Computability (15/11/2021)

Def: (universal function)

Given $K \geq 1$ the universal function of arity K is

$$\Psi_U : \mathbb{N}^{K+1} \to \mathbb{N}$$

$$\Psi_U (e, \vec{x}) = \varphi_e^{(K)}(\vec{x}) \qquad \text{well-defined}$$

Theorem: $\Psi_U^{(K)} : \mathbb{N}^{K+1} \to \mathbb{N}$ is computable

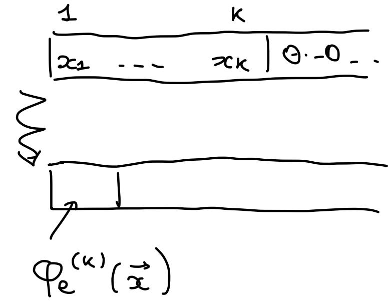
proof

given $e \in \mathbb{N}$ $\vec{x} \in \mathbb{N}^K$

we want $\Psi_U^{(K)}(e, \vec{x}) = \varphi_e^{(K)}(\vec{x})$

idea : → get the program $P_e = \gamma^{-1}(e)$

→ execute P_e



→ configuration of register

r_1 ---, r_K --- r_m	0	0	---
	$m+1$		

$$C = \prod_{i \geq 1} p_i^{r_i}$$

$$r_i = (C)_i$$

$$C_K : \mathbb{N}^{K+2} \to \mathbb{N}$$

$$C_K(e, \vec{x}, t) = \begin{cases} \text{configuration after } t \text{ steps of computation of } P_e(\vec{x}) \\ \text{(if } P_e \text{ does not stop on } \vec{x} \text{ in } t \text{ or fewer steps)} \\ \text{final configuration if } P_e(\vec{x}) \text{ stops in} \\ t \text{ or fewer steps} \end{cases}$$

program ↗ input ↑ number of steps ↖

$$J_K : \mathbb{N}^{K+2} \to \mathbb{N}$$

$$J_K(e, \vec{x}, t) = \begin{cases} \text{number of instruction to be executed after } t \\ \text{steps of } P_e(\vec{x}) \\ 0 \quad \text{if } P_e(\vec{x}) \text{ stops in } t \text{ or fewer steps} \end{cases}$$

assuming that c_k, J_k are computable

if $P_e(\vec{x})\downarrow$ then it stops in $\mu t.\ J_k(e,\vec{x},t)$ steps

$$\varphi_e^{(k)}(\vec{x}) = \left(c_k\left(e, \vec{x}, \mu t.\ J_k(e,\vec{x},t)\right)\right)_1$$

if $P_e(\vec{x})\uparrow$ then

$$\mu t.\ J_k(e,\vec{x},t)\uparrow$$

hence $\varphi_e^{(k)}(\vec{x})\uparrow = \left(c_k\left(e, \vec{x}, \mu t.\ J_k(e,\vec{x},t)\right)\right)_1$

Hence in all cases

$$\psi_U^{(k)}(e,\vec{x}) = \varphi_e^{(k)}(\vec{x}) = \left(c_k\left(e,\vec{x}, \mu t.\ J_k(e,\vec{x},t)\right)\right)_1$$

if computable $\Rightarrow$ $\psi_U^{(k)}$ is computable

AIM: prove that c_k, J_k are computable

* given $i \in \mathbb{N}$ instruction code $i = \beta(\text{Instr})$

$Z\text{arg}(i) = qt(4,i)+1$ $\qquad Z(n) \rightsquigarrow (n-1)*4$

$T\text{arg}_1(i) = \pi_1(qt(4,i))+1$ $\qquad T(m,n) \qquad \pi(m-1,n-1)*4+2$

$T\text{arg}_2(i) = \pi_2(qt(4,i))+1$

$S\text{arg}(i) = \text{- - -}$ $\qquad$ computable

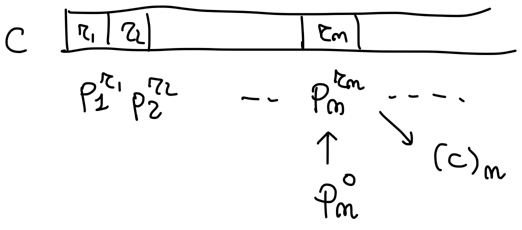
$J\text{arg}_1(i) = \text{- - -}$

$J\text{arg}_2(i) = \text{- - -}$

$J\text{arg}_2(i) = \text{- - -}$

* effect of executing algebraic instructions on a configuration

$\text{zero}(c,m) = qt\left(p_m^{(c)_m}, c\right)$



$\text{succ}(c,m) = p_m \cdot c$

$\text{transf}(c,m,n) = p_n^{(c)_m} \cdot \text{zero}(c,n)$ $\qquad$ computable

* effect on configuration of executing instruction with code i

$$\text{change}(c, i) = \begin{cases} \text{zero}(c, \text{zarg}(i)) & rm(4,i) = 0 \\ \text{succ}(c, \text{sarg}(i)) & rm(4,i) = 1 \\ \text{transf}(c, \text{Targ}_1(i), \text{Targ}_2(i)) & rm(4,i) = 2 \\ & rm(4\,i) = 3 \\ c & (\text{otherwise}) \end{cases}$$

