

Computadoras libres de prefijos

Las *computadoras libres de prefijos*, en inglés *prefix-free machines* o *self-delimiting machines*, son un subconjunto propio de las funciones parcialmente computables.

Definición. Un conjunto $A \subseteq \mathcal{A}^*$ es libre de prefijos si y sólo si para toda $v, w \in \mathcal{A}^*$, si $w \in A$ y $v <_{\text{pref}} w$ entonces $v \notin A$.

Definición. Una función $M : \mathcal{A}^* \rightarrow \mathcal{A}^*$ es una computadora libre de prefijos si M es una función parcialmente computable y el dominio de M es un conjunto libre de prefijos.

Chaitin en su artículo [1] da además una definición mecánica de estas computadoras, que tienen como característica principal que la cinta de entrada no contiene un símbolo distinguido o 'blanco' para delimitar la entrada.

Usando solamente la definición abstracta veamos que al restringirnos a computadoras libres de prefijos no sacrificamos a ningún algoritmo. Es decir, con estas computadoras podemos calcular cualquier función parcialmente computable. Formalmente:

Proposición. Las computadoras libres de prefijos computan todas las funciones parcialmente computables: existe una inyección computable de todas las funciones parcialmente computables en las que tienen dominio libre de prefijos.

Demostración. Fijemos $\psi : \mathcal{A}^* \rightarrow \mathcal{A}^*$ cualquier función computable total inyectiva cuya imagen sea un conjunto libre de prefijos. Por ejemplo, sea $\psi(a_1 a_2 \dots a_n) = a_1 a_1 a_2 a_2 \dots a_{n-1} a_{n-1} a_n \bar{a}_n$, donde \bar{a} es el mayor símbolo del alfabeto en el orden lexicográfico distinto de a . Para cada función parcialmente computable $f : \mathcal{A}^* \rightarrow \mathcal{A}^*$ sea $g_f : \mathcal{A}^* \rightarrow \mathcal{A}^*$ tal que para cada $s \in \mathcal{A}^*$, $s \in \text{dominio}(f)$ si y sólo si $\psi(s) \in \text{dominio}(g_f)$, y $g_f(\psi(s)) = f(s)$. Dado que ψ es totalmente computable, ψ^{-1} también; luego usando que f es parcialmente computable concluimos que g_f también lo es. Y claramente $\text{dominio}(g_f)$ es un conjunto libre de prefijos porque es un subconjunto de la imagen de ψ . \square

Otra manera de expresar el mismo argumento de esta demostración es mediante la noción de *reducción* entre conjuntos. Recordemos (de lo visto en "Lógica y Computabilidad") que una función $r : \mathcal{A}^* \rightarrow \mathcal{A}^*$ es una *reducción* de un conjunto A a un conjunto B si r es una función total tal que para todo $s \in \mathcal{A}^*$, $s \in A$ si y solo si $r(s) \in B$. Exhibiendo una función inyectiva ψ que mapea \mathcal{A}^* en un conjunto libre de prefijos estamos dando una reducción uno-a-uno, del conjunto \mathcal{A}^* a un conjunto libre de prefijos. Para cada función parcialmente computable f , definimos la función $g_f(\psi(s)) = f(s)$, y debemos ver que $\text{grafo}(f) = \{(s, f(s)) \mid s \in \mathcal{A}^*\}$ es reducible de manera uno-a-uno al $\text{grafo}(g_f) = \{(\psi(s), g_f(\psi(s))) \mid s \in \mathcal{A}^*\}$. La función $r((s, t)) = (\psi(s), t)$ hace la reducción ya que, si $(s, t) \in \text{grafo}(f)$ entonces $t = f(s)$, por lo tanto $r((s, t)) = (\psi(s), t) = (\psi(s), f(s)) = (\psi(s), g_f(\psi(s))) = (\psi(s), f(s)) \in \text{grafo}(g_f)$. Si $(s, t) \notin \text{grafo}(f)$ entonces $t \neq f(s)$, por lo tanto $r((s, t)) = (\psi(s), t)$, con $t \neq g_f(\psi(s))$, por lo que $r((s, t)) \notin \text{grafo}(g_f)$.

Notación. Dada una función $f : \mathcal{A}^* \rightarrow \mathcal{A}^*$ parcialmente computable definimos la función $f_t : \mathcal{A}^* \times \mathbb{N} \rightarrow \mathcal{A}^*$ que da el resultado de computar t pasos de la función f .

Observemos que si $f(s)$ converge en t pasos, entonces para todo $u \geq t$, $f_u(s) = f_t(s) = f(s)$.

Proposición. El conjunto de computadoras libres de prefijos es computablemente enumerable.

Demostración. Debemos dar una enumeración computable $(M_i)_{i \in \mathbb{N}}$ de todas las funciones parcialmente computables con dominio libre de prefijos. Definiremos cada M_i como una modificación de la i -ésima función parcialmente computable de la enumeración $(\varphi_i)_{i \in \mathbb{N}}$ de todas las funciones parcialmente computables que fijamos de antemano. Cuando la función $\varphi_d(s) = x$ converge en t pasos, declaramos $M_{d,t}(s) = x$, a menos que $M_{d,u}(z) = x$ haya sido definida en un paso anterior $u < t$ para alguna palabra $z <_{\text{pref}} s$ o $s <_{\text{pref}} z$. Claramente M_d tiene dominio libre de prefijos y $M_d = f_d$ en caso de que f_d ya tuviera dominio libre de prefijos. \square

Teorema. Hay una computadora libre de prefijos universal y óptima.

Demostración. Fijemos $(M_i)_{i \in \mathbb{N}}$ una enumeración computable de todas las computadoras libres de prefijos. Definimos $U : \mathcal{A}^* \rightarrow \mathcal{A}^*$, $U(0^i 1p) = M_i(p)$. La definición garantiza que U es universal, ya que puede realizar el cómputo de cada M_i . La optimalidad de U también es directa de la definición ya que para cada computadora M_i , $\min\{|p| \mid U(p) = s\} \leq \min\{|p| \mid M_i(p) = s\} + i + 1$. Y claramente $\text{dominio}(U) = \bigcup_{i \geq 0} \{0^i 1p : p \in \text{dominio}(M_i)\}$ es un conjunto libre de prefijos. \square

Referencias

- [1] Gregory Chaitin, A theory of program size formally identical to information theory, *Journal of the ACM* 22, 329-340, 1975.
- [2] Rod Downey, Denis Hirschfeldt. *Algorithmic Randomness and Complexity*, Springer, 2010.
- [3] Andre Nies. *Computability and Randomness*, Oxford University Press, 2009.