Temas de “Lógica matemática y fundamentos” (2019–20)

José A. Alonso Jiménez
María J. Hidalgo Doblado

Grupo de Lógica Computacional
Dpto. de Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla
Sevilla, 7 de febrero de 2020

Índice general

Parte I

Lógica proposicional

Tema 1

Sintaxis y semántica de la lógica proposicional

Contenido

1.1. Introducción

1.1.1. Panorama de la lógica

- Objetivos de la lógica:
  - La formalización del lenguaje natural.
  - Los métodos de razonamiento.
- Sistemas lógicos:
  - Lógica proposicional.
  - Lógica de primer orden.
  - Lógicas de orden superior.
  - Lógicas modales.
  - Lógicas descriptivas.
- Aplicaciones de la lógica en computación:
  - Programación lógica.
  - Verificación y síntesis automática de programas.
  - Representación del conocimiento y razonamiento.
  - Modelización y razonamiento sobre sistemas.
- Lógica informática = Representación del conocimiento +
Razonamiento

1.1.2. Ejemplos de argumentos y formalizaciones

- Ejemplos de argumentos:
  - Ejemplo 1: Si el tren llega a las 7 y no hay taxis en la estación, entonces Juan llegará tarde a la reunión. Juan no ha llegado tarde a la reunión. El tren llegó a las 7. *Por tanto,* habían taxis en la estación.
  - Ejemplo 2: Si hay corriente y la lámpara no está fundida, entonces está encendida. La lámpara no está encendida. Hay corriente. *Por tanto,* la lámpara está fundida.
- Formalización:

- Simbolización:

Simb.	Ejemplo 1	Ejemplo 2
p	el tren llega a las 7	hay corriente
q	hay taxis en la estación	la lámpara está fundida
r	Juan llega tarde a la reunión	la lámpara está encendida

- Si p y no q, entonces r. No r. p. Por tanto, q.
- $p \wedge \neg q \rightarrow r,\ \neg r,\ p \models q$.

1.2. Sintaxis de la lógica proposicional

1.2.1. El lenguaje de la lógica proposicional

El lenguaje de la lógica proposicional

- Alfabeto proposicional:
  - variables proposicionales: $p_0, p_1, \dots; p, q, r$.
  - conectivas lógicas:
    - monaria: $\neg$ (negación),
    - binarias: $\wedge$ (conjunción), $\vee$ (disyunción),
$\rightarrow$ (condicional), $\leftrightarrow$ (bicondicional).
  - símbolos auxiliares: "(" y ")".
- Fórmulas proposicionales:
  - Definición:
    - Las variables proposicionales son fórmulas (fórmulas atómicas).
    - Si F y G son fórmulas, entonces también lo son
$\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$ y $(F \leftrightarrow G)$
  - Ejemplos:
    - Fórmulas: p, $(p \vee \neg q)$, $\neg(p \vee p)$, $((p \rightarrow q) \vee (q \rightarrow p))$
    - No fórmulas: (p), $p \vee \neg q$, $(p \vee \wedge q)$

Fórmulas proposicionales (BNF)

- Notaciones:
  - $p, q, r, \dots$ representarán variables proposicionales.
  - $F, G, H, \dots$ representarán fórmulas.

- VP representa el conjunto de los variables proposicionales.
- Prop representa el conjunto de las fórmulas.
- $*$ representa una conectiva binaria.

- Forma de Backus Naur (BNF) de las fórmula proposicionales:
  - $F ::= p \mid \neg G \mid (F \wedge G) \mid (F \vee G) \mid (F \rightarrow G) \mid (F \leftrightarrow G)$.

1.2.2. Recursión e inducción sobre fórmulas

Definiciones por recursión sobre fórmulas

- Número de paréntesis de una fórmula:
  - Def: El número de paréntesis de una fórmula F se define recursivamente por:

$$\text{np}(F) = \begin{cases} 0, & \text{si } F \text{ es atómica;} \\ \text{np}(G), & \text{si } F \text{ es } \neg G; \\ 2 + \text{np}(G) + \text{np}(H), & \text{si } F \text{ es } (G * H) \end{cases}$$

  - Ejemplos:
    - $\text{np}(p) = 0$
    - $\text{np}(q) = 0$
    - $\text{np}(\neg q) = 0$
    - $\text{np}((\neg q \vee p)) = 2$
    - $\text{np}((p \rightarrow (\neg q \vee p))) = 4$

Demostración por inducción sobre fórmulas

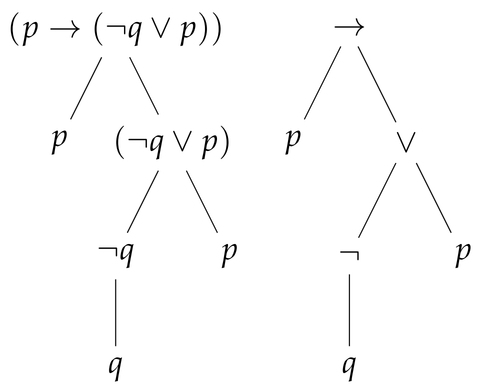
- Principio de inducción sobre fórmulas: Sea $\mathcal{P}$ una propiedad sobre las fórmulas que verifica las siguientes condiciones:
  - Todas las fórmulas atómicas tienen la propiedad $\mathcal{P}$.
  - Si F y G tienen la propiedad $\mathcal{P}$, entonces $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$ y $(F \leftrightarrow G)$, tienen la propiedad $\mathcal{P}$.

