TEMA 4: Lógica proposicional

(Apuntes elaborados por Juan Manuel Urbano Blanco)

Contents

1 Introducción

El primer objetivo de la Lógica es **representar** mediante símbolos **el conocimiento** humano para posteriormente **hacer deducciones o razonamientos**.

La **Lógica proposicional** es la aproximación más sencilla a este objetivo.

Los elementos del lenguaje humano que vamos a manejar son sentencias declarativas que pueden ser verdaderas o falsas (pero no ambas cosas o ninguna de las dos). Cada una de esas sentencias se construye a partir de ciertos enunciados más simples ó indescomponibles, denominados **enunciados atómicos**, y determinadas partículas que hacen de nexo ó unión entre enunciados.

Usaremos las letras mayúsculas P,Q,R,S,T, etc, llamadas **variables proposicionales**, para denotar enunciados atómicos, y las letras griegas $\alpha, \beta, \gamma, \delta, \ldots$ etc para representar enunciados en general. Cuando sea necesario, añadiremos subíndices a dichos símbolos.

En nuestro formalismo, denotaremos las partículas que hacen de nexo ó unión entre enunciados mediante los llamados **operadores o conectivas lógicas**. Los operadores lógicos que usaremos son los siguientes:

$\neg$	se lee **no** y representa la *negación*,
$\wedge$	se lee **y** y representa la *conjunción*,
$\vee$	se lee **o** y representa la *disyunción*,
$\rightarrow$	se lee **implica** y representa la *implicación*,
$\leftrightarrow$	se lee **equivale a** y representa la *equivalencia o doble implicación*.

Dado un conjunto $X \neq \varnothing$ de variables proposicionales, el conjunto de las **fórmulas ó proposiciones lógicas** sobre X, denotado por $\mathcal{F}(X)$, lo definimos recursivamente de la siguiente forma.

1. $X \subseteq \mathcal{F}(X)$, es decir, toda fórmula atómica es una proposición lógica.
2. Si $\alpha, \beta \in \mathcal{F}(X)$, entonces $\neg\alpha$, $\alpha\wedge\beta$, $\alpha\vee\beta$, $\alpha \to \beta$ y $\alpha\leftrightarrow\beta$ también pertenecen a $\mathcal{F}(X)$.
3. No hay más proposiciones lógicas en $\mathcal{F}(X)$ que las obtenidas aplicando un número finito de veces los apartados anteriores.

Si tenemos una fórmula del tipo $\alpha \to \beta$, entonces α se denomina **hipótesis o antecedente** y β se llama **tesis o consecuente**.

Las reglas anteriores mediante las cuales hemos definido $\mathcal{F}(X)$, han de ir acompañadas de unos **criterios de precedencia o prioridad entre conectivas**, así como el uso de paréntesis, para evitar ambigüedades. Para tal fin, establecemos lo siguiente:

1. La conectiva $\neg$ tiene prioridad sobre las otras conectivas. Por ejemplo,

   $\neg\alpha\wedge\beta$ representa $(\neg\alpha)\wedge\beta$ y es diferente de $\neg(\alpha\wedge\beta)$,

   $\neg\alpha\vee\beta$ representa $(\neg\alpha)\vee\beta$ y es diferente de $\neg(\alpha\vee\beta)$,

   $\neg\alpha \to \beta$ representa $(\neg\alpha) \to \beta$ y es diferente de $\neg(\alpha \to \beta)$,

   $\neg\alpha\leftrightarrow\beta$ representa $(\neg\alpha)\leftrightarrow\beta$ y es diferente de $\neg(\alpha\leftrightarrow\beta)$.

2. $\vee$ e $\wedge$ tienen igual precedencia y ambas tienen más precedencia que $\to$ y $\leftrightarrow$. En caso de ambigüedad, usaremos paréntesis. Por ejemplo,

   $\alpha\wedge\beta \to \gamma$ representa $(\alpha\wedge\beta) \to \gamma$ y es diferente de $\alpha\wedge(\beta \to \gamma)$,

   $\alpha \to \beta\vee\gamma$ representa $\alpha \to (\beta\vee\gamma)$ y es diferente de $(\alpha \to \beta)\vee\gamma$.

   Las proposiciones $(\alpha\wedge\beta)\vee\gamma$ y $\alpha\wedge(\beta\vee\gamma)$ tienen significados distintos, por lo cual no aceptaremos $\alpha\wedge\beta\vee\gamma$ como proposición.

   Tampoco aceptaremos $\alpha \to \beta \to \gamma$ como proposición, pues como veremos $(\alpha \to \beta) \to \gamma$ y $\alpha \to (\beta \to \gamma)$ tienen significados distintos.

3. Las conectivas $\to$ y $\leftrightarrow$ tienen igual precedencia.

   Por tanto la expresión $\alpha\leftrightarrow\beta \to \gamma$ no representará una fórmula, teniendo que usar paréntesis para eliminar la ambigüedad.

Existen otras notaciones en las cuales no es necesario usar paréntesis para establecer la prioridad de los operadores lógicos, como por ejemplo, **la notación polaca o prefija** (en honor al filósofo y lógico polaco J. Lukasiewicz, 1878-1956), en la cual cada conectiva precede a los argumentos a los que se aplica.

Si bien esta escritura resulta propicia para implementaciones en un ordenador, da como resultado expresiones de difícil lectura para un ser humano. Por ejemplo,

Notación infija ó usual	Notación polaca ó prefija
$P{\vee}Q$	$\vee PQ$
$(P{\vee}Q){\wedge}R$	$\wedge\vee PQR$
$P{\vee}(Q{\wedge}R)$	$\vee P{\wedge}QR$
$\neg\ (P \to R){\leftrightarrow}(Q{\wedge}\neg S)$	$\leftrightarrow\neg \to PR{\wedge}Q\neg S$

2 Traducción del lenguaje humano a la Lógica proposicional.

Una fórmula del tipo $\alpha \to \beta$ representa a cada una de las siguientes frases:

- Si α, entonces β.
- α implica β.
- β ocurre cuando ocurre α.
- β se sigue (se obtiene, se deduce) de α.
- β es una consecuencia lógica de α.
- β, si α.
- α sólo si β. La explicación es que si no ocurre β, entonces no ocurre α.
- α es condición suficiente para β.
- β es condición necesaria para α.
- β es *conditio sine qua non* para α.

Una fórmula del tipo $\alpha{\leftrightarrow}\beta$ se lee de cada una de las siguientes formas:

- α y β son equivalentes.
- α si y sólo si β.
- Si α entonces β, y si β entonces α.

Veamos unos ejemplos.

Ejemplo 1. Vamos a representar la frase “*Hace falta que haya nubes para que llueva*” mediante una proposición lógica.

Esta frase la podemos escribir de manera equivalente como “*Es necesario que haya nubes para que llueva*”, ó incluso de manera más clara, “*Si llueve, entonces hay nubes en el cielo*”.

Las variables proposicionales que usaremos son:

- P, que representa al enunciado atómico “*hay nubes*”, y
- Q, que representa al enunciado atómico “*llueve*”.

Obtenemos la proposición lógica $Q \to P$.

Claramente, el que haya nubes no asegura que llueva. □

Ejemplo 2. Para representar la frase “*El agua se evapora cuando hace calor y se congela cuando hace frío*”, usamos las variables proposicionales siguientes:

- P, que representa al enunciado atómico “*hace calor*”,
- Q, que representa al enunciado atómico “*hace frío*”,
- R, que representa al enunciado atómico “*el agua se evapora*”,
- S, que representa al enunciado atómico “*el agua se congela*”.

Resulta la proposición lógica siguiente:

$$(P \to R) \wedge (Q \to S).$$

□

3 Semántica de la Lógica proposicional.

Tal y como hemos dicho en la sección anterior, toda proposición lógica puede ser verdadera ó falsa. Consideramos el cuerpo finito $\mathbb{Z}_2 = \{0, 1\}$, donde el símbolo 1 se usará para indicar que una proposición es verdadera y el símbolo 0 para indicar que es falsa.

Sea X un conjunto de variables proposicionales y sea $\mathcal{F}(X)$ el conjunto de las proposiciones lógicas que se obtienen a partir de X. Una **interpretación** para $\mathcal{F}(X)$ es una aplicación $\mathrm{I} : \mathcal{F}(X) \to \mathbb{Z}_2$ que verifica:

1. $\mathrm{I}(\neg\alpha) = 1 \;\Leftrightarrow\; \mathrm{I}(\alpha) = 0$.
2. $\mathrm{I}(\alpha\wedge\beta) = 1 \;\Leftrightarrow\; \mathrm{I}(\alpha) = 1$ e $\mathrm{I}(\beta) = 1$.
3. $\mathrm{I}(\alpha\vee\beta) = 1 \;\Leftrightarrow\; \mathrm{I}(\alpha) = 1$ ó $\mathrm{I}(\beta) = 1$.

4. $\mathrm{I}(\alpha \to \beta) = 0 \;\Leftrightarrow\; \mathrm{I}(\alpha) = 1$ e $\mathrm{I}(\beta) = 0$.

5. $\mathrm{I}(\alpha{\leftrightarrow}\beta) = 1 \;\Leftrightarrow\; \mathrm{I}(\alpha) = \mathrm{I}(\beta)$.

$\square$

Diremos que $\mathrm{I}(\alpha)$ es el **valor de verdad** de la fórmula α bajo la interpretación I. Una fórmula α será verdadera bajo la interpretación I, si $\mathrm{I}(\alpha) = 1$, y será falsa si $\mathrm{I}(\alpha) = 0$.

Por tanto, en la definición previa, estamos dicidiendo que bajo la interpretación I:

1. $\neg\alpha$ es verdadera si y sólo si α es falsa.

2. $\alpha{\wedge}\beta$ es verdadera si y sólo si α y β son ambas verdaderas.

3. $\alpha{\vee}\beta$ es verdadera si y sólo si al menos una de las dos proposiciones α, β es verdadera.

   Así el operador $\vee$ representa el "o" inclusivo del lenguaje, frente al "o" exclusivo del lenguaje. El "o" exclusivo se considera verdadero si y sólo si exactamente uno de los dos argumentos a los que se aplica es verdadero y el otro es falso.

   Por ejemplo, si decimos "*Escucho la radio o me tomo un café* ", estamos usando un "o" inclusivo, ya que ambas acciones pueden ser ciertas simultáneamente, mientras que si decimos "*Hoy a las 20:00 horas estaré en Madrid o estaré en Barcelona*" usamos la partícula disyuntiva "o" con un sentido exclusivo.

4. $\alpha \to \beta$ es falsa sólo cuando α es verdadera y β es falsa; en el resto de casos, $\alpha \to \beta$ se considera verdadera.

5. $\alpha{\leftrightarrow}\beta$ es verdadera sólo cuando α y β son ambas verdaderas o ambas falsas; en los casos restantes, $\alpha{\leftrightarrow}\beta$ es falsa.

$\square$

Observamos que la definición dada para una interpretación I sobre $\mathcal{F}(X)$ podemos recordarla más fácilmente, si recordamos las tablas de los operadores booleanos siguientes definidos sobre el álgebra de Boole binaria $\mathbb{B} = \{0, 1\}$.

$-$	
0	1
1	0

$\cdot$	0	1
0	0	0
1	0	1

$+$	0	1
0	0	1
1	1	1

$\oplus$	0	1
0	0	1
1	1	0

$\to$	0	1
0	1	1
1	0	1

$\leftrightarrow$	0	1
0	1	0
1	0	1

De hecho, el significado de los operadores lógicos viene dado por los operadores booleanos, motivo por el cual empleamos (casi) los mismos símbolos que fueron empleados en el Tema 1.

Por tanto, una **interpretación** para $\mathcal{F}(X)$ es una aplicación $\mathrm{I} : \mathcal{F}(X) \to \mathbb{Z}_2$ que verifica:

1. $\mathrm{I}(\neg\alpha) = \overline{\mathrm{I}(\alpha)}$.
2. $\mathrm{I}(\alpha\wedge\beta) = \mathrm{I}(\alpha)\cdot\mathrm{I}(\beta)$.
3. $\mathrm{I}(\alpha\vee\beta) = \mathrm{I}(\alpha)+\mathrm{I}(\beta)$.
4. $\mathrm{I}(\alpha\to\beta) = \mathrm{I}(\alpha)\to\mathrm{I}(\beta) = \overline{\mathrm{I}(\alpha)}+\mathrm{I}(\beta)$.
5. $\mathrm{I}(\alpha\leftrightarrow\beta) = \mathrm{I}(\alpha)\leftrightarrow\mathrm{I}(\beta)$. □

Además sabemos que el conjunto de operadores booleanos $\{1,\oplus,\cdot\}$ es funcionalmente completo, motivo por el cual de forma equivalente también podemos definir una interpretación I sobre $\mathcal{F}(X)$ como una aplicación $\mathrm{I}:\mathcal{F}(X)\to\mathbb{Z}_2$ que verifica:

1. $\mathrm{I}(\neg\alpha) = 1\oplus\mathrm{I}(\alpha)$.
2. $\mathrm{I}(\alpha\vee\beta) = \mathrm{I}(\alpha)\oplus\mathrm{I}(\beta)\oplus\mathrm{I}(\alpha)\cdot\mathrm{I}(\beta)$.
3. $\mathrm{I}(\alpha\wedge\beta) = \mathrm{I}(\alpha)\cdot\mathrm{I}(\beta)$.
4. $\mathrm{I}(\alpha\to\beta) = 1\oplus\mathrm{I}(\alpha)\oplus\mathrm{I}(\alpha)\cdot\mathrm{I}(\beta)$.
5. $\mathrm{I}(\alpha\leftrightarrow\beta) = 1\oplus\mathrm{I}(\alpha)\oplus\mathrm{I}(\beta)$. □

Cualquier interpretación I sobre $\mathcal{F}(X)$ queda totalmente determinada en cuanto conocemos el valor de I sobre cada una de las fórmulas pertenecientes a X. Además si X consta de n elementos, entonces existirán 2^n interpretaciones sobre $\mathcal{F}(X)$.

Ejemplo 3. Sea $X=\{P,Q,R\}$ y sea I la interpretación sobre $\mathcal{F}(X)$ tal que

$$\mathrm{I}(P)=1,\quad \mathrm{I}(Q)=0\quad \text{e}\quad \mathrm{I}(R)=1.$$

Si $\alpha = Q\leftrightarrow((P\wedge Q)\to R)$, entonces

$$\mathrm{I}(\alpha)=\mathrm{I}(Q)\leftrightarrow\Big(\big(\mathrm{I}(P)\cdot\mathrm{I}(Q)\big)\to\mathrm{I}(R)\Big) = 0\leftrightarrow((1\cdot 0)\to 1) = 0\leftrightarrow(0\to 1) = 0\leftrightarrow 1 = 0,$$

por lo cual α es falsa bajo la interpretación I.

De forma equivalente podemos calcular

$$\begin{aligned}\mathrm{I}(\alpha) &= 1\oplus\mathrm{I}(Q)\oplus\mathrm{I}\big((P\wedge Q)\to R\big)\\ &= 1\oplus\mathrm{I}(Q)\oplus\Big(1\oplus\mathrm{I}(P\wedge Q)\oplus\mathrm{I}(P\wedge Q)\cdot\mathrm{I}(R)\Big)\\ &= 1\oplus\mathrm{I}(Q)\oplus\Big(1\oplus\mathrm{I}(P)\cdot\mathrm{I}(Q)\oplus\mathrm{I}(P)\cdot\mathrm{I}(Q)\cdot\mathrm{I}(R)\Big)\\ &= 1\oplus 0\oplus(1\oplus 1\cdot 0\oplus 1\cdot 0\cdot 1) = 0.\end{aligned}$$

Este segundo método no es recomendable en la práctica pues suelen aparecer expresiones muy grandes. □

Si α es una proposición lógica construida sobre las variables proposicionales $P_1, \ldots, P_n$, la **tabla de verdad** de α es una tabla formada por 2^n filas y por $n+1$ columnas. Las n primeras columnas están etiquetadas con $P_1, \ldots, P_n$ y la última con α. Cada fila representa una interpretación posible y el valor correspondiente de α.

Ejemplo 4. Obtenemos la tabla de verdad de la fórmula $\alpha = Q \leftrightarrow ((P \wedge Q) \rightarrow R)$.

P	Q	R	α
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

$\square$

A la hora de construir la tabla de verdad de una fórmula α, se recomienda hacerlo por etapas, es decir, ir construyendo las tablas de verdad de las subfórmulas de α. En el ejemplo anterior resulta:

P	Q	R	$P \wedge Q$	$(P \wedge Q) \rightarrow R$	α
0	0	0	0	1	0
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	0	1	1
1	0	0	0	1	0
1	0	1	0	1	0
1	1	0	1	0	0
1	1	1	1	1	1

Si $P_1, \ldots, P_n$ son las variables proposicionales que aparecen en una fórmula α, a veces interesa construir la tabla de verdad de α respecto de un conjunto mas ámplio de variables proposicionales, digamos $P_1, \ldots, P_n, P_{n+1}, \ldots, P_{n+m}$. Ésto lo usaremos cuando queramos comparar α con otras fórmulas cuyas variables proposicionales no sean exactamente $P_1, \ldots, P_n$. (Véase el Ejemplo 7.) En otras ocasiones construiremos tablas de verdad en las que no aparecen variables proposicionales, sino letras que representan subfórmulas más simples. (Véase el Ejemplo 5.)

4 Clasificación de fórmulas.

Una proposición lógica $\alpha \in \mathcal{F}(X)$ es:

- una **tautología** si para cualquier interpretación I sobre $\mathcal{F}(X)$ se tiene que $\mathrm{I}(\alpha) = 1$;
- una **contradicción** si para cualquier interpretación I sobre $\mathcal{F}(X)$ se tiene que $\mathrm{I}(\alpha) = 0$;
- **satisfacible** si existe al menos una interpretación I sobre $\mathcal{F}(X)$ tal que $\mathrm{I}(\alpha) = 1$;
- **refutable** si existe al menos una interpretación I sobre $\mathcal{F}(X)$ tal que $\mathrm{I}(\alpha) = 0$;
- **contingente** si es satisfacible y refutable.

Claramente, una fórmula α es una tautología si y sólo si $\neg\alpha$ es una contradicción, pues para cualquier interpretación I, se verifica que $\mathrm{I}(\alpha) = 1$ si y sólo si $\mathrm{I}(\neg\alpha) = 0$.

Desde el punto de vista de las tablas de verdad, una fórmula α es una tautología (respectivamente, una contradicción) si y sólo si la última columna de la tabla de verdad de α está formada sólo por 1s (respectivamente, 0s).

Ejemplo 5.
(1.) Para cualquier $\alpha \in \mathcal{F}(X)$, la fórmula $\beta = \alpha\vee\neg\alpha$ es una tautología. Podemos construir una especie de tabla de verdad para β como si α fuese una variable proposicional, pues en realidad no va a influir para nada quién sea la fórmula α:

α	β
0	1
1	1

De otra forma, podemos calcular

$$\begin{aligned}\mathrm{I}(\beta) = \mathrm{I}(\alpha\vee\neg\alpha) &= \mathrm{I}(\alpha) \oplus \mathrm{I}(\neg\alpha) \oplus \mathrm{I}(\alpha) \cdot \mathrm{I}(\neg\alpha) \\ &= \mathrm{I}(\alpha) \oplus (1 \oplus \mathrm{I}(\alpha)) \oplus (\mathrm{I}(\alpha) \cdot (1 \oplus \mathrm{I}(\alpha))) \\ &= \mathrm{I}(\alpha) \oplus 1 \oplus \mathrm{I}(\alpha) \oplus \mathrm{I}(\alpha) \oplus \mathrm{I}(\alpha) = 1,\end{aligned}$$

pues para todo $a \in \mathbb{Z}_2$, se verifica $a^2 = a$ y $0 = 2a = 4a$. Como $\mathrm{I}(\beta) = 1$ independientemente del valor de $\mathrm{I}(\alpha) \in \mathbb{Z}_2$, obtenemos que β es una tautología.

(2.) La proposición $\alpha \to (\beta \to \alpha)$ es una tautología. Lo comprobamos con la tabla de verdad:

α	β	$\beta \to \alpha$	$\alpha \to (\beta \to \alpha)$
0	0	1	1
0	1	0	1
1	0	1	1
1	1	1	1

(3.) La proposición $\alpha\wedge\neg\alpha$ es una contradicción, pues para cualquier interpretación I sobre $\mathcal{F}(X)$ tenemos

$$\mathrm{I}(\alpha\wedge\neg\alpha) = \mathrm{I}(\alpha) \cdot \mathrm{I}(\neg\alpha) = \mathrm{I}(\alpha) \cdot (1 \oplus \mathrm{I}(\alpha)) = \mathrm{I}(\alpha) \oplus \mathrm{I}(\alpha) \cdot \mathrm{I}(\alpha) = \mathrm{I}(\alpha) \oplus \mathrm{I}(\alpha) = 0,$$

pues en $\mathbb{Z}_2$ se verifica que $a^2 = a$ y $2a = 0$.

(4.) Sea $X = \{P, Q, R\}$. La proposición $\alpha = P \vee Q$ es satisfacible, pues existe la interpretación I sobre $\mathcal{F}(X)$ (entre otras posibilidades) dada por

$$\mathrm{I}(P) = 1, \quad \mathrm{I}(Q) = 0, \quad \mathrm{I}(R) = 1$$

tal que $\mathrm{I}(\alpha) = 1$.

También es α refutable, pues existe la interpretación I' sobre $\mathcal{F}(X)$ (entre otras posibilidades) dada por

$$\mathrm{I}'(P) = 0, \quad \mathrm{I}'(Q) = 0, \quad \mathrm{I}'(R) = 1$$

tal que $\mathrm{I}'(\alpha) = 0$. Por tanto α es una fórmula contingente. □

Un conjunto de fórmulas $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathcal{F}(X)$ se dice que es **satisfacible** si existe una interpretación I sobre $\mathcal{F}(X)$ tal que

$$\mathrm{I}(\alpha_1) = \cdots = \mathrm{I}(\alpha_n) = 1.$$

En tal caso se dice que **la interpretación** I **satisface al conjunto** Ω y se escribe $\mathrm{I}(\Omega) = 1$. Un conjunto de fórmulas es **insatisfacible** si no es satisfacible.

Ejemplo 6. Para $X = \{P, Q\}$, el conjunto

$$\Omega_1 = \Big\{ P \vee \neg Q, \ \neg P \vee Q \Big\}$$

es satisfacible pues existe la interpretación I sobre $\mathcal{F}(X)$ dada por

$$\mathrm{I}(P) = 0, \quad \mathrm{I}(Q) = 0$$

tal que

$$\mathrm{I}(P \vee \neg Q) = \mathrm{I}(\neg P \vee Q) = 1.$$

También sirve la interpretación

$$\mathrm{I}'(P) = 1, \quad \mathrm{I}'(Q) = 1$$

para obtener la misma conclusión.

El conjunto

$$\Omega_2 = \Big\{ P, \ P \vee Q, \ \neg P \Big\}$$

es insatisfacible, pues no existe ninguna interpretación I'' sobre $\mathcal{F}(X)$ tal que $\mathrm{I}''(P) = \mathrm{I}''(\neg P) = 1$. □

Proposición 1. Todas las interpretaciones satisfacen al conjunto vacío, y por tanto el conjunto vacío es satisfacible.

La justificación es la siguiente. Supongamos que existe alguna interpretación I que no satisface a $\varnothing$. Entonces existe alguna fórmula $\alpha \in \varnothing$ tal que $\mathrm{I}(\alpha) = 0$, lo cual es imposible, pues en el conjunto vacío no hay elementos.

Las dos siguientes propiedades son una consecuencia inmediata de las definiciones.

Proposición 2. Sean $\Omega \subseteq \mathcal{F}(X)$ y una tautología $\alpha \in \mathcal{F}(X)$. Entonces $\Omega \cup \{\alpha\}$ es satisfacible si y sólo si Ω es satisfacible. De modo equivalente, $\Omega \cup \{\alpha\}$ es insatisfacible si y sólo si Ω es insatisfacible.

Proposición 3. Un conjunto de fórmulas $\Omega = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subseteq \mathcal{F}(X)$ es insatisfacible si y sólo si la fórmula $\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_n$ es una contradicción.

Ejemplo 7. Veamos que el conjunto de proposiciones lógicas

$$\Omega = \Big\{P \to Q,\ Q \to R,\ P \wedge \neg R\Big\}$$

es insatisfacible. Una forma de comprobarlo es obteniendo las tablas de verdad de las fórmulas de Ω:

P	Q	R	$P \to Q$	$Q \to R$	$P \wedge \neg R$
0	0	0	1	1	0
0	0	1	1	1	0
0	1	0	1	0	0
0	1	1	1	1	0
1	0	0	0	1	1
1	0	1	0	1	0
1	1	0	1	0	1
1	1	1	1	1	0

Como no hay ninguna fila en la que todas las fórmulas de Ω valgan al mismo tiempo 1, ello implica que Ω es insatisfacible.

Otro camino muy engorroso sería comprobar que el valor de

$$\mathrm{I}\Big((P \to Q) \wedge (Q \to R) \wedge (P \wedge \neg R)\Big)$$

es idénticamente igual a 0, es decir, vale 0 independientemente del valor de $\mathrm{I}(P), \mathrm{I}(Q), \mathrm{I}(R)$ en $\mathbb{Z}_2$. □

Ejemplo 8. Sea el conjunto de proposiciones lógicas

$$\Omega' = \Big\{R \to R,\ P \to Q,\ Q \to R,\ P \wedge \neg R\Big\}.$$

Observamos que la fórmula $R \to R$ es una tautología, y por otra parte ya sabemos del ejemplo anterior que el conjunto

$$\Omega = \Big\{P \to Q,\ Q \to R,\ P \wedge \neg R\Big\}$$

es insatisfacible. Así, aplicamos la Proposición 2 y deducimos que Ω' es insatisfacible. □

Ejemplo 9. Sea la proposición lógica

$$\alpha = (P \to Q) \wedge (Q \to R) \wedge (P \wedge \neg R).$$

Por la Proposición 3, sabemos que α es una contradicción si, y sólo si, el conjunto de proposiciones

$$\Omega = \Big\{P \to Q,\ Q \to R,\ P \wedge \neg R\Big\},$$

es insatisfacible.

Como en el Ejemplo 7 ya hemos constatado esto último, obtenemos que α es una contradicción. $\square$

Dos **fórmulas** $\alpha, \beta \in \mathcal{F}(X)$ se dice que son **lógicamente equivalentes**, si para cualquier interpretación I sobre $\mathcal{F}(X)$ se cumple que $\mathrm{I}(\alpha) = \mathrm{I}(\beta)$.

Escribiremos $\alpha \equiv \beta$ para indicar que α y β son lógicamente equivalentes.

Se verifica la siguiente propiedad que se propone como un ejercicio.

Proposición 4. Para cualesquiera $\alpha, \beta \in \mathcal{F}(X)$, $\alpha \equiv \beta$ si y sólo si $\alpha \leftrightarrow \beta$ es una tautología.

En este bloque dedicado a la Lógica nos interesará transformar fórmulas dadas en otras (equivalentes) de estructura más simple. La siguiente propiedad da un primer paso en este sentido.

Proposición 5. Para cualesquiera $\alpha, \beta, \gamma \in \mathcal{F}(X)$, se verifica:

1. $\alpha \equiv \neg\neg\alpha$. (*Propiedad de la doble negación*)
2. $\alpha \equiv \alpha \vee \alpha$, $\alpha \equiv \alpha \wedge \alpha$. (*Propiedades de idempotencia*)
3. $\alpha \vee \beta \equiv \beta \vee \alpha$, $\alpha \wedge \beta \equiv \beta \wedge \alpha$. (*Propiedades conmutativas*)
4. $\alpha \vee (\beta \vee \gamma) \equiv (\alpha \vee \beta) \vee \gamma$, $\alpha \wedge (\beta \wedge \gamma) \equiv (\alpha \wedge \beta) \wedge \gamma$. (*Propiedades asociativas*)
5. $\alpha \equiv \alpha \wedge (\alpha \vee \beta)$, $\alpha \equiv \alpha \vee (\alpha \wedge \beta)$. (*Propiedades de absorción*)
6. $\alpha \wedge (\beta \vee \gamma) \equiv (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$, $\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$. (*Propiedades distributivas*)
7. $\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$, $\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$. (*Propiedades de De Morgan*)
8. $\alpha \to \beta \equiv \neg\alpha \vee \beta$. (*Equivalencia de la flecha*)
9. $\alpha \leftrightarrow \beta \equiv (\alpha \to \beta) \wedge (\beta \to \alpha)$. (*Equivalencia de la doble flecha*) $\square$

Para una proposición condicional o de implicación $\alpha \to \beta$:

- La implicación recíproca es $\beta \to \alpha$.

