

# Lógica Computacional, 2018-2

## Nota 10. Formas normales para la lógica de predicados.\*

Noé Salomón Hernández S.

### 1. Introducción

La *regla de resolución* es una regla de inferencia para la lógica de predicados análoga a la regla de resolución de la lógica proposicional. Tal regla de resolución funciona únicamente sobre expresiones en *forma clausular*.

### 2. FNCP y Forma Clausular

Ocuparemos la noción de literal que se introdujo en la nota pasada.

**Definición 1.** Una expresión está en *forma normal conjuntiva* syss es una conjunción de disyunciones de literales.

La forma normal conjuntiva se utilizó de forma implícita para obtener la forma clausular en la lógica proposicional.

**Definición 2.** Una fórmula está en *forma normal conjuntiva prenex (FNCP)* syss es de la forma:

$$Q_1x_1 \cdots Q_nx_n M$$

donde las  $Q_i$  son cuantificadores y  $M$  es una fórmula libre de cuantificadores en *forma normal conjuntiva*. A la secuencia  $Q_1x_1 \cdots Q_nx_n$  se le conoce como *prefijo* y a  $M$  como *matriz*.

**Definición 3.** Sea  $\varphi$  un enunciado, es decir, una fórmula *cerrada*, en FNCP cuyo prefijo consiste únicamente de cuantificadores universales. La *forma clausular* de  $\varphi$  consiste de la matriz de  $\varphi$  escrita como un conjunto de cláusulas.

**Definición 4.** Sean  $\varphi$  y  $\psi$  dos fórmulas.  $\varphi \approx \psi$  denota que  $\varphi$  es satisfacible syss  $\psi$  es satisfacible.

Es importante entender que  $\varphi \approx \psi$  ( $\varphi$  es satisfacible syss  $\psi$  es satisfacible) *no* implica que  $\varphi \equiv \psi$  ( $\varphi$  es lógicamente equivalente a  $\psi$ ). Por un lado,  $\varphi \approx \psi$  indica que *existe* un modelo para  $\varphi$  syss *existe* un modelo para  $\psi$ . Esto no es lo mismo que la equivalencia lógica  $\varphi \equiv \psi$ , lo cual significa que para *todo* modelo  $\mathcal{M}$ , se tiene que  $\mathcal{M}$  es un modelo para  $\varphi$  syss es también un modelo para  $\psi$ .

---

\*Esta nota se base en material elaborado por el prof. Favio Miranda y en el libro de Ben-Ari M., *Mathematical Logic for Computer Science*

**Teorema 1** (Skolem). *Sea  $\varphi$  una fórmula cerrada. Entonces existe una fórmula  $\psi$  en forma clausular tal que  $\varphi \approx \psi$ .*

Es claro que podemos transformar la  $\varphi$  del Teorema 1 en una fórmula lógicamente equivalente en FNCP. Es la eliminación de los cuantificadores existenciales lo que causa que la nueva fórmula no sea equivalente a la original. Esta eliminación se consigue definiendo nuevos símbolos de función.

### 3. Algoritmo de Skolem

Damos ahora un algoritmo para transformar una fórmula  $\varphi$  en una fórmula  $\psi$  en forma clausular.

**Algoritmo**

**Input:** Una fórmula *cerrada*  $\varphi$  de la lógica de predicados.

**Output:** Una fórmula  $\psi$  en forma clausular tal que  $\varphi \approx \psi$ .

(L) Se limpia la fórmula de entrada  $\varphi$ .

- Eliminar cuantificadores múltiples mediante las siguientes equivalencias:

$$\begin{array}{ll} \forall x \forall x \varphi \equiv \forall x \varphi & \exists x \forall x \varphi \equiv \forall x \varphi \\ \exists x \exists x \varphi \equiv \exists x \varphi & \forall x \exists x \varphi \equiv \exists x \varphi \end{array}$$

- Eliminar los cuantificadores vacuos mediante las siguientes equivalencias, donde  $x$  no está libre en  $\varphi$ :

$$\forall x \varphi \equiv \varphi \qquad \exists x \varphi \equiv \varphi$$

(I) Eliminar las implicaciones y bi-condicionales empleando las equivalencias con los operadores  $\wedge$ ,  $\vee$  y  $\neg$ .

$$\begin{array}{ll} \varphi \rightarrow \psi & \equiv \neg \varphi \vee \psi \\ \varphi \leftrightarrow \psi & \equiv (\neg \varphi \vee \psi) \wedge (\varphi \vee \neg \psi) \end{array}$$

(N) Meter el operador de negación para que tenga efecto sólo sobre fórmulas atómicas. Usar las siguientes equivalencias:

$$\begin{array}{ll} \neg \neg \varphi & \equiv \varphi \\ \neg(\varphi \wedge \psi) & \equiv \neg \varphi \vee \neg \psi \\ \neg(\varphi \vee \psi) & \equiv \neg \varphi \wedge \neg \psi \\ \neg \forall x \varphi & \equiv \exists x \neg \varphi \\ \neg \exists x \varphi & \equiv \forall x \neg \varphi \end{array}$$

(R) Renombrar variables ligadas para que ninguna variable se repita en dos o más cuantificadores.