  Entonces todas las fórmulas proposicionales tienen la propiedad $\mathcal{P}$.

- Propiedad: Todas las fórmulas proposicionales tienen un número par de paréntesis.
  - Demostración por inducción sobre las fórmulas.

- Base: F atómica $\Longrightarrow \text{np}(F) = 0$ es par.
- Paso: Supongamos que $\text{np}(F)$ y $\text{np}(G)$ es par (hipótesis de inducción). Entonces,
  $\text{np}(\neg F) = \text{np}(F)$ es par y
  $\text{np}((F * G)) = 2 + \text{np}(F) + \text{np}(G)$ es par,
  para cualquier conectiva binaria $\star$.

1.2.3. Árboles de análisis (o de formación)



1.2.4. Eliminación de paréntesis

Criterios de reducción de paréntesis

- Pueden eliminarse los paréntesis externos.
  $F \land G$ es una abreviatura de $(F \land G)$.

- Precedencia de asociación de conectivas: $\neg, \land, \lor, \to, \leftrightarrow$.
  $F \land G \to \neg F \lor G$ es una abreviatura de $((F \land G) \to (\neg F \lor G))$.

- Cuando una conectiva se usa repetidamente, se asocia por la derecha.
  $F \lor G \lor H$ abrevia $(F \lor (G \lor H))$
  $F \land G \land H \to \neg F \lor G$ abrevia $((F \land (G \land H)) \to (\neg F \lor G))$

1.2.5. Subfórmulas

Subfórmulas

- Def: El conjunto $\text{Subf}(F)$ de las subfórmulas de una fórmula F se define recursivamente por:

$$\text{Subf}(F) = \begin{cases} \{F\}, & \text{si } F \text{ es atómica;} \\ \{F\} \cup \text{Subf}(G), & \text{si } F \text{ es } \neg G; \\ \{F\} \cup \text{Subf}(G) \cup \text{Subf}(H), & \text{si } F \text{ es } G * H \end{cases}$$

- Ejemplos:
  - $\text{Subf}(p) = \{p\}$
  - $\text{Subf}(q) = \{q\}$
  - $\text{Subf}(\neg q) = \{\neg q, q\}$
  - $\text{Subf}(\neg q \lor p) = \{\neg q \lor p, \neg q, q, p\}$
  - $\text{Subf}(p \to \neg q \lor p) = \{p \to \neg q \lor p, p, \neg q \lor p, \neg q, q\}$

1.3. Semántica proposicional

1.3.1. Valores y funciones de verdad

- Valores de verdad ($\mathbb{B}$): 1: verdadero y 0: falso.

- Funciones de verdad:
  - $H_\neg : \{0,1\} \to \{0,1\}$ t.q. $H_\neg(i) = \begin{cases} 1, & \text{si } i = 0; \\ 0, & \text{si } i = 1. \end{cases}$
  - $H_\land : \{0,1\}^2 \to \{0,1\}$ t.q. $H_\land(i,j) = \begin{cases} 1, & \text{si } i = j = 1; \\ 0, & \text{en otro caso.} \end{cases}$
  - $H_\lor : \{0,1\}^2 \to \{0,1\}$ t.q. $H_\lor(i,j) = \begin{cases} 0, & \text{si } i = j = 0; \\ 1, & \text{en otro caso.} \end{cases}$
  - $H_\to : \{0,1\}^2 \to \{0,1\}$ t.q. $H_\to(i,j) = \begin{cases} 0, & \text{si } i = 1, j = 0; \\ 1, & \text{en otro caso.} \end{cases}$
  - $H_\leftrightarrow : \{0,1\}^2 \to \{0,1\}$ t.q. $H_\leftrightarrow(i,j) = \begin{cases} 1, & \text{si } i = j; \\ 0, & \text{en otro caso.} \end{cases}$

1.3.2. Interpretaciones

- Funciones de verdad mediante tablas de verdad:

i	$\neg i$
1	0
0	1

i	j	$i \wedge j$	$i \vee j$	$i \rightarrow j$	$i \leftrightarrow j$
1	1	1	1	1	1
1	0	0	1	0	0
0	1	0	1	1	0
0	0	0	0	1	1

- Interpretación:
  - Def.: Una interpretación es una aplicación $I : \mathsf{VP} \rightarrow \mathbb{B}$.
  - Prop: Para cada interpretación I existe una única aplicación $I' : \mathsf{Prop} \rightarrow \mathbb{B}$ tal que:
  $$I'(F) = \begin{cases} I(F), & \text{si } F \text{ es atómica;} \\ H_\neg(I'(G)), & \text{si } F = \neg G; \\ H_*(I'(G), I'(H)), & \text{si } F = G * H \end{cases}$$
  Se dice que $I'(F)$ es el valor de verdad de F respecto de I.