- La implicación contraria es $\neg\alpha \to \neg\beta$.
- La implicación contrarecíproca es $\neg\beta \to \neg\alpha$.

Proposición 6. Para dos proposiciones cualesquiera $\alpha, \beta \in \mathcal{F}(X)$, se verifica la llamada *Ley del contrarecíproco*:

$$\alpha \to \beta \equiv \neg\beta \to \neg\alpha.$$

□

Las justificaciones son inmediatas y las proponemos como ejercicio para el alumno.

5 Relación entre la Lógica de proposiciones y las álgebras de Boole.

Seguramente el alumno al ver los distintos apartados en la Proposición 5 recordará algunos de los axiomas y propiedades de las álgebras de Boole y se estará preguntando por la relación existente entre éstas y los conjuntos de fórmulas $\mathcal{F}(X)$.

Dado un conjunto X de variables proposicionales, hemos dicho anteriormente que dos fórmulas $\alpha, \beta \in \mathcal{F}(X)$ son equivalentes, si para cualquier interpretación I sobre $\mathcal{F}(X)$, se verifica que $\mathrm{I}(\alpha) = \mathrm{I}(\beta)$.

Como la propia palabra indica, estamos ante una relación de equivalencia sobre el conjunto $\mathcal{F}(X)$, cuyo conjunto cociente representamos por $\mathcal{B}(X)$. Denotamos como es usual por $[\gamma]$ la clase de equivalencia de la fórmula $\gamma \in \mathcal{F}(X)$ y definimos en $\mathcal{B}(X)$ las operaciones siguientes:

$$\overline{[\alpha]} = [\neg\alpha], \quad [\alpha]\vee[\beta] = [\alpha\vee\beta], \quad [\alpha]\wedge[\beta] = [\alpha\wedge\beta].$$

Estas operaciones sobre clases están bien definidas, es decir, el resultado no depende de los representantes elegidos dentro de cada clase de equivalencia.

Respecto de estas operaciones, el conjunto $\mathcal{B}(X)$ es un álgebra de Boole cuyo elemento cero es la clase de equivalencia en la que se encuentran todas las fórmulas de $\mathcal{F}(X)$ que son contradicciones y cuyo elemento uno es la clase de equivalencia en la que se encuentran todas las fórmulas de $\mathcal{F}(X)$ que son tautologías.

Intuitivamente, lo que estamos haciendo es agrupar en una misma clase de equivalencia todas las fórmulas pertenecientes a $\mathcal{F}(X)$ que tienen la misma tabla de verdad. Además, operar con clases se traduce en operar con las columnas de las correspondientes tablas de verdad mediante los operadores booleanos correspondientes.

Por ejemplo, si $X = \{P, Q, R\}$, entonces $\mathcal{B}(X)$ es el álgebra de Boole de ocho átomos. El elemento $[\neg P\wedge Q\wedge R]$ es un átomo de $\mathcal{B}(X)$, pues la tabla de verdad de la fórmula $\neg P\wedge Q\wedge R$ es nula salvo en una posición que vale 1.

6 Consecuencia lógica.

Dado un conjunto de fórmulas $\Omega \subseteq \mathcal{F}(X)$ y una fórmula $\beta \in \mathcal{F}(X)$, decimos que β es **consecuencia lógica** de Ω, si para toda interpretación I sobre $\mathcal{F}(X)$ se verifica que

$$\mathrm{I}(\Omega) = 1 \quad \Rightarrow \quad \mathrm{I}(\beta) = 1.$$

También se dice que Ω **implica semánticamente a** β, y se escribe $\Omega \models \beta$.

A las fórmulas del conjunto Ω se les llama **premisas** o hipótesis, y a la fórmula β la **conclusión** o tesis.

Si $\Omega = \{\alpha_1, \ldots, \alpha_n\}$ y $\Omega \models \beta$, escribimos de manera más simple

$$\alpha_1, \ldots, \alpha_n \models \beta.$$

Es inmediato que si $\beta \in \Omega$, entonces $\Omega \models \beta$.

Cuando $\Omega = \varnothing$, se escribe simplemente $\models \beta$ en vez de $\varnothing \models \beta$, y se dice que β es consecuencia lógica del conjunto vacío.

La implicación semántica mostrada en la proposición siguiente es muy común. Para su justificación, véase el Ejercicio 19.

Proposición 7 (Ley del Modus Ponendo Ponens ó Modus Ponens)**.** Para cualesquiera $\alpha, \beta \in \mathcal{F}(X)$, se verifica:

$$\Big\{\alpha,\ \alpha \to \beta\Big\} \models \beta.$$

□

La siguiente propiedad es una consecuencia inmediata de la Proposición 1.

Proposición 8. Para cualquier $\beta \in \mathcal{F}(X)$, β es una tautología si y sólo si $\models \beta$.

La siguiente propiedad se deduce fácilmente de las definiciones.

Proposición 9. Dadas dos fómulas $\beta_1, \beta_2 \in \mathcal{F}(X)$ tales que $\beta_1 \equiv \beta_2$, y dado un conjunto de fórmulas $\Omega \subseteq \mathcal{F}(X)$, se verifica que $\Omega \models \beta_1$ si y sólo si $\Omega \models \beta_2$.

En esta parte de la asignatura **el problema central** es:

Dados $\Omega \subseteq \mathcal{F}(X)$ y $\beta \in \mathcal{F}(X)$, **decidir si** $\Omega \models \beta$.

El teorema siguiente reduce este problema a otro equivalente que es el que nosotros terminaremos resolviendo.

Teorema 1. Sean $\Omega = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subseteq \mathcal{F}(X)$ y $\beta \in \mathcal{F}(X)$. Son equivalentes:
(1) $\Omega \models \beta$,
(2) $\Omega \cup \{\neg\beta\}$ es insatisfacible,
(3) La fórmula $\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_n \wedge \neg\beta$ es una contradicción.

□

Comentario 1. Un método usual de demostración en Matemáticas es el denominado **Método de demostración por reducción al absurdo**. Si queremos demostrar que una proposición lógica α implica a otra β por dicho método, suponemos la hipótesis α cierta y negamos la tesis β. Entonces, a partir de α y de $\neg\beta$ tratamos de llegar a una contradicción, es decir, a la afirmación de un hecho y de su negado. (Véase el Ejercicio 18 de la relación de ejercicios de este tema.)

Precisamente, el Teorema 1 para demostrar que el conjunto de hipótesis Ω implica semánticamente a la proposición β, considera el conjunto $\Omega \cup \{\neg\beta\}$ y trata de ver que éste es insatisfacible. La insatisfacibilidad de un conjunto de proposiciones lógicas significa que dicho conjunto implica semánticamente a una proposición, γ, y por otra parte también a su negada, $\neg\gamma$, lo cual es lo que llamamos una contradicción.

Por consiguiente, la equivalencia entre los apartados (1) y (2) en el Teorema 1 refleja el Método de demostración por reducción al absurdo en el caso de la Lógica proposicional.

Así, para demostrar $\Omega \models \beta$, si comprobamos que el conjunto $\Omega \cup \{\neg\beta\}$ es insatisfacible, diremos que hemos hecho una **demostración por refutación o por reducción al absurdo**. □

Comentario 2. La equivalencia entre los apartados (2) y (3) del teorema anterior es debida a la Proposición 3. Nótese además que la fórmula $\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_n \wedge \neg\beta$ es una contradicción, si y sólo si para toda interpretación I sobre $\mathcal{F}(X)$, se verifica

$$\mathrm{I}(\alpha_1) \cdot \mathrm{I}(\alpha_2) \cdots \mathrm{I}(\alpha_n) \cdot (1 \oplus \mathrm{I}(\beta)) = 0.$$

□

Comentario 3. Si $\Omega \subseteq \mathcal{F}(X)$ es un conjunto insatisfacible, y $\beta \in \mathcal{F}(X)$ es una fórmula cualquiera, entonces el conjunto $\Omega \cup \{\neg\beta\}$ es insatisfacible, y por consiguiente $\Omega \models \beta$. Ésto nos dice que a partir de algo falso se puede deducir cualquier cosa. □

Ejemplo 10. Sean $X = \{P, Q, R\}$ y $\Omega = \{P,\ \neg Q \to \neg P,\ Q \to R\}$. Queremos probar que

$$\Omega \models R.$$

<u>Método 1. Usar tablas de verdad.</u>

Construímos la tabla de verdad de cada una de las fórmulas de Ω y de la fórmula conclusión que es R.

P	Q	R	P	$\neg Q \to \neg P$	$Q \to R$	R
0	0	0	0	1	1	0
0	0	1	0	1	1	1
0	1	0	0	1	0	0
0	1	1	0	1	1	1
1	0	0	1	0	1	0
1	0	1	1	0	1	1
1	1	0	1	1	0	0
1	1	1	1	1	1	1

Ahora debemos fijarmos en aquellas filas en las que todas las fórmulas de Ω valen 1, y para cada una de ellas comprobar si R también vale 1.

Sólo en la octava fila todas las fórmulas de Ω valen 1 y en dicho caso R también vale 1, por lo que $\Omega \models R$.

Este método es exhaustivo ya que construye todas las interpretaciones posibles a partir de las variables proposicionales que aparecen en las fórmulas dadas, por lo cual es recomendable únicamente cuando el número de tales variables proposicionales sea pequeño.

Método 2. Resolver ecuaciones en $\mathbb{Z}_2$.

Planteamos el sistema de ecuaciones siguiente en $\mathbb{Z}_2$, cuyas incógnitas son $\mathrm{I}(P), \mathrm{I}(Q), \mathrm{I}(R)$:

$$\left.\begin{array}{l} \mathrm{I}(P) = 1 \\ \mathrm{I}(\neg Q \to \neg P) = 1 \\ \mathrm{I}(Q \to R) = 1 \end{array}\right\}$$

Queremos ver que toda solución del sistema verifica $\mathrm{I}(R) = 1$. Un forma de resolverlo se basa en la definición de $\mathrm{I}(\alpha \to \beta)$.

Comenzamos resolviendo el sistema formado por las dos primeras ecuaciones:

$$\left.\begin{array}{l} \mathrm{I}(P) = 1 \\ \mathrm{I}(\neg Q \to \neg P) = 1 \end{array}\right\}.$$

Al ser

$$\mathrm{I}(\neg Q \to \neg P) = 1 \oplus \mathrm{I}(\neg Q) \oplus \mathrm{I}(\neg Q) \cdot \mathrm{I}(\neg P),$$

la segunda ecuación queda como

$$1 \oplus (1 \oplus \mathrm{I}(Q)) \oplus (1 \oplus \mathrm{I}(Q)) \cdot (1 \oplus \mathrm{I}(P)) = 1.$$

Aplicamos la propiedad distributiva del producto respecto de la suma exclusiva y resulta

$$1 \oplus 1 \oplus \mathrm{I}(Q) \oplus 1 \oplus \mathrm{I}(Q) \oplus \mathrm{I}(P) \oplus \mathrm{I}(Q) \cdot \mathrm{I}(P)) = 1,$$

que equivale a
$$1 \oplus \mathrm{I}(P) \oplus \mathrm{I}(P) \cdot \mathrm{I}(Q) = 1,$$
es decir,
$$\mathrm{I}(P) \oplus \mathrm{I}(P) \cdot \mathrm{I}(Q) = 0,$$
pues recuérdese que estamos operando con la aritmética del cuerpo $\mathbb{Z}_2$.

Ahora usamos la primera ecuación $\mathrm{I}(P) = 1$ y obtenemos que
$$1 \oplus 1 \cdot \mathrm{I}(Q) = 0,$$
lo que implica que
$$\mathrm{I}(Q) = 1.$$

A continuación procedemos de manera similar con las ecuaciones
$$\mathrm{I}(Q) = 1 \qquad \text{y} \qquad \mathrm{I}(Q \to R) = 1,$$
y llegamos a que $\mathrm{I}(R) = 1$, que es lo que queríamos obtener.

Método 3. Aplicar el apartado (3) del Teorema 1.

Comprobamos que la fórmula
$$P \wedge (P \to Q) \wedge (Q \to R) \wedge \neg R$$
es una contradicción. Para ello consideramos una interpretación I cualquiera sobre $\mathcal{F}(X)$ y calculamos
$$\mathrm{I}(P) \cdot \mathrm{I}(P \to Q) \cdot \mathrm{I}(Q \to R) \cdot (1 \oplus \mathrm{I}(R)).$$
Si aplicamos las reglas de cálculo para I, representamos $\mathrm{I}(P), \mathrm{I}(Q), \mathrm{I}(R)$ por p, q, r, respectivamente, y usamos por ejemplo Maxima, resulta:
$$p^2\,q^2\,r^2 \oplus p^2\,q\,r^2 \oplus p\,q\,r^2 \oplus 2\,p^2\,q^2\,r \oplus 3\,p^2\,q\,r \oplus$$
$$2\,p\,q\,r \oplus p^2\,r \oplus p\,r \oplus p^2\,q^2 \oplus 2\,p^2\,q \oplus p\,q \oplus p^2 \oplus p.$$
Ahora usamos las reglas de simplificación en $\mathbb{Z}_2$ y resulta:
$$pqr \oplus pqr \oplus pqr \oplus 0pqr \oplus pqr \oplus 0pqr \oplus$$
$$pr \oplus pr \oplus pq \oplus 0pq \oplus pq \oplus p \oplus p =$$
$$4pqr \oplus 2pr \oplus 2pq \oplus 2p = 0.$$

Como se puede ver, de esta forma los cálculos son muy engorrosos.

En este ejemplo particular, si observamos que la proposición $\neg Q \to \neg P$ es equivalente a $P \to Q$ por la Ley del contrarecíproco, entonces la implicación semática
$$\Omega \models R$$

se verifica si y sólo si se verifica

$$\Big\{P,\ P \to Q,\ Q \to R\Big\} \models R.$$

Ahora bien, por la Regla del Modus ponens, se cumple que

$$\Big\{P,\ P \to Q\Big\} \models Q,$$

y por tanto

$$\Big\{P,\ P \to Q,\ Q \to R\Big\} \models Q.$$

De nuevo por la Regla del Modus ponens, tenemos que

$$\Big\{Q,\ Q \to R\Big\} \models R,$$

con lo cual

$$\Big\{P,\ P \to Q,\ Q \to R\Big\} \models R.$$

$\square$

Ejemplo 11. Sean $X = \{P, Q, R\}$ y $\Omega = \{P \to Q, Q \to R\}$. Queremos comprobar que

$$\Omega \models (P \vee Q) \to R.$$

Método 1.

Construímos la tabla de verdad de cada una de las fórmulas de Ω y de la fórmula conclusión que es $(P \vee Q) \to R$.

P	Q	R	$P \to Q$	$Q \to R$	$(P \vee Q) \to R$
0	0	0	1	1	1
0	0	1	1	1	1
0	1	0	1	0	0
0	1	1	1	1	1
1	0	0	0	1	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	1

Ahora debemos fijarmos en aquellas filas en las que todas las fórmulas de Ω valen 1, y para cada una de ellas comprobar si $(P \vee Q) \to R$ también vale 1.

En las filas primera, segunda, cuarta y octava, todas las fórmulas de Ω valen 1 y en dicho caso $(P \vee Q) \to R$ también, por lo que

$$\Omega \models (P \vee Q) \to R.$$

Método 2.

Partimos del sistema de ecuaciones siguiente en $\mathbb{Z}_2$:

$$\left.\begin{array}{l} \mathrm{I}(P \to Q) = 1 \\ \mathrm{I}(Q \to R) = 1 \end{array}\right\}$$

Queremos ver que toda solución del sistema verifica $\mathrm{I}((P \vee Q) \to R) = 1$.

Las soluciones de la ecuación $\mathrm{I}(P \to Q) = 1$ son:

$$(0,0,0),\ (0,0,1),\ (0,1,0),\ (0,1,1),\ (1,1,0),\ (1,1,1).$$

De éstas, son además solución de la ecuación $\mathrm{I}(Q \to R) = 1$, y por tanto del sistema anterior, las siguientes:

$$(0,0,0),\ (0,0,1),\ (0,1,1),\ (1,1,1).$$

Es inmediato comprobar que cada una de estas soluciones verifica la ecuación

$$\mathrm{I}((P \vee Q) \to R) = 1.$$

$\square$

Como se ha puesto de manifiesto en los ejemplos anteriores, los métodos que tenemos disponibles hasta este momento para decidir la insatisfacibilidad de un conjunto de proposiciones lógicas son bastante limitados. En la parte final de este tema estudiaremos el Algoritmo de Davis-Putnam que nos permite resolver dicho problema de forma más efectiva.

La propiedad siguiente es una consecuencia de la Proposición 2 y del Teorema 1, y nos dice que ante un problema de implicación semántica, podemos suprimir todas las tautologías pertenecientes al conjunto de premisas sin que ello altere la respuesta al problema inicial.

Proposición 10. Sean $\Omega \subseteq \mathcal{F}(X)$ y $\alpha \in \mathcal{F}(X)$, una tautología. Entonces

$$\Omega \cup \{\alpha\} \models \beta \text{ si y sólo si } \Omega \models \beta.$$

Una herramienta que a veces resulta útil en los problemas de consecuencia lógica es el denominado **Teorema de la deducción** que mostramos a continuación.

Teorema 2 (Teorema de la deducción)**.** Dados un conjunto de fórmulas $\Omega \subseteq \mathcal{F}(X)$ y $\alpha, \beta \in \mathcal{F}(X)$, las siguientes afirmaciones son equivalentes:

1. $\Omega \models \alpha \to \beta$.
2. $\Omega \cup \{\alpha\} \models \beta$.

$\square$

Ejemplo 12. Probamos que la fórmula

$$(P \to Q) \to ((P \to (Q \to R)) \to (P \to R))$$

es una tautología, es decir,

$$\models (P \to Q) \to ((P \to (Q \to R)) \to (P \to R)).$$

Aplicamos varias veces el Teorema de la deducción, y obtenemos las siguientes implicaciones semánticas equivalentes entre sí:

1ª vez: $\left\{P \to Q\right\} \models (P \to (Q \to R)) \to (P \to R)$.

2ª vez: $\left\{P \to Q,\ P \to (Q \to R)\right\} \models P \to R$.

3ª vez: $\left\{P \to Q,\ P \to (Q \to R),\ P\right\} \models R$.

Llegados a este punto, hemos de aplicar alguno de los métodos anteriores. Por ejemplo, planteamos el sistema de ecuaciones siguiente en $\mathbb{Z}_2$:

$$\left.\begin{array}{l} \mathrm{I}(P \to Q) = 1 \\ \mathrm{I}(P \to (Q \to R)) = 1 \\ \mathrm{I}(P) = 1 \end{array}\right\}.$$

Para resolverlo, comenzamos con el subsistema

$$\left.\begin{array}{l} \mathrm{I}(P \to Q) = 1 \\ \mathrm{I}(P) = 1 \end{array}\right\}$$

cuyas soluciones son:

$$(1,1,0),\ (1,1,1).$$

De éstas, sólo la segunda verifica la ecuación $\mathrm{I}(P \to (Q \to R)) = 1$.

Por tanto $(1,1,1)$ es la única solución del sistema

$$\left.\begin{array}{l} \mathrm{I}(P \to Q) = 1 \\ \mathrm{I}(P \to (Q \to R)) = 1 \\ \mathrm{I}(P) = 1 \end{array}\right\}.$$

Como además esta solución verifica la ecuación $\mathrm{I}(R) = 1$, obtenemos que la implicación semántica

$$\left\{P \to Q,\ P \to (Q \to R),\ P\right\} \models R$$

es cierta, lo que concluye la comprobación.

Otro camino alternativo sería, una que vez hemos planteado problema $\Omega \models R$, donde

$$\Omega := \Big\{ P \to Q,\ P \to (Q \to R),\ P \Big\},$$

observar que

$$\Omega \models Q \qquad \text{y} \qquad \Omega \models Q \to R,$$

por la Regla del Modus ponens. Como además

$$\Big\{ Q,\ Q \to R \Big\} \models R,$$

concluímos finalmente que $\Omega \models R$. $\square$

Ejemplo 13. Comprobamos la Ley del contrarecíproco haciendo uso del Teorema de la deducción, es decir, que para cualesquiera dos fórmulas $\alpha, \beta \in \mathcal{F}(X)$, la fórmula

$$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$$

es una tautología. Ésto equivale a

$$\models (\alpha \to \beta) \to (\neg\beta \to \neg\alpha).$$

Aplicamos el Teorema de la deducción dos veces y llegamos a

$$\Big\{ \alpha \to \beta, \neg\beta \Big\} \models \neg\alpha,$$

lo cual por el Teorema 1, equivale a que el conjunto

$$\Big\{ \alpha \to \beta, \neg\beta,\ \neg\neg\alpha \Big\}$$

sea insatisfacible, es decir, que el conjunto

$$\Omega := \Big\{ \alpha \to \beta, \neg\beta,\ \alpha \Big\}$$

sea insatisfacible.

Como por la Regla del Modus ponens se cumple que

$$\Omega \models \beta,$$

y por otra parte

$$\Omega \models \neg\beta,$$

ésto significa que Ω es insatisfacible.

$\square$

7 Cláusulas.

Un **literal** en $\mathcal{F}(X)$ es una variable proposicional o su negada.

Por ejemplo, si $X = \{P, Q, R\}$, entonces P y $\neg Q$ son literales. No es un literal la fórmula $\neg\neg Q$, aún cuando esta fórmula es equivalente a Q que sí es un literal.

Si λ es un literal, denotamos por λ^c el literal que es equivalente a $\neg\lambda$. Por ejemplo, si $\lambda = P$, entonces $\lambda^c = \neg P$ y si $\lambda = \neg P$, entonces $\lambda^c = P$.

Una **cláusula** es una disyunción de literales. Aclaramos que todo literal es una cláusula. También se considera como cláusula a la "disyunción de cero literales", denominándose ésta la **cláusula vacía**. Denotamos la cláusula vacía por $\square$.

Así podemos decir que una cláusula es una disyunción de cero o más literales.

La cláusula vacía va a representar una situación de insatisfacibilidad. Por definición, **todo conjunto de cláusulas que contenga a la cláusula vacía, será insatisfacible**.

Ejemplo 14. Dado $X = \{P, Q, R, S\}$, las siguientes fórmulas de $\mathcal{F}(X)$ son cláusulas:

$$P, \quad \neg Q, \quad \neg P \vee Q, \quad R \vee P \vee R \vee R, \quad R \vee Q \vee S, \quad \neg P \vee \neg Q \vee \neg R \vee \neg S.$$

$\square$

Ya que, $\alpha \vee \alpha \equiv \alpha$, siempre que en una cláusula C aparezca un literal λ más de una vez, al escribir C sólo escribiremos una ocurrencia del literal λ.

Así, en el ejemplo anterior nos quedamos con $R \vee P$ ó bien con $P \vee R$, en vez de $R \vee P \vee R \vee R$.

Si en una cláusula C aparecen los literales λ y λ^c, se dice que C es una **cláusula tautológica**.

Por ejemplo,

$$P \vee \neg P, \quad P \vee Q \vee \neg P \vee R,$$

son cláusulas tautológicas.

Una fórmula de $\mathcal{F}(X)$ se dice que está en **forma clausulada** si es una cláusula o está expresada como conjunción de cláusulas.

Ejemplo 15. Dado $X = \{P, Q, R, S\}$, son fórmulas en forma clausulada,

$$P, \quad \neg R, \quad \neg Q \vee P, \quad P \vee Q \vee \neg R,$$

pues todas ellas son cláusulas. Las siguientes fórmulas también están en forma clausula, al ser conjunción de dos ó más cláusulas:

$$\neg Q \wedge P, \quad (\neg Q \vee P) \wedge P, \quad P \wedge \neg P, \quad (P \vee \neg R) \wedge (\neg Q \vee \neg P) \wedge Q.$$

$\square$

Teorema 3. Toda fórmula α_1 puede ser transformada en una fórmula α_2 lógicamente equivalente con α_1, de modo que α_2 esté en forma clausulada.

La justificación del teorema anterior se basa en las equivalencias lógicas dadas en la Proposición 5 y consiste en aplicar las siguientes reglas tantas veces como haga falta:

1. Si aparece alguna subfórmula del tipo $\alpha\leftrightarrow\beta$, la reemplazamos por $(\alpha \to \beta)\wedge(\beta \to \alpha)$.
2. Si hay alguna subfórmula del tipo $\alpha \to \beta$, la reemplazamos por $\neg\alpha\vee\beta$.
3. Toda subfórmula del tipo $\neg\neg\alpha$ la substituímos por α.
4. Interiorizamos las negaciones dentro de los paréntesis mediante las leyes de De Morgan:
$$\neg(\alpha\wedge\beta) \equiv \neg\alpha\vee\neg\beta, \quad \neg(\alpha\vee\beta) \equiv \neg\alpha\wedge\neg\beta.$$
   De esta manera llegamos a una fórmula en la que sólo pueden aparecer algunas de las conectivas siguientes: $\neg, \vee, \wedge$.

   Además, si aparece $\neg$, ésta sólo actúa sobre variables proposicionales.
5. Ahora puede ser necesario el uso de las leyes distributivas para conseguir que dentro de los paréntesis no aparezca ninguna conectiva $\wedge$.
6. Si es posible, aplicamos la regla $(\alpha\vee\beta)\wedge\alpha \equiv \alpha$ (Ley de absorción).

De este modo obtenemos una forma clausulada de la fórmula inicial.

A continuación podemos aplicar las siguientes **optimizaciones**:

1. Si la fórmula inicial no era tautología, suprimimos las cláusulas tautológicas que haya.
2. Si en alguna de las cláusulas resultantes aparece un literal λ dos o más veces, lo escribimos una sóla vez en dicha cláusula, pues $\lambda\vee\lambda \equiv \lambda$. □

Ejemplo 16. Obtenemos una forma clausulada para la fórmula $\alpha = (P \to Q)\leftrightarrow(\neg P\wedge Q)$.

$$\alpha \equiv \Big((P \to Q) \to (\neg P\wedge Q)\Big)\wedge\Big((\neg P\wedge Q) \to (P \to Q)\Big) \equiv$$
$$\Big(\neg(\neg P\vee Q)\vee(\neg P\wedge Q)\Big)\wedge\Big(\neg(\neg P\wedge Q)\vee(\neg P\vee Q)\Big) \equiv$$
$$\Big((P\wedge\neg Q)\vee(\neg P\wedge Q)\Big)\wedge\Big((P\vee\neg Q)\vee(\neg P\vee Q)\Big) \equiv$$
$$(P\vee\neg P) \wedge (\neg Q\vee\neg P) \wedge (P\vee Q) \wedge (\neg Q\vee Q) \wedge (P\vee\neg Q\vee\neg P\vee Q).$$

Ésto ya es una forma clausulada de α. Finalmente suprimimos las cláusulas tautológicas y obtenemos una forma clausulada de α más simple:

$$(\neg Q\vee\neg P) \wedge (P\vee Q).$$

□

Como se ha visto en el ejemplo anterior, la forma clausulada no es única. Por ejemplo, en algunas cláusulas podríamos insertar variables proposicionales que no aparecen en tales cláusulas:

$$(\neg Q \vee \neg P) \wedge (P \vee Q) \equiv (\neg Q \vee \neg P) \wedge (P \vee Q \vee R) \wedge (P \vee Q \vee \neg R).$$

Obtenemos así otra forma clausulada de α, aunque como es lógico, nos quedamos con la que ya teníamos que es más simple.

Ejemplo 17. Obtenemos una forma clausulada para $\alpha = (\neg P \wedge Q \wedge R) \vee (\neg P \wedge \neg Q)$.

Comenzamos aplicando la propiedad distributiva.

$\alpha \equiv \Big((\neg P \wedge Q \wedge R) \vee \neg P\Big) \wedge \Big((\neg P \wedge Q \wedge R) \vee \neg Q\Big)$.

