Math Logic: Model Theory & Computability
Lecture 29

Cor. (a) Primitive recursive relations form a Boolean algebra, i.e. are closed under complements and finite unions/intersections.

(b) Primitive recursive functions are closed under definitions by cases.

Proof. HW.

Prop. The class of primitive recursive functions is closed under bounded search, i.e. if $R \subseteq \mathbb{N}^k \times \mathbb{N}$ is primitive recursive, then so is the function $f: \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ defined by $f(\vec{a}, b) := \mu_{x<b}(R(\vec{a}, x)) := \begin{cases} \mu_x(R(\vec{a},x)) & \text{if } \exists x < b\ R(\vec{a},x). \\ b & \text{otherwise} \end{cases}$

In particular, the class of primitive recursive relations is closed under bdd quantification, i.e. if $R \subseteq \mathbb{N}^k \times \mathbb{N}$ prim. rec. then so are $\exists y < z\ R(\vec{x}, y)$ and $\forall y < z\ R(\vec{x}, y)$.

Proof. $f(\vec{a}, 0) = 0$ and $f(\vec{a}, b+1) = \begin{cases} f(\vec{a}, b) & \text{if } f(\vec{a}, b) < b \\ b & \text{if } f(\vec{a}, b) = b \text{ and } R(\vec{a}, b) \\ b+1 & \text{otherwise} \end{cases}$.

For bdd quantification, it's enough to prove that $Q(\vec{x}, z) :\Leftrightarrow \exists y < z\ R(\vec{x}, y)$ is prim. rec. But $\mathbb{1}_Q(\vec{x}, z) = \mathbb{1}_<(\mu_{y<z}(R(\vec{x}, y)), z)$. □

Cor. The Gödel β function as well as all coding/decoding functions are primitive recursive.

Proof. The search operation involved in the definitions of these functions is bounded. Details left as HW. ⊠

Cor (Normal form). Every computable function $f: \mathbb{N}^k \to \mathbb{N}$ is of the form
$$f(\vec{a}) = (\mu_x(R(\vec{a}, x)))_0 = (\mu_x(g(\vec{a}, x) = 0))_0,$$

where $R \subseteq \mathbb{N}^k \times \mathbb{N}$ is primitive recursive and the search μ_x is safe, i.e. for each $\vec{a} \in \mathbb{N}^k$ there is $x \in \mathbb{N}$ with $R(\vec{a}, x)$, and $g := \overline{\mathrm{bit}} \circ \mathbb{1}_R$ is also prim. rec.

Proof. We prove by induction on the inductive definition/complexity of computable functions. First note that if f is already primitive recursive, then it is of the desired form because:

$$f(\vec{a}) = \left(\mu_x \left((x)_0 = f(\vec{a})\right)\right)_0 = \left(\mu_x \left(\mathbb{1}_{\neq}((x)_0, f(\vec{a})) = 0\right)\right)_0.$$

Since we have already shown that all basic computable functions in (C1) are prim. recursive, we're done with the base case.

For (C2), suppose that $f = g(h_1, \dots, h_\ell)$ where each $h_i : \mathbb{N}^k \to \mathbb{N}$ and $g : \mathbb{N}^\ell \to \mathbb{N}$ are computable and are of the desired form:

$$g(\vec{b}) = \left(\mu_y \left(R(\vec{b}, y)\right)\right)_0 \text{ and } h_i(\vec{a}) = \left(\mu_{x_i} \left(R_i(\vec{a}, x_i)\right)\right)_0.$$

Then
$$f(\vec{a}) = \Big(\mu_z \Big(\mathrm{lh}(z) = \ell+2 \wedge (z)_0 = \big(\overbrace{(z)_{\ell+1}}^{y}\big)_0 \wedge \forall i < \ell \big((R_i(\vec{a}, (z)_{i+1}) \text{ and } \forall x_i < (z)_{i+1} \neg R_i(\vec{a}, x_i)\big) \wedge R\big(((z)_1)_0, \dots, ((z)_\ell)_0, (z)_{\ell+1}\big) \wedge \forall y' < y \neg R\big(((z)_1)_0, \dots, ((z)_\ell)_0, y'\big)\Big)\Big)_0.$$

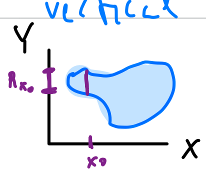
(C3) is handled even easier, if $f(\vec{a}) = \mu_x (g(\vec{a}, x) = 0)$ where $g : \mathbb{N}^{k+1} \to \mathbb{N}$ is computable and is of the form $g(\vec{a}, x) = \left(\mu_y \left(h(\vec{a}, x, y) = 0\right)\right)_0$, then

$$f(\vec{a}) = \Big(\mu_z \Big((z)_0 = \big(\underbrace{(z)_1}_{x}\big)_0 \wedge h(\vec{a}, (z)_1, (z)_2) = 0 \wedge ((z)_2)_0 = 0 \wedge \forall u < (z)_2 \; h(\vec{a}, (z)_1, u) \neq 0\Big)\Big)_0.$$

$\square$

Parameterization of computable and primitive recursive functions.