- Ejemplo: Sea $F = (p \vee q) \wedge (\neg q \vee r)$
  - valor de F en una interpretación I_1 tal que $I_1(p) = I_1(r) = 1, I_1(q) = 0$
  $$\begin{array}{ccccccc} (p & \vee & q) & \wedge & (\neg q & \vee & r) \\ (1 & \vee & 0) & \wedge & (\neg 0 & \vee & 1) \\ & 1 & & \wedge & (1 & \vee & 1) \\ & 1 & & \wedge & & 1 & \\ & & & 1 & & & \end{array}$$
  - valor de F en una interpretación I_2 tal que $I_2(r) = 1, I_2(p) = I_2(q) = 0$
  $$\begin{array}{ccccccc} (p & \vee & q) & \wedge & (\neg q & \vee & r) \\ 0 & 0 & 0 & 0 & 10 & 1 & 1 \end{array}$$

- Prop.: Sea F una fórmula y I_1, I_2 dos interpretaciones. Si $I_1(p) = I_2(p)$ para todos las variables proposicionales de F, entonces $I_1'(F) = I_2'(F)$.

- Notación: Se escribe $I(F)$ en lugar de $I'(F)$.

1.3.3. Modelos, satisfacibilidad y validez

Modelos y satisfacibilidad

- Modelo de una fórmula
  - Def.: I es modelo de F si $I(F) = 1$.

- Notación: $I \models F$.
- Ejemplo (continuación del anterior):
  – si $I_1(p) = I_1(r) = 1, I_1(q) = 0$, entonces $I_1 \models (p \lor q) \land (\neg q \lor r)$
  – si $I_2(r) = 1, I_2(p) = I_2(q) = 0$, entonces $I_2 \not\models (p \lor q) \land (\neg q \lor r)$.

- Fórmulas satisfacibles e insatisfacibles
  - Def.: F es satisfacible si F tiene algún modelo.
  - Ejemplo: $(p \to q) \land (q \to r)$ es satisfacible
    $I(p) = I(q) = I(r) = 0$.
  - Def.: F es insatisfacible si F no tiene ningún modelo.
  - Ejemplo: $p \land \neg p$ es insatisfacible

p	$\neg p$	$p \land \neg p$
1	0	0
0	1	0

Tautologías y contradicciones

- Def.: F es una tautología (o válida) si toda interpretación es modelo de F. Se representa por $\models F$.
- Def.: F es una contradicción si ninguna interpretación es modelo de F.
- Def.: F es contingente si no es tautología ni contradicción.
- Ejemplos:
  1. $(p \to q) \lor (q \to p)$ es una tautología.
  2. $(p \to q) \land \neg(p \to q)$ es una contradicción.
  3. $p \to q$ es contingente.

p	q	$p \to q$	$q \to p$	$(p \to q) \lor (q \to p)$	$\neg(p \to q)$	$(p \to q) \land \neg(p \to q)$
1	1	1	1	1	0	0
1	0	0	1	1	1	0
0	1	1	0	1	0	0
0	0	1	1	1	0	0

Clasificaciones de fórmulas

Todas las fórmulas		
Tautologías	Contigentes	Contradicciones
Verdadera en todas las interpretaciones (ej. $p \lor \neg p$)	Verdadera en algunas interpretaciones y falsa en otras (ej. $p \rightarrow q$)	Falsa en todas las interpretaciones (ej. $p \land \neg p$)
Safisfacibles		Insatisfacibles
Todas las fórmulas		

Satisfacibilidad y validez

- Los problemas SAT y TAUT:
  - Problema SAT: Dada F determinar si es satisfacible.
  - Problema TAUT: Dada F determinar si es una tautología.
- Relaciones entre satisfacibilidad y tautologicidad:
  - F es tautología $\iff$ $\neg F$ es insatisfacible.
  - F es tautología $\implies$ F es satisfacible.
  - F es satisfacible $\not\Longrightarrow$ $\neg F$ es insatisfacible.

    $p \rightarrow q$ es satisfacible.

    $$I(p) = I(q) = 1$$

    $\neg(p \rightarrow q)$ es satisfacible.

    $$I(p) = 1, I(q) = 0.$$

1.3.4. Algoritmos para satisfacibilidad y validez

- Tabla de verdad para $\models (p \rightarrow q) \lor (q \rightarrow p)$:

p	q	$(p \rightarrow q)$	$(q \rightarrow p)$	$(p \rightarrow q) \lor (q \rightarrow p)$
1	1	1	1	1
1	0	0	1	1
0	1	1	0	1
0	0	1	1	1

- Tabla de verdad simplificada para $\models (p \to q) \lor (q \to p)$:

p	q	$(p$	$\to$	$q)$	$\lor$	$(q$	$\to$	$p)$
1	1	1	1	1	1	1	1	1
1	0	1	0	0	1	0	1	1
0	1	0	1	1	1	1	0	0
0	0	0	1	0	1	0	1	0

- Método de Quine para $\models (p \to q) \lor (q \to p)$

$(p$	$\to$	$q)$	$\lor$	$(q$	$\to$	$p)$
			0			
	0				0	
				1		0
0		1				
	1					

- Método de Quine para $\models (p \to q) \lor (q \to p)$

$(p$	$\to$	$q)$	$\lor$	$(q$	$\to$	$p)$
0	0	1	0	1	0	0
	$1*$					

- Tablas de verdad para $\not\models (p \leftrightarrow q) \lor (q \leftrightarrow p)$

p	q	$(p \leftrightarrow q)$	$(q \leftrightarrow p)$	$(p \leftrightarrow q) \lor (q \leftrightarrow p)$
1	1	1	1	1
1	0	0	0	0
0	1	0	0	0
0	0	1	1	1