Ahora podemos aplicar una ley de absorción, pues $(\neg P \wedge Q \wedge R) \vee \neg P \equiv \neg P$. Resulta:

$\alpha \equiv \neg P \wedge \Big((\neg P \wedge Q \wedge R) \vee \neg Q\Big)$.

De nuevo usamos una propiedad distributiva. Obtenemos

$\neg P \wedge \Big((\neg P \wedge Q \wedge R) \vee \neg Q\Big) \equiv \neg P \wedge \Big((\neg P \vee \neg Q) \wedge (Q \vee \neg Q) \wedge (R \vee \neg Q)\Big)$.

Eliminamos las cláusulas tautológicas y obtenemos una forma clausulada de α:

$$\alpha \equiv \neg P \wedge (\neg P \vee \neg Q) \wedge (R \vee \neg Q).$$

Incluso si aplicamos la ley de absorción, podemos obtener otra forma clausulada más simple todavía para α:

$$\alpha \equiv \neg P \wedge (R \vee \neg Q).$$

$\square$

8 El Algoritmo de Davis-Putnam.

Presentamos en esta última sección un nuevo método para resolver el problema de la implicación semántica, es decir, dados un conjunto de fórmulas $\Omega \subseteq \mathcal{F}(X)$ y una fórmula $\alpha \in \mathcal{F}(X)$, decidir si $\Omega \models \alpha$.

Tal y como hemos visto en el Teorema 1,

$$\Omega \models \alpha \quad \Leftrightarrow \quad \Omega \cup \{\neg \alpha\} \text{ es insatisfacible.}$$

La propiedad siguiente nos dice que estudiar la insatisfacibilidad de un subconjunto de fórmulas de $\mathcal{F}(X)$ es equivalente a estudiar la insatisfacibilidad de un subconjunto de cláusulas de $\mathcal{F}(X)$.

Proposición 11. Sean $\Delta \subseteq \mathcal{F}(X)$ un conjunto de fórmulas, Δ' el conjunto que se obtiene a partir de Δ reemplazando cada fórmula por una forma clausulada correspondiente, y sea Δ'' el conjunto que resulta de substituir cada fórmula en Δ' por las cláusulas que la componen. Entonces:

$$\Delta \text{ es insatisfacible } \Leftrightarrow \Delta' \text{ es insatisfacible } \Leftrightarrow \Delta'' \text{ es insatisfacible.}$$

Por tanto, vemos que **el problema de decidir si una fórmula es consecuencia lógica de un conjunto de fórmulas se reduce al problema de decidir si un conjunto de cláusulas es insatisfacible o no**.

Ejemplo 18. Con la notación de la propiedad anterior, si

$$\Delta = \Big\{P \rightarrow Q,\ R \rightarrow (P \wedge Q)\Big\},$$

entonces

$$\Delta' = \Big\{\neg P \vee Q,\ (\neg R \vee P) \wedge (\neg R \vee Q)\Big\},$$

y

$$\Delta'' = \Big\{\neg P \vee Q,\ \neg R \vee P,\ \neg R \vee Q\Big\}.$$

□

Presentamos a continuación el **Algoritmo de Davis-Putnam**, el cual permite decidir si un conjunto finito de cláusulas es o no satisfacible. Para ello damos la siguiente notación:

Si C es una cláusula y λ es un literal que aparece en C, denotamos por $C-\lambda$ la cláusula que resulta de suprimir todas las ocurrencias de λ en C.

Por ejemplo, si

$$C = P \vee \neg Q \vee R,$$

entonces

$$C - \neg Q = P \vee R.$$

Si

$$C = P,$$

entonces

$$C - P = \square,$$

la cláusula vacía.

El algoritmo de Davis-Putnam se basa en **las tres reglas** siguientes, las cuales permiten ir reduciendo el conjunto de cláusulas.

1. Sea Σ un conjunto de cláusulas y supongamos que en Σ hay una <u>cláusula unitaria</u>, es decir, aquella que consiste en un único literal λ. Sea Σ' el conjunto que resulta de Σ al suprimir todas las cláusulas que contengan el literal λ y substituir las cláusulas

C que contengan a λ^c por $C - \lambda^c$. Las cláusulas de Σ en las que no aparece λ ni λ^c se añaden "intactas" a Σ'. Es decir, si

$$\Sigma = \Big\{\lambda, \lambda \vee C_1, \ldots, \lambda \vee C_m,$$

$$\lambda^c \vee C_{m+1}, \ldots, \lambda^c \vee C_{m+n},$$

$$C_{m+n+1}, \ldots, C_{m+n+p}\Big\},$$

donde ninguna de las cláusulas $C_{m+n+1}, \ldots, C_{m+n+p}$ contiene a λ ó a λ^c, entonces

$$\Sigma' = \Big\{C_{m+1}, \ldots, C_{m+n}, C_{m+n+1}, \ldots, C_{m+n+p}\Big\}.$$

En tal caso se verifica que Σ es satisfacible si y sólo si Σ' es satisfacible.

2. Supongamos que en el conjunto Σ existe un literal puro, es decir, un literal λ tal que λ^c no aparece en ninguna cláusula de Σ. Sea entonces Σ' el conjunto de cláusulas que resulta de suprimir todas las cláusulas de Σ en las que aparece el literal λ. Es decir, si
$$\Sigma = \Big\{\lambda \vee C_1, \ldots, \lambda \vee C_m, C_{m+1}, \ldots, C_{m+n}\Big\},$$
donde ninguna de las cláusulas $C_{m+1}, \ldots, C_{m+n}$ contiene a λ, entonces
$$\Sigma' = \Big\{C_{m+1}, \ldots, C_{m+n}\Big\}.$$
En tal caso se verifica que Σ es satisfacible si y sólo si Σ' es satisfacible.

3. Sea Σ un conjunto de cláusulas y a una proposición atómica. Supongamos que Σ viene dado por:
$$\Sigma = \Big\{a \vee C_1, \ldots, a \vee C_m,$$
$$\neg a \vee C_{m+1}, \ldots, \neg a \vee C_{m+n},$$
$$C_{m+n+1}, \ldots, C_{m+n+p}\Big\}.$$
donde ninguna de las cláusulas $C_{m+n+1}, \ldots, C_{m+n+p}$ contiene a a ó a $\neg a$. Sean
$$\Sigma_1 = \{C_{m+1}, \ldots, C_{m+n}, C_{m+n+1}, \ldots, C_{m+n+p}\}$$
y
$$\Sigma_2 = \{C_1, \ldots, C_m, C_{m+n+1}, \ldots, C_{m+n+p}\}.$$
En tal caso, Σ es satisfacible si y sólo si Σ_1 es satisfacible ó Σ_2 es satisfacible. De modo equivalente, Σ es insatisfacible si y sólo si Σ_1 y Σ_2 son ambos insatisfacibles. $\square$

Obsérvese que en la Regla (3.) anterior, Σ_1 es el conjunto que se obtendría a partir de Σ como si hubiésemos aplicado la Regla 1 con $\lambda = a$ y Σ_2 es el conjunto que se obtendría a partir de Σ como si hubiésemos aplicado la Regla 1 con $\lambda = \neg a$.

Recordamos que por definición todo conjunto en el que aparezca la cláusula vacía es siempre insatisfacible, ya que bajo cualquier interpretación I se verifica que $\mathrm{I}(\square) = 0$.

Vemos ya el Algoritmo de Davis-Putnam para decidir la insatisfacibilidad de un conjunto de cláusulas Σ:

1. Tratamos de aplicar la Regla 1, es decir, comprobamos si hay alguna cláusula unitaria. Si la respuesta es afirmativa, y λ es dicha cláusula, reducimos el conjunto de cláusulas tal y como hemos visto en la Regla 1, anotamos el literal λ y volvemos al principio con el nuevo conjunto de cláusulas.

2. Si la respuesta es negativa, comprobamos si existe un literal puro. De existir, llamémoslo λ, procedemos como en la Regla 2, anotamos el literal λ y volvemos al inicio con el nuevo conjunto de cláusulas.

3. De no existir tampoco un literal puro, elegimos una variable proposicional a que aparezca en alguna de las cláusulas del conjunto actual, abrimos dos ramas, tal y como hemos visto en la Regla 3, y analizamos cada una de las ramas. Al analizar la rama con Σ_1, anotamos el literal a, mientras que al analizar Σ_2, anotamos el literal $\neg a$.

4. El algoritmo acaba cuando obtenemos un conjunto satisfacible, en cuyo caso Σ también es satisfacible y una interpretación que satisface a Σ es aquella que vale 1 sobre todos los literales que hemos ido anotando, o bien, si en todas las ramas se obtiene un conjunto insatisfacible, en cuyo caso el conjunto inicial Σ será insatisfacible. $\square$

Ejemplo 19. El conjunto $\Sigma = \{P\}$ es claramente satisfacible, pues tomando cualquier interpretación I tal que $\mathrm{I}(P) = 1$, la única fórmula en Σ se hace verdadera. Veamos cómo se puede obtener esa misma conclusión aplicando el Algoritmo de Davis-Putnam.

En Σ hay un cláusula unitaria. Si aplicamos la Regla 1 con $\lambda = P$, obtenemos el conjunto

$$\Sigma' = \{\} = \varnothing,$$

el cual es satisfacible. (Recuérdese la Proposición 1.) Por tanto Σ es satisfacible. $\square$

Ejemplo 20. El conjunto $\Sigma = \{P, \neg P\}$ es claramente insatisfacible. Veamos cómo la aplicación de las reglas anteriores conducen a la misma conclusión.

Podemos aplicar la Regla 1, pues tanto P como $\neg P$ son cláusulas unitarias. Si aplicamos la Regla 1 con $\lambda = P$, y puesto que

$$\neg P - \lambda^c = \neg P - \neg P = \square,$$

obtenemos que

$$\Sigma' = \Big\{\Box\Big\},$$

que es insatisfacible, y por tanto Σ es también insatisfacible. $\Box$

Ejemplo 21. Al conjunto

$$\Sigma = \Big\{P \vee Q,\ P \vee \neg Q\Big\}$$

no le podemos aplicar la Regla 1 al no haber ninguna cláusula unitaria. Sin embargo la Regla 2 sí es aplicable pues $\lambda = P$ es un literal puro en Σ. Resulta

$$\Sigma' = \{\},$$

que es satisfacible, por lo cual Σ es satisfacible.

Esta misma conclusión la podríamos haber obtenido eligiendo cualquier interpretación I tal que $\mathrm{I}(P) = 1$, independientemente del valor de $\mathrm{I}(Q)$. $\Box$

Ejemplo 22. Al conjunto

$$\Sigma = \Big\{P \vee \neg Q, \neg P \vee Q\Big\}$$

no le podemos aplicar la Regla 1 ni la Regla 2. No contiene cláusulas unitarias ni literales puros.

Aplicamos la Regla 3 con $a = P$ y obtenemos que Σ es satisfacible si y sólo si $\Sigma_1 = \{Q\}$ es satisfacible ó $\Sigma_2 = \{\neg Q\}$ es satisfacible.

En este caso Σ_1 y Σ_2 son ambos evidentemente satisfacibles. Deducimos así que Σ es satisfacible.

Una interpretación que satisface a Σ es

$$\mathrm{I}(P) = 0,\ \mathrm{I}(Q) = 0.$$

Otra es

$$\mathrm{I}(P) = 1,\ \mathrm{I}(Q) = 1.$$

$\Box$

Ejemplo 23. Sea el conjunto de proposiciones

$$\Sigma = \Big\{P \vee Q,\ P \vee \neg Q,\ \neg P \vee R,\ \neg P \vee \neg R\Big\}.$$

Vemos que no es aplicable la Regla 1 ni la 2. Aplicamos la Regla 3 con $a = P$ y obtenemos los conjuntos

$$\Sigma_1 = \Big\{R, \neg R\Big\}$$

y

$$\Sigma_2 = \Big\{Q, \neg Q\Big\}.$$

Como cada uno de ellos es insatisfacible, resulta que Σ es insatisfacible. $\Box$

Ejemplo 24. Dado
$$\Sigma = \Big\{P\vee\neg Q,\ \neg P\vee R,\ \neg P\vee\neg R\Big\},$$
no le podemos aplicar la Regla 1 al no haber ninguna cláusula unitaria.

Sí hay un literal puro $\lambda = \neg Q$. Aplicamos la Regla 2 a Σ y resulta
$$\Sigma' = \Big\{\neg P\vee R,\ \neg P\vee\neg R\Big\}.$$

No podemos aplicar la Regla 1 a Σ'. Sin embargo $\lambda = \neg P$ es un literal puro para Σ'. Aplicamos la Regla 2 a Σ' y obtenemos
$$\Sigma'' = \varnothing,$$
que es satisfacible.

Así, Σ es satisfacible. Además una interpretación que satisface a Σ se obtiene a partir de los literales anotados. Concretamente $\mathrm{I}(Q) = 0$, $\mathrm{I}(P) = 0$ e $\mathrm{I}(R)$ indistintamente 0 ó 1. $\square$

Ejemplo 25. Dado el conjunto
$$\Omega = \Big\{\neg P \to (Q\wedge R),\ R \to (\neg P \to R),\ (Q\vee S) \to R\Big\},$$
estudiamos si es cierto ó no que $\Omega \models R\wedge P$.

Por el Teorema 1, ésto equivale a que el conjunto
$$\Omega' = \Big\{\neg P \to (Q\wedge R),\ R \to (\neg P \to R),\ (Q\vee S) \to R,\ \neg(R\wedge P)\Big\}$$
sea insatisfacible.

Calculamos una forma clausulada para cada una de las fórmulas pertenecientes a Ω'.

- $\neg P \to (Q\wedge R) \equiv P \vee (Q\wedge R) \equiv (P\vee Q) \wedge (P\vee R)$.
- $R \to (\neg P \to R) \equiv \neg R\vee P\vee R$.
- $(Q\vee S) \to R \equiv \neg(Q\vee S) \vee R \equiv (\neg Q\wedge\neg S) \vee R \equiv (\neg Q\vee R) \wedge (\neg S\vee R)$.
- $\neg(R\wedge P) \equiv \neg R\vee\neg P$.

En el punto segundo hemos obtenido una cláusula tautológica, la cual descartamos. Por tanto, resulta el conjunto siguiente de cláusulas al cual le aplicamos el Algoritmo de Davis-Putnam:

$$\Sigma = \Big\{P{\vee}Q,\ P{\vee}R,\ \neg Q{\vee}R,\ \neg S{\vee}R,\ \neg R{\vee}\neg P\Big\}.$$

En Σ no hay cláusulas unitarias. Sin embargo $\lambda = \neg S$ es literal puro, por lo que aplicamos la Regla 2 y resulta

$$\Sigma' = \Big\{P{\vee}Q,\ P{\vee}R,\ \neg Q{\vee}R,\ \neg R{\vee}\neg P\Big\}.$$

En Σ' no hay cláusulas unitarias ni literales puros, por lo que aplicamos la Regla 3 con $a = P$ y obtenemos

$$\Sigma'_1 = \Big\{\neg R,\ \neg Q{\vee}R\Big\}$$

y

$$\Sigma'_2 = \Big\{Q,\ R,\ \neg Q{\vee}R\Big\}.$$

Analizamos Σ'_1, al que le aplicamos la Regla 1 con $\lambda = \neg R$ y resulta $\Sigma''_1 = \Big\{\neg Q\Big\}$ que es satisfacible.

Por consiguiente Σ' es satisfacible y así Σ también lo es. Deducimos de todo ésto que la implicación semántica propuesta, $\Omega \models R{\wedge}P$, no se verifica.

Además, usando las anotaciones de los literales obtenemos una interpretación I que satisface a Σ, y por tanto también a Ω, pero $\mathrm{I}(R{\wedge}P) = 0$.

De la satisfacibilidad de Σ''_1 resulta $\mathrm{I}(Q) = 0$. De la obtención de Σ''_1 a partir de Σ'_1 tenemos $\mathrm{I}(R) = 0$. Al haber considerado Σ'_1 desde Σ', tenemos $\mathrm{I}(P) = 1$, pues $a = P$. De la obtención de Σ' a partir de Σ, resulta $\mathrm{I}(S) = 0$.

Por tanto $\mathrm{I}(P) = 1$, $\mathrm{I}(Q) = 0$, $\mathrm{I}(R) = 0$ e $\mathrm{I}(S) = 0$. □

Tema 6: Formas normales, unificación, resolución.

(Apuntes elaborados por Juan Manuel Urbano Blanco)

Índice

1 Introducción

En este tema desarrollamos técnicas apropiadas para decidir si un conjunto de fórmulas de un lenguaje de predicados de primer orden $\mathcal{L}$, es insatisfacible ó inconsistente, y por tanto, basándonos en el Teorema 2 de la Sección 7 del Tema 5, podremos saber si una fórmula de $\mathcal{L}$ es o no consecuencia lógica de un subconjunto de $\mathcal{L}$.

2 Formas normales

Dado un conjunto de fórmulas Γ, vamos a aplicarle a cada fórmula en Γ ciertas transformaciones de manera que el conjunto resultante de fórmulas sea insatisfacible si y sólo si Γ también lo es, y con la ventaja adicional de que las transformaciones darán lugar a **fórmulas más "simples"** que las fórmulas de partida.

Para cada fórmula $\alpha \in \Gamma$ comenzamos calculando una forma normal prenexa, y entonces a partir de ésta última, obtenemos una forma normal de Skolem de α. Por último, nos restringiremos sólo a conjuntos formados por sentencias, y a partir de las correspondientes formas normales de Skolem, obtendremos sus formas clausuladas. La idea de todo este proceso es un generalización de la teoría estudiada en el Tema 4.

2.1 Forma normal prenexa

Una fórmula α está en **forma normal prenexa**, si en ella no aparecen cuantificadores o bien es de la forma

$$C_1\ x_1 \ldots C_n\ x_n\ \beta,$$

donde cada C_i es un cuantificador $\forall$ ó $\exists$, y β es una fórmula en la cual no hay cuantificadores.

Ejemplo 1. Sea $\mathcal{L}$ un lenguaje de predicados de primer orden tal que

$$\text{Cons}(\mathcal{L}) = \{a\}, \quad \text{Func}(\mathcal{L}) = \{f^1\} \ \text{ y } \ \text{Rel}(\mathcal{L}) = \{P^1, Q^2, R^0\}.$$

Las siguientes fórmulas de $\mathcal{L}$ están en forma normal prenexa:

$$R, \quad Q(x, f(a)), \quad P(x) \to R, \quad \forall x P(x), \quad \forall x \exists y \Big(Q(a,x) \wedge P(f(y)) \Big), \quad \exists y \exists x \forall y \neg P(a).$$

Las siguientes no están en forma normal prenexa:

$$\forall x P(x) \to P(f(a)), \quad P(a) \to \exists x P(x), \quad \neg \exists y \Big(P(y) \wedge Q(a,y) \Big), \quad \forall x \neg \exists y Q(x,y).$$

$\square$

Teorema 1. Para toda fórmula α existe otra fórmula α^* en forma normal prenexa semánticamente equivalente a α. $\square$

La demostración se hace aplicando las propiedades de equivalencia lógica estudiadas al final del tema anterior.

Ejemplo 2. Obtengamos una forma normal prenexa para la fórmula siguiente

$$\alpha := P(y) \to \forall x R(x, a).$$

Sabemos que

$$\alpha \equiv \neg P(y) \vee \forall x R(x, a).$$

Como en la primera parte de la fórmula, es decir, en $\neg P(y)$, no hay ocurrencias libres de x, ampliamos el radio de acción del cuantificador $\forall x$, y obtenemos que

$$\neg P(y) \vee \forall x R(x,a) \ \equiv \ \forall x \Big(\neg P(y) \vee R(x,a) \Big).$$

Por consiguiente, la fórmula

$$\forall x \Big(\neg P(y) \vee R(x,a) \Big)$$

es una forma normal prenexa de α.

Nótese que la fórmula siguiente, equivalente a la anterior, es también una forma normal prenexa de α:

$$\forall x \Big(P(y) \to R(x,a) \Big).$$

$\square$

Ejemplo 3. Calculamos una forma normal prenexa para la fórmula siguiente

$$\alpha := \exists x P(x) \to R(a, f(b)).$$

En primer lugar tenemos que

$$\alpha \equiv \neg \exists x P(x) \vee R(a, f(b)).$$

Interiorizamos la negación, y obtenemos que

$$\neg \exists x P(x) \vee R(a, f(b)) \;\equiv\; \forall x \neg P(x) \vee R(a, f(b)).$$

Como en la segunda parte de la fórmula obtenida, es decir, en $R(a, f(b))$, no hay ocurrencias libres de x, ampliamos el radio de acción del cuantificador $\forall x$, y obtenemos que

$$\forall x \neg P(x) \vee R(a, f(b)) \;\equiv\; \forall x \Big(\neg P(x) \vee R(a, f(b)) \Big).$$

Por tanto, la fórmula

$$\forall x \Big(\neg P(x) \vee R(a, f(b)) \Big)$$

es una forma normal prenexa de α.

La fórmula siguiente, equivalente a la anterior, es también una forma normal prenexa de α:

$$\forall x \Big(P(x) \to R(a, f(b)) \Big).$$

□

Ejemplo 4. Son formas normales prenexas de la fórmula

$$\alpha := \forall y P(y) \to \forall x R(x, a),$$

tanto la fórmula

$$\forall x \exists y \Big(\neg P(y) \vee R(x, a) \Big)$$

como la fórmula

$$\exists y \forall x \Big(\neg P(y) \vee R(x, a) \Big).$$

En este ejemplo, el orden de extracción de los cuantificadores es indiferente, pues las dos partes $\forall y P(y)$ y $\forall x R(x, a)$ de α son independientes, en el sentido de que no comparten ninguna variable. Como veremos en otros ejemplos, éste no será el caso. □

Ejemplo 5. La fórmula

$$\alpha := \exists x P(x) \to \forall x R(x, a)$$

es equivalente a

$$\neg \exists x P(x) \vee \forall x R(x, a),$$

la cual a su vez equivale a
$$\forall x \neg P(x) \lor \forall x R(x,a).$$
Los dos cuantificadores universales que aparecen en la fórmula resultante, aunque cuantifican a la misma variable, es posible que tengan significados diferentes, pues la variable x que aparece en la subfórmula $\neg P(x)$ posiblemente no tenga nada que ver con la variable x que aparece en la subfórmula $R(x,a)$.

Recordemos aquí que en general no son equivalentes las fórmulas
$$\forall x \alpha_1 \lor \forall x \alpha_2 \qquad \text{y} \qquad \forall x(\alpha_1 \lor \alpha_2).$$

Así, la fórmula
$$\forall x \neg P(x) \lor \forall x R(x,a),$$
puede que no sea equivalente a la fórmula
$$\forall x \Big(\neg P(x) \lor R(x,a) \Big),$$
con lo cual no podemos asegurar que ésta última sea una forma normal prenexa de α.

Por tanto, por ahora tenemos la fórmula
$$\forall x \neg P(x) \lor \forall x R(x,a).$$

Decidimos comenzar extrayendo el primero de los dos cuantificadores. Como en la subfórmula
$$\forall x R(x,a)$$
no hay ocurrencias libres de x, ampliamos el radio de acción del primer cuantificador $\forall x$, y resulta
$$\forall x \neg P(x) \lor \forall x R(x,a) \;\equiv\; \forall x \Big(\neg P(x) \lor \forall x R(x,a) \Big).$$
Si ahora nos proponemos extraer el cuantificador universal que hay dentro de los paréntesis, no lo podemos hacer tal cual, pues le afectaría a la variable x que aparece en la subfórmula precedente $\neg P(x)$, a la cual realmente no le afectaba.

Por tanto, hemos de renombrar variables. Obtenemos que
$$\forall x R(x,a) \;\equiv\; \forall y R(y,a),$$
con lo cual, resulta
$$\forall x \Big(\neg P(x) \lor \forall x R(x,a) \Big) \;\equiv\; \forall x \Big(\neg P(x) \lor \forall y R(y,a) \Big).$$
Ya sí podemos extraer el cuantificador $\forall y$, pues ahora en la subfórmula $\neg P(x)$ no hay ocurrencias libres de la variable y. Resulta por tanto
$$\forall x \Big(\neg P(x) \lor \forall y R(y,a) \Big) \;\equiv\; \forall x \forall y \Big(\neg P(x) \lor R(y,a) \Big),$$

siendo la fórmula

$$\forall x \forall y \Big(\neg P(x) \vee R(y,a) \Big)$$

una forma normal prenexa de α. $\square$

Ejemplo 6. Si tenemos la fórmula

$$\alpha := \forall x P(x) \to \exists x R(x,a),$$

entonces

$$\alpha \;\equiv\; \neg \forall x P(x) \vee \exists x R(x,a) \;\equiv\; \exists x \neg P(x) \vee \exists x R(x,a) \;\equiv\; \exists x \Big(\neg P(x) \vee R(x,a) \Big).$$

La última equivalencia se basa en el Lema 4 de la Sección 8 del Tema 5. Por tanto una forma normal prenexa de α es

$$\exists x \Big(\neg P(x) \vee R(x,a) \Big).$$

Un camino alternativo hubiese sido renombrar en la fórmula

$$\exists x \neg P(x) \vee \exists x R(x,a)$$

para obtener la fórmula equivalente

$$\exists x \neg P(x) \vee \exists y R(y,a),$$

y de ésta, la fórmula

$$\exists x \exists y \Big(\neg P(x) \vee R(y,a) \Big),$$

la cual es otra forma normal prenexa de α. $\square$

Ejemplo 7. Si tenemos la fórmula

$$\alpha := \neg \exists z \exists x R(x,y) \vee \exists y P(f(y)),$$

comenzamos interiorizando la negación de la subfórmula de la izquierda, y resulta

$$\alpha \;\equiv\; \forall z \forall x \neg R(x,y) \vee \exists y P(f(y)).$$

Por una parte, el cuantificador $\forall z$ es superfluo y lo podemos suprimir, con lo cual obtenemos

$$\alpha \;\equiv\; \forall x \neg R(x,y) \vee \exists y P(f(y)).$$

Como en la subfórmula de la izquierda hay una ocurrencia libre de la variable y, para extraer el cuantificador $\exists y$ primero hemos de renombrar todas las ocurrencias de la variable y en la subfórmula de la derecha. De este modo llegamos a que

$$\alpha \;\equiv\; \forall x \neg R(x,y) \vee \exists y_1 P(f(y_1)),$$

donde y_1 es un nuevo símbolo de variable, distinto de y. Por tanto,

$$\alpha \ \equiv \ \exists y_1 \Big(\forall x \neg R(x,y) \vee P(f(y_1)) \Big) \ \equiv \ \exists y_1 \forall x \Big(\neg R(x,y) \vee P(f(y_1)) \Big),$$

siendo la fórmula

$$\exists y_1 \forall x \Big(\neg R(x,y) \vee P(f(y_1)) \Big)$$

una forma normal prenexa de α.

Como veremos en la sección siguiente, al obtener una forma normal prenexa para una fórmula, es preferible, siempre que se pueda, extraer primero (de modo que queden más a la izquierda) los cuantificadores existenciales. □

Recordemos también que en general, no son equivalentes las fórmulas

$$\exists x \beta_1 \wedge \exists x \beta_2 \qquad \text{y} \qquad \exists x (\beta_1 \wedge \beta_2).$$

Ésto lo tendremos en cuenta en el ejemplo siguiente.