* config. of registers starting from c, after executing instruction t of program P_e

$$\underset{\uparrow\ \ \uparrow\ \ \uparrow}{\text{next conf}\,(e, c, t)} = \begin{cases} \text{change}(c, a(e,t)) & \text{if } 1 \le t \le \ell(e) \\ c & \text{otherwise} \end{cases}$$

↖ computable

* number of next instruction if I execute $i = \beta(\text{Instr})$ and this is in position t in the program

$$\underset{\uparrow}{n}i\,(c, i, t) = \begin{cases} t+1 \swarrow & \text{if } rm(4,i) \neq 3 \text{ or } (rm(4,i) = 3 \text{ and } (c)_{\text{Jarg}_1(i)} \neq (c)_{\text{Jarg}_2(i)}) \\ \text{Jarg}_3(i) & \text{otherwise} \end{cases}$$

* next instruction, if we execute instruction t in program P_e in configuration c

$$\text{next instr}\,(e, c, t) = \begin{cases} ni(c, a(e,t), t) & \text{if } 1 \le t \le \ell(e) \text{ and } 1 \le ni(c, a(e,t), t) \le \ell(e) \\ 0 & \text{otherwise} \end{cases}$$

Now

$$c_K(e, \vec{x}, 0) = \prod_{i=1}^{K} p_i^{x_K}$$

1		K	
x_1	---	x_K	0 --

$$J_K(e, \vec{x}, 0) = 1$$

$$c_K(e, \vec{x}, t+1) = \text{next conf}\,(e, c_K(e, \vec{x}, t), J_K(e, \vec{x}, t))$$

$$J_K(e, \vec{x}, t+1) = \text{next instr}\,(e, c_K(e, \vec{x}, t), J_K(e, \vec{x}, t))$$

↑ primitive recursion of computable functions $\Rightarrow \in \mathcal{C} = \mathcal{R}$

(looking closer $\in \mathcal{PR}$)

↳ CK, JK computable

$$\Rightarrow \quad \psi_{U}^{(k)}(e, \vec{x}) = \left(CK\left(e, \vec{x}, \mu t.\ JK(e, \vec{x}, t)\right)\right)_1$$

computable ■

<u>Corollary</u>: The following predicates are decidable

(a) $H_K(e, \vec{x}, t) \equiv$ " $P_e(\vec{x})\downarrow$ in t steps or less " ↙

(b) $S_K(e, \vec{x}, \underline{y}, t) \equiv$ " $P_e(\vec{x})\downarrow y$ in t steps or less " ↙

<u>proof</u>

(a) $\chi_{H_K} : \mathbb{N}^{K+2} \to \mathbb{N}$

$$\chi_{H_K}(e, \vec{x}, t) = \begin{cases} 1 & \text{if } H_K(e, \vec{x}, t) \\ 0 & \text{otherwise} \end{cases} \qquad \text{computable}$$

$$= \overline{sg}\left(\underbrace{JK(e, \vec{x}, t)} \right)$$

↳ $\neq 0$ if $P_e(\vec{x})$ does not stop in t steps

0 otherwise

computable by composition

(b) $\chi_{S_K}(e, \vec{x}, y, t)$

$$= \chi_{H_K}(e, \vec{x}, t) \cdot \overline{sg}\left(\left| \left(CK(e, \vec{x}, t) \right)_1 - y \right| \right)$$

computable by composition

$\Rightarrow$ if $K = 1$ we omit it $H(e, x, t)$ for $H_1(e, x, t)$

* Exercises

* Given $f: \mathbb{N} \to \mathbb{N}$ ~~total~~ computable injective

then

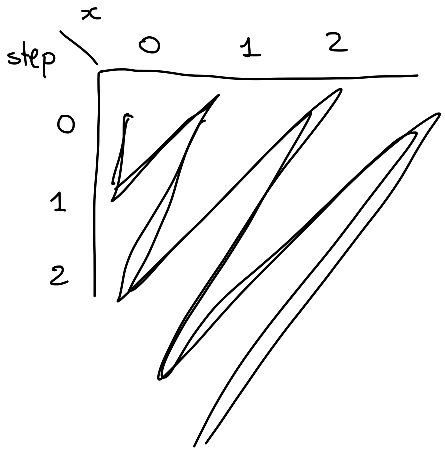
$$f^{-1} : \mathbb{N} \to \mathbb{N}$$

$$f^{-1}(y) = \begin{cases} x & \text{such that } f(x) = y \text{ if it exists} \\ \uparrow & \text{otherwise} \end{cases}$$

is computable

proof $f(y) =$ ~~$\mu x. \, |f(x) - y|$~~ □

Idea :



try n steps
on input x
for varying n , x

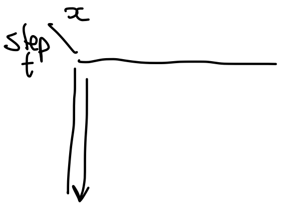
f is computable iff there exist $e \in \mathbb{N}$ program for f