- Método de Quine para $\not\models (p \leftrightarrow q) \lor (q \leftrightarrow p)$

$(p$	$\leftrightarrow$	$q)$	$\lor$	$(q$	$\leftrightarrow$	$p)$
0	0	1	0	1	0	0
1	0	0	0	0	0	1

1.3.5. Selección de tautologías

1.	$F \to F$	(ley de identidad).
2.	$F \lor \neg F$	(ley del tercio excluido).
3.	$\neg(F \land \neg F)$	(principio de no contradicción).
4.	$(\neg F \to F) \to F$	(ley de Clavius).
5.	$\neg F \to (F \to G)$	(ley de Duns Scoto).
6.	$((F \to G) \to F) \to F$	(ley de Peirce).
7.	$(F \to G) \land F \to G$	(modus ponens).
8.	$(F \to G) \land \neg G \to \neg F$	(modus tollens).

1.3.6. Equivalencia lógica

Fórmulas equivalentes

- Def.: F y G son equivalentes si $I(F) = I(G)$ para toda interpretación I. Representación: $F \equiv G$.
- Ejemplos de equivalencias notables:
  1. Idempotencia: $F \lor F \equiv F$; $F \land F \equiv F$.
  2. Conmutatividad: $F \lor G \equiv G \lor F$; $F \land G \equiv G \land F$.
  3. Asociatividad: $F \lor (G \lor H) \equiv (F \lor G) \lor H$;
     $F \land (G \land H) \equiv (F \land G) \land H$
  4. Absorción: $F \land (F \lor G) \equiv F$; $F \lor (F \land G) \equiv F$.
  5. Distributividad: $F \land (G \lor H) \equiv (F \land G) \lor (F \land H)$;
     $F \lor (G \land H) \equiv (F \lor G) \land (F \lor H)$.
  6. Doble negación: $\neg\neg F \equiv F$.
  7. Leyes de De Morgan: $\neg(F \land G) \equiv \neg F \lor \neg G$;
     $\neg(F \lor G) \equiv \neg F \land \neg G$
  8. Leyes de tautologías: Si F es una tautología, $F \land G \equiv G$; $F \lor G \equiv F$.
  9. Leyes de contradicciones: Si F es una contradicción $F \land G \equiv F$; $F \lor G \equiv G$.

Propiedades de la equivalencia lógica

- Relación entre equivalencia y bicondicional:
  - $F \equiv G$ syss $\models F \leftrightarrow G$.
- Propiedades básicas de la equivalencia lógica:

- Reflexiva: $F \equiv F$.
- Simétrica: Si $F \equiv G$, entonces $G \equiv F$.
- Transitiva: Si $F \equiv G$ y $G \equiv H$, entonces $F \equiv H$.

- Principio de sustitución de fórmulas equivalentes:
  - Prop.: Si en la fórmula F se sustituye una de sus subfórmulas G por una fórmula G' lógicamente equivalente a G, entonces la fórmula obtenida, F', es lógicamente equivalente a F.
  - Ejemplo:
$$\begin{aligned} F &= \neg(p \land q) \to \neg(p \land \neg\neg r) \\ G &= \neg(p \land q) \\ G' &= \neg p \lor \neg q \\ F' &= (\neg p \lor \neg q) \to \neg(p \land \neg\neg r) \end{aligned}$$

1.3.7. Modelos de conjuntos de fórmulas

- Notación:
  - $S, S_1, S_2, \dots$ representarán conjuntos de fórmulas.
- Modelo de un conjunto de fórmulas:
  - Def.: I es modelo de S si para toda $F \in S$ se tiene que $I \models F$.
  - Representación: $I \models S$.
  - Ejemplo: Sea $S = \{(p \lor q) \land (\neg q \lor r), q \to r\}$
    La interpretación I_1 tal que $I_1(p) = 1, I_1(q) = 0, I_1(r) = 1$ es modelo de S ($I_1 \models S$).
$$\begin{array}{cccccccccccc} \{(p & \lor & q) & \land & (\neg & q & \lor & r), & & q & \to & r\} \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & & 0 & 1 & 1 \end{array}$$
    La interpretación I_2 tal que $I_2(p) = 0, I_2(q) = 1, I_2(r) = 0$ no es modelo de S ($I_2 \not\models S$).
$$\begin{array}{cccccccccccc} \{(p & \lor & q) & \land & (\neg & q & \lor & r), & & q & \to & r\} \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & & 1 & 0 & 0 \end{array}$$

1.3.8. Consistencia y consecuencia lógica

Conjunto consistente de fórmulas

- Def.: S es consistente si S tiene algún modelo.
- Def.: S es inconsistente si S no tiene ningún modelo.