Ejemplo 8. Obtenemos una forma normal prenexa de

$$\alpha := \forall x \Big(R(x,y) \rightarrow \forall x R(x,a) \Big) \rightarrow \forall x \forall z R(x,b).$$

Aplicamos pasos similares a los que hemos aplicado en los ejemplos anteriores y obtenemos:

$$\alpha \ \equiv \ \neg \forall x \Big(\neg R(x,y) \vee \forall x R(x,a) \Big) \vee \forall x R(x,b),$$

$$\alpha \ \equiv \ \exists x \neg \Big(\neg R(x,y) \vee \forall x R(x,a) \Big) \vee \forall x R(x,b),$$

$$\alpha \ \equiv \ \exists x \Big(\neg\neg R(x,y) \wedge \neg \forall x R(x,a) \Big) \vee \forall x R(x,b),$$

$$\alpha \ \equiv \ \exists x \Big(R(x,y) \wedge \exists x \neg R(x,a) \Big) \vee \forall x R(x,b),$$

$$\alpha \ \equiv \ \exists x \Big(R(x,y) \wedge \exists x_1 \neg R(x_1,a) \Big) \vee \forall x R(x,b),$$

$$\alpha \ \equiv \ \exists x \exists x_1 \Big(R(x,y) \wedge \neg R(x_1,a) \Big) \vee \forall x R(x,b),$$

$$\alpha \ \equiv \ \exists x \exists x_1 \Big(R(x,y) \wedge \neg R(x_1,a) \Big) \vee \forall x_2 R(x_2,b),$$

$$\alpha \ \equiv \ \exists x \exists x_1 \forall x_2 \Big(\big(R(x,y) \wedge \neg R(x_1,a) \big) \vee R(x_2,b) \Big).$$

Por tanto la fórmula

$$\exists x \exists x_1 \forall x_2 \Big(\big(R(x,y) \wedge \neg R(x_1,a) \big) \vee R(x_2,b) \Big)$$

es una forma normal prenexa de α. □

Conviene conocer la propiedad siguiente de cara a nuestros propósitos.

Proposición 1. Dado un conjunto de fórmulas Γ, considérese el conjunto Γ^* formado al escoger una fórmula en forma normal prenexa para cada fórmula de Γ. Entonces Γ es insatisfacible, si y sólo si, Γ^* es insatisfacible. □

2.2 Forma normal de Skolem

Sea α^* una fórmula en forma normal prenexa. Grosso modo, una forma normal de Skolem para α^* será obtenida reemplazando las variables cuantificadas existencialmente en α^* por ciertos términos apropiados y "eliminando" los cuantificadores existenciales que aparecen en α^*.

Dada $\alpha^* \in \mathrm{Form}(\mathcal{L})$ en forma normal prenexa, una forma normal de Skolem de α^* es una fórmula obtenida al ir substituyendo cada variable x_i, cuantificada existencialmente, por un término $f(x_{i_1}, \ldots, x_{i_m})$, donde f es un símbolo de función de aridad m que no aparece en α^* ni ha sido usado hasta el momento para la substitución de otra variable, y los símbolos $x_{i_1}, \ldots, x_{i_m}$ son las variables cuantificadas universalmente que preceden a $\exists x_i$ (es decir, aparacen a su izquierda) en la escritura de α^*; de no haber ninguna, la substitución se hace por un símbolo de constante que no aparezca en α^* ni tampoco haya sido utilizado hasta el momento.

Ejemplo 9.

	F.N.P.	F.N.S.
1.	$P(x)$	$P(x)$
2.	$P(a)$	$P(a)$
3.	$\forall x P(x)$	$\forall x P(x)$
4.	$\exists x P(x)$	$P(a)$
5.	$\exists x Q(a,x)$	$Q(a,b)$
6.	$\exists x \exists y R(a,x,f(y))$	$R(a,b,f(c))$
7.	$\forall x \exists y Q(x,y)$	$\forall x Q(x,f(x))$
8.	$\forall x \exists y Q(f(x),y)$	$\forall x Q(f(x),g(x))$
9.	$\forall x \forall y \exists z R(x,y,z)$	$\forall x \forall y R(x,y,f(x,y))$
10.	$\forall x \forall y \exists z R(f(x),y,z)$	$\forall x \forall y R(f(x),y,g(x,y))$
11.	$\exists x \forall y \exists z R(h(a,x),y,z)$	$\forall y R(h(a,b),y,f(y))$
12.	$\forall x \exists y \forall x_1 \exists y_1 R(h(x,y),y_1,f(x_1))$	$\forall x \forall x_1 R(h(x,g_1(x)),g_2(x,x_1),f(x_1))$

Los reemplazamientos que hemos aplicado son los siguientes:

1. ninguno.
2. ninguno.
3. ninguno.
4. $x := a$.
5. $x := b$.
6. $x := b,\ y := c$.
7. $y := f(x)$.
8. $y := g(x)$.
9. $z := f(x, y)$.
10. $z := g(x, y)$.
11. $x := b,\ z := f(y)$.
12. $y := g_1(x),\ y_1 := g_2(x, x_1)$.

□

Ejemplo 10. Dada la fórmula

$$\alpha^* := \exists x P(x)$$

en forma normal prenexa, una forma normal de Skolem para α^* es

$$\beta := P(a),$$

la cual ha sido obtenida aplicando el reemplazamiento $x := a$.

Consideramos la estructura $\mathcal{E}$ dada por:

- $\mathrm{D} = \mathbb{Z}$,
- $a^{\mathcal{E}} = 5$,
- $P^{\mathcal{E}}(r) = \mathbf{1}$ si y sólo si r es par.

Entonces para cualquier asignación v en $\mathcal{E}$, se cumple $\mathrm{I}^v(\alpha^*) = \mathbf{1}$, mientras que $\mathrm{I}^v(\beta) = \mathbf{0}$. Por tanto α^* y β no son equivalentes.

Sin embargo, los conjuntos de fórmulas $\{\alpha^*\}$ y $\{\beta\}$ son ambos satisfacibles. Que $\{\alpha^*\}$ es satisfacible, lo acabamos de poner de manifiesto con la interpretación anterior. El conjunto $\{\beta\}$ es también satisfacible. Basta considerar la estructura siguiente:

- $\mathrm{D} = \mathbb{Z}$,
- $a^{\mathcal{E}} = 5$,
- $P^{\mathcal{E}}(r) = \mathbf{1}$ si y sólo si r es impar.

□

Vemos con este ejemplo que al "skolemizar" de esta forma, algunas fórmulas ya no son equivalentes a su forma normal de Skolem, es decir, se puede perder la equivalencia semántica, pero el carácter de satisfacibilidad ó insatisfacibilidad del conjunto formado por ellas, que es lo que realmente nos interesa, se conserva.

Proposición 2. Sea Γ^* un conjunto de fórmulas en forma normal prenexa. Calculamos una forma normal de Skolem para cada fórmula en Γ^*, y obtenemos el conjunto Γ^{**}. Entonces Γ^* es insatisfacible, si y sólo si, Γ^{**} es insatisfacible. $\square$

Ya comentamos en un ejemplo previo, que a la hora de obtener una forma normal de Skolem de una fórmula dada, interesa extraer primero los cuantificadores existenciales, siempre que ello sea posible.

Ejemplo 11. Consideramos la fórmula

$$\alpha := \forall x_1 P(x_1) \wedge \forall x_2 P(x_2) \wedge \forall x_3 P(x_3) \wedge \exists x_4 P(x_4) \wedge \exists x_5 P(x_5).$$

Una forma normal prenexa de α es

$$\forall x_1 \forall x_2 \forall x_3 \exists x_4 \exists x_5 \Big(P(x_1) \wedge P(x_2) \wedge P(x_3) \wedge P(x_4) \wedge P(x_5) \Big),$$

que conduce a la forma normal de Skolem

$$\forall x_1 \forall x_2 \forall x_3 \Big(P(x_1) \wedge P(x_2) \wedge P(x_3) \wedge P(f(x_1, x_2, x_3)) \wedge P(g(x_1, x_2, x_3)) \Big),$$

con los reemplazamientos $x_4 := f(x_1, x_2, x_3)$ y $x_5 := g(x_1, x_2, x_3)$.

Claramente es preferible obtener la forma normal prenexa

$$\exists x_4 \exists x_5 \forall x_1 \forall x_2 \forall x_3 \Big(P(x_1) \wedge P(x_2) \wedge P(x_3) \wedge P(x_4) \wedge P(x_5) \Big),$$

a partir de la cual, con los reemplazamientos $x_4 := a$, $x_5 := b$, se obtiene la forma normal de Skolem

$$\forall x_1 \forall x_2 \forall x_3 \Big(P(x_1) \wedge P(x_2) \wedge P(x_3) \wedge P(a) \wedge P(b) \Big),$$

más sencilla que la previamente obtenida. $\square$

Recalcamos que cuando vayamos a transformar varias fórmulas que ya están en forma normal prenexa a forma normal de Skolem, procuraremos que los nuevos símbolos introducidos para cada una de las fórmulas (ya sean de constante o de función) no aparezcan ni hayan sido utilizados en ninguna de las otras fórmulas dadas.

Ejemplo 12. Supongamos el conjunto de fórmulas en forma normal prenexa:

$$\Gamma^* := \Big\{ \forall y \neg Q(a, y),\ \exists x \forall y Q(x, y) \Big\},$$

y sea el conjunto Γ^{**} siguiente formado por una forma normal de Skolem para cada una de las fórmulas de Γ^*, pero donde no hemos tenido en cuenta el criterio anterior acerca de los símbolos nuevos:

$$\Gamma^{**} := \Big\{\forall y \neg Q(a, y),\ \forall y Q(a, y)\Big\}.$$

Es evidente que Γ^{**} es insatisfacible, ya que es imposible que $Q(a, y)$ sea al mismo tiempo verdadero y falso bajo una interpretación cualquiera. Veamos que Γ^* es satisfacible. Definimos una estructura $\mathcal{E}$ como sigue:

$\mathrm{D}_{\mathcal{E}} = \{1, 2, 3\}$.
$a^{\mathcal{E}} = 3$.
Para $r, s \in \mathrm{D}_{\mathcal{E}}$, $Q^{\mathcal{E}}(r, s) = \begin{cases} \mathbf{1} & \text{si } (r, s) \in \{(1, 1), (1, 2), (1, 3)\}, \\ \mathbf{0} & \text{en otro caso.} \end{cases}$

Ya que las dos fórmulas en Γ son sentencias, para cualquier asignación v en $\mathcal{E}$, se verifica que

$$\mathrm{I}^v\Big(\forall y \neg Q(a, y)\Big) = \mathrm{I}^v\Big(\neg\exists y Q(a, y)\Big) = \mathbf{1},$$

pues no existe ningún elemento $r \in \mathrm{D}_{\mathcal{E}}$ tal que $P^{\mathcal{E}}(3, r) = \mathbf{1}$. Además

$$\mathrm{I}^v\Big(\exists x \forall y Q(x, y)\Big) = \mathbf{1},$$

pues existe el elemento $r = 1 \in \mathrm{D}_{\mathcal{E}}$ el cual está relacionado con cualquier elemento perteneciente a $\mathrm{D}_{\mathcal{E}}$. Por tanto, el conjunto Γ^* es satisfacible. $\square$

2.3 Forma normal clausulada

En esta sección nos restringiremos a fórmulas que sean **sentencias**. Recuérdese que una fórmula es una sentencia si no contiene ocurrencias libres de ninguna variable. Ya comentamos en el Tema 5, al traducir frases a lenguajes de predicados de primer orden, que en las fómulas provenientes del lenguaje hablado no aparecen ocurrencias libres de variables. Tal y como veremos en los ejercicios, la restricción que hemos impuesto a las fórmulas no afecta a nuestro objetivo.

Así pues vamos a partir de sentencias que están ya en forma normal de Skolem, digamos

$$\alpha := \forall x_1 \cdots \forall x_n \beta,$$

donde no aparece ningún cuantificador en β.

Obtenemos una forma normal clausulada de α simplemente calculando una forma clausulada de β de manera análoga a como vimos en el Tema 4 para la Lógica de proposiciones.

Si tenemos una cláusula C y L es un literal que aparece en C varias veces, reduciremos todas las ocurrencias del literal L en C a una sóla. De este modo obtenemos otra cláusula más simple que es equivalente a C.

Por ejemplo, la cláusula

$$R(x, f(y)) \vee \neg S(x, a) \vee R(x, f(y)) \vee R(y, f(y))$$

la reducimos a

$$R(x, f(y)) \vee \neg S(x, a) \vee R(y, f(y)).$$

Si $C_1 \wedge \cdots \wedge C_m$ es una fórmula equivalente a β, siendo cada C_i una disyunción de (uno o más) literales, decimos que la fórmula

$$\forall x_1 \cdots \forall x_n \Big(C_1 \wedge \cdots \wedge C_m \Big)$$

es una **forma normal clausulada** de α. También decimos que las fórmulas

$$\underbrace{\forall x_1 \cdots \forall x_n C_1}, \ldots, \underbrace{\forall x_1 \cdots \forall x_n C_m}$$

son las cláusulas de α.

Nótese que por las equivalencias estudiadas en el Tema 5, se cumple que

$$\forall x_1 \cdots \forall x_n \Big(C_1 \wedge \cdots \wedge C_m \Big) \equiv \Big(\forall x_1 \cdots \forall x_n C_1 \Big) \wedge \cdots \wedge \Big(\forall x_1 \cdots \forall x_n C_m \Big).$$

De hecho, de ahora en adelante al escribir una forma normal clausulada de una fórmula dada (y en particular al escribir las cláusulas que la componen) omitiremos los cuantificadores universales precedentes, sin olvidarnos de que todas las variables estarán cuantificadas universalmente. Por este motivo nos hemos restringido a sentencias, para que no haya ambigüedad. Así pues, si escribimos la cláusula

$$R(x, a) \vee \neg P(z)$$

realmente nos estamos refiriendo a

$$\forall x \forall z \Big(R(x, a) \vee \neg P(z) \Big).$$

Siguiendo con la nomenclatura anterior, diremos también que el conjunto $\{C_1, \ldots, C_m\}$ es una forma normal clausulada de α.

Ejemplo 13. Si tenemos la fórmula ya en forma normal clausulada,

$$\alpha := \forall x \forall y \Big(\big(Q(x, a) \vee \neg P(a) \big) \wedge \big(\neg Q(a, y) \vee R \big) \Big),$$

entonces

$$\forall x \forall y \Big(Q(x, a) \vee \neg P(a) \Big) \qquad \text{y} \qquad \forall x \forall y \Big(\neg Q(a, y) \vee R \Big)$$

son las cláusulas de α.

También diremos que el conjunto de fórmulas

$$\Big\{Q(x,a)\vee\neg P(a),\ \neg Q(a,y)\vee R\Big\}$$

es una forma normal clausulada de α. □

Ejemplo 14. En el Ejemplo 8 vimos que una forma normal prenexa de la fórmula

$$\forall x\Big(R(x,y)\to\forall x R(x,a)\Big)\to\forall x\forall z R(x,b),$$

es

$$\exists x\exists x_1\forall x_2\Big(\big(R(x,y)\wedge\neg R(x_1,a)\big)\vee R(x_2,b)\Big).$$

Por tanto, una forma normal prenexa de la fórmula

$$\beta:=\forall y\Big(\forall x\Big(R(x,y)\to\forall x R(x,a)\Big)\to\forall x\forall z R(x,b)\Big),$$

es

$$\forall y\exists x\exists x_1\forall x_2\Big(\Big(R(x,y)\wedge\neg R(x_1,a)\Big)\vee R(x_2,b)\Big).$$

Posiblemente haya otras formas normales prenexas para β más favorables que la que hemos obtenido, aunque ello no nos importa en este momento.

Así, una forma normal de Skolem para β es

$$\forall y\forall x_2\Big(\Big(R(f(y),y)\wedge\neg R(g(y),a)\Big)\vee R(x_2,b)\Big),$$

donde hemos aplicado los reemplazamientos $x:=f(y)$, $x_1:=g(y)$.

Para obtener una forma normal clausulada, aplicamos una propiedad distributiva dentro del paréntesis grande. Resulta

$$\forall y\forall x_2\Big(\Big(R(f(y),y)\vee R(x_2,b)\Big)\wedge\Big(\neg R(g(y),a)\big)\vee R(x_2,b)\Big)\Big),$$

que es una forma normal clausulada de β. De acuerdo con las definiciones anteriores, las cláusulas de β son

$$\forall y\forall x_2\Big(R(f(y),y)\vee R(x_2,b)\Big),\ \forall y\forall x_2\Big(\neg R(g(y),a)\big)\vee R(x_2,b)\Big).$$

También decimos que el conjunto

$$\Big\{R(f(y),y)\vee R(x_2,b),\ \neg R(g(y),a)\big)\vee R(x_2,b)\Big\}$$

es una forma normal clausulada de β. □

Se verifica el siguiente teorema.

Teorema 2. Dado un conjunto de <u>sentencias</u> Γ, sea el conjunto Γ' resultante de considerar para cada fórmula de Γ una forma normal clausulada, y sea Γ'' el conjunto formado por todas las cláusulas que aparecen en las fórmulas de Γ'. Entonces Γ es insatisfacible, si y sólo si, Γ'' es insatisfacible.

Así, probar la insatisfacibilidad ó inconsistencia de un conjunto de sentencias, es equivalente a probar la insconsistencia de un conjunto de cláusulas, que son fórmulas más sencillas. Como se puede apreciar, es la misma idea que ya vimos en el Tema 4, pero ahora en el contexto de la Lógica de predicados.

Ejemplo 15. Dado el conjunto de sentencias

$$\Gamma := \Big\{\forall x(P(x) \to Q(x)),\ \exists y(P(y) \wedge P(a)),\ \neg\exists z Q(z)\Big\},$$

obtenemos una forma normal clausulada para cada una de ellas:

$\forall x(P(x) \to Q(x))$	$\forall x(\neg P(x) \vee Q(x))$
$\exists y(P(y) \wedge P(a))$	$P(b) \wedge P(a)$
$\neg\exists z Q(z)$	$\forall z \neg Q(z)$

Entonces, siguiendo la notación del teorema anterior, tenemos

$$\Gamma' := \Big\{\forall x(\neg P(x) \vee Q(x)),\ P(b) \wedge P(a),\ \forall z \neg Q(z)\Big\}.$$

Al descomponer cada una de sus fórmulas en cláusulas y omitir cuantificadores, resulta

$$\Gamma'' := \Big\{\neg P(x) \vee Q(x),\ P(b),\ P(a),\ \neg Q(z)\Big\}.$$

Según el teorema anterior, el conjunto de sentencias Γ es insatisfacible, si y sólo si, el conjunto de cláusulas Γ'' es insatisfacible. Aplicando los algoritmos que veremos en la última sección del tema, obtendremos fácilmente que el conjunto de cláusulas Γ'' es insatisfacible. $\square$

3 Unificación

Recordemos que un **literal** en un lenguaje de predicados de primer orden puede ser una fórmula atómica o bien la negación de una fórmula atómica.

Dados dos literales, el proceso de reemplazar determinados símbolos de variable en dichos literales de forma que ambos resulten idénticos, se denomina **unificación**. Por ejemplo, si tenemos los literales

$$R(x, f(f(a))) \quad \text{y} \quad R(y, f(z)),$$

ambos se unifican aplicando el reemplazamiento

$$\begin{cases} x := y \\ z := f(a) \end{cases}$$

para obtener el literal

$$R(y, f(f(a))).$$

Por supuesto hay otros reemplazamientos que también unifican a dichos literales, como por ejemplo,

$$\begin{cases} x := g(h(z_1)) \\ y := g(h(z_1)) \\ z := f(a) \end{cases}$$

para obtener

$$R(g(h(z_1)), f(f(a))),$$

ó bien

$$\begin{cases} x := b \\ y := b \\ z := f(a) \end{cases}$$

para obtener

$$R(b, f(f(a))).$$

Recalcamos que se trata de un proceso de reemplazamiento de unas letras por otras secuencias de letras, es decir, es un proceso puramente sintáctico, en el que no se evalúa nada. A continuación damos las definiciones precisas.

Una **substitución** en un lenguaje de predicados $\mathcal{L}$ viene dada por una aplicación

$$\sigma : \mathrm{Var}(\mathcal{L}) \to \mathrm{Term}(\mathcal{L})$$

la cual se extiende a todo término (de forma parecida a como ocurría con las asignaciones en el Tema 5). Esta nueva aplicación del conjunto $\mathrm{Term}(\mathcal{L})$ en $\mathrm{Term}(\mathcal{L})$ también la denotaremos por σ. Nótese que si c es un símbolo de constante, entonces $\sigma(c) = c$.

Dada la substitución σ que verifica

$$\begin{cases} \sigma(x_1) & = & t_1, \\ & \vdots & \\ \sigma(x_n) & = & t_n, \\ \sigma(y) & = & y, \text{ si } y \in \mathrm{Var}(\mathcal{L}) \setminus \{x_1, \ldots, x_n\}, \end{cases}$$

donde como es usual x_i son símbolos de variable del lenguaje $\mathcal{L}$ y t_j son términos de $\mathcal{L}$, escribiremos

$$\sigma = (x_1|t_1, \ldots, x_n|t_n).$$

Así, al dar una substitución σ, asumiremos que los símbolos de variable que no aparecen como una parte izquierda, se quedan fijos.

Denotaremos a la substitución identidad como $\psi = ()$.

Ejemplo 16. Si tenemos un lenguaje de predicados $\mathcal{L}$ tal que

$$\mathrm{Cons}(\mathcal{L}) = \{a\}, \quad \mathrm{Func}(\mathcal{L}) = \{f^1\} \;\text{ y }\; \mathrm{Rel}(\mathcal{L}) = \{P^1, Q^2, R^0\},$$

entonces la aplicación $\sigma : \mathrm{Var}(\mathcal{L}) \to \mathrm{Term}(\mathcal{L})$ determinada por las condiciones

$$\sigma(x) = f(f(a)), \quad \sigma(y) = f(y)$$

es un substitución en $\mathcal{L}$ que denotamos como

$$\sigma = \big(x|f(f(a)),\ y|f(y)\big).$$

A partir de esta información calculamos

$$\begin{aligned}
&\sigma\Big(f(x)\Big) = f(f(f(a))),\\
&\sigma(a) = a,\\
&\sigma(f(y)) = f(f(y)).
\end{aligned}$$

Además tenemos que

$$\sigma(z) = z, \quad \sigma(x_1) = x_1,$$

con lo cual

$$\begin{aligned}
&\sigma(f(z)) = f(z),\\
&\sigma(f(f(x_1))) = f(f(x_1)).
\end{aligned}$$

□

Dado un literal L, ya sea de la forma $P(t_1, \ldots, t_n)$ o bien $\neg P(t_1, \ldots, t_n)$, y una substitución σ, denotamos por $\sigma(L)$ el literal que resulta de reemplazar cada símbolo de variable x_i que aparece en L por el valor $\sigma(x_i)$. Es decir, si

$$L = P(t_1, \ldots, t_n),$$

entonces

$$\sigma(L) = P(\sigma(t_1), \ldots, \sigma(t_n)),$$

y si

$$L = \neg P(t_1, \ldots, t_n),$$

entonces

$$\sigma(L) = \neg P(\sigma(t_1), \ldots, \sigma(t_n)).$$

Ejemplo 17. Si tenemos la substitución $\sigma = \big(x|f(x,y),\ y|a\big)$, y el literal

$$L := R(x, g(y), z),$$

entonces

$$\sigma(L) = R(f(x,y), g(a), z),$$

mientras que si

$$L := \neg R(x, g(y), z),$$

entonces

$$\sigma(L) = \neg R(f(x,y), g(a), z).$$

□

Ejemplo 18. Sean la substitución

$$\sigma = \big(x|f(a,b),\ y|z\big)$$

y el literal

$$L := P(x, g(b), y, g(z)).$$

Entonces

$$\sigma(L) = P\Big(f(a,b), g(b), z, g(z)\Big).$$

□

Ejemplo 19. Si $\sigma = \big(x|y,\ y|x\big)$ y $L := P(x, g(b), y, g(z))$, entonces

$$\sigma(L) = P\Big(y, g(b), x, g(z)\Big)$$

y

$$\sigma^2(L) = \sigma\Big(\sigma(L)\Big) = L.$$

□

Una substitución σ es un **unificador** para un conjunto de literales $\Gamma = \{L_1, \ldots, L_n\}$, si

$$\sigma(L_1) = \cdots = \sigma(L_n).$$

Un conjunto de literales Γ se dice **unificable**, si existe al menos un unificador para Γ.

Ejemplo 20. El conjunto de literales

$$\Big\{P(x),\ P(a),\ P(y)\Big\}$$

es unificable tomando la substitución

$$\sigma = \big(x|a,\ y|a\big),$$

pues

$$\begin{aligned}
&\sigma(P(x)) = P(\sigma(x)) = P(a),\\
&\sigma(P(a)) = P(a),\\
&\sigma(P(y)) = P(\sigma(y)) = P(a).
\end{aligned}$$

□

Ejemplo 21. El conjunto de literales

$$\Big\{P(x,z),\ P(f(y),y)\Big\}$$

es unificable, pues si tomamos la substitución

$$\sigma = \big(x|f(y),\ z|y\big)$$

resulta

$$\sigma(P(x,z)) = P(\sigma(x),\sigma(z)) = P(f(y),y)$$

y

$$\sigma(P(f(y),y)) = P(f(y),y),$$

pues se supone que $\sigma(y) = y$. □

Ejemplo 22. El conjunto de literales

$$\Big\{P(x,a),\ P(b,x)\Big\}$$

no es unificable, pues no existe ninguna substitución σ tal que $\sigma(x) = b$ y $\sigma(x) = a$, ya que a y b son símbolos de constante distintos. □

Una vez que tenemos un unificador para un conjunto de literales Γ, en general, podremos construir a partir de él infinitos unificadores para Γ. De entre todos los posibles unificadores para Γ, habrá unos que son más simples que los demás.

Un unificador σ para un conjunto de literales Γ se dice que es un **unificador principal** o **de máxima generalidad** para Γ, si para cualquier otro unificador ρ para Γ existe una substitución μ de forma que $\rho = \mu \circ \sigma$, es decir, ρ se obtiene componiendo μ con σ.

Ejemplo 23. El conjunto de literales

$$\Gamma := \Big\{P(a),\ P(x)\Big\}$$

es unificable mediante el unificador $\sigma = \big(x|a\big)$, el cual es un unificador principal para Γ. □

Ejemplo 24. Dado el conjunto de literales

$$\Gamma := \Big\{P(x), P(y)\Big\},$$

y las substituciones siguientes,

1. $\sigma_1 = (y|x)$,
2. $\sigma_2 = (x|z, y|z)$,
3. $\sigma_3 = (x|f(y), y|f(y))$,
4. $\sigma_4 = (x|a, y|a)$,

todas ellas son unificadores para Γ, de los cuales sólo σ_1 es unificador principal para Γ. $\square$

3.1 Cálculo de un unificador principal mediante la resolución de sistemas de ecuaciones en términos

Dados los literales L_1 y L_2 de un lenguaje de predicados $\mathcal{L}$, para que puedan ser unificables es necesario que ambos vayan negados o bien ambos sin negar, y que además empiecen por el mismo símbolo de relación. Esto es, L_1 y L_2 deben ser,

$$L_1 = R(t_1, \dots, t_n), \quad L_2 = R(t'_1, \dots, t'_n),$$

o bien

$$L_1 = \neg R(t_1, \dots, t_n), \quad L_2 = \neg R(t'_1, \dots, t'_n).$$

Queda claro pues que L_1 y L_2 son unificables si y sólo si existe una substitución σ en $\mathcal{L}$ tal que $\sigma(t_i) = \sigma(t'_i)$ para todo $i = 1, \dots, n$. Esta idea da lugar a la siguiente definición y proporciona un método para resolver el problema de la unificación.

Una **ecuación** en un lenguaje de predicados $\mathcal{L}$ es una pareja de términos de $\mathcal{L}$, es decir, (t_1, t_2) con $t_1, t_2 \in \mathrm{Term}(\mathcal{L})$. La ecuación (t_1, t_2) la escribiremos como

$$t_1 = t_2.$$

Una **solución** de una ecuación (t_1, t_2) es una substitución σ en $\mathcal{L}$ que cumple

$$\sigma(t_1) = \sigma(t_2).$$

Ejemplo 25. Cada una de las substituciones

$$\sigma_1 = \big(x|f(x),\ y|f(x)\big), \qquad \sigma_2 = \big(x|y\big), \qquad \sigma_3 := \big(y|x\big)$$

es una solución de la ecuación $x = y$.

La ecuación

$$f(x) = g(x, y)$$

no tiene solución. Tampoco tiene solución la ecuación

$$a = f(x).$$

$\square$

Un **sistema de ecuaciones** E es un conjunto finito de ecuaciones.

Ejemplo 26. El sistema $E := \{x = y,\ y = a,\ z = b\}$ consta de tres ecuaciones. □

Una **solución** de un sistema de ecuaciones es una substitución σ que es solución de cada una de las ecuaciones del sistema. Un sistema de ecuaciones en términos se dice compatible, si posee al menos una solución. En caso contrario, se denomina incompatible.

