Computability theory

FAMAF - UNC

SLP

1 Computational paradigms

Pro^{Σ} is a $(\Sigma \cup \Sigma_p)$-p.r. Since each $\mathcal{P} \in Pro^{\Sigma}$ corresponds to a Σ-recursive function, this entails $\mathcal{R}$ is Σ-p.r. Which means there is an enumeration $\varphi_1, \varphi_2, \ldots$ such that every Σ-recursive function f satisfies that $f = \varphi_i$ for some $i \in \mathbb{N}$.

2 Halting function

Recall that we say that $\mathcal{P}$ halts with input $\overrightarrow{s}, \overrightarrow{p}$ in t steps when:

$$S^t_{\mathcal{P}}(1, \overrightarrow{s}, \overrightarrow{p}) = \left(n(\mathcal{P}) + 1, \overrightarrow{x}, \overrightarrow{w}\right)$$

for any $\overrightarrow{x}, \overrightarrow{w}$. Recall too that, if $\mathcal{P}$ halts with input $\omega^n \times \Sigma^{*m}$, we use $\Psi^{n,m,\#}_{\mathcal{P}}$ to denote the value of $N1$ in the halting state. This lead to our definition of Σ-computability, where we said that f $(n, m, \#)$ is Σ-computable if and only if there is a program $\mathcal{P}$ such that

$$f = \Psi^{n,m,\#}_{\mathcal{P}}$$

(and analogously for the alphabetic case). The computability of sets was analogously defined, with S being Σ-computable if and only if $\chi_S^{\omega^n \times \Sigma^{*m}}$ is Σ-computable. With regards to Σ-enumerability, we say a set $S \subseteq \omega^n \times \Sigma^{*m}$ is Σ-enumerable if it is empty or if there is a functon $F : \omega \to \omega^n \times \Sigma^{*m}$ such that $I_F = S$ and $F_{(i)}$ is Σ-computable. In other words, if there are programs $\mathcal{P}_1, \ldots, \mathcal{P}_{n+m}$ such that

$$S = \operatorname{Im}\left[\Psi^{1,0,\#}_{\mathcal{P}_1}, \ldots, \Psi^{1,0,\#}_{\mathcal{P}_n}, \Psi^{1,0,*}_{\mathcal{P}_{n+1}}, \ldots, \Psi^{1,0,*}_{\mathcal{P}_{n+m}}\right]$$

and the domain of each $\Psi_{\mathcal{P}^{n,m,z}}$ is ω.

Given $n, m \in \omega$, we define:

$$\mathrm{Halt}^{n,m} = \lambda t \overrightarrow{x} \overrightarrow{\alpha} \mathcal{P} \left[i^{n,m}(t, \overrightarrow{x}, \overrightarrow{\alpha}, \mathcal{P}) = n(\mathcal{P}) + 1\right]$$