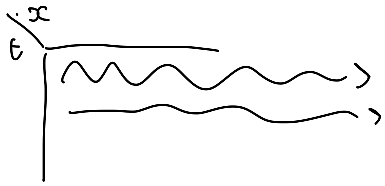
$$f = \varphi_e$$

look for x input
n number of steps s.t. $\underbrace{\varphi_e(x) \downarrow y \text{ in } t \text{ steps}}_{S(e,\, x,\, y,\, t)}$

$f^{-1}(y) =$ ~~$\mu x. \, \mu t \, . \, S(e, x, y, t)$~~

~~$\mu t. \, \mu x \, . \, S(e, x, y, t)$~~





$$= \text{“} \mu (x,t).\quad S(e,x,y,t) \text{”}$$

$$= \pi_1\left(\mu\omega.\quad S(e, \underbrace{\pi_1(\omega)}, y, \pi_2(\omega))\right)$$

$$\wr$$

$$\omega = \pi(x,t)$$

more precisely

$$f^{-1}(y) = \underset{\uparrow}{\pi_1} \left(\mu\omega. \quad \left| \underset{\uparrow}{\chi_S} (e, \pi_1(\omega), y, \pi_2(\omega)) - 1 \right| \right)$$

$$\omega$$

$$\wr$$

$$\boxed{(\omega)_1, (\omega)_2}, (\omega)_3, (\omega)_4 \; \ldots$$

$$f^{-1}(y) = \left(\mu\omega. \quad \left| \chi_S (e, \underset{\uparrow}{(\omega)_1}, y, \underset{\uparrow}{(\omega)_2}) - 1 \right| \right)_1$$

OBSERVATION : function which is total and not computable

$$f(x) = \begin{cases} \boxed{\varphi_x(x) + 1} & \text{if } \boxed{x \in W_x} \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} \varphi_U(x,x) + 1 & \text{if } \boxed{x \in W_x} \\ 0 & \text{otherwise} \end{cases}$$

problem

EXERCISE :

The predicate $\mathrm{Halt}(x) = \begin{cases} \text{true} & \text{if } x \in W_x \quad (\varphi_x(x) \downarrow) \\ \text{false} & \text{otherwise} \end{cases}$

is undecidable

EXERCISE : Totality

$Tot(x) \equiv$ "$W_x = \mathbb{N}$" $\equiv$ "φ_x is total" not decidable

proof

We want to prove that

$$\chi_{Tot}(x) = \begin{cases} 1 & \text{if } \varphi_x \text{ is total} \\ 0 & \text{otherwise} \end{cases}$$

is not computable

Assume χ_{TOT} is computable

Define

$$f(x) = \begin{cases} \varphi_x(x) + 1 & \text{if } \boxed{\varphi_x \text{ is total}} \\ 0 & \text{otherwise} \end{cases}$$

* f is total

* $\forall x$ if φ_x is total then $\varphi_x \neq f$

since $f(x) = \varphi_x(x) + 1 \neq \varphi_x(x)$

$\Rightarrow$ it is not computable

* f is computable [contradiction]

REWORK

$$f(x) \overset{NO}{\boxed{=}} \underbrace{(\varphi_x(x) + 1)}_{\text{undefined}} \cdot \underbrace{\chi_{Tot}(x)}_{0}$$

if φ_x total, $\chi_{TOT}(x) = 1$ $\rightsquigarrow$ $\varphi_x(x)+1$

if φ_x is not total $\chi_{TOT}(x) = 0$ $\rightsquigarrow$ 0

$$= (\Psi_U(x, x) + 1) \; \chi_{TOT}(x)$$

computable, by composition

$$f(x) = \Big(\mu w \,.\, \big(S(x,x,(w)_1,(w)_2) \wedge Tot(x) \wedge (w)_3 = (w)_2 + 1\big) \vee \big((w)_3 = 0 \wedge \neg Tot(x)\big)\Big)$$

<u>EXERCISE</u> : Given $Q(x)$ decidable

$f_1, f_2 : \mathbb{N} \to \mathbb{N}$ computable ~~total~~

$\Rightarrow$ $f : \mathbb{N} \to \mathbb{N}$

$$f(x) = \begin{cases} f_1(x) & \text{if } Q(x) \\ f_2(x) & \text{if } \neg Q(x) \end{cases} \qquad \text{computable}$$

$$= f_1(x) \cdot \cancel{\chi_Q(x) + f_2(x)} \cdot \chi_{\neg Q}(x)$$

complete the proof in absence of totality hypothesis. □