- Ejemplos:
  - $\{(p \lor q) \land (\neg q \lor r), p \to r\}$ es consistente (con modelos I_4, I_6, I_8)
  - $\{(p \lor q) \land (\neg q \lor r), p \to r, \neg r\}$ es inconsistente

	p	q	r	$(p \lor q)$	$(\neg q \lor r)$	$(p \lor q) \land (\neg q \lor r)$	$p \to r$	$\neg r$
I_1	0	0	0	0	1	0	1	1
I_2	0	0	1	0	1	0	1	0
I_3	0	1	0	1	0	0	1	1
I_4	0	1	1	1	1	1	1	0
I_5	1	0	0	1	1	1	0	1
I_6	1	0	1	1	1	1	1	0
I_7	1	1	0	1	0	0	0	1
I_8	1	1	1	1	1	1	1	0

Consecuencia lógica

- Def.: F es consecuencia de S si todos los modelos de S son modelos de F.
- Representación: $S \models F$.
- Ejemplos: $\{p \to q, q \to r\} \models p \to r$ y $\{p\} \not\models p \land q$

	p	q	r	$p \to q$	$q \to r$	$p \to r$
I_1	0	0	0	1	1	1
I_2	0	0	1	1	1	1
I_3	0	1	0	1	0	1
I_4	0	1	1	1	1	1
I_5	1	0	0	0	1	0
I_6	1	0	1	0	1	1
I_7	1	1	0	1	0	0
I_8	1	1	1	1	1	1

p	q	$p \land q$
1	1	1
1	0	0
0	1	0
0	0	0

Propiedades de la consecuencia

- Propiedades básicas de la relación de consecuencia:
  - Reflexividad: $S \models S$.
  - Monotonía: Si $S_1 \models F$ y $S_1 \subseteq S_2$, entonces $S_2 \models F$.
  - Transitividad: Si $S \models F$ y $\{F\} \models G$, entonces $S \models G$.
- Relación entre consecuencia, validez, satisfacibilidad y consistencia:

- Las siguientes condiciones son equivalentes:
  1. $\{F_1, \dots, F_n\} \models G$
  2. $\models F_1 \wedge \dots \wedge F_n \to G$
  3. $\neg(F_1 \wedge \dots \wedge F_n \to G)$ es insatisfacible
  4. $\{F_1, \dots, F_n, \neg G\}$ es inconsistente

1.3.9. Argumentaciones y problemas lógicos

Ejemplo de argumentación

- Problema de los animales: Se sabe que
  1. Los animales con pelo o que dan leche son mamíferos.
  2. Los mamíferos que tienen pezuñas o que rumian son ungulados.
  3. Los ungulados de cuello largo son jirafas.
  4. Los ungulados con rayas negras son cebras.

  Se observa un animal que tiene pelos, pezuñas y rayas negras. Por consiguiente, se concluye que el animal es una cebra.

- Formalización:

$$\begin{array}{l} \{ \quad \text{tiene\_pelos} \vee \text{da\_leche} \to \text{es\_mamífero}, \\ \quad\;\; \text{es\_mamífero} \wedge (\text{tiene\_pezuñas} \vee \text{rumia}) \to \text{es\_ungulado}, \\ \quad\;\; \text{es\_ungulado} \wedge \text{tiene\_cuello\_largo} \to \text{es\_jirafa}, \\ \quad\;\; \text{es\_ungulado} \wedge \text{tiene\_rayas\_negras} \to \text{es\_cebra}, \\ \quad\;\; \text{tiene\_pelos} \wedge \text{tiene\_pezuñas} \wedge \text{tiene\_rayas\_negras}\} \\ \models \text{es\_cebra} \end{array}$$

Problemas lógicos: veraces y mentirosos

- Enunciado: En una isla hay dos tribus, la de los veraces (que siempre dicen la verdad) y la de los mentirosos (que siempre mienten). Un viajero se encuentra con tres isleños A, B y C y cada uno le dice una frase
  1. A dice "B y C son veraces syss C es veraz"
  2. B dice "Si A y C son veraces, entonces B y C son veraces y A es mentiroso"
  3. C dice "B es mentiroso syss A o B es veraz"

  Determinar a qué tribu pertenecen A, B y C.

- Simbolización: a: "A es veraz", b: "B es veraz", c: "C es veraz".

- Formalización:
  $F_1 = a \leftrightarrow (b \land c \leftrightarrow c)$, $F_2 = b \leftrightarrow (a \land c \to b \land c \land \neg a)$ y $F_3 = c \leftrightarrow (\neg b \leftrightarrow a \lor b)$.
- Modelos de $\{F_1, F_2, F_3\}$:
  Si I es modelo de $\{F_1, F_2, F_3\}$, entonces $I(a) = 1, I(b) = 1, I(c) = 0$.
- Conclusión: A y B son veraces y C es mentiroso.

Bibliografía

1. C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal.* (Ariel, 2000)

   Cap. 0 (Introducción), 6 (Sintaxis de la lógica proposicional), 7 (Semántica de la lógica proposicional), 9 (Consecuencia lógica) y 11 (Lógica proposicional y lenguaje natural).

2. M. Ben–Ari, *Mathematical logic for computer science (2nd ed.).* (Springer, 2001)

   Cap. 1 (Introduction) y 2 (Propositional calculus: formulas, models, tableaux).

3. J.A. Díez *Iniciación a la Lógica*, (Ariel, 2002)

   Cap. 2 (El lenguaje de la lógica proposicional) y 3 (Semántica formal. Consecuencia lógica).

4. M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems.* (Cambridge University Press, 2000)

   Cap. 1 (Propositional logic).