Dos **sistemas** de ecuaciones son **equivalentes** si tienen las mismas soluciones.

Ejemplo 27. La substitución $\sigma = \big(x|a,\ y|a,\ z|b\big)$ es solución del sistema

$$E := \{x = y,\ y = a,\ z = b\}.$$

□

Un sistema $E := \{e_1, e_2, \ldots, e_n\}$ está en **forma resuelta**, si cada ecuación e_i es de la forma

$$x_i = t_i,$$

donde

- como es usual los símbolos x_i son símbolos de variable,
- $x_i \neq x_j$ para $i \neq j$,
- los símbolos t_i son términos de $\mathcal{L}$, y además
- ninguna de las variables que aparecen en los términos t_i pertenece al conjunto

$$\Big\{x_1, x_2, \ldots, x_n\Big\}.$$

Si el sistema

$$E := \Big\{x_1 = t_1,\ x_2 = t_2, \ldots,\ x_n = t_n\Big\}$$

está en forma resuelta (donde como es habitual $x_1, x_2, \ldots, x_n$ son símbolos de variable y $t_1, t_2, \ldots, t_n$ denotan términos), definimos la substitución

$$\sigma_E = \Big(x_1|t_1,\ x_2|t_2, \ldots, x_n|t_n\Big).$$

Teorema 3. Si tenemos un conjunto de literales Γ, a partir del cual obtenemos un sistema de ecuaciones en términos E el cual está en forma resuelta, entonces Γ es unificable y la substitución σ_E es un unificador principal para Γ. □

A continuación damos una serie de propiedades que nos van a permitir transformar un sistema de ecuaciones en otro equivalente a él, de forma que, tras un número finito de transformaciones lleguemos a un sistema en forma resuelta o a un sistema incompatible.

Lema 1. Los sistemas $E \cup \{t = t\}$ y E son equivalentes. □

Lema 2. Los sistemas $E \cup \{t_1 = t_2\}$ y $E \cup \{t_2 = t_1\}$ son equivalentes. □

Lema 3. Los sistemas $E \cup \{f(t_1, \ldots, t_n) = f(u_1, \ldots, u_n)\}$ y $E \cup \{t_1 = u_1, \ldots, t_n = u_n\}$ son equivalentes. □

Lema 4. Dado un símbolo de variable x y un término t tales que x no aparece en t, entonces los sistemas $E \cup \{x = t\}$ y $\sigma(E) \cup \{x = t\}$, con $\sigma = (x|t)$, son equivalentes. ($\sigma(E)$ denota el sistema obtenido al aplicar σ a cada una de las ecuaciones en E.) □

Lema 5. El sistema $E \cup \{f(t_1, \ldots, t_n) = g(u_1, \ldots, u_m)\}$, con $f \neq g$, es incompatible. □

Lema 6. El sistema $E \cup \{f(t_1, \ldots, t_n) = a\}$, con a un símbolo de constante, es incompatible. □

Lema 7. Si x es un símbolo de variable, t es un término tal que $t \neq x$, de forma que x aparece en t, entonces el sistema $E \cup \{x = t\}$ no tiene solución. □

Teorema 4. Para todo sistema compatible existe un sistema equivalente que está en forma resuelta. □

Ejemplo 28. Determinar si el conjunto de literales

$$\Gamma := \Big\{P(a, x, f(g(y))),\ P(z, f(z),\ f(u))\Big\}$$

es unificable o no lo es. Caso de serlo, encontrar un unificador principal para él.

El sistema de ecuaciones asociado a este problema es

$$\left.\begin{array}{l} a = z \\ x = f(z) \\ f(g(y)) = f(u) \end{array}\right\}$$

que es equivalente a

$$\left.\begin{array}{l} z = a \\ x = f(z) \\ f(g(y)) = f(u) \end{array}\right\}$$

el cual equivale a

$$\left.\begin{array}{l} z = a \\ x = f(a) \\ f(g(y)) = f(u) \end{array}\right\}$$

equivalente a

$$\left.\begin{array}{l} z = a \\ x = f(a) \\ g(y) = u \end{array}\right\}$$

y por último equivalente a

$$\left.\begin{array}{l} z = a \\ x = f(a) \\ u = g(y) \end{array}\right\}$$

que está en forma resuelta. Por tanto Γ es unificable y un unificador principal para Γ es

$$\sigma_E = \Big(z|a,\ x|f(a),\ u|g(y)\Big).$$

$\square$

Ejemplo 29. Determinar si el conjunto de literales

$$\Gamma := \Big\{P(x,y),\ P(f(z),x),\ P(u,f(x))\Big\}$$

es unificable o no, y en caso de serlo encontrar un unificador principal para él.

Un sistema de ecuaciones asociado a este problema es:

$$\left.\begin{array}{l} x = f(z) \\ y = x \\ x = u \\ y = f(x) \end{array}\right\}$$

Este sistema es equivalente a

$$\left.\begin{array}{l} x = f(z) \\ y = f(z) \\ f(z) = u \\ y = f(f(z)) \end{array}\right\}$$

que es equivalente a

$$\left.\begin{array}{l} x = f(z) \\ y = f(z) \\ u = f(z) \\ y = f(f(z)) \end{array}\right\}$$

equivalente a

$$\left.\begin{array}{l} x = f(z) \\ y = f(z) \\ u = f(z) \\ f(z) = f(f(z)) \end{array}\right\}$$

que por último es equivalente a

$$\left.\begin{array}{l} x = f(z) \\ y = f(z) \\ u = f(z) \\ z = f(z) \end{array}\right\}$$

el cual al contener la ecuación $z = f(z)$, no tiene solución. Por tanto el conjunto Γ no es unificable. $\square$

4 Resolución

Hemos visto ya anteriormente la siguiente propiedad, llamada *Modus ponendo ponens*

$$\Big\{\alpha,\ \alpha \to \beta\Big\} \models \beta,$$

donde α y β son dos fórmulas cualesquiera de un lenguaje de predicados $\mathcal{L}$.

Ésta propiedad la podemos expresar de modo equivalente como

$$\Big\{\alpha,\ \neg\alpha\vee\beta\Big\} \models \beta,$$

y nos dice que siempre que tengamos en nuestro conjunto de fórmulas dos fórmulas α y $\neg\alpha\vee\beta$, a partir de ellas podemos deducir la fórmula β.

Así pues, tenemos una *regla de inferencia ó de deducción*, la cual a partir de dos fórmulas apropiadas nos permite obtener otra que es consecuencia lógica de las dos anteriores

Esta regla de deducción se puede generalizar de la manera siguiente.

Lema 8. Dadas las fórmulas

$$\alpha \vee \beta_1 \vee \cdots \vee \beta_m$$

y

$$\neg\alpha \vee \beta_1' \vee \cdots \vee \beta_n',$$

se verifica que

$$\Big\{\alpha \vee \beta_1 \vee \cdots \vee \beta_m,\ \neg\alpha \vee \beta_1' \vee \cdots \vee \beta_n'\Big\} \models \beta_1 \vee \cdots \vee \beta_m \vee \beta_1' \vee \cdots \vee \beta_n'.$$

□

Recordemos que un literal de $\mathcal{L}$ es una fórmula atómica ó la negación de una fórmula atómica. Además, si L es un literal, entonces su literal complementario se define como

$$L^c = \begin{cases} L' & \text{si } L = \neg L', \\ \neg L' & \text{si } L = L', \end{cases}$$

donde L' denota una fórmula atómica.

De forma parecida a lo que hicimos en el Tema 4 cuando estudiamos el Algoritmo de Davis-Putnam, aquí también trabajaremos con cláusulas, motivo por el cual damos la siguiente versión del Lema 8.

Lema 9. Dadas las cláusulas $L \vee L_1 \vee\cdots\vee L_m$ y $L^c \vee L_1' \vee\cdots\vee L_n'$, se verifica que

$$\Big\{L \vee L_1 \vee\cdots\vee L_m,\ L^c \vee L_1' \vee\cdots\vee L_n'\Big\} \models L_1 \vee\cdots\vee L_m \vee L_1' \vee\cdots\vee L_n'.$$

□

Si C es una cláusula y L es un literal que aparece en C, notamos por $C - L$ a la cláusula que tiene todos los literales de C excepto L. En el caso en que $C = L$, a $C - L$ lo representamos por $\Box$, y nos referimos a ella como la **cláusula vacía**. Recordemos que todo conjunto de cláusulas que contenga a la cláusula vacía se considera insatisfacible ó inconsistente.

Sean C_1 y C_2 dos cláusulas *sin variables comunes* y sean L_1 y L_2 dos literales de C_1 y C_2, respectivamente. Si L_1 y L_2^c tienen un unificador principal, σ, entonces la cláusula

$$\Big(\sigma(C_1) - \sigma(L_1)\Big) \ \vee \ \Big(\sigma(C_2) - \sigma(L_2)\Big)$$

es una **resolvente binaria** de C_1 y C_2. En tal caso diremos que L_1 y L_2 son los literales sobre los que se resuelve.

Como se supone que C_1 y C_2 son sentencias, donde sus variables aparecen cuantificadas universalmente, si es necesario, siempre podemos renombrar variables. Por ello podemos suponer que las cláusulas consideradas no tienen variables en común.

Lema 10. Si C es una resolvente binaria de las cláusulas C_1 y C_2, entonces

$$\Big\{C_1, C_2\Big\} \ \models \ C.$$

$\Box$

Ejemplo 30. Sean las cláusulas

$$\begin{aligned} C_1 \equiv &\ \ R(f(x), b) \vee S(x, y) \vee P(a), \\ C_2 \equiv &\ \ \neg R(f(z), b) \vee \neg P(f(z)), \end{aligned}$$

es decir,

$$\begin{aligned} C_1 \equiv &\ \ \forall x \forall y \Big(R(f(x), b) \vee S(x, y) \vee P(a) \Big), \\ C_2 \equiv &\ \ \forall z \Big(\neg R(f(z), b) \vee \neg P(f(z)) \Big). \end{aligned}$$

Observamos que las dos cláusulas no tienen variables en común.

Consideramos el literal $L_1 := R(f(x), b)$ de C_1 y el literal $L_2 := \neg R(f(z), b)$ de C_2. Un unificador principal para los literales L_1 y L_2^c, es decir, para el conjunto de literales

$$\Big\{ R(f(x), b), \ R(f(z), b) \Big\},$$

es

$$\sigma = (z|x).$$

(También hubiese sido correcto utilizar el unificador principal para L_1 y L_2^c dado por $\sigma' = (x|z)$.)

Tenemos por una parte que

$$\sigma(C_1) \;=\; C_1 \;=\; R(f(x),b) \vee S(x,y) \vee P(a) \qquad \text{y} \qquad \sigma(L_1) \;=\; L_1 \;=\; R(f(x),b),$$

y por otra que

$$\sigma(C_2) \;=\; \neg R(f(x),b) \vee \neg P(f(x)) \qquad \text{y} \qquad \sigma(L_2) \;=\; \neg R(f(x),b).$$

De aquí,

$$\sigma(C_1) - \sigma(L_1) \;=\; S(x,y) \vee P(a),$$

y

$$\sigma(C_2) - \sigma(L_2) \;=\; \neg P(f(x)).$$

Por tanto obtenemos la resolvente binaria siguiente para C_1 y C_2:

$$S(x,y) \vee P(a) \vee \neg P(f(x)).$$

Nótese que no es posible resolver sobre los literales $P(a)$ y $\neg P(f(z))$ respectivos de C_1 y C_2, al no ser los literales $P(a)$ y $P(f(z))$ unificables. □

Ejemplo 31. Sean las cláusulas

$$\begin{aligned} C_1 \equiv&\;\; R(f(x),b) \vee S(x,y) \vee P(a), \\ C_2 \equiv&\;\; \neg R(f(x),b) \vee \neg P(f(z)). \end{aligned}$$

Vemos que ambas cláusulas comparten variables, aunque en este ejemplo particular es posible calcular una resolvente binaria sobre los literales $L_1 := R(f(x),b)$ de C_1 y $L_2 := \neg R(f(x),b)$ de C_2. En este caso utilizamos el unificador principal identidad que transforma cada variable en ella misma. La resolvente binaria que obtenemos es

$$S(x,y) \vee P(a) \vee \neg P(f(z)).$$

□

En el ejemplo siguiente se muestra por qué hemos de procurar que cláusulas distintas no compartan variables.

Ejemplo 32. Sean las cláusulas

$$\begin{aligned} C_1 \equiv&\;\; R(x,b) \vee Q(x,a), \\ C_2 \equiv&\;\; \neg R(a,x) \vee P(x). \end{aligned}$$

En el literal $R(x,b)$ de C_1 y en el literal $\neg R(a,x)$ de C_2 aparece x, y ello hace que no podamos encontrar un unificador principal para el conjunto de literales

$$\Big\{ R(x,b),\;\; R(a,x) \Big\},$$

pues el sistema de ecuaciones en términos

$$\begin{cases} x = a \\ b = x \end{cases}$$

es incompatible. Por consiguiente, de esta forma no podemos obtener una resolvente binaria para las cláusulas C_1 y C_2.

Si renombramos en C_2 cambiando el símbolo de variable x por z, resulta

$$\begin{aligned} C_1 \equiv &\quad R(x,b) \vee Q(x,a), \\ C_2 \equiv &\quad \neg R(a,z) \vee P(z). \end{aligned}$$

Tomando $L_1 := R(x,b), L_2 := \neg R(a,z)$ y $\sigma = (x|a, z|b)$, obtenemos la resolvente binaria

$$Q(a,a) \vee P(b).$$

$\square$

Es razonable pensar que si un conjunto de cláusulas Γ es insatisfacible, entonces de alguna forma ha de ser posible obtener la cláusula vacía calculando resolventes binarias a partir de cláusulas de Γ ó de otras resolventes binarias previamente calculadas.

Ejemplo 33. Sea el conjunto de cláusulas

$$\Gamma := \Big\{\neg P(x) \vee Q(x),\ \neg Q(y) \vee R(y),\ P(a),\ \neg R(a)\Big\}.$$

Intuitivamente vemos que Γ es insatisfacible, pues si escribimos las fórmulas de Γ como

$$\begin{aligned} &\forall x(P(x) \to Q(x)), \\ &\forall y(Q(y) \to R(y)), \\ &P(a), \\ &\neg R(a), \end{aligned}$$

observamos que a partir de las tres primeras se obtiene la conclusión $R(a)$, la cual es "incompatible" con la cuarta fórmula $\neg R(a)$. Dicho de otro modo, si Γ fuese satisfacible, entonces por el Lema 9 obtendríamos que el conjunto de cláusulas $\{R(a),\ \neg R(a)\}$ sería satisfacible, cosa que no es cierta.

Ahora vamos a obtener la cláusula vacía calculando resolventes binarias a partir de Γ. Para ello llamamos

$$\begin{aligned} C_1 &\equiv \neg P(x) \vee Q(x), \\ C_2 &\equiv \neg Q(y) \vee R(y), \\ C_3 &\equiv P(a), \\ C_4 &\equiv \neg R(a). \end{aligned}$$

De C_1 y C_3, con $\sigma_1 = (x|a)$, obtenemos la resolvente binaria

$$C_5 \equiv Q(a).$$

A partir de ésta y C_2, con $\sigma_2 = (y|a)$, obtenemos la resolvente binaria

$$C_6 \equiv R(a).$$

Por último, de C_4 y C_6, con $\sigma_3 = ()$, obtenemos

$$C_7 \equiv \square.$$

$\square$

Sin embargo, con la definición de resolvente que tenemos hasta este momento, existen conjuntos de cláusulas insatisfacibles a partir de los cuales no es posible obtener la cláusula vacía mediante el cálculo de resolventes binarias sucesivas.

Ejemplo 34. El conjunto de cláusulas

$$\Gamma := \Big\{P(x) \vee P(y),\ \neg P(z) \vee \neg P(u)\Big\}$$

claramente es inconsistente pues $\Gamma \models P(a)$ y $\Gamma \models \neg P(a)$.

Sin embargo es imposible obtener la cláusula vacía mediante el cálculo de resolventes binarias sucesivas, pues según las definiciones, para calcular una resolvente binaria a partir de dos cláusulas, tomamos un literal en cada una de ellas y calculamos un unificador principal que aplicamos a ambas cláusulas. Finalmente los transformados de los literales seleccionados son suprimidos.

En nuestro caso, cualquier resolvente binaria que obtengamos usando cláusulas de Γ ó resolventes binarias previamente calculadas, consta siempre de dos literales. Vemos así que nunca obtendremos la cláusula vacía. Por ello necesitamos ampliar el concepto de resolvente. $\square$

Si dos o más literales de una cláusula C tienen un unificador principal σ, entonces se dice que $\sigma(C)$ es un **factor** de la cláusula C. También se suele decir que $\sigma(C)$ se obtiene a partir de C aplicando la regla de disminución. Si $\sigma(C)$ tiene un sólo literal, diremos que $\sigma(C)$ es un **factor unitario** de C.

Ejemplo 35. Sea la cláusula

$$C_1 \equiv P(x) \vee P(a) \vee Q(f(x), y)$$

y la substitución

$$\sigma_1 = (x|a).$$

Entonces la cláusula

$$\sigma_1(C_1) \equiv P(a) \vee Q(f(a), y)$$

es un factor de C_1. Así hemos unificado los literales $P(x)$ y $P(a)$ que aparecen en C_1 para obtener la cláusula $\sigma_1(C_1)$.

A partir de la cláusula

$$C_2 \equiv P(f(x)) \vee \neg Q(y, a) \vee P(y) \vee P(z)$$

podemos obtener varios factores según nos interese. Por ejemplo, una posibilidad consiste en unificar los literales $P(f(x))$ y $P(y)$ mediante la substitución

$$\sigma_2 = (y|f(x)),$$

para obtener el factor

$$\sigma_2(C_2) = P(f(x)) \vee \neg Q(f(x), a) \vee P(z).$$

Otra posibilidad consiste en unificar los literales $P(f(x))$, $P(y)$ y $P(z)$ mediante la substitución

$$\sigma_3 = (y|f(x), z|f(x)).$$

Resulta entonces la cláusula

$$\sigma_3(C_2) = P(f(x)) \vee \neg Q(f(x), a).$$

$\square$

Lema 11. Si $\sigma(C)$ es un factor de una cláusula C, entonces $C \models \sigma(C)$. $\square$

Ya estamos en condiciones de dar el concepto de resolvente que vamos a adoptar.

Una **resolvente** de dos cláusulas C_1 y C_2 es una de las siguientes resolventes binarias:

1. Una resolvente binaria de C_1 y C_2.
2. Una resolvente binaria de C_1 y un factor de C_2.
3. Una resolvente binaria de C_2 y un factor de C_1.
4. Una resolvente binaria de un factor de C_1 y un factor de C_2.

Ejemplo 36. Obtenemos todas las resolventes posibles para las cláusulas

$$\left.\begin{array}{ll} C_1 \equiv & P(x) \vee P(f(y)) \vee R(x, y) \\ C_2 \equiv & \neg P(f(g(a))) \vee Q(z) \end{array}\right\}.$$

1. Resolvente binaria de C_1 y C_2 sobre los literales respectivos $P(x)$ y $\neg P(f(g(a)))$. Usamos el unificador principal $\sigma = (x|f(g(a)))$ y obtenemos
$$P(f(y)) \vee R(f(g(a)), y) \vee Q(z).$$

2. Resolvente binaria de C_1 y C_2 sobre los literales respectivos $P(f(y))$ y $\neg P(f(g(a)))$. Usamos el unificador principal $\sigma = (y|g(a))$ y obtenemos
$$P(x) \vee R(x, g(a)) \vee Q(z).$$

3. Resolvente binaria de un factor de C_1 con la cláusula C_2. Previamente unificamos los literales
$$L_1 := P(x),\ L_2 := P(f(y))$$
de C_1 usando el unificador principal
$$\sigma = (x|f(y))$$
y obtenemos el factor de C_1 dado por
$$\sigma(C_1) \equiv P(f(y)) \vee R(f(y), y).$$
Finalmente calculamos la resolvente binaria de $\sigma(C_1)$ y C_2. Para ello usamos el unificador principal
$$\delta = (y|g(a))$$
y obtenemos
$$R(f(g(a)), g(a)) \vee Q(z),$$
que es una resolvente para C_1 y C_2.

   Otra forma de obtener este mismo resultado consiste en calcular un unificador principal σ' para el conjunto de literales
$$\Big\{P(x),\ P(f(y)),\ P(f(g(a)))\Big\}.$$
Resulta
$$\sigma' = (x|f(g(a)), y|g(a)),$$
de donde
$$\sigma'(C_1) \equiv P(f(g(a))) \vee R(f(g(a)), g(a)),$$

$$\sigma'(C_2) \equiv \neg P(f(g(a))) \vee Q(z),$$
y de aquí la resolvente
$$R(f(g(a)), g(a)) \vee Q(z).$$

□

La siguiente propiedad es la generalización del Lema 10.

Lema 12. Si C es una resolvente de las cláusulas C_1 y C_2, entonces

$$\left\{C_1, C_2\right\} \models C.$$

□

A continuación damos la definición formal de **deducción de una cláusula a partir de un conjunto de cláusulas**, la cual ya la hemos comentado intuitivamente en algunos ejemplos anteriores.

> Dado un conjunto de cláusulas, $\Gamma \cup \{C\}$, una deducción de C a partir de Γ es una secuencia finita de cláusulas $C_1, C_2, \ldots, C_m$ tal que $C_m = C$ y para todo $i < m$, se cumple que C_i es un elemento de Γ ó bien es una resolvente de dos cláusulas C_j y C_k, con j y k ambos menores que i.

De acuerdo con la definición anterior, a las cláusulas pertenecientes a Γ se les llama hipótesis.

En la práctica, cuando escribamos una deducción, numeraremos y describiremos las fórmulas que la componen, ya sean hipótesis ó resolventes, debiendo en este último caso indicar las cláusulas intervinientes y el unificador principal utilizado.

Ejemplo 37. Consideramos el conjunto de cláusulas siguiente:

$$\Gamma := \left\{\neg P(x) \vee Q(x),\ \neg Q(y) \vee R(y),\ P(a),\ \neg R(a)\right\}.$$

La secuencia siguiente es una deducción de la cláusula $R(a)$ a partir del conjunto Γ:

1. $\neg P(x) \vee Q(x)$, hipótesis.
2. $P(a)$, hipótesis.
3. $Q(a)$, resolvente de (1.) y (2.); $\sigma_1 = (x|a)$.
4. $\neg Q(y) \vee R(y)$, hipótesis.
5. $R(a)$, resolvente de (3.) y (4.); $\sigma_2 = (y|a)$.

□

Ejemplo 38. Dado el conjunto de cláusulas

$$\Gamma := \left\{\neg P(x) \vee Q(x),\ P(a)\right\},$$

la secuencia siguiente es una deducción (muy poco interesante) de la cláusula $P(a)$ a partir del conjunto Γ:

1. $\neg P(x) \vee Q(x)$, hipótesis.
2. $P(a)$, hipótesis.
3. $P(a)$, hipótesis.

$\square$

Un **refutación para un conjunto de cláusulas** Γ es una deducción de la cláusula vacía a partir de Γ.

Un conjunto de cláusulas Γ se denomina **refutable**, si existe alguna refutación para Γ.

Ejemplo 39. Consideramos de nuevo el conjunto de cláusulas

$$\Gamma := \Big\{\neg P(x) \vee Q(x),\ \neg Q(y) \vee R(y),\ P(a),\ \neg R(a)\Big\}.$$

La secuencia siguiente es una refutación para Γ:

1. $\neg P(x) \vee Q(x)$, hipótesis.
2. $P(a)$, hipótesis.
3. $Q(a)$, resolvente de (1.) y (2.); $\sigma_1 = (x|a)$.
4. $\neg Q(y) \vee R(y)$, hipótesis.
5. $R(a)$, resolvente de (3.) y (4.); $\sigma_2 = (y|a)$.
6. $\neg R(a)$, hipótesis.
7. $\square$, resolvente de (5.) y (6.); $\sigma_3 = ()$.

Por tanto, Γ es refutable. $\square$

Ejemplo 40. El conjunto de cláusulas siguiente no es refutable:

$$\Gamma := \Big\{\neg P(x) \vee Q(x),\ P(a),\ \neg Q(b)\Big\}.$$

Si intentamos encontrar una refutación para Γ y comenzamos con la secuencia

1. $\neg P(x) \vee Q(x)$, hipótesis.
2. $P(a)$, hipótesis.
3. $Q(a)$, resolvente de (1.) y (2.); $\sigma_1 = (x|a)$.
4. $\neg Q(b)$, hipótesis.
5. $\neg P(b)$, resolvente de (1.) y (4.); $\sigma_2 = (x|b)$.

vemos que ya no podemos obtener nuevas resolventes. Como no hemos obtenido $\square$, el conjunto no es refutable. $\square$

El Principio de resolución es una regla de inferencia ó de deducción que consiste en generar resolventes a partir de un conjunto de cláusulas. Así, para demostrar la inconsistencia de un conjunto de cláusulas intentaremos encontrar una deducción de $\square$ a partir del conjunto dado. Esta idea se sustenta en el siguiente resultado.

Teorema 5. (**Completitud del Principio de resolución**). Un conjunto de cláusulas Γ es inconsistente, si y sólo si, Γ es refutable, es decir, existe una deducción de $\square$ a partir de Γ. $\square$

Ejemplo 41. En el Ejemplo 34 hemos estudiado el conjunto de cláusulas

$$\Gamma := \Big\{P(x)\vee P(y),\ \neg P(z)\vee\neg P(u)\Big\},$$

para el cual hemos argumentado que es inconsistente. Ahora, basándonos en el Teorema de Completitud del principio de resolución, obtenemos esta misma conclusión. Para ello basta considerar la siguiente refutación para Γ:

1. $P(x)$, factor de la hipótesis $P(x)\vee P(y)$; $\sigma_1 = (y|x)$.
2. $\neg P(z)$, factor de la hipótesis $\neg P(z)\vee\neg P(u)$; $\sigma_2 = (u|z)$.
3. $\square$, resolvente de (1.) y (2.); $\sigma_3 = (z|x)$.

$\square$

Ejemplo 42. Consideramos el conjunto de cláusulas

$$\Gamma := \Big\{R(a,x)\vee\neg P(x),\ \neg R(y,y)\vee P(y)\Big\},$$

y llamamos

$$C_1 \equiv R(a,x)\vee\neg P(x),$$

$$C_2 \equiv \neg R(y,y)\vee P(y).$$

De C_1 y C_2, resolvemos sobre los literales $R(a,x)$ y $\neg R(y,y)$, usando $\sigma_1 = (x|a,\ y|a)$, y obtenemos

$$C_3 \equiv \neg P(a)\vee P(a).$$

De C_1 y C_2, resolvemos sobre los literales $\neg P(x)$ y $P(y)$, usando $\sigma_2 = (y|x)$, y obtenemos

$$C_4 \equiv R(a,x)\vee\neg R(x,x).$$

Nótese que $\square$ no se puede obtener como resolvente de C_1 y C_2.

De hecho, Γ es satisfacible. Para verlo, consideramos la estructura $\mathcal{E}$ tal que:

$$\begin{aligned}
&\mathrm{D}_{\mathcal{E}} := \{1, 2\} \subseteq \mathbb{N},\\
&a^{\mathcal{E}} := 1,\\
&P^{\mathcal{E}}(r) = \mathbf{1} \text{ si y sólo si } r = 2,\\
&R^{\mathcal{E}}(r, s) = \mathbf{1} \text{ si y sólo si } (r, s) \in \{(1, 2), (2, 2)\}.
\end{aligned}$$

Obsérvese que

$$C_1 \equiv \forall x\Big(P(x) \to R(a, x)\Big),$$

$$C_2 \equiv \forall y\Big(R(y, y) \to P(y)\Big).$$

Entonces, para cualquier asignación v en $\mathcal{E}$ tenemos que

$$\mathrm{I}^v\,(C_1) = \mathrm{I}^v\,(C_2) = \mathbf{1}.$$

$\square$

Ejemplo 43. Consideramos de nuevo el conjunto de cláusulas del Ejemplo 40 el cual hemos visto que no es refutable:

$$\Gamma := \Big\{\neg P(x) \vee Q(x),\ P(a),\ \neg Q(b)\Big\}.$$

Por el Teorema 5, Γ es consistente. $\square$

Ejemplo 44. Sea el conjunto de cláusulas

$$\Gamma := \Big\{R(a, x) \vee P(x),\ \neg R(f(y), y) \vee P(y),\ \neg P(a)\Big\}.$$

Llamamos

$$C_1 \equiv R(a, x) \vee P(x),$$

$$C_2 \equiv \neg R(f(y), y) \vee P(y),$$

$$C_3 \equiv \neg P(a).$$

Nótese que cláusulas distintas no tienen variables en común.

