Lecture #14, Nov, 4

0.0.1 Remark. We saw that *a Boolean wff, is also a 1st-order wff.* *We view Boolean formulas as abstractions of 1st-order ones*.

How is this Abstraction manifesting itself?

Well, in any given 1st-order wff we just "hide" all 1st-order features. That is, look at any wff like the following three forms as Boolean variables.

1. $t = s$
2. $\phi t_1 t_2 t_3 \ldots t_n$
3. $(\forall x)A$

Why so? You see, if you "live" inside Boolean logic, you know these configurations are "*statements*" but you *cannot say what they say*:

You do not understand the symbols.

So an inhabitant of Boolean logic can USE the above if connected with Boolean glue.

Examples.

- You see this "$x = y \to x = y \vee x = z$" as "$\boxed{x = y} \to \boxed{x = y} \vee \boxed{x = z}$" where the first a and second box is the same —say variable p— while the last one is different. You recognize a tautology!

- You see this "$x = x$" as "$\boxed{x = x}$". Just a Boolean variable. Not a tautology.
- The same goes for this "$(\forall x)x = y \to x = y$" which the Boolean citizen views as "$\boxed{(\forall x)x = y} \to \boxed{x = y}$", that is, a Boolean wff $p \to q$. Not a tautology.

Process of abstraction: We only abstract the expressions 1.–3. above in order to turn a 1st-order wff into a Boolean wff.

The three forms above are know in logic as **Prime Formulas**.

More Boolean abstraction examples:

- If A is

$$p \to x = y \lor (\forall x)\phi x \land q \; \big(\text{note that } q \textit{ is not} \text{ in the scope of } (\forall x)\big)$$

then we abstract as

$$p \to \boxed{x = y} \lor \boxed{(\forall x)\phi x} \land q \tag{1}$$

so the Boolean citizen sees

$$p \to p' \lor p'' \land q$$

If we ask "show all the prime formulas in A by boxing them" then we —who understand 1st-order language and can see inside scopes— would have also boxed ϕx above. The Boolean citizen cannot see ϕx in the scope of $(\forall x)$ so the boxing for such a person would be as we gave in (1)

- First box all prime formulas in (2) below.

$$(\forall x)(x = y \to (\forall z)z = a \lor q)$$

Here it is.

$$\boxed{(\forall x)(\boxed{x = y} \to \boxed{(\forall z)\boxed{z = a}} \lor q)}$$

Now abstract the above for Boolean inhabitants:

$$\boxed{(\forall x)(x = y \to (\forall z)z = a \lor q)}$$

They see no glue at all!

The abstraction is

$$p$$

- $x = y \to x = y$ abstracts as $\boxed{x = y} \to \boxed{x = y}$. That is, $p \to p$ —*a tautology.*

Why bother with abstractions? Well, the last example is a tautology so a Boolean citizen can prove it.

However $x = x$ and $(\forall x)x = y \to x = y$ are not tautologies and we need predicate logic techniques to settle their theoremhood. □

We can now define:

0.0.2 Definition. (Tautologies and Tautological Implications) We say that a (1st-order) wff, A, *is a tautology and write* $\models_{taut} A$, iff its *Boolean abstraction* is.

In 1st-order Logic $\Gamma \models_{\mathrm{taut}} A$ is applied to the Boolean abstraction of A and the wff in Γ.

Goes without saying that ALL the *identical* occurrences of $\boxed{\ldots}$ in $\Gamma \cup \{A\}$ will stand for the same Boolean variale.

For example, $x = y \models_{taut} x = y \vee z = v$ is correct as we see from

$$\overbrace{\boxed{x = y}}^{p} \models_{taut} \overbrace{\boxed{x = y}}^{p} \vee \overbrace{\boxed{z = v}}^{q}$$

□

Substitutions

A substitution is a *textual substitution*.

In $A[\mathbf{x} := t]$ we will replace all occurrences of a *free* $\mathbf{x}$ in A by the term t: *Find and replace*.

In $A[\mathbf{p} := B]$ we will replace all occurrences of a $\mathbf{p}$ in A by B: *Find and replace*.

0.0.3 Example. (What to avoid) Consider the substitution below

$$\Big((\exists x)\neg x = y\Big)[y := x]$$

If we go ahead with it *as a brute force "find and replace"* *asking no questions*, then we are met by a *serious problem*:

The result

$$(\exists x)\neg x = x \tag{1}$$

says *something other than* what the original formula says!

The latter says "for any choice of y-value there is a *fresh* (i.e., other than y) new x-value".

The above is true in any application of logic *where we have infinitely many objects*. For example, it is true of real numbers and natural numbers.

(1) though is *NEVER* true! It says that there is an object that is *different from itself*! □

0.0.4 Definition. (Substitution) Each of

1. In $A[\mathbf{x} := t]$ replace all occurrences of a free $\mathbf{x}$ in A by the term t: *Find and replace*.

2. In $A[\mathbf{p} := B]$ replace all occurrences of a $\mathbf{p}$ in A by B: *Find and replace*.

dictates that we do a *find and replace*.