Tema 2

Deducción natural proposicional

Contenido

2.1. Reglas de deducción natural

2.1.1. Reglas de la conjunción

- Regla de introducción de la conjunción: $\dfrac{F \quad G}{F \land G} \land i$

- Reglas de eliminación de la conjunción: $\dfrac{F \land G}{F} \land e_1 \quad \dfrac{F \land G}{G} \land e_2$

- Ejemplo: $p \land q, r \vdash q \land r$:

1	$p \land q$	premisa
2	r	premisa
3	q	$\land e_2$ 1
4	$q \land r$	$\land i$ 2, 3

- Adecuación de las reglas de la conjunción:
  - $\land i : \{F, G\} \models F \land G$
  - $\land e_1 : F \land G \models F$
  - $\land e_2 : F \land G \models G$

2.1.2. Reglas de la doble negación

- Regla de eliminación de la doble negación: $\dfrac{\neg\neg F}{F} \neg\neg e$

- Regla de introducción de la doble negación: $\dfrac{F}{\neg\neg F} \neg\neg i$

- Ejemplo: $p, \neg\neg(q \land r) \vdash \neg\neg p \land r$:

1	p	premisa
2	$\neg\neg(q \land r)$	premisa
3	$\neg\neg p$	$\neg\neg i$ 1
4	$q \land r$	$\neg\neg e$ 2
5	r	$\land e_2$ 4
6	$\neg\neg p \land r$	$\land i$ 3, 5

- Adecuación de las reglas de la doble negación:
  - $\neg\neg e : \{\neg\neg F\} \models F$
  - $\neg\neg i : \{F\} \models \neg\neg F$

2.1.3. Regla de eliminación del condicional

- Regla de eliminación del condicional: $\dfrac{F \quad F \to G}{G} \to e$

- Ejemplo: $\neg p \wedge q, \neg p \wedge q \to r \vee \neg p \vdash r \vee \neg p$:

1	$\neg p \wedge q$	premisa
2	$\neg p \wedge q \to r \vee \neg p$	premisa
3	$r \vee \neg p$	$\to$e $1,2$

- Ejemplo: $p, p \to q, p \to (q \to r) \vdash r$:

1	p	premisa
2	$p \to q$	premisa
3	$p \to (q \to r)$	premisa
4	q	$\to$e $1,2$
5	$q \to r$	$\to$e $1,3$
6	r	$\to$e $4,5$

- Adecuación de la eliminación del condicional: $\{F, F \to G\} \models G$

2.1.4. Regla derivada de modus tollens (MT)

- Regla derivada de modus tollens: $\dfrac{F \to G \quad \neg G}{\neg F}\ MT$

- Ejemplo: $p \to (q \to r), p, \neg r \vdash \neg q$:

1	$p \to (q \to r)$	premisa
2	p	premisa
3	$\neg r$	premisa
4	$q \to r$	$\to$e $1,2$
5	$\neg q$	MT $3,4$

- Ejemplo: $\neg p \to q, \neg q \vdash p$:

1	$\neg p \to q$	premisa
2	$\neg q$	premisa
3	$\neg\neg p$	MT $1,2$
4	p	$\neg\neg$e 3

2.1.5. Regla de introducción del condicional

- Regla de introducción del condicional: $$\frac{\boxed{\begin{array}{c} F \\ \vdots \\ G \end{array}}}{F \to G} \to i$$

- Ejemplo: $p \to q \vdash \neg q \to \neg p$:

1	$p \to q$	premisa
2	$\neg q$	supuesto
3	$\neg p$	MT $1, 2$
4	$\neg q \to \neg p$	$\to$i $2 - 3$

- Adecuación de la regla de introducción del condicional: Si $F \models G$, entonces $\models F \to G$.

- Ejemplo: $\neg q \to \neg p \vdash p \to \neg\neg q$:

1	$\neg q \to \neg p$	premisa
2	p	supuesto
3	$\neg\neg p$	$\neg\neg$i 2
4	$\neg\neg q$	MT $1, 3$
5	$p \to \neg\neg q$	$\to$i $2 - 4$

- Ejemplo (de teorema): $\vdash p \to p$:

1	p	supuesto
2	$p \to p$	$\to$i $1 - 1$

- Ejemplo: $\vdash (q \to r) \to ((\neg q \to \neg p) \to (p \to r))$:

1	$q \to r$	supuesto
2	$\neg q \to \neg p$	supuesto
3	p	supuesto
4	$\neg\neg p$	$\neg\neg$i 3
5	$\neg\neg q$	MT 2, 4
6	q	$\neg\neg$e 5
7	r	$\to$e 1, 6
8	$p \to r$	$\to$i 3 − 7
9	$(\neg q \to \neg p) \to (p \to r)$	$\to$i 2 − 8
10	$(q \to r) \to ((\neg q \to \neg p) \to (p \to r))$	$\to$i 1 − 9

2.1.6. Reglas de la disyunción

- Reglas de introducción de la disyunción: $\dfrac{F}{F \lor G}\,\lor i_1 \qquad \dfrac{G}{F \lor G}\,\lor i_2$

- Regla de eliminación de la disyunción: $$\dfrac{F \lor G \quad \boxed{\begin{matrix} F \\ \vdots \\ H \end{matrix}} \quad \boxed{\begin{matrix} G \\ \vdots \\ H \end{matrix}}}{H}\,\lor e$$

- Ejemplo: $p \lor q \vdash q \lor p$:

