Guía 8: Batallas entre paradigmas

- En esta guía se probará que cada uno de los paradigmas vistos *vence* al otro (en el sentido de que incluye por lo menos todas las funciones que el otro contiene en su modelización del concepto de función Σ-efectivamente computable).
  - Esto nos dirá que los tres son equivalentes

Neumann vence a Godel

- *Teorema*: Si h es Σ-recursiva entonces es Σ-computable.
- *Corolario*: Si

$$\begin{gathered} f : D_f \subseteq \omega^n \times \Sigma^{*m} \to \omega \\ g : D_g \subseteq \omega^n \times \Sigma^{*m} \to \Sigma^* \\ P : D_P \subseteq \omega^n \times \Sigma^{*m} \to \{0,1\} \end{gathered}$$

  son Σ-r. entonces hay macros

$$\begin{aligned} &[V\overline{n+1} \leftarrow f(V1,..,V\bar{n},W1,..,W\bar{m})] \\ &[W\overline{m+1} \leftarrow g(V1,..,V\bar{n},W1,..,W\bar{m})] \\ &[\text{IF } P(V1,..,V\bar{n},W1,..,W\bar{m}) \text{ GOTO } A1] \end{aligned}$$

  - Esto quiere decir que hay macros para todas las funciones Σ-mixtas y predicados Σ-mixtos que hemos trabajado hasta el momento en la materia, y que todos eran Σ-p.r.
- *Lemma*: Si $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$ son Σ-enumerables, entonces $S_1 \cup S_2$ y $S_1 \cap S_2$ son Σ-enumerables.
- *Lemma*: Si $S \subseteq \omega^n \times \Sigma^{*m}$ es Σ-computable, entonces S es Σ-enumerable.

Godel vence a Neumann

- *Consideraciones*:
  - Sean $n, m \in \omega$, definimos las siguientes funciones:

$$\begin{gathered} i^{n,m} : \omega \times \omega^n \times \Sigma^{*m} \times Pro^\Sigma \to \omega \\ E_\#^{n,m} : \omega \times \omega^n \times \Sigma^{*m} \times Pro^\Sigma \to \omega^{[N]} \\ E_*^{n,m} : \omega \times \omega^n \times \Sigma^{*m} \times Pro^\Sigma \to \Sigma^{[N]} \end{gathered}$$

    de modo que $(i^{n,m}(t,\vec{x},\vec{\alpha},\mathcal{P}), E_\#^{n,m}(t,\vec{x},\vec{\alpha},\mathcal{P}), E_*^{n,m}(t,\vec{x},\vec{\alpha},\mathcal{P}))$ es la descripción instantánea que se obtiene luego de correr $\mathcal{P}$ una cantidad t de pasos partiendo del estado $||x_1,..,x_n,\alpha_1,..,\alpha_m||$.
  - Definimos también las funciones

$$E^{n,m}_{\#j} : \omega \times \omega^n \times \Sigma^{*m} \times Pro^\Sigma \to \omega$$
$$E^{n,m}_{*j} : \omega \times \omega^n \times \Sigma^{*m} \times Pro^\Sigma \to \Sigma^*$$

que marcan el valor de la j-ésima componente de $E^{n,m}_{\#}$ y $E^{n,m}_{*}$, respectivamente.
  - *Proposición*: Sean $n, m \geq 0$, $i^{n,m}$, $E^{n,m}_{\#j}$ y $E^{n,m}_{*j}$ son $(\Sigma \cup \Sigma_p)$-p.r. para todo $j \in N$.
- *Consideraciones para $Halt^{n,m}$* y $T^{n,m}$
  - *Definición*: Dados $n, m \in \omega$, definimos $Halt^{n,m} = \lambda t\vec{x}\vec{\alpha}\mathcal{P}[i^{n,m}(t, \vec{x}, \vec{\alpha}, \mathcal{P}) = n(\mathcal{P}) + 1]$
    - Básicamente, $Halt^{n,m}$ es un predicado que dice si $\mathcal{P}$ se detiene luego de t pasos partiendo del estado $||x_1, .., x_n, \alpha_1, .., \alpha_m||$.
  - *Lema*: Pro^Σ es un conjunto $(\Sigma \cup \Sigma_p)$-p.r., y $\lambda\mathcal{P}[n(\mathcal{P})]$ y $\lambda i\mathcal{P}[I^{\mathcal{P}}_i]$ son funciones $(\Sigma \cup \Sigma_p)$-p.r.
  - *Lema*: $Halt^{n,m}$ es $(\Sigma \cup \Sigma_p)$-p.r.
  - *Definición*: Definimos $T^{n,m} = M(Halt^{n,m})$
    - $D_{T^{n,m}} = \{(\vec{x}, \vec{\alpha}, \mathcal{P}) : \mathcal{P} \text{ se detiene partiendo de } ||x_1, .., x_n, \alpha_1, .., \alpha_m||\}$
    - Para $(\vec{x}, \vec{\alpha}, \mathcal{P}) \in D_{T^{n,m}}$, $T^{n,m}(\vec{x}, \vec{\alpha}, \mathcal{P})$ indica la cantidad de pasos necesarios para que $\mathcal{P}$ se detenga partiendo de $||x_1, .., x_n, \alpha_1, .., \alpha_m||$.
  - *Lema*: $T^{n,m}$ es $(\Sigma \cup \Sigma_p)$-recursiva
  - *Proposición*: $T^{n,m}$ no es $(\Sigma \cup \Sigma_p)$-p.r.
    - *Consecuencia*: La minimización de un predicado $(\Sigma \cup \Sigma_p)$-p.r. no necesariamente es $(\Sigma \cup \Sigma_p)$-p.r.
- *Teorema*: Si $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$ (con $O \in \{\omega, \Sigma^*\}$) es Σ-computable, entonces f es Σ-recursiva.

Godel vence a Turing

- *Teorema*: Si $f : S \subseteq \omega^n \times \Sigma^{*m} \to O$ es Σ-Turing computable, entonces f es Σ-recursiva.

Turing vence a Neumann

- *Teorema*: Si $f : S \subseteq \omega^n \times \Sigma^{*m} \to O$ es Σ-computable, entonces f es Σ-Turing computable.

La tesis de Church

- *Teorema*: Sea Σ un alfabeto finito, dada una función f, las siguientes son equivalentes:
  - f es Σ-Turin computable
  - f es Σ-recursiva
  - f es Σ-computable
- *Teorema*: Sea Σ un alfabeto finito, dado un conjunto $S \subseteq \omega^n \times \Sigma^{*m}$, las siguientes son equivalentes:
  - S es Σ-Turing enumerable

    - S es Σ-recursivamente enumerable
    - S es Σ-enumerable
- *Teorema*: Sea Σ un alfabeto finito, dado un conjunto $S \subseteq \omega^n \times \Sigma^{*m}$, las siguientes son equivalentes:
    - S es Σ-Turing computable
    - S es Σ-recursivo
    - S es Σ-computable
- *TESIS DE CHURCH*: Toda función Σ-efectivamente computable es Σ-recursiva.