Constructive Logic (15-317), Spring 2022
Recitation 4: Quantifiers and Heyting Arithmetic (2022-02-09)
clogic staff

1 Quantifiers

Up to now, we have been vague about what, exactly, our atomic propositions A are representing. In order to discuss quantification, however, we need to be precise over what, exactly, we are quantifying over. We do this via a new judgment $t : \tau$, where τ is some to-be-defined type. Oftentimes, we are interested in some particular type, like the type of natural numbers or the type of Turing Machines, but the meaning of the $\exists$ and $\forall$ connectives are independent of this.
The rules for verifying these are as follows:

$$\dfrac{\begin{array}{c}\overline{a : A}\\ \vdots \\ A(a)\uparrow\end{array}}{\forall x : \tau.\, A(x)\uparrow}\ \forall\mathsf{I}^a \qquad\qquad \dfrac{\forall x : \tau.\, A(x)\downarrow \quad t : \tau}{A(t)\downarrow}\ \forall\mathsf{E}$$

$$\dfrac{t : \tau \quad A(t)\uparrow}{\exists x : \tau.\, A(x)\uparrow}\ \exists\mathsf{I} \qquad\qquad \dfrac{\exists x : \tau.\, A(x)\downarrow \quad \begin{array}{c}\overline{a : \tau} \quad \overline{A(a)\downarrow}^{\,u}\\ \vdots \\ C\uparrow\end{array}}{C\uparrow}\ \exists\mathsf{E}^{a,u}$$

By now, you should be comfortable with erasing the arrows to recover the rules defining these connectives for natural deduction. The intuition for these rules should be straightforward – to prove that some proposition $A(x)$ is true for all $x : \tau$, we should be able to derive $A(c)$ *true* for some arbitrary $c : \tau$. Similarly, we can introduce an existential by demonstrating some object satisfying the proposition.

Eliminating foralls is similarly simple. To eliminate an existential, however, we must do a little more work. If we have $\exists x : \tau.\, A(x)$, then we may not assume anything else about the witness! It must be an object of type τ, and also that it satisfies $A(x)$, but any other properties must be abstracted out, to be replaced with an arbitrary object with the known properties.

2 Examples with quantifiers

Consider predicates $A(x)$ and $B(x)$ which depend on $x : \tau$.

Task 1. Show $\forall x : \tau.\, A(x) \wedge B(x) \supset \forall x : \tau.\, A(x) \wedge \forall x : \tau.\, B(x)$ *true.*

Next, let $A(x, y)$ be a formula with two variables $x : \tau$ and $y : \sigma$.

Task 2. Show that you can "swap" an existential and universal. Do a verification proof. To prove this, show that $\exists x : \tau.\forall y : \sigma.A(x, y) \supset \forall y : \sigma.\exists x : \tau.A(x, y)$.

3 Heyting Arithmetic

Now that we have fully explored the surrounding machinery, let's try and look at a more sophisticated system of logic.

$$\dfrac{n\colon \mathsf{nat} \quad C(0)\ \mathit{true} \quad \begin{array}{c} \overline{x : \mathsf{nat}} \quad \overline{C(x)\ \mathit{true}}^{\,u} \\ \vdots \\ C(S\ x)\ \mathit{true} \end{array}}{C(n)\ \mathit{true}}\ \mathsf{natE}^{x,u}$$

The other was the *rule of primitive recursion*, which introduces a new term constructor R for each type τ:

$$\dfrac{n\colon \mathsf{nat} \quad t_0\colon \tau \quad \begin{array}{c} \overline{x : \mathsf{nat}} \quad \overline{r : \tau} \\ \vdots \\ t_s\colon \tau \end{array}}{R(n, t_0, x.\,r.\,t_s)\colon \tau}\ \mathsf{natE}^{x,r}$$