1	$p \lor q$	premisa
2	p	supuesto
3	$q \lor p$	$\lor i_2$ 2
4	q	supuesto
5	$q \lor p$	$\lor i_1$ 4
6	$q \lor p$	$\lor e$ 1, 2 − 3, 4 − 5

- Ejemplo: $q \to r \vdash p \lor q \to p \lor r$:

1	$q \to r$	premisa
2	$p \lor q$	supuesto
3	p	supuesto
4	$p \lor r$	$\lor i_1$ 3
5	q	supuesto
6	r	$\to e$ 1, 5
7	$p \lor r$	$\lor i_2$ 6
8	$p \lor r$	$\lor e$ 2, 3 − 4, 5 − 7
9	$p \lor q \to p \lor r$	$\to i$ 2 − 8

2.1.7. Regla de copia

- Ejemplo (usando la regla hyp): $\vdash p \to (q \to p)$:

1	p	supuesto
2	q	supuesto
3	p	hyp 1
4	$q \to p$	$\to i$ 2 − 3
5	$p \to (q \to p)$	$\to i$ 1 − 4

2.1.8. Reglas de la negación

- Extensiones de la lógica para usar falso:
  - Extensión de la sintaxis: $\bot$ es una fórmula proposicional.
  - Extensión de la semántica: $I(\bot) = 0$ en cualquier interpretación I.
- Reglas de la negación:
  - Regla de eliminación de lo falso: $\dfrac{\bot}{F}\ \bot e$
  - Regla de eliminación de la negación: $\dfrac{F \quad \neg F}{\bot}\ \neg e$
- Adecuación de las reglas de la negación:
  - $\bot \models F$

- $\{F, \neg F\} \models \bot$

- Ejemplo: $\neg p \lor q \vdash p \to q$:

1	$\neg p \lor q$	premisa
2	p	supuesto
3	$\neg p$	supuesto
4	$\bot$	$\neg$e $2, 3$
5	q	$\bot$e 4
6	q	supuesto
7	q	$\lor$e $1, 3-5, 6-6$
8	$p \to q$	$\to$i $2-7$

- Regla de introducción de la negación: $$\frac{\boxed{\begin{array}{c} F \\ \vdots \\ \bot \end{array}}}{\neg F} \neg \text{i}$$

- Adecuación: Si $F \models \bot$, entonces $\models \neg F$.

- Ejemplo: $p \to q, p \to \neg q \vdash \neg p$:

1	$p \to q$	premisa
2	$p \to \neg q$	premisa
3	p	supuesto
4	q	$\to$e $1, 3$
5	$\neg q$	$\to$e $2, 3$
6	$\bot$	$\neg$e $4, 5$
7	$\neg p$	$\neg$i $3-6$

2.1.9. Reglas del bicondicional

- Regla de introducción del bicondicional: $$\frac{F \to G \quad G \to F}{F \leftrightarrow G} \leftrightarrow \text{i}$$

- Ejemplo: $p \land q \leftrightarrow q \land p$:

1	$p \land q$	supuesto
2	p	$\land e_1$ 1
3	q	$\land e_2$ 1
4	$q \land p$	$\land i$ 2,3
5	$p \land q \to q \land p$	$\to i$ 1 − 4
6	$q \land p$	supuesto
7	q	$\land e_2$ 6
8	p	$\land e_1$ 6
9	$p \land q$	$\land i$ 7,8
10	$q \land p \to p \land q$	$\to i$ 6 − 9
11	$p \land q \leftrightarrow q \land p$	$\leftrightarrow i$ 5,10

- Eliminación del bicondicional: $\dfrac{F \leftrightarrow G}{F \to G} \leftrightarrow e_1$ $\quad \dfrac{F \leftrightarrow G}{G \to F} \leftrightarrow e_2$
- Ejemplo: $p \leftrightarrow q, p \lor q \vdash p \land q$:

1	$p \leftrightarrow q$			premisa
2	$p \lor q$			premisa
3	p	supuesto	q	supuesto
4	$p \to q$	$\leftrightarrow e_1$ 1	$q \to p$	$\leftrightarrow e_2$ 1
5	q	$\to e$ 4,3	p	$\to e$ $4', 3'$
6	$p \land q$	$\land i$ 3,5	$p \land q$	$\land i$ $3', 5'$
7	$p \land q$			$\lor e$ $2, 3-6, 3'-6'$

2.2. Reglas derivadas

2.2.1. Regla del modus tollens

- Regla derivada de modus tollens (MT): $\dfrac{F \to G \quad \neg G}{\neg F} MT$
- Derivación:

1	$F \to G$	premisa
2	$\neg G$	premisa
3	F	supuesto
4	G	$\to$e $1, 3$
5	$\bot$	$\neg$e $2, 4$
6	$\neg F$	$\neg$i $2 - 4$

2.2.2. Regla de introducción de doble negación

- Regla de introducción de la doble negación: $$\frac{F}{\neg\neg F}\ \neg\neg\text{i}$$
- Derivación:

1	F	premisa
2	$\neg F$	supuesto
3	$\bot$	$\neg$e $1, 2$
4	$\neg\neg F$	$\neg$i $2 - 3$

2.2.3. Regla de reducción al absurdo

- Regla de reducción al absurdo: $$\frac{\boxed{\begin{array}{c}\neg F\\ \vdots\\ \bot\end{array}}}{F}\ RAA$$
- Derivación:

