Extensions of PA

Lecture 7

Case and induction commands of CL

PA proves $x = 0 \vee \exists y\, x = y'$
this **justifies** a **case analysis** command *case* $N_1; z$ which behaves as:

$$\frac{}{z = 0 \mid z = y'}$$

Typically used when in assumptions $z \neq 0$ and we wish its predecessor.
PA proves $x + 1 = x'$ and, and hence $x = 0 \vee \exists y\, x = y + 1$
this **justifies** a **case analysis** command *case* $N; z$ which behaves as:

$$\frac{}{z = 0 \mid z = y + 1}$$

We also have N-induction rule: *ind* $N; x;$

$$\frac{}{\phi[0]* \;\Big|\; \begin{array}{c}\phi[x] \\ \phi[x+1]*\end{array}}$$

Definitional extensions with predicate symbols

$$x < y \leftrightarrow \exists z\, x + z' = y$$
$$x \leq y \leftrightarrow x < y \lor x = y$$

PA now proves

- both symbols **transitive**,
- $<$ **irreflexive**: $x \not< x$,
- $\leq$ **antisymmetric**: $x \leq y \land y \leq x \rightarrow x = y$
- $<$ **linear**: $x < y \lor x = y \lor y < x$,
- $\leq$ **linear**: $x \leq y \lor y < x$

The last two **justify** the **case rules** of

Trichotomy $\mathit{case}\ \mathit{Trich}; x, y \ \dfrac{}{x < y \mid x = y \mid x > y}$

Dichotomy $\mathit{case}\ \mathit{Dich}; x, y \ \dfrac{}{x \leq y \mid x > y}$

Induction with measure

For any formula $\phi[\vec{x}]$ and term $\mu[\vec{x}]$ PA proves the **induction with measure**

$$\forall\vec{x}(\forall\vec{y}(\mu[\vec{y}] < \mu[\vec{x}] \rightarrow \phi[\vec{y}]) \rightarrow \phi[\vec{x}]) \rightarrow \phi[\vec{x}]$$

This **justifies** the induction **rule**: *indm* $\mu[\vec{x}]$

$$\frac{}{\forall\vec{y}(\mu[\vec{y}] < \mu[\vec{x}] \rightarrow \phi[\vec{y}])}$$
$$\phi[\vec{x}]*$$

A **special** case when the measure $\mu[\vec{x}]$ is just x we have **complete induction**:

$$\forall x(\forall y(y < x \rightarrow \phi[y]) \rightarrow \phi[x]) \rightarrow \phi[x]$$

and the **rule**: *indm x*

$$\frac{}{\forall y(y < x \rightarrow \phi[y])}$$
$$\phi[x]*$$

Definitional extensions with function symbols

In arbtritrary theory T a new f by its defining axiom:

$$f(\vec{x}) = y \leftrightarrow \phi[\vec{x}, y]$$

provided T proves the **existence** and **uniqueness** conditions:

$$\forall\vec{x}\exists y\phi[\vec{x}, y]$$
$$\phi[\vec{x}, y_1] \land \phi[\vec{x}, y_2] \rightarrow y_1 = y_2$$

If PA proves the **existence** $\forall\vec{x}\exists y\phi[\vec{x}, y]$ then we can **uniquely** pick the **least number** y **such that** $\phi[\vec{x}, y]$.
Extension of PA by **minimization** adds two new axioms

$$\phi[\vec{x}, f(\vec{x})] \qquad z < f(\vec{x}) \rightarrow \neg\phi[\vec{x}, z]$$

This is done by **defining** $f(\vec{x}) = \mu_y[\phi[\vec{x}, y]]$.
CL command *use* f makes the two axioms accessible.
Minimization is **equivalent** to extending PA with

$$f(\vec{x}) = y \leftrightarrow \phi[\vec{x}, y] \land \forall z(z < y \rightarrow \neg\phi[\vec{x}, z])$$

where both the **existence** and **uniqueness** are provable.

Introduction of modified subtraction

PA proves $\forall x \forall y \exists d(y \leq x \rightarrow y + d = x)$
We can thus extend PA by minimization:

$$x \dot{-} y = \mu_d[y \leq x \rightarrow y + d = x]$$

and the two axioms are accessible as

$$y \leq x \rightarrow y + (x \dot{-} y) = x$$
$$z < x \dot{-} y \rightarrow \neg(y \leq x \rightarrow y + z = x)$$

From the last we get $x < y \rightarrow x \dot{-} y \leq z$ and then

$$x < y \rightarrow x \dot{-} y = 0$$

Introduction of division and remainder functions

PA proves

$$\forall x \forall y \exists q \exists r (y > 0 \rightarrow x = q \cdot y + r \wedge r < y)$$

This is the existence condition for two extensions by minimizations:

$$x \div y = \mu_q[y > 0 \rightarrow \exists r(x = q \cdot y + r \wedge r < y)]$$
$$x \bmod y = \mu_r[y > 0 \rightarrow \exists q(x = q \cdot y + r \wedge r < y)]$$

Towards the recursive CL definitions

In order to be able to introduce functions defined in CL by recursion we need some kind of **coding** of sequences, i.e. **lists** in PA.
Cantor's pairing function does not suffice, we need also **list concatenation**!
We can introduce a **dyadic pairing** $x; y$ **and concatenation** $x \boxplus y$ functions s.t

$$x_1; y_1 = x_2; y_2 \rightarrow x_1 = x_2 \land y_1 = y_2$$
$$x < x; y \land y < x; y$$

Defining: $Atom(x) \leftrightarrow \forall y \forall z\, x \neq y; z$
we can introduce $\boxplus$ to satisfy:

$$Atom(x) \rightarrow x \boxplus z = z$$
$$(x; y) \boxplus z = x; y \boxplus z \ .$$

Predicates of divisibility and powers of two

We explicitly define the predicate of **divisibility** $x \mid y$:

$$x \mid y \leftrightarrow \exists z\, y = z \cdot x$$

and then the predicate $Pow_2(p)$ holding iff $p = 2^x$ for some x. In absence of exponentiation (it has a recursive definition) we can define the predicate of p **is a power of two** explicitly as:

$$Pow_2(p) \leftrightarrow \forall d (d \mid p \rightarrow d = 1 \lor 2 \mid d)$$