Def. For a subset $R \subseteq X \times Y$, where X, Y are sets, and $x_0 \in X$, $y_0 \in Y$, we call $R_{x_0} := \{y \in Y : (x_0, y) \in R\}$ and $R^{y_0} := \{x \in X : (x, y_0) \in R\}$ the vertical section of R at x_0 and the horizontal section of R at y_0, resp.



For a function $f: X \times Y \to Z$, where X, Y, Z are sets, and $x_0 \in X$, $y_0 \in Y$, we call the functions

$$f_{x_0} : Y \to Z, \quad y \mapsto f(x_0, y) \qquad \text{and} \qquad f^{y_0} : X \to Z, \quad x \mapsto f(x, y_0)$$

the *vertical section* of f at x_0 and the *horizontal section* of f at y_0.

Def. For a class Γ of relations on $\mathbb{N}^k$, a *parameterization* of Γ is a relation $P \subseteq \mathbb{N} \times \mathbb{N}^k$ such that for all $R \in \Gamma$, there is $c \in \mathbb{N}$ such that $R = P_c$. Similarly, for a class Γ of functions $\mathbb{N}^k \to \mathbb{N}$, a *parameterization* of Γ is a function $F : \mathbb{N} \times \mathbb{N}^k \to \mathbb{N}$ such that for each $f \in \Gamma$, there is $c \in \mathbb{N}$ such that $f = F_c$.

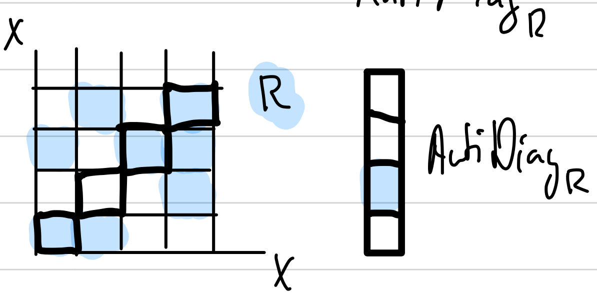
The following method due to Cantor gives a way to prove that some classes of relations/functions that are closed under "complements" do not admit a parameterization that belongs to the same class.

Diagonalization (Cantor). For any set X and any $R \subseteq X \times X$, the set

$$\mathrm{AntiDiag}_R := \{x \in X : (x, x) \notin R\}$$

is not a vertical or horizontal fiber of R, i.e. $\nexists\, x_0, y_0 \in X$ s.t.

$$\mathrm{AntiDiag}_R = R_{x_0} \quad \text{or} \quad = R^{y_0}.$$



Proof. If $\mathrm{AntiDiag}_R = R_{x_0}$ then

$$x_0 \in \mathrm{AntiDiag}_R \iff x_0 \in R_{x_0} \iff (x_0, x_0) \in R \iff x_0 \notin \mathrm{AntiDiag}_R. \qquad \square$$

We can do the same with functions:

Function Diagonalization. For any sets X, Y with distinct $y_1, y_2 \in Y$, and any function

$F: X \times X \to Y$, the function $\mathrm{AntiDiag}_F : X \to Y$, defined by $x \mapsto \begin{cases} y_1 & \text{if } F(x,x) \neq y_1 \\ y_2 & \text{otherwise} \end{cases}$,

is not a vertical or horizontal fiber of F, i.e. $\nexists\, x_0 \in X$ and $y_0 \in Y$ s.t. $\mathrm{AntiDiag}_F = F_{x_0}$ or $\mathrm{AntiDiag}_F = F^{y_0}$.

Proof. If $\mathrm{AntiDiag}_F = F_{x_0}$ then $\mathrm{AntiDiag}_F(x_0) = F(x_0, x_0) \overset{\text{by the def. of } \mathrm{AntiDiag}_F}{\neq} \mathrm{AntiDiag}_F(x_0)$. $\square$