Its behaviour is captured by the following reduction rules:

$$\begin{aligned} R(0, t_0, x.\,r.\,t_s) &\Longrightarrow_R t_0, \\ R(S\ n', t_0, x.\,r.\,t_s) &\Longrightarrow_R [R(n', t_0, x.\,r.\,t_s)/r][n'/x]\,t_s. \end{aligned}$$

These rules R indicate that R describes a recursive function "$R(n)$" on the first parameter, with value t_0 when $n = 0$, and value $[R(n')/r][n'/x]t_s$ when $n = S\ n'$. This motivates the more readable *schema of primitive recursion*, where we define the function (call it "f" to avoid confusion) f by cases:

$$\begin{aligned} f(0) &= t_0, \\ f(S\ x) &= t_s(x, f(x)). \end{aligned}$$

We can recover the recursor version of the definition as follows:

$$f = (\mathsf{fn}\ n \Rightarrow R(n, t_0, x.r.t_s(x, r))).$$

3.1 All of the rules in one place!

Here are all of the Heyting arithmetic rules.

$$\frac{}{0 : \mathsf{nat}}\ \mathsf{nat}I_0 \qquad \frac{x : \mathsf{nat}}{\mathsf{s}\ x : \mathsf{nat}}\ \mathsf{nat}I_S$$

$$\frac{x : \mathsf{nat} \quad C(0)\ \mathit{true} \quad \begin{array}{c} \overline{y : \mathsf{nat}} \quad \overline{C(y)\ \mathit{true}} \\ \vdots \\ C(\mathsf{s}\ y)\ \mathit{true} \end{array}}{C(x)\ \mathit{true}}\ \mathsf{nat}E^{y,u}$$

$$\frac{}{0 = 0\ \mathit{true}} = I_{00} \qquad \frac{x = y\ \mathit{true}}{\mathsf{s}\ x = \mathsf{s}\ y\ \mathit{true}} = I_{SS}$$

$$\frac{0 = \mathsf{s}\ x\ \mathit{true}}{C\ \mathit{true}} = E_{0S} \qquad \frac{\mathsf{s}\ x = 0\ \mathit{true}}{C\ \mathit{true}} = E_{S0} \qquad \frac{\mathsf{s}\ x = \mathsf{s}\ y\ \mathit{true}}{x = y\ \mathit{true}} = E_{SS}$$

$$\frac{}{R(0, t_0, x.r.t_S) \Rightarrow_R t_0} \Rightarrow_R I_0 \qquad \frac{}{R(\mathsf{s}\ n, t_0, x.r.t_S) \Rightarrow_R [R(n, t_0, x.r.t_S)/r][n/x]t_S} \Rightarrow_R I_S$$

$$\frac{A(x)\ \mathit{true} \quad x \Rightarrow_R y}{A(y)\ \mathit{true}} \Rightarrow_R E_1 \qquad \frac{A(y)\ \mathit{true} \quad x \Rightarrow_R y}{A(x)\ \mathit{true}} \Rightarrow_R E_2$$

3.2 Working with these ideas

Task 3. The judgmental form of the principle of induction can be used to show the following more traditional formulation that uses universal quantification:

$$\forall n : \mathsf{nat}.\, C(0) \supset (\forall x : \mathsf{nat}.\, C(x) \supset C(S\ x)) \supset C(n)\ \mathit{true}.$$

Task 4. (BONUS) Prove

$$\forall n : \mathsf{nat}.\, R(n, 0, x.r.S\ (S\ r)) = R(n, n, x.r.S\ r)\ \mathit{true}$$

You may assume for the purposes of this proof that $R(n, S\ y, x.r.S\ r) \Rightarrow_R R(S\ (R(n, y, x.r.S\ r)))$ (note that while they are equivalent, neither side actually reduces to the other).
Furthermore, *although this is not a rule*, assume you may step underneath successors, as if you have a rule $\Rightarrow_R I^*$with the premise $x \Rightarrow_R y$ and conclusion $S\ x \Rightarrow_R S\ y$.