Calculable ⟹ récursive [Carton p 185][Wolper p 135][Lassaigne-R p 148]

(Th) Soit $f : \mathbb{N} \to \mathbb{N}$ calculable au sens strict par MT=M. Alors $\exists f_M$ une fonction récursive totale qui coïncide avec f.

<u>démo</u> • On note $M = (Q, \Sigma, \Gamma, \delta, I, F, \#)$

On peut supposer que M est déterministe à 1 seul état initial et que $\Sigma = \{1\}$

(cad entrée et sortie sont codées en base 1 : au début on a 1 en base n sur le ruban et à la fin $f(n)$ en base 1)

• On encode ensuite M

Q est identifié à $[\![0, |Q|-1]\!]$; et on note $k = |\Gamma|+1$

$I = \{0\}$; $\#$ est codé par 0 et les éléments de Γ qui st $\neq$ de $\#$ sont codés par $[\![1, |\Gamma|]\!]$

• Si (g, q, d) est une configuration de M (avec g = mot avant tête de lecture, q = état courant et d = mot après la tête de lecture large cad $g_1 \dots g_k\, d_1 \dots d_e$)

alors on représente (g, q, d) par le triplet d'entiers $(g_k, q, {}^t d_k)$

où g_k = entier dont l'écriture en base k est g ; q = codage de q ; ${}^t d_k$ = entier dont l'écriture en base k est ${}^t d$ avec ${}^t d$ = transposé de d = d renversé

On rq l'infinité de $\#$ en fin de ruban est ignorée car interprétée comme 1 infinité de zéros devant l'écriture de ${}^t d$ en base k.

<u>ex</u> $(aab, q_2, ab\#\#\dots)$ est une configuration qui devient $(14, 2, \overset{7}{\cancel{5}})$

(ici, $k = 3$; $a \to 1$; $b \to 2$, $\# \to 0$, $g = aab = 112$; $d = ab = 12$ et ${}^t d = 21$)

• On construit maintenant f_M avec des fonctions prim rec et éventuellement minimisat° non bornée mais de prédicats sûrs

* Comme F est $< \infty$, $1\!\!1_F$ est prim rec (teste l'$\in$ à F)

* Comme $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\to, \leftarrow\}$ a un ensemble de déf $< \infty$, elle est aussi prim rec (définition par cas)

* init $\begin{cases} \mathbb{N} \longrightarrow \mathbb{N}^3 \\ n \longmapsto \left(0, 0, \text{l'entier dont l'écriture en base } k \text{ est } \underbrace{1 \dots 1}_{n \text{ fois}} = \sum_{i=0}^{n-1} k^i \right) \end{cases}$

est prim rec car c'est le cas de l'exponentiation et de la somme.

Elle construit 1 représentation de l'état initial $(g = \varepsilon,\ q = 0,\ d = n)$

* transition : $\begin{cases} \mathbb{N}^3 \longrightarrow \mathbb{N}^3 \\ (g, q, d) \longmapsto \begin{cases} \text{on note } \delta(q, \overbrace{d \bmod k}^{= 1^{\text{ère}} \text{ lettre de } d = \text{celle sous tête lecture}}) = (q', s', dir) \\ \text{si } dir = \rightarrow \text{ alors } \left(g \times k + s',\ q',\ \frac{d}{k}\right) \\ \text{si } dir = \leftarrow \text{ alors } \left(\frac{g}{k},\ q',\ d \times k + s'\right) \end{cases} \end{cases}$

si on va à droite, on ajoute s' au mot d et on retire la 1ère lettre de d

est rec prim car div eucl, mult, somme, δ (f def par cas) le sont

Elle calcule la config obtenue en 1 pas à partir de sa config en argument

$n\text{-transit}^{\circ}(x, i) = \begin{cases} x \text{ si } i = 0 \\ \text{transition}(n\text{-transit}^{\circ}(x, i-1)) \text{ sinon} \end{cases}$

* n_transitions : $\begin{cases} \mathbb{N}^4 \longrightarrow \mathbb{N}^3 \\ ((g, q, d), i) \longrightarrow R_{f,g}((g, q, d), i) \text{ avec } f = id \text{ et } g : c, n, k \rightarrow \text{transition}(k) \end{cases}$

est rec prim et calcule la config obtenue en i pas à partir de la config (g, q, d)

* arrêt $\begin{cases} \mathbb{N}^3 \longrightarrow \mathbb{N} \\ c = (g, q, d) \longrightarrow \mu i \left(\mathbb{1}_F \left(\pi_2^3 (n\text{-transitions}(c, i)) \right) \right) \end{cases}$

renvoie le nb d'étapes nécessaires à l'arrêt de M sur l'entrée c. Elle est rec totale car le prédicat est sûr car M s'arrête toujours (/ def Turing calculable)

* somme $\begin{cases} \mathbb{N} \rightarrow \mathbb{N} \\ n \longmapsto \mu i \leq n \ (k^i > n) \end{cases}$ est rec prim et renvoie l'entier i = nb de symboles dans l'écriture en base k de n (ex $k = 2$, $n = 6 = 110 \rightarrow i = 3$ car $8 = 2^3 > 6 > 2^2$)

On va s'en servir car à la fin, on a des entiers écrits en base k qui sont l'écriture en base k de $f(n) = 1 \dots 1$ sur le ruban. Pour obtenir $f(n)$, il suffit donc de compter les 1 card K taille de e

- $f_M : n \mapsto \underbrace{\text{somme}\left(\pi_1^3 \left(\overbrace{n\text{-transitions}(\text{init}(n), \underbrace{\text{arrêt}(\text{init}(n))}_{\text{calcule en combien d'étapes on s'arrête sur init}(n)})}^{\text{effectue tres les transit}^{\circ} \text{ nécessaires}} \right) \right)}_{\text{extrait le } g \text{ final et somme le traduit}} + \underbrace{\text{somme}\left(\pi_3^3 \left(n\text{-transitions}(\text{init}(n), \text{arrêt}(\text{init}(n))) \right) \right)}_{\text{idem mais pour } d \text{ final}}$ est rec totale et coïncide avec f

(2) rec totale $\Rightarrow$ Turing calculable se fait en construisant des machines pour 0, S et π_i^j puis par induct° pour la composit° récurs° minimisat°