(E) Extraer cuantificadores de la matriz. Escoger un cuantificador más externo, es decir, un cuantificador que no está dentro del alcance de otro. Extraer el cuantificador usando las siguientes equivalencias, donde  $Q$  es un cuantificador, y  $op$  es  $\vee$  o  $\wedge$ :

$$\psi \text{ op } Qx \varphi \equiv Qx (\psi \text{ op } \varphi) \qquad Qx \varphi \text{ op } \psi \equiv Qx (\varphi \text{ op } \psi)$$

Repetir hasta que todos los cuantificadores aparezcan en el prefijo para que la matriz se encuentre libre de cuantificadores. Las equivalencias se pueden aplicar porque ninguna variable figura en dos o más cuantificadores.

- (D) Usar las leyes distributivas para transformar la matriz a su FNC. La fórmula está ahora en FNCP.

$$\begin{aligned}\varphi \vee (\psi \wedge \chi) &\equiv (\varphi \vee \psi) \wedge (\varphi \vee \chi) \\ (\varphi \wedge \psi) \vee \chi &\equiv (\varphi \vee \chi) \wedge (\psi \vee \chi)\end{aligned}$$

- (S) Recorrer el prefijo de izquierda a derecha. Para todo cuantificador existencial  $\exists x$  en  $\varphi$ , sean  $y_1, \dots, y_n$  las variables cuantificadas universalmente que preceden a  $\exists x$ , y sea  $f$  un **nuevo** símbolo de función de aridad  $n$ . Eliminar  $\exists x$  y reemplazar cada presencia de  $x$  por  $f(y_1, \dots, y_n)$ . Cualquier función definida de esta forma se llama *función de Skolem*. Si no hay cuantificadores universales precediendo a  $\exists x$ , reemplazar  $x$  por una nueva constante, la constante usada para reemplazar la variable existencial se llama *constante de Skolem*. El proceso de reemplazar cuantificadores existenciales por funciones se llama *Skolemización*.

- ▷ La fórmula puede ser escrita en forma clausular al dejar a un lado los cuantificadores universales y al escribir la matriz como un conjunto de cláusulas.

## 4. Ejemplo

Transformar el siguiente enunciado a forma clausular.

$$\neg \exists y (P(y) \wedge \forall z (R(z) \rightarrow Q(y, z)))$$

- (L) La fórmula queda igual.

(I)  $\neg \exists y (P(y) \wedge \forall z (\neg R(z) \vee Q(y, z)))$ .

- (N)

$$\begin{aligned}\neg \exists y (P(y) \wedge \forall z (\neg R(z) \vee Q(y, z))) &\equiv \forall y \neg (P(y) \wedge \forall z (\neg R(z) \vee Q(y, z))) \\ &\equiv \forall y (\neg P(y) \vee \neg \forall z (\neg R(z) \vee Q(y, z))) \\ &\equiv \forall y (\neg P(y) \vee \exists z \neg (\neg R(z) \vee Q(y, z))) \\ &\equiv \forall y (\neg P(y) \vee \exists z (\neg \neg R(z) \wedge \neg Q(y, z))) \\ &\equiv \forall y (\neg P(y) \vee \exists z (R(z) \wedge \neg Q(y, z)))\end{aligned}$$

- (R) La fórmula queda igual.

(E)  $\forall y \exists z (\neg P(y) \vee (R(z) \wedge \neg Q(y, z)))$ .

(D)  $\forall y \exists z ((\neg P(y) \vee R(z)) \wedge (\neg P(y) \vee \neg Q(y, z)))$ .

(S)  $\forall y ((\neg P(y) \vee R(f(y))) \wedge (\neg P(y) \vee \neg Q(y, f(y))))$ .

- ▷  $\{\{\neg P(y), R(f(y))\}, \{\neg P(y), \neg Q(y, f(y))\}\}$ , esta es la forma clausular que buscamos.