However we *abort* the substitution 1 or 2 if it so happens that going ahead with it makes a free variable $\mathbf{y}$ of t or B bound because *t or B ended up in the scope of a $(\forall \mathbf{y})$ or $(\exists \mathbf{y})$*.

We say that the substitution is undefined and that the reason is that *we had a "free variable capture"*.

There is a variant of substution 2, above:

3. In $A[\mathbf{p} \setminus B]$ replace all occurrences of a $\mathbf{p}$ in A by B: *Find and replace*.

For technically justified reasons to be learnt later, we never abort this one, capture or not.

We call the substitutions 1. and 2. *conditional*, while the substitution 3. unconditional.

There is NO unconditional version of 1.

$[\mathbf{x} := t], [\mathbf{p} := B], [\mathbf{p} \setminus B]$ have higher priority that all connectives $\forall, \exists, \neg, \wedge, \vee, \rightarrow, \equiv$. They associate from LEFT to RIGHT that is $A[\mathbf{x} := t][\mathbf{p} := B]$ means

$$\Big(\Big(A[\mathbf{x} := t]\Big)[\mathbf{p} := B]\Big)$$

□

0.0.5 Example. Several substitutions based on Definition 0.0.4.

(1) $(y = x)[y := x]$.

The red brackets are META brackets. I need them to show the substitution applies to the whole formula.

The result is $x = x$.

(2) $\Big((\forall x)x = y\Big)[y := x]$. By 0.0.4, this is undefined because if I go ahead then x is captured by $(\forall x)$.

(3) $(\forall x)(x = y)[y := x]$. According to priorities, this means $(\forall x)\Big\{(x = y)[y := x]\Big\}$.

That is, "apply the quantifier $(\forall x)$ to $x = x$", which is all right.

Result is $(\forall x)x = x$.

(4) $\Big((\forall x)(\forall y)\phi(x, y)\Big)[y := x]$. This says

- Do $\Bigg((\forall x)\Big((\forall y)\phi(x, y)\Big)\Bigg)[y := x]$
- This is all right since y is not free in $\Big((\forall y)\phi(x, y)\Big)$ —so *not found; no replace!*

  Result is the original formula UNCHANGED.

(5) $\Big(z = a \vee (\forall x)x = y\Big)[y := x]$. *Abort*: x is captured when we attempt substitution in the subformula $(\forall x)x = y$.

(6) $\Big((\forall x)p\Big)[p \setminus x = y]$ Unconditional substitution. *Just find and replace, no questions asked!*

Result: $(\forall x)x = y$.

(7) $\Big((\forall x)p\Big)[p := x = y]$ Undefined. *x in $x = y$ will get captured if you go ahead!* $\square$

0.0.6 Definition. (Partial Generalisation) We say that B is *a partial generalisation* of A if B is formed *by adding as a PREFIX to A* zero or more strings of the form $(\forall \mathbf{x})$ for any choices whatsoever of the variable $\mathbf{x}$ —*repetitions allowed*. □

0.0.7 Example. Here is a small list of partial generalisations of the formula $x = z$:

$$x = z,$$

$$(\forall w)x = z,$$

$$(\forall x)(\forall x)x = z,$$

$$(\forall x)(\forall z)x = z,$$

$$(\forall z)(\forall x)x = z,$$

$$(\forall z)(\forall z)(\forall z)(\forall x)(\forall z)x = z.$$

□

0.1. Axioms and Rules for Predicate Logic

0.1.1 Definition. (1st-Order Axioms) These are all the partial generalisations of all the instances of the following schemata.

1. All tautologies
2. $(\forall \mathbf{x})A \to A[\mathbf{x} := t]$

Note that *we get an instance of this schema ONLY IF the substitution is not aborted*.

3. $A \to (\forall \mathbf{x})A$ —*PROVIDED* $\mathbf{x}$ *is not free in* A.
4. $(\forall \mathbf{x})(A \to B) \to (\forall \mathbf{x})A \to (\forall \mathbf{x})B$
5. $\mathbf{x} = \mathbf{x}$
6. $t = s \to (A[\mathbf{x} := t] \equiv A[\mathbf{x} := s])$

The set of all first-order axioms is named "Λ_1" —"1" for 1st-order. □

Our only INITIAL (or *Primary*) rule is **Modus Ponens**:

$$\frac{A, A \to B}{B} \qquad (MP)$$

You may think that including all tautologies as axioms is overkill.
However

1. It is customary to do so in the literature ([Tou08, Sho67, End72, Tou03])
2. After Post's Theorem we do know that every tautology is a theorem of Boolean logic. Adopting axiom one makes every tautology also a theorem of Predicate Logic outright!

   This is the easiest way to incorporate Boolean logic as a sublogic of 1st-order logic.

Bibliography

[End72] Herbert B. Enderton, *A mathematical introduction to logic*, Academic Press, New York, 1972.

[Sho67] Joseph R. Shoenfield, *Mathematical Logic*, Addison-Wesley, Reading, Massachusetts, 1967.

[Tou03] G. Tourlakis, *Lectures in Logic and Set Theory; Volume 1: Mathematical Logic*, Cambridge University Press, Cambridge, 2003.

[Tou08] ______, *Mathematical Logic*, John Wiley & Sons, Hoboken, NJ, 2008.