic is valid? No! It turns out that if such a method existed, it would provide a solution to the Halting problem as well...and we know no such solution exists. So there is no algorithm to determine whether a given formula is valid or not. So for a given signature, the language consisting of all WFFs for that signature is not recursive: membership of it is undecidable!

However, WFF validity in predicate logic is semi-decidable: for any given signature of predicate logic, there is a grammar that can generate all valid formulas of that signature. So the language of valid formulas is at least recursively enumerable. In theory, we could build a Turing Machine that generated all valid WFFs for a fixed signature, if we waited long enough...

On the plus side, propositional logic is decidable: use truth tables! On the minus side, so-called second order logic (where we can quantify over predicate symbols and functions as well) is not even semi-decidable! This is *Gödel's Incompleteness Theorem.*

Unfortunately, second order logic is needed to pin down the concept of the natural numbers precisely: it can't be done using standard (first order) predicate logic alone. We need an axiom that captures the principle of mathematical induction, and it can't be first order. Without that axiom, there will be non-standard models satisfying those axioms, not just the natural numbers. Let $s(x)$ be the successor function: $s(x) = x + 1$. Then the axiom of induction (Peano axiom) is

$$\forall P : P(0) \wedge (P(x) \to P(s(x))) \to \forall x (nat(x) \to P(x))).$$

It follows that the second order predicate logic of the natural numbers is not even semi-decidable: there is no TM that can crank out any valid formula given long enough.