

Capítulo IV: FUNCIONES RECURSIVAS

IV.4: FUNCIONES RECURSIVAS

Mario de J. Pérez Jiménez
Grupo de investigación en Computación Natural
Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

Lógica Matemática

Curso 2010-11



Minimización no acotada de funciones (I)

Sea $f : \mathbb{N}^{k+1} - \rightarrow \mathbb{N}$, ($k \geq 1$). La función definida por μ -recursión a partir de f es la función $f_\mu : \mathbb{N}^k - \rightarrow \mathbb{N}$ dada por

$$f_\mu(\vec{x}) = \begin{cases} \min\{y : f(\vec{x}, y) = 0 \wedge \forall z < y (f(\vec{x}, z) \downarrow)\} & \text{si existe} \\ \uparrow & \text{e.o.c.} \end{cases}$$

Notaremos $f_\mu(\vec{x}) = \mu y (f(\vec{x}, y) = 0)$.

- ▶ Si una función es de aridad $k + 1$, entonces f_μ es de aridad k .
- ▶ En la definición se ha considerado el menor valor de la función que vale 0. En lugar de 0 se podría haber elegido cualquier otro número natural.
- ▶ En la definición se ha considerado el menor valor respecto de la última componente. Se podría haber considerado cualquier otra componente.
- ▶ La condición $\forall z < y (f(\vec{x}, z) \downarrow)$ es esencial para que la construcción de f_μ sea mecánica.

Minimización no acotada de funciones (II)

- ▶ Existen funciones **totales** cuya función definida por μ -recursión **no** es **total**.

- ▶ Consideremos la función suma, $Sum(x, y) = x + y$. Entonces

$$Sum_{\mu}(x) = \mu y (Sum(x, y) = 0) = \begin{cases} 0 & \text{si } x = 0 \\ \uparrow & \text{e.o.c.} \end{cases}$$

- ▶ Existen funciones **parciales no totales** cuya función definida por μ -recursión **sí** es **total**.

- ▶ Consideremos la función $g(x, y) = \begin{cases} x - y & \text{si } y \leq x \\ \uparrow & \text{si } y > x \end{cases}$

Entonces $g_{\mu}(x) = \mu y (g(x, y) = 0) = x$, para cada $x \in \mathbb{N}$.

- ▶ Toda función definida por **minimización no acotada** a partir de funciones **GOTO-computables** es **GOTO-computable**.

- ▶ Sea $f : \mathbb{N}^{k+1} \rightarrow \mathbb{N}$ ($k \geq 1$) una función **GOTO-computable**.
El siguiente programa calcula f_{μ} :

```
[A]  Z ← f(X1, ..., Xk, Y)
      IF Z = 0 GOTO E
      Y ← Y + 1
      GOTO A
```

Minimización no acotada de predicados

Sea θ un predicado $(k + 1)$ -ario. La función k -aria definida por **minimización no acotada** a partir de θ es la función $\theta_\mu : \mathbb{N}^k \rightarrow \mathbb{N}$ dada por

$$\theta_\mu(\vec{x}) \equiv \mu y(\theta(\vec{x}, y)) = \begin{cases} \min\{y : \theta(\vec{x}, y) = 1\} & \text{si existe el minimo} \\ \uparrow & \text{e.c.o.c.} \end{cases}$$

Notaremos $\theta_\mu(\vec{x}) = \mu y(\theta(\vec{x}, y))$.

- ▶ La función obtenida por minimización no acotada a partir de un predicado **no tiene porqué ser un predicado**.
- ▶ Toda función definida por **minimización no acotada** a partir de un predicado **GOTO-computable** es una función **GOTO-computable**.

La clase de las **funciones recursivas**, \mathcal{P} , es la clase de funciones que verifica las condiciones siguientes:

- ▶ Toda función básica pertenece a \mathcal{P} .
- ▶ \mathcal{P} es cerrada bajo los procedimientos de composición, recursión primitiva y minimización no acotada.
- ▶ Si \mathcal{C} es una clase de funciones que:
 - ▶ contiene a todas las funciones básicas; y
 - ▶ es cerrada bajo los procedimientos de composición, recursión primitiva y minimización no acotada,entonces $\mathbf{P} \subseteq \mathcal{C}$.

Notaremos:

- ▶ $\mathcal{P}^{(n)}$ la clase de las funciones **recursivas n -arias**.
- ▶ \mathcal{R} la clase de **funciones recursivas totales**.

Obviamente:

- ▶ Toda función primitiva recursiva es recursiva y total.
- ▶ Toda función recursiva es GOTO-computable.

Se probará que toda función GOTO-computable es, también, recursiva.

Conjuntos y Predicados recursivos

Definición: Un conjunto $B \subseteq \mathbb{N}^k$ es **recursivo** si su función característica C_B es recursiva. Notaremos $B \in \Delta_1^0$.

Todo conjunto (resp. predicado) **PR** es un conjunto (resp. predicado) recursivo.

Se han establecido propiedades para funciones **PR** en las que sólo se usa el cierre de la clase por composición y recursión.

Todas esas propiedades serán, asimismo, válidas para funciones **recursivas totales**.

- ▶ Teorema de definición por casos para funciones recursivas totales.
- ▶ Los predicados recursivos son estables bajo las conectivas lógicas y bajo cuantificaciones acotadas.
- ▶ Los conjuntos recursivos son estables bajo unión, intersección y complementario.
- ▶ Las funciones recursivas totales son estables bajo suma y productos acotados.
- ▶ La función obtenida por minimización acotada de un predicado recursivo es una función recursiva.

Teorema del grafo para funciones recursivas totales.

Sea $f : \mathbb{N}^k \rightarrow \mathbb{N}$ una función total. Son equivalentes:

- (a) f es recursiva.
- (b) El grafo de f es un conjunto recursivo

Téngase presente que:

- ▶ $C_{G(f)}(\vec{x}, y) = \bar{s}g(|f(\vec{x}) - y|)$.
- ▶ $f(\vec{x}) = (\mu y)(C_{G(f)}(\vec{x}, y))$.