De C_1 y C_2, no podemos obtener ninguna resolvente puesto que el sistema siguiente de ecuaciones en términos,

$$\begin{cases} a = f(y) \\ x = y \end{cases}$$

es incompatible.

De C_1 y C_3, resolvemos sobre los literales $P(x)$ y $\neg P(a)$, usando $\sigma_1 = (x|a)$, y obtenemos

$$C_4 \equiv R(a, a).$$

De C_2 y C_3, resolvemos sobre los literales $P(y)$ y $\neg P(a)$, usando $\sigma_2 = (y|a)$, y obtenemos

$$C_5 \equiv \neg R(f(a), a).$$

En este punto del proceso, observamos que C_4 no se puede resolver con ninguna de las cláusulas C_1, C_2, C_3, C_5. Tampoco es posible resolver C_5 con ninguna de las cláusulas C_1, C_2, C_3, C_4.

Así pues, no es posible generar nuevas resolventes. Como $\Box$ no pertenece a Γ ni tampoco ha sido obtenida como resolvente, basándonos en el Teorema de Completitud del principio de resolución, concluimos que Γ es satisfacible ó consistente. $\Box$

En el Tema 4 estudiamos el Algoritmo de Davis-Putnam que permite decidir si un conjunto de cláusulas de la Lógica proposicional es o no insatisfacible. Como la Lógica de proposiciones se generaliza mediante la Lógica de predicados, es posible utilizar el Principio de resolución como método alternativo para decidir la insatisfacibilidad de un conjunto de cláusulas de la Lógica proposicional. Ahora las substituciones utilizadas son todas iguales a la substitución identidad $\sigma = ()$.

Ejemplo 45. Comprobemos mediante el Principio de resolución que, tal y como vimos en el Tema 4, el conjunto de cláusulas siguiente es insatisfacible,

$$\Sigma = \Big\{ P \vee Q,\ P \vee \neg Q,\ \neg P \vee R,\ \neg P \vee \neg R \Big\}.$$

Para ello basta construir la siguiente refutación para Σ:

1. $P \vee Q$, hipótesis.
2. $P \vee \neg Q$, hipótesis.
3. P, resolvente de (1.) y (2.).
4. $\neg P \vee R$, hipótesis.
5. $\neg P \vee \neg R$, hipótesis.
6. $\neg P$, resolvente de (4.) y (5.).
7. $\Box$, resolvente de (3.) y (6.).

Por el Principio de resolución, Σ es insatisfacible. $\Box$

Ejemplo 46. En el Tema 4 vimos mediante el Algoritmo de Davis-Putnam que el conjunto siguiente de cláusulas es satisfacible,

$$\Sigma = \Big\{ P \vee \neg Q,\ \neg P \vee R,\ \neg P \vee \neg R \Big\}.$$

Ahora obtenemos esta misma conclusión aplicando el Principio de resolución. Para ello llamamos

$$C_1 := P \vee \neg Q, \quad C_2 := \neg P \vee R, \quad C_3 := \neg P \vee \neg R.$$

Primero calculamos todas las resolventes posibles entre ellas.

- $C_4 := \neg Q \vee R$, resolvente de C_1 y C_2.
- $C_5 := \neg Q \vee \neg R$, resolvente de C_1 y C_3.
- $C_6 := \neg P$, resolvente de C_2 y C_3.

Ahora calculamos todas las resolventes posibles entre C_4 y alguna cláusula del conjunto $\{C_1, C_2, C_3, C_5, C_6\}$.

- $C_7 := \neg Q \vee \neg P$, resolvente de C_4 y C_3.
- $C_8 := \neg Q$, resolvente de C_4 y C_5.

Ahora todas las resolventes de C_5 con alguna cláusula del conjunto $\{C_1, C_2, C_3, C_6\}$. Sólo se obtiene C_7 como resolvente de C_5 con C_2.

A continuación obtenemos todas las resolventes de C_6 con alguna cláusula del conjunto $\{C_1, C_2, C_3\}$. Sólo se obtiene C_8 como resolvente de C_6 con C_1.

Por tanto, hasta este momento hemos calculado todas las resolventes posibles entre dos cláusulas del conjunto $\{C_1, C_2, C_3, C_4, C_5, C_6\}$. A continuación intentamos obtener una resolvente de C_7 con cada una de las cláusulas del conjunto $\{C_1, C_2, C_3, C_4, C_5, C_6\}$, y análogamente para C_8. También hemos de intentar resolver C_7 con C_8.

De todas éstas posibilidades, sólo se puede calcular la resolvente de C_7 con C_1, que produce de nuevo C_8.

Por tanto no se pueden generar nuevas resolventes. Como no hemos obtenido la cláusula vacía, por el Principio de resolución, el conjunto Σ es satisfacible. $\square$

5 Estrategias de resolución

Si partimos de un conjunto de cláusulas Γ y queremos estudiar si es satisfacible ó insatisfacible, necesitamos una **estrategia** que nos vaya guiando en el proceso de obtención de resolventes.

Como un primer paso, suprimiremos las cláusulas tautológicas que aparezcan en el conjunto inicial, así como aquellas que se vayan obteniendo.

La estrategia más simple para generar resolventes, es la llamada **estrategia de saturación**. Su descripción es la siguiente:

1. $\Gamma_0 := \Gamma$.
2. Para cada $i \geq 0$, calcular el conjunto Δ_i formado por todas las resolventes obtenibles a partir de dos cláusulas cualesquiera $C_{i,j}, C_{i,k} \in \Gamma_i$, distintas. Definir $\Gamma_{i+1} := \Gamma_i \cup \Delta_i$.
3. Si $\square \in \Gamma_{i+1}$, entonces se detiene el proceso y se concluye que Γ es insatisfacible.

4. Si $\Gamma_{i+1} = \Gamma_i$, entonces se detiene el proceso y se concluye que Γ es satisfacible.

5. Hacer $i := i + 1$, y volver al paso (2.).

$\square$

La estrategia de saturación suele ser muy ineficiente, pues es posible que obtengamos una misma cláusula varias veces además de muchas cláusulas inútiles de cara a la obtención de la cláusula vacía.

Esta estrategia la hemos usado ya en los Ejemplos 40 y 46.

Otra posibilidad que puede presentarse en la práctica, es que sea posible generar una infinidad de resolventes distintas. En este caso, si no se ha obtenido $\square$, el proceso debería detenerse en algún momento, y concluir que muy probablemente Γ sea satisfacible.

Ello nos lleva a que hemos de poner algún tipo de cota ó límite sobre el máximo número de resolventes a generar.

Ejemplo 47. Sea el conjunto de cláusulas

$$\Gamma := \Big\{\neg P(x) \vee P(f(x)),\ P(a)\Big\}.$$

A partir de Γ resulta la sucesión de resolventes

$$P(f(a)),\ P(f(f(a))),\ P(f(f(f(a)))),\ P(f(f(f(f(a))))), \ldots.$$

Se puede comprobar que Γ es satisfacible, con lo cual nunca obtendremos $\square$. Por tanto, tras generar un número suficientemente grande de resolventes, concluímos que muy probablemente Γ es satisfacible. $\square$

Entre otras ideas que se pueden utilizar para generar resolventes, mencionamos las siguientes:

1. *Elección de la cláusula más corta.* En cada momento del proceso, de entre todas las resolventes posibles que se pueden obtener, se calcula una cuyo número de literales sea el menor posible.

2. *Preferencia de cláusulas unitarias.* En cada momento del proceso, se intenta que la resolvente obtenida sea tal que alguna de las cláusulas intervinientes en su obtención conste de un sólo literal.

3. *Técnica del conjunto soporte.* Si estamos intentando dar respuesta al problema $\Omega \models \alpha$, y para ello queremos saber si el conjunto de fórmulas $\Omega \cup \{\neg\alpha\}$ es insatisfacible, obtenemos un conjunto de cláusulas Γ a partir de las fórmulas en $\Omega \cup \{\neg\alpha\}$. Supongamos que Δ es el subconjunto de Γ formado por aquellas cláusulas que proceden de la fórmula $\neg\alpha$. Como Ω se supone satisfacible, ya que de lo contrario sería inmediato que $\Omega \models \alpha$, la idea de esta estrategia consiste en que toda resolvente obtenida se base en alguna cláusula que proceda, aunque sea de forma indirecta, del subconjunto Δ.

Estas ideas también pueden ser aplicadas de forma combinada.

Establecemos en este punto la metodología que se deduce de todas las ideas expuestas en este tema:

Para probar que un conjunto de fórmulas Ω *implica semánticamente a una fórmula* α, *tratamos de ver que el conjunto de fórmulas* $\Omega \cup \{\neg\alpha\}$ *es inconsistente. Para ello, obtenemos un conjunto de formas clausuladas correspondientes, y a partir de éste, un conjunto de cláusulas* Γ. *Entonces aplicamos el Principio de resolución a* Γ *tratando de obtener una refutación para él.*

Este método para probar una implicación semántica se denomina **refutación por resolución**.

Ejemplo 48. Queremos demostrar la siguiente implicación semántica mediante una refutación por resolución:

$$\Big\{\forall x(R(x) \to R(f(x))),\ R(a),\ \forall x P(x) \vee \forall x Q(x)\Big\} \models \forall x\big(P(x) \vee Q(x)\big).$$

Para ello consideramos el conjunto

$$\Sigma = \Big\{\forall x(R(x) \to R(f(x))),\ R(a),\ \forall x P(x) \vee \forall x Q(x),\ \neg\forall x\big(P(x) \vee Q(x)\big)\Big\},$$

el cual queremos ver que es inconsistente. A partir de éste obtenemos un conjunto de formas clausuladas correspondientes:

$$\Sigma'' = \Big\{\forall x(\neg R(x) \vee R(f(x))),\ R(a),\ \forall y \forall z(P(y) \vee Q(z)),\ \neg P(a) \wedge \neg Q(a)\Big\}.$$

Finalmente llegamos al conjunto de cláusulas

$$\Gamma = \Big\{\neg R(x) \vee R(f(x)),\ R(a),\ P(y) \vee Q(z),\ \neg P(a),\ \neg Q(a)\Big\}.$$

Si aplicamos la estrategia de elección de la cláusula más corta o bien la de preferencia de cláusulas unitarias, podríamos calcular, entre otras posibilidades, una resolvente a partir de las cláusulas

$$\neg R(x) \vee R(f(x)) \qquad \text{y} \qquad R(a).$$

Obtenemos la cláusula $R(f(a))$, a partir de la cual podríamos generar también las resolventes

$$R(f(f(a))),\ R(f(f(f(a)))),\ R(f(f(f(f(a))))), \ldots,$$

sin llegar nunca a la cláusula vacía (Véase el Ejemplo 47).

Es preferible aplicar la Técnica del conjunto soporte, y hacer intervenir en nuestros cálculos de resolventes siempre alguna cláusula que proceda directa ó indirectamente de las cláusulas obtenidas al negar la fórmula $\forall x\big(P(x) \vee Q(x)\big)$ que queremos demostrar.

Así pues, calculamos una resolvente a partir de las cláusulas

$$P(y) \vee Q(z) \qquad \text{y} \qquad \neg P(a),$$

y obtenemos la cláusula $Q(a)$, la cual sólo se puede resolver con la cláusula $\neg Q(a)$. Llegamos de este modo a $\square$, probando así la inconsistencia de Γ y por tanto la implicación semántica propuesta. $\square$

Cuando partimos de un conjunto de cláusulas Γ, y comenzamos a generar resolventes, estamos generando un árbol, llamado el **árbol de las deducciones**. La forma en la que vamos obteniendo resolventes determina la forma en la que vamos recorriendo dicho árbol. Por ejemplo, la estrategia de saturación estudiada anteriormente, da lugar a lo que se denomina **exploración del árbol primero en anchura**. Otra posibilidad, es la llamada **exploración del árbol primero en profundidad**, que se produce cuando al obtener la resolvente $C_{i,j+1}$ se utiliza $C_{i,j}$, y así para todo $j \geq 1$. Si no hemos obtenido la cláusula vacía, podemos retroceder hasta otro nodo más arriba, en el que aplicamos de nuevo la misma idea. En este caso se habla de **exploración del árbol primero en profundidad con retroceso**.

Al margen de estas cuestiones algorítmicas, otra alternativa consiste en restringir el concepto de deducción.

> Una **deducción lineal-input** de una cláusula C a partir de un conjunto de cláusulas Γ, es una secuencia finita de cláusulas $C_1, C_2, \ldots, C_m$, tal que $C_m = C$ y para todo i, con $2 \leq i \leq m$, se cumple que C_i es una resolvente calculada a partir de C_{i-1} y otra cláusula perteneciente a Γ.

Nótese que la propia definición de deducción lineal-input conlleva que el árbol de las deducciones sea recorrido primero en profundidad.

Consideramos de nuevo el Ejemplo 37 en el cual, sin darnos cuenta, hicimos una deducción lineal-input de la cláusula $R(a)$ a partir del conjunto Γ dado.

Tenemos el conjunto de cláusulas

$$\Gamma := \Big\{ \neg P(x) \vee Q(x),\ \neg Q(y) \vee R(y),\ P(a),\ \neg R(a) \Big\},$$

y la secuencia siguiente de cláusulas:

1. $\neg P(x) \vee Q(x)$, hipótesis.
2. $P(a)$, hipótesis.
3. $Q(a)$, resolvente de (1.) y (2.); $\sigma_1 = (x|a)$.
4. $\neg Q(y) \vee R(y)$, hipótesis.

5. $R(a)$, resolvente de (3.) y (4.); $\sigma_2 = (y|a)$.

Una **refutación lineal-input** de un conjunto de cláusulas Γ, es una deducción lineal-input de la cláusula vacía a partir de Γ.

Veamos de nuevo el Ejemplo 39 en el que se lleva a cabo una refutación lineal-input para el conjunto Γ dado.

Teníamos
$$\Gamma := \Big\{\neg P(x) \vee Q(x),\ \neg Q(y) \vee R(y),\ P(a),\ \neg R(a)\Big\},$$
y la secuencia siguiente de cláusulas que finaliza en $\square$:

1. $\neg P(x) \vee Q(x)$, hipótesis.
2. $P(a)$, hipótesis.
3. $Q(a)$, resolvente de (1.) y (2.); $\sigma_1 = (x|a)$.
4. $\neg Q(y) \vee R(y)$, hipótesis.
5. $R(a)$, resolvente de (3.) y (4.); $\sigma_2 = (y|a)$.
6. $\neg R(a)$, hipótesis.
7. $\square$, resolvente de (5.) y (6.); $\sigma_3 = ()$.

6 Conjuntos de Horn

Un literal de la forma $R(t_1, \ldots, t_n)$ se dice que es un **literal positivo**, mientras que un literal de la forma $\neg R(t_1, \ldots, t_n)$ se denomina un **literal negativo**.

Un **cláusula negativa** es aquella en la que todos sus literales son negativos.

Un **cláusula de Horn** es la que tiene exactamente un literal positivo, y todos los literales restantes, si los hay, son negativos.

Un **conjunto de Horn** es aquel de la forma $\{C_0\} \cup \Delta$, donde C_0 es una cláusula negativa y Δ es un conjunto de cláusulas de Horn.

Ejemplo 49. Con las notaciones usuales, las siguientes cláusulas son negativas:
$$\neg R(f(x,y),z),\quad \neg Q(a),\quad \neg R(f(x,y),z) \vee \neg P(b),\quad \neg R(y,z) \vee \neg P(b) \vee \neg R(a, g(g(y))).$$

Las siguientes son cláusulas de Horn:
$$R(x, g(y)),\quad P(x) \vee \neg Q(h(x)),\quad R(x,a) \vee \neg P(y) \vee \neg Q(b) \vee \neg R(z, g(z)).$$

Ninguna de las cláusulas siguientes es de Horn:
$$P(a) \vee Q(y),\quad \neg R(a,y) \vee \neg Q(g(z)).$$

Todos los conjuntos siguientes son de Horn:

- $\Big\{\neg Q(x) \vee\neg P(a)\Big\}$.
- $\Big\{\neg Q(x) \vee\neg P(a),\ Q(y) \vee \neg R(y,h(x))\Big\}$.
- $\Big\{\neg Q(x) \vee\neg P(a),\ Q(z) \vee \neg R(y,h(x)),\ R(a,b) \vee \neg P(z) \vee \neg Q(f(x,y))\Big\}$.

El conjunto $\Big\{\neg Q(x) \vee\neg P(a),\ Q(a) \vee \neg R(y,h(x)) \vee P(z)\Big\}$ no es de Horn. □

Observamos que una resolvente obtenida a partir de una cláusula negativa y una cláusula de Horn, es de nuevo una cláusula negativa.

El siguiente resultado establece la importancia de los conjuntos de Horn.

Proposición 3. Un conjunto de Horn $\{C_0\} \cup \Delta$ es insatisfacible, si y sólo si, existe una refutación lineal-input para $\{C_0\} \cup \Delta$ que comienza en la cláusula C_0. □

En el Ejemplo 39 ya vimos un conjunto de Horn y una refutación lineal-input para él.

Hay conjuntos de cláusulas que no son de Horn para los que existen refutaciones lineales-input, como por ejemplo,

$$\Gamma = \Big\{P{\vee}Q,\ \neg P,\ \neg Q\Big\}.$$

Una refutación lineal-input para Γ es:

1. $P{\vee}Q$, hipótesis.
2. $\neg P$, hipótesis.
3. Q, resolvente de (1.) y (2.).
4. $\neg Q$, hipótesis.
5. $\Box$, resolvente de (3.) y (4.).

Los conceptos estudiados en este tema forman la base del sistema PROLOG, que es un lenguaje de programación ideado a principios de los años 70 en la Universidad de Aix-Marseille I (en Marsella, Francia) por Alain Colmerauer y Philippe Roussel. El nombre PROLOG proviene del francés *PROgrammation en LOGique*.

PROLOG trabaja con conjuntos de Horn con los que trata de obtener refutaciones lineales-input, también llamadas refutaciones lineales-ordenadas. Para ello aplica una estrategia de búsqueda denominada *búsqueda primero en profundidad con retroceso*.

Tema 5: Lógica de predicados de primer orden. Semántica.

(Apuntes elaborados por Juan Manuel Urbano Blanco)

Índice

1 Introducción. Definiciones básicas.

La Lógica proposicional puede no ser apropiada para expresar ciertos tipos de conocimiento. Por ejemplo:

Algunas personas hablan cinco idiomas.

Esta afirmación no está especificando ninguna persona de manera explícita, con nombre y apellidos, sólo está diciendo que existe un subconjunto no vacío dentro del conjunto de todas las personas, tal que cada uno de sus elementos habla cinco idiomas.

La Lógica proposicional tampoco permite designar objetos de forma indirecta en los enunciados, como por ejemplo cuando decimos "*el padre de Manolo*" ó "*el coche del padre de Manolo*".

Por otra parte, existen razonamientos que no pueden ser llevados a cabo con el formalismo de la Lógica proposicional. Por ejemplo:

Premisa 1:	*Todos los gatos trepan a los árboles.*
Premisa 2:	*Garfield es un gato.*
Conclusión:	*Garfield trepa a los árboles.*

Si intentamos aplicar la Lógica de proposiciones para hacer la deducción anterior, representaríamos la Premisa 1, la Premisa 2 y la Conclusión mediante sendas variables proposicionales P, Q, R, y como se aprecia, R no es consecuencia lógica del conjunto $\{P, Q\}$.

Denotemos por A el conjunto de todos los gatos, por B el conjunto de los seres que trepan a los árboles, y por a el gato Garfield. Entonces el razonamiento anterior lo podemos escribir simbólicamente como:

$$\left.\begin{array}{l} A \subseteq B \\ a \in A \end{array}\right\} \implies a \in B.$$

El problema es que en la Lógica de proposiciones no se contempla el hacer referencia a un objeto concreto perteneciente a un conjunto, como es en este caso Garfield que se supone es un gato concreto, conocido para el hablante.

La Lógica de predicados de primer orden soluciona estos problemas mediante el uso de cuantificadores, ya sea universales ó existenciales, como por ejemplo, en la frases "*Todos los pájaros tienen plumas*" y "*Algunos pájaros no comen insectos*".

Además la Lógica de predicados de primer orden permite representar propiedades de objetos a través de funciones y predicados, tal y como veremos en los ejemplos subsiguientes.

2 Lenguajes de predicados.

Un lenguaje de predicados $\mathcal{L}$ tiene por alfabeto los siguientes elementos.

- **Símbolos de variable**, que los denotaremos por $x, y, z, \ldots$, así como por $x_1, x_2, \ldots$, $y_1, y_2, \ldots$, etc., cuando ello sea necesario. Escribiremos $\mathrm{Var}(\mathcal{L})$ para representar el conjunto de todos los símbolos de variable del lenguaje $\mathcal{L}$.

- **Símbolos de constante**, que los representaremos por las letras $a, b, c, \ldots$ así como por $a_1, a_2, \ldots$, $b_1, b_2, \ldots$, etc. cuando ello sea necesario. Escribiremos $\mathrm{Cons}(\mathcal{L})$ para designar el conjunto de los símbolos de constante de $\mathcal{L}$.

- **Símbolos de función**, denotados como $f^{m_1}, g^{m_2}, h^{m_3}, \ldots$, donde $m_1, m_2, m_3, \ldots$, son números naturales llamados aridades. Cuando sea conveniente añadiremos subíndices, escribiendo en tal caso $f_1^{m_1}, f_2^{m_2}, \ldots$, $g_1^{n_1}, g_2^{n_2}, \ldots$, etc. La aridad de un símbolo de función es el número de argumentos (también llamados términos) a los que se supone se aplica dicho símbolo de función. Para simplificar la notación escribiremos por ejemplo f cuando la aridad sea conocida, en vez de f^{m_1}. El conjunto de todos los símbolos de función de $\mathcal{L}$ lo simbolizaremos por $\mathrm{Func}(\mathcal{L})$.

- **Símbolos de relación o de predicado**, que los representaremos siempre con letras mayúsculas $P^{m_1}, Q^{m_2}, R^{m_3}, \ldots$, donde $m_1, m_2, m_3, \ldots$, son las aridades correspondientes. Cuando ello sea necesario, incorporaremos subíndices $R_1^{a_1}, R_2^{a_2}, \ldots$, etc. Los símbolos de predicado van a representar relaciones que se definen para un número fijo de objetos, es cual es la aridad de dicho símbolo de predicado. También omitiremos las aridades cuando éstas sean conocidas. El conjunto de símbolos de relación lo denotaremos por $\mathrm{Rel}(\mathcal{L})$.

- **Operadores o conectivas lógicas**: Éstos son los operadores $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$, que ya conocemos del Tema 4, así como el cuantificador existencial denotado por $\exists$, y el cuantificador universal representado mediante el símbolo $\forall$.

Cuando especifiquemos un lenguaje de predicados de primer orden, indicaremos sólo los conjuntos $\mathrm{Cons}(\mathcal{L})$, $\mathrm{Func}(\mathcal{L})$ y $\mathrm{Rel}(\mathcal{L})$, omitiendo la especificación del conjunto $\mathrm{Var}(\mathcal{L})$, el cual es algo común a todos los lenguajes de predicados de primer orden.

Como hemos mencionado anteriormente, la aridad de un símbolo de función o de predicado es el número de elementos (llamados términos) a los que ésta ó éste se aplica. Normalmente sólo indicaremos los superíndices de aridad al principio, es decir, a la hora de definir el lenguaje. Posteriormente cada vez que se use un símbolo, ya sea de función o de predicado, omitiremos los superíndices de aridad.

Nótese que es posible definir símbolos de predicado 0-arios. Éstos representan afirmaciones que no se aplican a ningún término y que pueden ser verdaderas o falsas. Por ejemplo, la frase "*El Sol luce cada mañana*" podríamos representarla mediante un símbolo de predicado P con aridad 0. De esta forma la Lógica proposicional, queda inmersa dentro de la Lógica de predicados.

A partir de los símbolos anteriores definimos los términos, las fórmulas atómicas y las fórmulas. Los términos harán el papel de los objetos en el lenguaje hablado. Las fórmulas vendrán a representar los enunciados, siendo las fórmulas atómicas los enunciados más simples o indescomponibles.

Los **términos** del lenguaje $\mathcal{L}$ se definen recursivamente de la siguiente forma.

1. Todo símbolo de constante es un término.
2. Todo símbolo de variable es un término.
3. Si f es un símbolo de función de aridad n y $t_1, \ldots, t_n$ son términos, entonces $f(t_1, \ldots, t_n)$ es un término.
4. Todos los términos se generan aplicando las reglas anteriores un número finito de veces.

El conjunto de los términos de un lenguaje de primer orden $\mathcal{L}$ lo denotaremos por $\mathrm{Term}(\mathcal{L})$.

Una **fórmula atómica** o **átomo** es una expresión de la forma $R(t_1, \ldots, t_n)$, con R un símbolo de relación de aridad n y $t_1, \ldots, t_n$ términos.

Las **fórmulas** de $\mathcal{L}$ se definen recursivamente como sigue.

1. Todo átomo es una fórmula.
2. Si α y β son fórmulas, entonces $\neg\alpha$, $\alpha\vee\beta$, $\alpha\wedge\beta$, $\alpha \to \beta$ y $\alpha\leftrightarrow\beta$ también son fórmulas.
3. Si α es una fórmula y x es un símbolo de variable, entonces $\forall x\alpha$ y $\exists x\alpha$ son fórmulas.
4. Toda fórmula se genera aplicando las reglas anteriores un número finito de veces.

Denotamos el conjunto de todas las fórmulas del lenguaje $\mathcal{L}$ por $\mathrm{Form}(\mathcal{L})$.

Como en Lógica proposicional, se definen unas reglas de prioridad o precedencia entre las conectivas. Se mantienen los criterios que ya conocemos así como el uso de paréntesis. Sólo queda añadir que los cuantificadores y la negación tienen prioridad sobre el resto, y cuando varios de éstos aparecen juntos entonces se consideran de derecha a izquierda en la fórmula sobre la que actúan. Aclaramos todos estos conceptos con algunos ejemplos.

Ejemplo 1. Sea $\mathcal{L}$ el lenguaje de primer orden tal que

$$\mathrm{Cons}(\mathcal{L}) = \{a\}, \quad \mathrm{Func}(\mathcal{L}) = \{f^1\} \;\text{ y }\; \mathrm{Rel}(\mathcal{L}) = \{P^1, Q^2, R^0\}.$$

Las siguientes expresiones son términos de $\mathcal{L}$:

$$a, \quad x, \quad y, \quad f(x), \quad f(f(a)), \quad f(f(f(x_2))).$$

Las siguientes expresiones son fórmulas atómicas de $\mathcal{L}$:

$$Q(x, f(f(y))), \qquad R, \qquad P(a), \qquad P(f(f(f(z)))).$$

Las siguientes expresiones son fórmulas, no atómicas, de $\mathcal{L}$:

$$\exists y P(f(f(x))), \qquad \neg Q(a, y) \to \exists x P(x), \qquad \forall x \exists y \Big(Q(x, f(y)) \wedge R\Big) \to P(f(a)).$$

Las siguientes expresiones no son fórmulas de $\mathcal{L}$:

$$P(a, y), \quad P(x)\vee f(y), \quad R(a) \to \forall x P(y), \quad P(R), \quad \exists P\ \forall x\ (Q(x, x)\vee P(x)),$$

pues en la primera fórmula no estamos respetando la aridad del símbolo de predicado P, en la segunda estamos usando el término $f(y)$ como una fórmula, en la tercera no respetamos la aridad del símbolo de predicado R, en la cuarta estamos usando el símbolo de predicado R como si fuese un término, y en la quinta estamos cuantificando un predicado.

Con respecto a la prioridad de los operadores, por ejemplo, si escribimos la fórmula

$$\forall x P(x) \to Q(f(x), a)$$

nos estamos refiriendo a la fórmula

$$\Big(\forall x P(x)\Big) \to Q(f(x), a)$$

que no es la misma que
$$\forall x\Big(P(x) \to Q(f(x), a)\Big).$$

Tampoco es igual la fórmula
$$\forall x \exists y Q(x, y)$$
que la fórmula
$$\exists y \forall x Q(x, y),$$
pues tal y como veremos al final del tema, el orden de los cuantificadores es importante. □

Ya vimos en el Tema 4 cómo se traducen algunas frases del lenguaje natural al formalismo de la Lógica de proposiciones. Ahora ampliamos estas nociones con los ejemplos siguientes en los que aparecen cuantificadores y objetos designados de manera indirecta.

Ejemplo 2. Traducimos a un lenguaje apropiado de primer orden las frases siguientes.

1. *Todos los osos polares son de color blanco.*

   Usamos tres símbolos de predicado monarios: $P(x)$ designa que x vive en el Polo norte, $Q(x)$ indica que x es un oso, y $R(x)$ que x es un objeto o cosa de color blanco. Obtenemos la fórmula siguiente:
$$\forall x\Big(P(x) \wedge Q(x) \to R(x)\Big).$$

2. *Algunos osos no son de color blanco.*

   Aprovechamos los mismos predicados del apartado anterior y obtenemos:
$$\exists x\Big(Q(x) \wedge \neg R(x)\Big).$$

   Es evidente que esta fórmula también representa a cada una de las frases siguientes:

   - "*Existen osos que no son de color blanco*",
   - "*Hay osos que no son de color blanco*",
   - "*No todos los osos son de color blanco*".

3. *El coche del padre de Manolo es de color azul.*

   El substantivo "Manolo" designa un elemento conocido y concreto, por lo que lo representamos mediante un símbolo de constante a. Usamos dos símbolos de función f^1 y g^1, de modo que $f(x)$ designa el padre de x y $g(x)$ designa el coche de x. Finalmente definimos el predicado $P(x)$ que significa que x es de color azul. La frase se representa entonces como
$$P(g(f(a))).$$

También podíamos haber definido $f(x)$ para significar el coche del padre de x, en cuyo caso obtendríamos

$$P(f(a)).$$

La tercera posibilidad consiste en utilizar un símbolo de constante, digamos a, para designar el coche del padre de Manolo, con lo cual la representación sería

$$P(a).$$

El elegir una representación u otra, depende del uso que hagamos, el cual vendrá dado por las restantes frases.

4. *Los amigos de los amigos de Pepe son amigos de Pepe.*

   Representamos a "Pepe" mediante un símbolo de constante a y usamos el símbolo de predicado P^2 de modo que $P(x, y)$ designa que y es amigo de x. Una solución es

$$\forall x \forall y \Big(P(a, x) \wedge P(x, y) \to P(a, y) \Big).$$

5. *Puedes engañar a todo el mundo algún tiempo. Puedes engañar a algunos todo el tiempo. Pero no puedes engañar a todo el mundo todo el tiempo* (Abraham Lincoln).

   Definimos el símbolo de predicado $M(x, t)$ que designa que es posible engañar a x en el instante de tiempo t. Entonces una traducción posible sería:

$$\exists t \forall x M(x, t) \wedge \exists x \forall t M(x, t) \wedge \neg \forall x \forall t M(x, t).$$

□

3 Ocurrencias libres y ligadas. Variables libres y ligadas. Sentencias.

El **radio de acción** de un cuantificador en una fórmula α es la subfórmula de α a la cual ese cuantificador afecta. Así en la fórmula

$$\forall x \beta,$$

el cuantificador universal $\forall$ cuantifica al símbolo de variable x, y su radio de acción es la subfórmula β. Análogamente en la fórmula

$$\exists x \gamma,$$

el radio de acción del cuantificador $\exists$ es la subfórmula γ.

Ejemplo 3. En la fórmula

$$\forall x R(x) \to Q(x, y)$$

el radio de acción del cuantificador $\forall$ es $R(x)$. □

Ejemplo 4. En la fórmula
$$\forall x\Big(R(x) \wedge \exists y Q(x,y)\Big)$$
el radio de acción del cuantificador $\forall$ es $R(x) \wedge \exists y Q(x,y)$, y el radio de acción del cuantificador $\exists$ es $Q(x,y)$. □

Ejemplo 5. En la fórmula
$$\forall x \exists y\Big(R(x) \wedge Q(x,y)\Big)$$
el radio de acción del cuantificador $\forall$ es $\exists y\Big(R(x) \wedge Q(x,y)\Big)$, y el radio de acción del cuantificador $\exists$ es $R(x) \wedge Q(x,y)$. □

Una **ocurrencia** de una variable x en una fórmula α es una aparición de x en la escritura de α. Una ocurrencia de una variable x en una fórmula α se denomina **ligada** si aparece inmediatamente detrás de un cuantificador, o bien aparece en el radio de acción de un cuantificador (universal o existencial) cuya variable acompañante sea x. Una ocurrencia de una variable en una fórmula es **libre**, si no es ligada.

Ejemplo 6. En la fórmula
$$\forall x\Big(R(x,z) \wedge \exists y \forall z Q(x,y,z)\Big)$$
todas las ocurrencias de x son ligadas; la primera ocurrencia de z es libre y las dos siguientes son ligadas; las dos ocurrencias de y son ligadas. □

Ejemplo 7. En la fórmula
$$\forall x \exists y P(x) \wedge Q(x,y)$$
de izquierda a derecha, las dos primeras ocurrencias de x son ligadas mientras que la tercera es libre; la primera ocurrencia de y es ligada y la segunda es libre. □

Una fórmula es una **sentencia** si todas las ocurrencias de todas sus variables son ligadas. Dicho de otra forma, cuando no contiene ocurrencias libres de ninguna variable.

Ejemplo 8. La fórmula
$$\forall x P(x) \wedge \exists y Q(x,y)$$
no es una sentencia, mientras que
$$\forall x \exists y\Big(P(x) \wedge Q(x,y)\Big)$$
sí lo es. □

Resulta que todas las fórmulas que proceden de traducir un enunciado del lenguaje hablado tienen todas sus variables ligadas, y por tanto son sentencias. Mire de nuevo los distintos apartados del Ejemplo 2 y comprobará ésto que acabamos de decir.

4 Estructuras e interpretaciones.

En esta sección generalizamos el concepto de interpretación dado en el tema anterior. Ahora el valor de verdad de una fórmula dependerá del significado que demos a los símbolos de relación, de función y de constante que aparecen en dicha fórmula, así como del dominio en el que se consideren sus variables.

Para un lenguaje de primer orden $\mathcal{L}$, una **estructura**, también llamada **$\mathcal{L}$-estructura**, es una cuaterna

$$\mathcal{E} = \Big(\mathrm{D}, \{c_i^{\mathcal{E}}\}, \{f_j^{\mathcal{E}}\}, \{R_k^{\mathcal{E}}\}\Big),$$

donde:

1. D es un conjunto <u>no vacío</u> llamado **dominio** o **universo** de la $\mathcal{L}$-estructura,
2. a cada símbolo de constante $c_i \in \mathrm{Cons}(\mathcal{L})$ le corresponde un elemento $c_i^{\mathcal{E}} \in \mathrm{D}$,
3. a cada símbolo de función $f_j^n \in \mathrm{Func}(\mathcal{L})$ le corresponde una aplicación:
$$f_j^{\mathcal{E}} : \mathrm{D}^n \to \mathrm{D},$$
4. a cada símbolo de relación $R_k^m \in \mathrm{Rel}(\mathcal{L})$ le corresponde una aplicación
$$R_k^{\mathcal{E}} : \mathrm{D}^m \to \mathbb{Z}_2.$$

Al igual que en el tema anterior usamos el conjunto $\mathbb{Z}_2 = \{\mathbf{0}, \mathbf{1}\}$ formado por los dos valores de verdad, donde **1** indicará verdadero y **0** falso. Por tanto al definir una estructura sobre un lenguaje de predicados $\mathcal{L}$, le estamos dando un significado concreto a cada símbolo del lenguaje, salvo a los símbolos de variable.

A veces para denotar el dominio de una estructura $\mathcal{E}$ escribiremos $\mathrm{D}_{\mathcal{E}}$.

Ejemplo 9. Damos una estructura $\mathcal{E}$ para el lenguaje de primer orden $\mathcal{L}$ definido por $\mathrm{Cons}(\mathcal{L}) = \{a\}$, $\mathrm{Func}(\mathcal{L}) = \{f^1\}$ y $\mathrm{Rel}(\mathcal{L}) = \{P^1, Q^2, R^0\}$:

1. $\mathrm{D} = \mathbb{R}$.
2. $a^{\mathcal{E}} = \sqrt{2}$.
3. $f^{\mathcal{E}} : \mathbb{R} \to \mathbb{R},\ f^{\mathcal{E}}(r) = r^2 + \cos(r) - 1$.
4. - Para $r \in \mathbb{R},\ P^{\mathcal{E}}(r)$ es verdad si y sólo si $r \in \mathbb{Z}$.

     Dicho de otra forma, $P^{\mathcal{E}} : \mathbb{R} \to \mathbb{Z}_2$ es la aplicación tal que
$$P^{\mathcal{E}}(r) = \begin{cases} \mathbf{1} & \text{si } r \in \mathbb{Z}, \\ \mathbf{0} & \text{en otro caso.} \end{cases}$$

- Para $r, s \in \mathbb{R}$, $Q^{\mathcal{E}}(r,s)$ es verdad si y sólo si $r \leq s$.

  Es decir, $Q^{\mathcal{E}} : \mathbb{R} \times \mathbb{R} \to \mathbb{Z}_2$ es la aplicación tal que

$$Q^{\mathcal{E}}(r,s) = \begin{cases} \mathbf{1} & \text{si } r \leq s, \\ \mathbf{0} & \text{en otro caso.} \end{cases}$$

- $R^{\mathcal{E}}$ es falso.

  En este caso, $R^{\mathcal{E}} : \varnothing \to \mathbb{Z}_2$ es la aplicación constante $\mathbf{0}$.

$\square$

A continuación nos ocupamos de la interpretación de los símbolos de variable que aparecen en las fórmulas.

Dados un lenguaje $\mathcal{L}$ y una $\mathcal{L}$-estructura $\mathcal{E}$, una **asignación** en $\mathcal{E}$ es una aplicación

$$v : \mathrm{Var}(\mathcal{L}) \to \mathrm{D},$$

donde D es el dominio de $\mathcal{E}$.

Toda asignación v en una $\mathcal{L}$-estructura $\mathcal{E}$ se extiende de forma única al conjunto de los términos del lenguaje $\mathcal{L}$ de la siguiente manera. Para ello consideramos la aplicación

$$v' : \mathrm{Term}(\mathcal{L}) \to \mathrm{D}$$

definida por

$$v'(t) = \begin{cases} c^{\mathcal{E}} & \text{si } t = c \in \mathrm{Cons}(\mathcal{L}), \\ v(x) & \text{si } t = x \in \mathrm{Var}(\mathcal{L}), \\ f^{\mathcal{E}}(v'(t_1), \ldots, v'(t_n)) & \text{si } t = f(t_1, \ldots, t_n). \end{cases}$$

De esta forma a cada término del lenguaje $\mathcal{L}$ le estamos asignando un valor en $\mathrm{D}_{\mathcal{E}}$. A partir de ahora, vamos a identificar v con su extensión v', ya que v' queda unívocamente determinada una vez que se conoce v. Además especificaremos únicamente los valores de v para los símbolos de variable que nos interesen, es decir, aquellos que intervienen en las fórmulas que estamos considerando.

Ejemplo 10. Sea el lenguaje $\mathcal{L}$ definido por

$$\mathrm{Cons}(\mathcal{L}) = \{a\},\ \mathrm{Func}(\mathcal{L}) = \{f^2\} \ \text{ y } \ \mathrm{Rel}(\mathcal{L}) = \{Q^2\},$$

y la $\mathcal{L}$-estructura $\mathcal{E}$ siguiente:

1. $\mathrm{D} = \mathbb{Z}$.

2. $a^{\mathcal{E}} = 8$.

3. $f^{\mathcal{E}} : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z},\ f^{\mathcal{E}}(r,s) = r^2 - s$.

4. Para $r, s \in \mathbb{Z},\ Q^{\mathcal{E}}(r,s)$ es verdad si y sólo si $r \leq s$.

Sea v la asignación en la $\mathcal{L}$-estructura $\mathcal{E}$ tal que:

$$v(x) = 3,\ v(y) = -1.$$

Entonces

$$v(a) = a^{\mathcal{E}} = 8;$$

$$v(f(x,a)) = f^{\mathcal{E}}(v(x), v(a)) = f^{\mathcal{E}}(3,8) = 3^2 - 8 = 1;$$

$$v(f(a, f(y,x))) = f^{\mathcal{E}}(v(a), v(f(y,x))) = f^{\mathcal{E}}(v(a), f^{\mathcal{E}}(v(y), v(x))) = \\ f^{\mathcal{E}}(8, f^{\mathcal{E}}(-1,3)) = f^{\mathcal{E}}(8, (-1)^2 - 3) = f^{\mathcal{E}}(8,-2) = 8^2 - (-2) = 66.$$

□

Dada una asignación v para un lenguaje $\mathcal{L}$ en una estructura $\mathcal{E}$, un símbolo de variable $x \in \mathrm{Var}(\mathcal{L})$ y un elemento $r \in \mathrm{D}$, definimos $v(x|r)$ como la asignación que actúa sobre todos los símbolos de variable de $\mathcal{L}$ igual que lo hace v, excepto sobre x al cual le asigna el valor $r \in \mathrm{D}$.

Análogamente se define $v(x_1|r_1, \ldots, x_n|r_n)$, que sería igual que v en los símbolos de variable $y \notin \{x_1, \ldots, x_n\}$ y $v(x_1|r_1, \ldots, x_n|r_n)(x_i) = r_i$ para cada $i \in \{1, \ldots, n\}$.

Ejemplo 11. Para el lenguaje $\mathcal{L}$ y la estructura $\mathcal{E}$ del ejemplo anterior, hemos considerado una asignación v tal que

$$v(x) = 3,\ v(y) = -1.$$

Ahora consideramos la asignación w_1 definida a partir de v como $w_1 = v(x|7)$. Entonces

$$w_1(x) = 7,\ w_1(y) = -1.$$

Análogamente, si definimos $w_2 = v(x|7, y|0)$, entonces

$$w_2(x) = 7,\ w_2(y) = 0.$$

□

Estamos ya en condiciones de definir formalmente lo que vamos a entender por interpretación de una fórmula, que no es otra cosa que una extensión "natural" de las interpretaciones que vimos en el capítulo anterior para la Lógica proposicional.

Una **$\mathcal{L}$-interpretación** ó interpretación para un lenguaje de predicados de primer orden $\mathcal{L}$, es un triple ordenado $(\mathcal{E}, v, \mathrm{I}^v)$, donde $\mathcal{E}$ es una $\mathcal{L}$-estructura, v es una asignación en $\mathcal{E}$ e I^v es una aplicación

$$\mathrm{I}^v : \mathrm{Form}(\mathcal{L}) \to \mathbb{Z}_2,$$

que verifica las siguientes propiedades:

1. $\mathrm{I}^v(R(t_1, \ldots, t_n)) = R^{\mathcal{E}}(v(t_1), \ldots, v(t_n))$, con $t_1, \ldots, t_n$ términos y R un símbolo de relación n-ario de $\mathcal{L}$.
2. $\mathrm{I}^v(\neg\alpha) = 1 \oplus \mathrm{I}^v(\alpha)$;
3. $\mathrm{I}^v(\alpha \vee \beta) = \mathrm{I}^v(\alpha) \oplus \mathrm{I}^v(\beta) \oplus \mathrm{I}^v(\alpha) \cdot \mathrm{I}^v(\beta)$;
4. $\mathrm{I}^v(\alpha \wedge \beta) = \mathrm{I}^v(\alpha) \cdot \mathrm{I}^v(\beta)$;
5. $\mathrm{I}^v(\alpha \to \beta) = \mathbf{1} \oplus \mathrm{I}^v(\alpha) \oplus \mathrm{I}^v(\alpha) \cdot \mathrm{I}^v(\beta)$;
6. $\mathrm{I}^v(\alpha \leftrightarrow \beta) = \mathbf{1} \oplus \mathrm{I}^v(\alpha) \oplus \mathrm{I}^v(\beta)$;
7. $$\mathrm{I}^v(\forall x\alpha) = \begin{cases} \mathbf{1}, & \text{si para todo } r \in \mathrm{D} \text{ se verifica que } \mathrm{I}^{v(x|r)}(\alpha) = \mathbf{1}, \\ \mathbf{0}, & \text{en caso contrario.} \end{cases}$$
8. $$\mathrm{I}^v(\exists x\alpha) = \begin{cases} \mathbf{1}, & \text{si existe } r \in \mathrm{D} \text{ tal que } \mathrm{I}^{v(x|r)}(\alpha) = \mathbf{1}, \\ \mathbf{0}, & \text{en caso contrario.} \end{cases}$$

El apartado (1) es el caso base y nos dice cómo se interpretan las fórmulas atómicas. Concretamente una fórmula atómica es cierta si y sólo si los elementos que son asignados a los argumentos que acompañan al símbolo de relación R están relacionados mediante la relación concreta que se le asocia al símbolo R. Los apartados (2)-(6) de la definición recogen la idea de cómo se deben interpretar las conectivas. Obsérvese que es exactamente igual a la definición de interpretación en Lógica proposicional. Por último, los apartados (7) y (8) corresponden a la interpretación de los cuantificadores. El séptimo apartado dice que la fórmula $\forall x\alpha$ es cierta en la $\mathcal{L}$-estructura $\mathcal{E}$ considerada y bajo la asignación v, si para cualquier valor que le asignemos al símbolo de variable x (de ahí el cambio de asignación de v por $v(x|r)$) la fórmula α es cierta en la $\mathcal{L}$-estructura dada con la nueva asignación. El octavo apartado es análogo al anterior.

Importante: Dada una $\mathcal{L}$-interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ y una fórmula $\alpha \in \mathrm{Form}(\mathcal{L})$, las ocurrencias libres de las variables que aparecen en α se interpretan haciendo uso de la asignación v mientras que las ocurrencias ligadas de las variables que aparecen en α se interpretan atendiendo al cuantificador correspondiente tal y como se indica en los apartados (7) y (8) de la definición anterior.

En este sentido, es interesante también conocer el siguiente resultado.

Lema 1 (**Lema de coincidencia**). Sea α una fórmula de un lenguaje de predicados $\mathcal{L}$ y sean $x_1, \ldots, x_n \in \mathrm{Var}(\mathcal{L})$ los símbolos de variable para cada uno de los cuales aparece en α al menos una ocurrencia libre. Sea $\mathcal{E}$ una $\mathcal{L}$-estructura y sean v_1 y v_2 dos asignaciones en $\mathcal{E}$ tales que $v_1(x_i) = v_2(x_i)$ para todo $i = 1, \ldots, n$. Entonces $\mathrm{I}^{v_1}(\alpha) = \mathrm{I}^{v_2}(\alpha)$. $\square$

Ejemplo 12. Para el lenguaje de primer orden $\mathcal{L}$ definido por

$$\mathrm{Cons}(\mathcal{L}) = \{a\},\ \mathrm{Func}(\mathcal{L}) = \{f^1\}\ \text{ y }\ \mathrm{Rel}(\mathcal{L}) = \{P^1, Q^2\},$$

consideramos la estructura $\mathcal{E}$ siguiente:

1. $\mathrm{D} = \mathbb{R}$.
2. $a^{\mathcal{E}} = \sqrt{5}$.
3. $f^{\mathcal{E}} : \mathbb{R} \to \mathbb{R},\ f^{\mathcal{E}}(r) = r^2$.
4. Para $r \in \mathbb{R},\ P^{\mathcal{E}}(r) = \mathbf{1}$ si y sólo si $r \in \mathbb{Z}$;

   Para $r, s \in \mathbb{R},\ Q^{\mathcal{E}}(r, s) = \mathbf{1}$ si y sólo si $r \leq s$.

También consideramos la asignación v tal que $v(x) = 4$, $v(y) = \frac{1}{100}$, y calculamos las interpretaciones siguientes:

1. $\mathrm{I}^v(P(a)) = P^{\mathcal{E}}(v(a)) = P^{\mathcal{E}}(a^{\mathcal{E}}) = P^{\mathcal{E}}(\sqrt{5}) = \mathbf{0}$.
2. $\mathrm{I}^v(P(x)) = P^{\mathcal{E}}(v(x)) = P^{\mathcal{E}}(4) = \mathbf{1}$.
3. $\mathrm{I}^v(Q(x, y)) = Q^{\mathcal{E}}(v(x), v(y)) = Q^{\mathcal{E}}(4, \frac{1}{100}) = \mathbf{0}$.
4. La fórmula $\exists x Q(x, y)$ nos dice que existe algún elemento del dominio, es decir, un número real x el cual es menor o igual que el valor asignado a y, es decir, $v(y) = \frac{1}{100}$. Ésto es claramente verdadero, pues basta tomar como x cualquier número real menor o igual que $\frac{1}{100}$. Por consiguiente

   $$\mathrm{I}^v(\exists x Q(x, y)) = \mathbf{1}.$$

   De forma más rigurosa, tomamos cualquier $r \in \mathrm{D} = \mathbb{R}$ tal que $r \leq 1/100$ y tenemos

   $$\mathrm{I}^{v(x|r)}(Q(x, y)) = Q^{\mathcal{E}}(w(x), w(y)) = Q^{\mathcal{E}}(r, \frac{1}{100}) = \mathbf{1},$$

   donde hemos denotado la asignación $v(x|r)$ por w. Por tanto

   $$\mathrm{I}^v(\exists x Q(x, y)) = \mathbf{1}.$$

5. La fórmula $\exists y P(x)$ nos dice que es posible asignarle al símbolo de variable y algún valor en D tal que $v(x)$ sea un número entero. Ésto es cierto pues según la definición de v, sabemos que $v(x) = 4$, que evidentemente pertenece a $\mathbb{Z}$. Por tanto $\mathrm{I}^v(\exists y P(x)) = \mathbf{1}$. De manera más formal, cogemos cualquier elemento $r \in \mathbb{R}$ y tenemos que
$$\mathrm{I}^{v(y|r)}(P(x)) = P^{\mathcal{E}}(w(x)) = P^{\mathcal{E}}(4) = \mathbf{1},$$
donde w representa $v(y|r)$. Por tanto
$$\mathrm{I}^v(\exists y P(x)) = \mathbf{1}.$$

6. $\mathrm{I}^v(\exists x P(x)) = \mathbf{1}$, ya que si cogemos cualquier $r \in \mathbb{Z}$, tenemos que
$$\mathrm{I}^{v(x|r)}(P(x)) = P^{\mathcal{E}}(w(x)) = P^{\mathcal{E}}(r) = \mathbf{1},$$
donde hemos denotado $v(x|r)$ por w. Por tanto
$$\mathrm{I}^v(\exists x P(x)) = \mathbf{1}.$$

7. $\mathrm{I}^v(P(f(a))) = \mathbf{1}$, pues estamos diciendo que $(\sqrt{5})^2 = 5 \in \mathbb{Z}$. De manera más precisa,
$$\mathrm{I}^v(P(f(a))) = P^{\mathcal{E}}(v(f(a))) = P^{\mathcal{E}}(f^{\mathcal{E}}(v(a))) = P^{\mathcal{E}}(f^{\mathcal{E}}(a^{\mathcal{E}})) = P^{\mathcal{E}}((\sqrt{5})^2) = P^{\mathcal{E}}(5) = \mathbf{1}.$$

8. La fórmula $\forall x \exists y Q(f(y), x)$ nos dice que para todo $x \in \mathbb{R}$ existe $y \in \mathbb{R}$ tal que $y^2 \leq x$. Sabemos que para cualquier $r \in \mathbb{R}$ se verifica que $r^2 \geq 0$; si la variable x en la fórmula anterior toma un valor negativo, no existirá $y \in \mathbb{R}$ tal que $y^2 \leq x$. Por consiguiente
$$\mathrm{I}^v\Big(\forall x \exists y Q(f(y), x\Big) = \mathbf{0}.$$

9. La fórmula $\forall x \exists y Q(x, f(y))$ nos dice que para todo $x \in \mathbb{R}$ existe (al menos) un elemento $y \in \mathbb{R}$ tal que $x \leq y^2$. Esta propiedad es cierta pues si $x \leq 0$, tomamos $y := 0$, mientras que si $x > 0$, cogemos $y := x + 1$. Por tanto
$$\mathrm{I}^v\Big(\forall x \exists y Q(x, f(y))\Big) = \mathbf{1}.$$

10. $\mathrm{I}^v\Big(\forall x \exists y Q(x, f(y)) \to \forall x \exists y Q(f(y), x)\Big) = \mathbf{0}$ como consecuencia de los dos apartados anteriores.

$\square$

5 Validez y satisfacibilidad.

Sean un lenguaje de predicados $\mathcal{L}$ y una fórmula $\alpha \in \mathrm{Form}(\mathcal{L})$. Entonces:

- α es **satisfacible** si existe alguna interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ de $\mathcal{L}$ tal que $\mathrm{I}^v(\alpha) = \mathbf{1}$. En dicho caso se dice que la interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ satisface a α así como que es un **modelo** para α.

- α se dice **satisfacible en una estructura** $\mathcal{E}$, si existe alguna interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ que satisface a α.

- Cuando una fórmula no es satisfacible, esto es, cuando es falsa bajo cualquier interpretación, la llamamos **contradicción**.

- α es **refutable** si $\neg\alpha$ es satisfacible. De modo equivalente α es **refutable** si existe una interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ de $\mathcal{L}$ tal que $\mathrm{I}^v(\alpha) = \mathbf{0}$.

- α es **válida** en una $\mathcal{L}$-estructura $\mathcal{E}$ si para toda asignación v en $\mathcal{E}$ se verifica que $\mathrm{I}^v(\alpha) = \mathbf{1}$.

- α es **universalmente válida** si es válida en toda $\mathcal{L}$-estructura.

Comentarios:

1. Podemos extender las definiciones anteriores, diciendo que un conjunto de fórmulas $\Omega \subseteq \mathrm{Form}(\mathcal{L})$ es satisfacible, si existe alguna interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ de $\mathcal{L}$ tal que $\mathrm{I}^v(\alpha) = \mathbf{1}$ para toda $\alpha \in \Omega$. En tal caso diremos que la interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ satisface a Ω, así como que $(\mathcal{E}, v, \mathrm{I}^v)$ es un modelo para Ω, y escribiremos $\mathrm{I}^v(\Omega) = \mathbf{1}$.

   El conjunto de fórmulas Ω es satisfacible en la estructura $\mathcal{E}$, si existe alguna interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ que satisface a Ω.

2. Observamos que cualquier fórmula de $\mathcal{L}$ puede encuadrarse en uno de los siguientes tipos:

   - Universalmente válida.
   - Satisfacible y refutable.
   - Contradicción.

3. Si α es válida en una $\mathcal{L}$-estructura $\mathcal{E}$, en particular α es satisfacible en $\mathcal{E}$. Sin embargo puede ocurrir que α sea satisfacible en $\mathcal{E}$ sin que sea válida en $\mathcal{E}$.

4. α es universalmente válida si y sólo si $\neg\alpha$ es una contradicción.

5. Para probar que una fórmula es satisfacible o refutable hemos de buscar una interpretación concreta bajo la cual la fórmula se haga verdadera o falsa, respectivamente. Sin embargo para probar que es universalmente válida o una contradicción tenemos que dar un razonamiento de tipo general que sea válido para cualquier interpretación posible de dicha fórmula.

Ejemplo 13. Sea $\mathcal{L}$ un lenguaje de predicados de primer orden.

1. Para cualquier fórmula $\alpha \in \mathrm{Form}(\mathcal{L})$, la fórmula $\alpha \to \alpha$ es universalmente válida, pues para toda interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ de $\mathcal{L}$, por la definición de la aplicación I^v tenemos que
$$\mathrm{I}^v(\alpha \to \alpha) = \mathbf{1} \oplus \mathrm{I}^v(\alpha) \oplus \mathrm{I}^v(\alpha) \cdot \mathrm{I}^v(\alpha) = \mathbf{1}.$$
Este ejemplo se puede generalizar de la manera siguiente. A partir de una tautología β de la Lógica proposicional obtenemos fórmulas universalmente válidas de la Lógica de predicados simplemente reemplazando cada variable proposicional que aparece en β por una fórmula cualquiera de $\mathcal{L}$.

   Por ejemplo, consideremos la proposición lógica $\beta := P \to (Q \to P)$ construida a partir de las variables proposicionales P y Q. Es inmediato comprobar que β es una tautología. Por otra parte, si sabemos que $\exists x R(x, y)$ y $P(f(a))$ son fórmulas de un lenguaje de predicados $\mathcal{L}$, entonces la fórmula siguiente, también perteneciente a $\mathcal{L}$,
$$\exists x R(x, y) \to \Big(P(f(a) \to \exists x R(x, y)\Big)$$
es universalmente válida.

   Decir también que existen fórmulas universalmente válidas que no son obtenibles a partir de ninguna tautología proposicional. Véase el apartado (6.) más abajo.

2. Para cualquier $\alpha \in \mathrm{Form}(\mathcal{L})$, la fórmula $\alpha \wedge \neg\alpha$ es una contradicción, pues
$$\mathrm{I}^v(\alpha \wedge \neg\alpha) = \mathrm{I}^v(\alpha) \cdot (\mathbf{1} \oplus \mathrm{I}^v(\alpha)) = \mathrm{I}^v(\alpha) \oplus \mathrm{I}^v(\alpha) = \mathbf{0}$$
para toda interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ de $\mathcal{L}$.

3. La fórmula $P(x) \to P(y)$ es válida en toda estructura $\mathcal{E}$ cuyo dominio tenga cardinal 1, es decir, $\mathrm{D}_{\mathcal{E}} = \{r\}$, pues en tal caso cualquier asignación v en $\mathcal{E}$ verifica que $v(x) = v(y) = r$. Comprobemos ésto:
$$\begin{aligned}
&\mathrm{I}^v\Big(P(x) \to P(y)\Big) = \\
&\mathbf{1} \oplus \mathrm{I}^v(P(x)) \oplus \mathrm{I}^v(P(x)) \cdot \mathrm{I}^v(P(y)) = \\
&\mathbf{1} \oplus P^{\mathcal{E}}(v(x)) \oplus P^{\mathcal{E}}(v(x)) \cdot P^{\mathcal{E}}(v(y)) = \\
&\mathbf{1} \oplus P^{\mathcal{E}}(r) \oplus P^{\mathcal{E}}(r) \cdot P^{\mathcal{E}}(r) = \\
&\mathbf{1} \oplus P^{\mathcal{E}}(r) \oplus P^{\mathcal{E}}(r) = \\
&\mathbf{1}.
\end{aligned}$$

4. También puede comprobarse de manera similar que $P(x) \to P(y)$ es válida en toda estructura en la cual la aplicación $P^{\mathcal{E}}$ (es decir, la forma de interpretar el símbolo de predicado P) es constante en el dominio $\mathrm{D}_{\mathcal{E}}$.

5. Por el apartado anterior, la fórmula $P(x) \to P(y)$ es satisfacible. Ésta también es refutable. Para ver ésto último, consideramos la estructura dada por

$$\begin{aligned}
&\mathrm{D}_{\mathcal{E}} = \mathbb{Z}, \\
&P^{\mathcal{E}}(r) = \mathbf{1} \;\Leftrightarrow\; r \text{ es par},
\end{aligned}$$

   y la asignación

$$v(x) = 2, \quad v(y) = 3.$$

   Entonces

$$\mathrm{I}^v\Big(P(x) \to P(y)\Big) = \mathbf{1} \oplus \mathrm{I}^v(P(x)) \oplus \mathrm{I}^v(P(x)) \cdot \mathrm{I}^v(P(y)) = \mathbf{1} \oplus \mathbf{1} \oplus \mathbf{1} \cdot \mathbf{0} = \mathbf{0}.$$

   Recuérdese que las ocurrencias libres de variables se interpretan mediante la asignación utilizada.

6. La fórmula $\forall x P(x) \to \exists y P(y)$ es universalmente válida, pues intuitivamente, si aceptamos que el símbolo de predicado P se verifica para todos los elementos del dominio considerado D, como D nunca es vacío, ciertamente existe algún elemento perteneciente a D para el cual P se hace verdad.

   De manera más rigurosa, si tenemos una interpretación cualquiera $(\mathcal{E}, v, \mathrm{I}^v)$ para $\mathcal{L}$, pueden ocurrir dos posibilidades excluyentes:

   - $\mathrm{I}^v(\forall x P(x)) = \mathbf{0}$. Entonces

$$\begin{aligned}
&\mathrm{I}^v(\forall x P(x) \to \exists y P(y)) = \\
&\mathbf{1} \oplus \mathrm{I}^v(\forall x P(x)) \oplus \mathrm{I}^v(\forall x P(x)) \cdot \mathrm{I}^v(\exists y P(y)) = \\
&\mathbf{1} \oplus \mathbf{0} \oplus \mathbf{0} \cdot \mathrm{I}^v(\exists y P(y)) = \\
&\mathbf{1}.
\end{aligned}$$

   - $\mathrm{I}^v(\forall x P(x)) = \mathbf{1}$, lo cual significa que para todo elemento $r \in \mathrm{D}_{\mathcal{E}}$ se cumple que $P^{\mathcal{E}}(r) = \mathbf{1}$. Como $\mathrm{D}_{\mathcal{E}} \neq \varnothing$, ésto implica que $\mathrm{I}^v(\exists y P(y)) = \mathbf{1}$. Por tanto resulta que

$$\begin{aligned}
&\mathrm{I}^v(\forall x P(x) \to \exists y P(y)) = \\
&\mathbf{1} \oplus \mathrm{I}^v(\forall x P(x)) \oplus \mathrm{I}^v(\forall x P(x)) \cdot \mathrm{I}^v(\exists y P(y)) = \\
&\mathbf{1} \oplus \mathbf{1} \oplus \mathbf{1} \cdot \mathbf{1} = \\
&\mathbf{1}.
\end{aligned}$$

Como en ambos casos obtenemos el valor lógico **1**, deducimos que la fórmula propuesta es universalmente válida.

En el tema siguiente veremos un método sistemático para decidir cuándo una fórmula es universalmente válida.

□

En la sección anterior hemos visto el Lema de coincidencia el cual implica que si una fórmula α es una **sentencia**, es decir, una fórmula sin variables libres, entonces la asignación no es relevante para calcular la interpretación de α.

Incluímos aquí las dos propiedades siguientes que se deducen del Lema de coincidencia.

Corolario 1. Dada una sentencia α y $\mathcal{E}$ una $\mathcal{L}$-estructura, son equivalentes:

1. α es válida en $\mathcal{E}$,
2. α es satisfacible en $\mathcal{E}$.

□

Corolario 2. Dada una sentencia α y $\mathcal{E}$ una $\mathcal{L}$-estructura, entonces es cierta una de las siguientes afirmaciones, y sólo una:

1. α es válida en $\mathcal{E}$,
2. $\neg\alpha$ es válida en $\mathcal{E}$.

□

6 Consecuencia Lógica. El Teorema de la deducción.

Sea $\mathcal{L}$ un lenguaje de predicados y $\Gamma \cup \{\beta\} \subseteq \text{Form}(\mathcal{L})$. Decimos que β es **consecuencia lógica** de Γ, o bien que Γ **implica semánticamente** a β, y lo notaremos por $\Gamma \models \beta$, si para toda $\mathcal{L}$-interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ que satisface a Γ (es decir, que verifica $\mathrm{I}^v(\alpha) = \mathbf{1}$ para toda fórmula $\alpha \in \Gamma$), se cumple que $\mathrm{I}^v(\beta) = \mathbf{1}$.

Si $\Gamma = \varnothing$, entonces escribiremos $\models \beta$ en lugar de $\varnothing \models \beta$. Se puede comprobar que $\models \beta$ equivale a decir que β sea universalmente válida.

Como se puede ver, la idea es esencialmente la misma que la dada en el tema de Lógica proposicional pero ahora adaptada al contexto de la Lógica de predicados. Análogamente a como vimos allí, si ahora $\Gamma = \{\alpha_1, \dots, \alpha_n\}$ y $\Gamma \models \beta$, escribiremos simplemente

$$\alpha_1, \dots, \alpha_n \models \beta.$$

Como la Lógica de proposiciones está incluída en la Lógica de predicados, todos los ejemplos de implicación semántica vistos allí, siguen siendo ciertos ahora.

Ejemplo 14. Si α y β son dos fórmulas de un lenguaje de predicados $\mathcal{L}$, tenemos el esquema de *Modus ponens*, es decir,

$$\Big\{\alpha,\ \alpha \to \beta\Big\} \models \beta.$$

Para comprobarlo, tomemos $(\mathcal{E}, v, \mathrm{I}^v)$ una $\mathcal{L}$-interpretación cualquiera tal que

$$\mathrm{I}^v(\alpha) = \mathbf{1} \quad \text{y} \quad \mathrm{I}^v(\alpha \to \beta) = \mathbf{1}.$$

Entonces

$$\mathbf{1} = \mathbf{1} \oplus \mathrm{I}^v(\alpha) \oplus \mathrm{I}^v(\alpha) \cdot \mathrm{I}^v(\beta) = \mathbf{1} \oplus \mathbf{1} \oplus \mathrm{I}^v(\beta),$$

de donde

$$\mathrm{I}^v(\beta) = \mathbf{1}.$$

□

Ejemplo 15. Veamos que en cualquier lenguaje de predicados $\mathcal{L}$ se verifica que

$$\forall x P(x) \vee \forall x Q(x) \models \forall x\big(P(x) \vee Q(x)\big).$$

Sea $(\mathcal{E}, v, \mathrm{I}^v)$ una $\mathcal{L}$-interpretación de forma que

$$\mathrm{I}^v\Big(\forall x P(x) \vee \forall x Q(x)\Big) = \mathbf{1}.$$

Ésto implica que

$$\mathrm{I}^v\Big(\forall x P(x)\Big) = \mathbf{1} \quad \text{o bien} \quad \mathrm{I}^v\Big(\forall x Q(x)\Big) = \mathbf{1}.$$

Supongamos que $\mathrm{I}^v(\forall x P(x)) = \mathbf{1}$. El caso $\mathrm{I}^v(\forall x Q(x)) = \mathbf{1}$ se resuelve de manera similar. Entonces para todo $r \in \mathrm{D}_{\mathcal{E}}$ tenemos que

$$P^{\mathcal{E}}(r) = \mathbf{1}.$$

Ahora bien, ya sea $Q^{\mathcal{E}}(r) = \mathbf{1}$ ó $Q^{\mathcal{E}}(r) = \mathbf{0}$, se cumple que

$$P^{\mathcal{E}}(r) \oplus Q^{\mathcal{E}}(r) \oplus P^{\mathcal{E}}(r) \cdot Q^{\mathcal{E}}(r) = \mathbf{1} \oplus Q^{\mathcal{E}}(r) \oplus \mathbf{1} \cdot Q^{\mathcal{E}}(r) = \mathbf{1}$$

para todo $r \in \mathrm{D}_{\mathcal{E}}$, lo cual significa que

$$\mathrm{I}^v\Big(\forall x\big(P(x) \vee Q(x)\big)\Big) = \mathbf{1}.$$

□

El método empleado en el ejemplo anterior es bastante artesanal, pues se basa únicamente en la definición de implicación semántica. En el tema siguiente veremos un método sistemático para decidir cuándo una fórmula es consecuencia lógica de un conjunto de fórmulas.

Si queremos demostrar que una fórmula β no es consecuencia lógica de un conjunto Γ, es decir,

$$\Gamma \not\models \beta,$$

tenemos que encontrar una interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ que sea un modelo para Γ pero no lo sea para β.

Ejemplo 16. Veamos que

$$\forall x\big(P(x) \vee Q(x)\big) \not\models \forall x P(x) \vee \forall x Q(x).$$

Consideramos la estructura $\mathcal{E}$ dada por

$$\begin{aligned} &\mathrm{D} = \mathbb{Z}, \\ &P^{\mathcal{E}}(r) = \mathbf{1} \Leftrightarrow r \text{ es par}, \\ &Q^{\mathcal{E}}(r) = \mathbf{1} \Leftrightarrow r \text{ es impar}. \end{aligned}$$

Entonces $\forall x(P(x) \vee Q(x))$ nos dice que dado un número entero cualquiera, éste siempre será par o impar, lo cual es verdadero, mientras que $\forall x P(x) \vee \forall x Q(x)$ nos dice que todos los números enteros son pares o bien todos son impares, lo cual es falso. $\square$

Aquí también tenemos el correspondiente Teorema de la deducción cuya utilidad ya quedó patente en el tema anterior a la hora de resolver ejercicios sobre implicación semántica.

Teorema 1 (Teorema de la deducción). Dado un conjunto de fórmulas $\Gamma \cup \{\alpha, \beta\}$ en un lenguaje de predicados, son equivalentes las afirmaciones siguientes:

- $\Gamma \cup \{\alpha\} \models \beta$,
- $\Gamma \models \alpha \to \beta$.

$\square$

Ejemplo 17. Decir que la fórmula

$$\alpha = \forall x P(x) \vee \forall x Q(x) \to \forall x\big(P(x) \vee Q(x)\big)$$

es universalmente válida, es lo mismo que afirmar

$$\models \forall x P(x) \vee \forall x Q(x) \to \forall x\big(P(x) \vee Q(x)\big),$$

lo cual por el Teorema de la deducción es equivalente a que

$$\forall x P(x) \vee \forall x Q(x) \models \forall x\big(P(x) \vee Q(x)\big).$$

Pero ésto ya lo constatamos en un ejemplo anterior, por lo que α es universalmente válida. $\square$

Nótese que la fórmula $\forall x(P(x) \to Q(x))$ no tiene el formato $\alpha \to \beta$, por lo cual, si tenemos un conjunto de fórmulas Γ y nos planteamos

$$\Gamma \models \forall x\big(P(x) \to Q(x)\big),$$

no podemos aplicar el Teorema de la deducción.

7 Consecuencia lógica y conjuntos insatisfacibles.

En lo sucesivo nos vamos a centrar en intentar resolver el problema de la implicación semántica, es decir, decidir si $\Gamma \models \beta$.

De forma similar a la definición dada en el tema anterior, un conjunto de fórmulas se dice que es **insatisfacible** o **inconsistente**, si no existe ninguna interpretación que haga ciertas simultáneamente todas las fórmulas del conjunto.

El problema de decidir la consecuencia lógica para una fórmula se puede transformar en el de decidir la insatisfacibilidad de un conjunto de fórmulas usando el siguiente resultado.

Teorema 2. Sea $\Gamma \cup \{\beta\}$ un conjunto de fórmulas en un lenguaje de predicados de primer orden. Entonces las siguientes afirmaciones son equivalentes:

1. $\Gamma \models \beta$,
2. $\Gamma \cup \{\neg\beta\}$ es insatisfacible.

$\square$

En el próximo capítulo nos ocuparemos de transformar un conjunto de fórmulas inicial en otros que conserven el carácter de satisfacibilidad ó insatisfacibilidad de dicho conjunto y que sean más sencillos de estudiar. Daremos algoritmos para determinar, en algunos casos, la insatisfacibilidad de tales conjuntos.

8 Equivalencia lógica.

Dos fórmulas $\alpha, \beta \in \mathrm{Form}(\mathcal{L})$ son **equivalentes**, si para cualquier $\mathcal{L}$-interpretación $(\mathcal{E}, v, \mathrm{I}^v)$ se verifica que $\mathrm{I}^v(\alpha) = \mathrm{I}^v(\beta)$, es decir, si la fórmula $\alpha \leftrightarrow \beta$ es universalmente válida. Escribiremos $\alpha \equiv \beta$ para indicar que las fórmulas α y β son equivalentes.

Como es de esperar, la equivalencia lógica define una relación de equivalencia en el conjunto $\mathrm{Form}(\mathcal{L})$.

La siguiente propiedad nos dice que la relación de equivalencia lógica es compatible con las conectivas u operadores lógicos.

Lema 2. Sean $\alpha_1, \alpha_2, \beta_1, \beta_2 \in \mathrm{Form}(\mathcal{L})$. Entonces:

1. $\alpha_1 \equiv \beta_1 \implies \neg\alpha_1 \equiv \neg\beta_1$,

2. $$\left.\begin{array}{l}\alpha_1 \equiv \beta_1\\ \alpha_2 \equiv \beta_2\end{array}\right\} \implies \alpha_1{\wedge}\alpha_2 \equiv \beta_1{\wedge}\beta_2,$$

3. $$\left.\begin{array}{l}\alpha_1 \equiv \beta_1\\ \alpha_2 \equiv \beta_2\end{array}\right\} \implies \alpha_1{\vee}\alpha_2 \equiv \beta_1{\vee}\beta_2,$$

4. $\alpha_1 \equiv \beta_1 \implies \forall x\alpha_1 \equiv \forall x\beta_1$, para cualquier $x \in \mathrm{Var}(\mathcal{L})$,

5. $\alpha_1 \equiv \beta_1 \implies \exists x\alpha_1 \equiv \exists x\beta_1$, para cualquier $x \in \mathrm{Var}(\mathcal{L})$.

□

Todas las equivalencias lógicas estudiadas en el tema anterior, aquí siguen siendo válidas. Ahora estudiamos otras que involucran cuantificadores.

Lema 3. Para cualesquiera $\alpha, \beta \in \mathrm{Form}(\mathcal{L})$:

1. $\neg\forall x\alpha \equiv \exists x\neg\alpha$,

2. $\neg\exists x\alpha \equiv \forall x\neg\alpha$,

3. $\exists x\alpha \equiv \neg\forall x\neg\alpha$,

4. $\forall x\alpha \equiv \neg\exists x\neg\alpha$.

□

Estas propiedades nos indican cómo se relaciona la conectiva $\neg$ con los cuantificadores. Nos dice **cómo se interioriza** (y también cómo se extrae) **la conectiva $\neg$ en una fórmula cuantificada**.

Ejemplo 18. Dada la fórmula

$$\alpha := \neg\exists x\Big(R(x,a) \wedge \neg Q(f(x))\Big),$$

hacemos uso del lema anterior, de las Leyes de De Morgan y de las Leyes de doble negación, y obtenemos

$$\alpha \;\equiv\; \forall x\Big(\neg R(x,a) \vee Q(f(x))\Big).$$

□

Ejemplo 19. La fórmula

$$\neg\exists x\forall y\Big(R(x,y)\to Q(g(x,y))\Big)$$

es equivalente a

$$\forall x\neg\forall y\Big(R(x,y)\to Q(g(x,y))\Big),$$

la cual es equivalente a

$$\forall x\exists y\neg\Big(R(x,y)\to Q(g(x,y))\Big).$$

Si recordamos que

$$\alpha\to\beta\;\equiv\;\neg\alpha\vee\beta,$$

entonces

$$\forall x\exists y\neg\Big(R(x,y)\to Q(g(x,y))\Big)\equiv\forall x\exists y\neg\Big(\neg R(x,y)\,\vee\,Q(g(x,y))\Big).$$

Usamos de nuevo las Leyes de De Morgan y las Leyes de doble negación, y obtenemos

$$\forall x\exists y\neg\Big(\neg R(x,y)\,\vee\,Q(g(x,y))\Big)\equiv\forall x\exists y\Big(R(x,y)\,\wedge\,\neg Q(g(x,y))\Big).$$

Así pues, la fórmula inicial es equivalente a ésta última. □

Lema 4. Para cualesquiera $\alpha,\beta\in\mathrm{Form}(\mathcal{L})$:

1. $\forall x\alpha\wedge\forall x\beta\equiv\forall x(\alpha\wedge\beta)$,
2. $\exists x\alpha\vee\exists x\beta\equiv\exists x(\alpha\vee\beta)$.

□

El apartado (1.) en la propiedad anterior nos dice que **el cuantificador universal distribuye al operador** $\wedge$, mientras que el apartado (2.) nos dice que **el cuantificador existencial distribuye al operador** $\vee$.

Ejemplo 20. La fórmula

$$\forall xP(x)\,\wedge\,\forall xR(x,a)$$

es equivalente a

$$\forall x\Big(P(x)\,\wedge\,R(x,a)\Big).$$

La fórmula

$$\forall xP(x)\,\wedge\,\forall xR(x,a)\,\wedge\,\forall x\exists yQ(f(x),y)$$

es equivalente a

$$\forall x\Big(P(x)\,\wedge\,R(x,a)\,\wedge\,\exists yQ(f(x),y)\Big).$$

La fórmula

$$\exists x P(x) \vee \exists x R(x,a)$$

es equivalente a

$$\exists x \Big(P(x) \vee R(x,a) \Big).$$

□

Lema 5. Sean $\alpha, \beta \in \text{Form}(\mathcal{L})$ y supongamos que no hay ocurrencias libres de x en β. Entonces:

1. $\forall x \alpha \wedge \beta \equiv \forall x(\alpha \wedge \beta)$,
2. $\exists x \alpha \wedge \beta \equiv \exists x(\alpha \wedge \beta)$,
3. $\forall x \alpha \vee \beta \equiv \forall x(\alpha \vee \beta)$,
4. $\exists x \alpha \vee \beta \equiv \exists x(\alpha \vee \beta)$.

□

La propiedad anterior nos dice **cómo incluir los operadores $\vee$ e $\wedge$ en el radio de acción de un cuantificador** supuesto que la parte a incluir no interfiere con la variable cuantificada.

Ejemplo 21. Consideremos la fórmula

$$\delta := \forall x P(x) \vee \forall x \exists y Q(x,y).$$

Podemos aplicar el apartado (3.) del lema anterior con

$$\alpha := P(x) \qquad \text{y} \qquad \beta := \forall x \exists y Q(x,y).$$

Entonces

$$\delta \;\equiv\; \forall x \Big(P(x) \vee \forall x \exists y Q(x,y) \Big).$$

Nótese que los dos cuantificadores universales que aparecen en esta última fórmula "juegan papeles diferentes" dentro de dicha fórmula. Es posible que el primero se refiera a objetos de cierto conjunto, como por ejemplo personas, y el segundo a objetos de otro conjunto diferente, como por ejemplo automóviles. □

Ejemplo 22. En la fórmula

$$\forall x P(x) \vee Q(x)$$

no podemos aplicar el apartado (3.) del lema anterior, pues si hacemos las identificaciones

$$\alpha := P(x) \qquad \text{y} \qquad \beta := Q(x),$$

vemos que x aparece libre en β.

De hecho, se puede constatar que la fórmula inicial no es equivalente a la fórmula

$$\forall x \Big(P(x) \vee Q(x) \Big).$$

□

Lema 6. Sean $\alpha, \beta \in \text{Form}(\mathcal{L})$ y supongamos que el símbolo de variable y no aparece en β (ni libre ni ligada). Entonces:

1. $\forall x\beta \equiv \forall y\beta^*$,

2. $\exists x\beta \equiv \exists y\beta^*$,

donde β^* es la fórmula que se obtiene reemplazando todas las ocurrencias libres de x en β por la variable y. □

Esta propiedad nos dice que **podemos renombrar o cambiar los nombres de las variables cuantificadas** que aparecen en una fórmula, lo cual combinado con otras propiedades anteriores nos permite incluir todas las componentes de una fórmula dentro del radio de acción de un cuantificador. En particular **podemos extraer los cuantificadores de una fórmula de modo que todos ellos aparezcan en la parte de la izquierda**. Recordamos también que **las ocurrencias libres de variables son intocables, es decir, no podemos renombrarlas**.

Ejemplo 23. La fórmula $\forall xR(x, y)$ es equivalente a la fórmula $\forall zR(z, y)$.

Sin embargo, la fórmula $\forall xR(x, y)$ no es equivalente a $\forall yR(y, y)$.

En la fórmula $\forall xR(x, y)$ la única ocurrencia de y es libre, mientras que en la fórmula $\forall yR(y, y)$, dicha ocurrencia la hemos convertido en ligada, con lo cual su significado ahora viene dado por el cuantificador universal. En la relación de problemas se propone justificar que ambas fórmulas no son equivalentes. □

Ejemplo 24. Sea la fórmula

$$\alpha := \forall xR(x, y) \vee \exists yP(f(y)).$$

Como en la subfórmula $\exists yP(f(y))$ no aparece la variable x, entonces resulta

$$\alpha \;\equiv\; \forall x\Big(R(x, y) \vee \exists yP(f(y))\Big).$$

Si nos proponemos en primer lugar extraer a la izquierda el cuantificador existencial de modo que afecte a toda la fórmula, hemos de renombrar primero la variable cuantificada por dicho cuantificador. Por una parte aplicando el Lema 6 tenemos que

$$\exists yP(f(y)) \;\equiv\; \exists zP(f(z)),$$

y por otra aplicando el Lema 2, resulta

$$\alpha = \forall xR(x, y) \vee \exists yP(f(y)) \;\equiv\; \forall xR(x, y) \vee \exists zP(f(z)).$$

Ya sólo queda aplicar el Lema 5 para obtener el objetivo propuesto

$$\alpha \;\equiv\; \exists z\Big(\forall xR(x, y) \vee P(f(z))\Big).$$

Sobre la fórmula obtenida podemos aplicar de nuevo el Lema 5 y llegamos a

$$\alpha \;\equiv\; \exists z \forall x \Big(R(x,y) \,\vee\, P(f(z)) \Big).$$

Así hemos obtenido una fórmula equivalente a la inicial, en la cual todos los cuantificadores aparecen agrupados en la parte de la izquierda. □

En general en una fórmula no podemos alterar el orden en el que aparecen los cuantificadores, pues la fórmula resultante podría no ser equivalente a la fórmula inicial.

Ejemplo 25. Las fórmulas $\forall x \exists y R(x,y)$ y $\exists y \forall x R(x,y)$ no son equivalentes, pues si consideramos la estructura $\mathcal{E}$ dada por

$$\begin{aligned} &\mathrm{D} = \mathbb{R}, \\ &R^{\mathcal{E}}(x,y) := \text{“ } x \leq y \text{ ”}, \end{aligned}$$

entonces para cualquier asignación v en $\mathcal{E}$ tenemos que

$$\mathrm{I}^v \Big(\forall x \exists y R(x,y) \Big) = \mathbf{1},$$

pues podemos asignarle a y el valor que toma x, mientras que

$$\mathrm{I}^v \Big(\exists y \forall x R(x,y) \Big) = \mathbf{0},$$

ya que $\mathbb{R}$ no está acotado superiormente. □

La propiedad siguiente nos dice que podemos cambiar el orden de cuantificadores idénticos siempre que vayan consecutivos.

Lema 7. Si $\alpha \in \mathrm{Form}(\mathcal{L})$, entonces:

1. $\forall x \forall y \alpha \equiv \forall y \forall x \alpha$,

2. $\exists x \exists y \alpha \equiv \exists y \exists x \alpha$.

□

Por último vemos que a veces es posible eliminar **cuantificadores superfluos** sin que ello afecte a la interpretación de la fórmula.

Lema 8. Si $\alpha \in \mathrm{Form}(\mathcal{L})$, entonces:

1. $\forall x \forall x \alpha \equiv \forall x \alpha$,

2. $\exists x \forall x \alpha \equiv \forall x \alpha$,

3. $\forall x \exists x \alpha \equiv \exists x \alpha$,

4. $\exists x \exists x \alpha \equiv \exists x \alpha$.

$\square$

Con la notación del lema anterior, el cuantificador que afecta a α es el más próximo por la derecha a α.

Lema 9. Si $\alpha \in \mathrm{Form}(\mathcal{L})$ e y es un símbolo de variable que no aparece en α, entonces:

1. $\forall y \alpha \equiv \alpha$,
2. $\exists y \alpha \equiv \alpha$.

$\square$

Ejemplo 26. Tenemos las siguientes equivalencias:

$$\begin{array}{lcl} \exists x P(y) & \equiv & P(y), \\ \forall y \exists x \neg P(y) & \equiv & \forall y \neg P(y), \\ \forall z \forall y \exists x \neg P(y) & \equiv & \forall y \neg P(y), \\ \forall x \forall y R(x,y) & \equiv & \forall y \forall x R(x,y). \end{array}$$

$\square$