1	$\neg F \to \bot$	premisa
2	$\neg F$	supuesto
3	$\bot$	$\to$e $1, 2$
4	$\neg\neg F$	$\neg$i $2 - 3$
5	F	$\neg$e $\neg 4$

2.2.4. Ley del tercio excluido

- Ley del tercio excluido (LEM): $$\frac{}{F \vee \neg F}\ LEM$$

- Derivación:

1	$\neg(F \vee \neg F)$	supuesto
2	F	supuesto
3	$F \vee \neg F$	$\vee$i$_1$ 2
4	$\perp$	$\neg$e 1,3
5	$\neg F$	$\neg$i 2 − 4
6	$F \vee \neg F$	$\vee$i$_2$ 5
7	$\perp$	$\neg$e 1,6
8	$F \vee \neg F$	RAA 1 − 7

- Ejemplo: $p \rightarrow q \vdash \neg p \vee q$:

1	$p \rightarrow q$	premisa
2	$p \vee \neg p$	LEM
3	p	supuesto
4	q	$\rightarrow$e 1,3
5	$\neg p \vee q$	$\vee$i$_2$ 4
6	$\neg p$	supuesto
7	$\neg p \vee q$	$\vee$i$_1$ 6
8	$\neg p \vee q$	$\vee$e 2,3 − 5,6 − 7

2.3. Resumen de reglas de deducción natural

	Introducción	Eliminación
$\land$	$\dfrac{F \quad G}{F \land G}\land i$	$\dfrac{F \land G}{F}\land e_1 \qquad \dfrac{F \land G}{G}\land e_2$
$\lor$	$\dfrac{F}{F \lor G}\lor i_1 \qquad \dfrac{G}{F \lor G}\lor i_2$	$\dfrac{F \lor G \quad \boxed{\begin{array}{c} F \\ \vdots \\ H \end{array}} \quad \boxed{\begin{array}{c} G \\ \vdots \\ H \end{array}}}{H}\lor e$
$\rightarrow$	$\dfrac{\boxed{\begin{array}{c} F \\ \vdots \\ G \end{array}}}{F \rightarrow G}\rightarrow i$	$\dfrac{F \quad F \rightarrow G}{G}\rightarrow e$
$\neg$	$\dfrac{\boxed{\begin{array}{c} F \\ \vdots \\ \bot \end{array}}}{\neg F}\neg i$	$\dfrac{F \quad \neg F}{\bot}\neg e$
$\bot$		$\dfrac{\bot}{F}\bot e$
$\neg\neg$		$\dfrac{\neg\neg F}{F}\neg\neg e$
$\leftrightarrow$	$\dfrac{F \rightarrow G \quad G \rightarrow F}{F \leftrightarrow G}\leftrightarrow i$	$\dfrac{F \leftrightarrow G}{F \rightarrow G}\leftrightarrow e_1 \qquad \dfrac{F \leftrightarrow G}{G \rightarrow F}\leftrightarrow e_2$

- Adecuación y completitud del cálculo de deducción natural.

Bibliografía

1. C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal.* (Ariel, 2000).

   Cap. 16: Cálculo deductivo.

2. R. Bornat *Using ItL Jape with X* (Department of Computer Science, QMW, 1998).
3. J.A. Díez *Iniciación a la Lógica*, (Ariel, 2002).

Cap. 4: Cálculo deductivo. Deducibilidad.

4. M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems.* (Cambridge University Press, 2000)

   Cap. 1: Propositional logic.

5. E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)

   Cap. 3.6: El método de la deducción natural.

Tema 3

Sintaxis y semántica de la lógica de primer orden

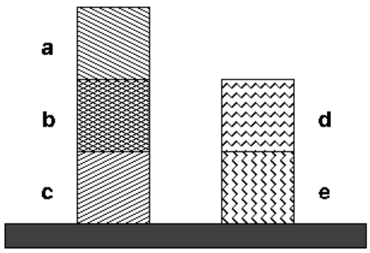
Contenido

3.1. Representación del conocimiento en lógica de primer orden

3.1.1. Representación de conocimiento geográfico

- Ejemplo 1: *Si Sevilla es vecina de Cádiz, entonces Cádiz es vecina de Sevilla. Sevilla es vecina de Cádiz. Por tanto, Cádiz es vecina de Sevilla*
  - Representación en lógica proposicional:
    $\{SvC \rightarrow CvS, \quad SvC\} \models CvS$
- Ejemplo 2: *Si una ciudad es vecina de otra, entonces la segunda es vecina de la primera. Sevilla es vecina de Cádiz. Por tanto, Cádiz es vecina de Sevilla*
  - Representación en lógica proposicional: Imposible
  - Representación en lógica de primer orden:
    $\{\forall x \ \forall y \ [vecina(x, y) \rightarrow vecina(y, x)], \quad vecina(Sevilla, Cadiz)\}$
    $\models vecina(Cadiz, Sevilla)$

3.1.2. Representación del mundo de los bloques



- Simbolización:
  - sobre(x, y) se verifica si el bloque x está colocado sobre el bloque y
  - sobre_mesa(x) se verifica si el bloque x está sobre la mesa
- Situación del ejemplo:
  sobre(a, b), sobre(b, c), sobre_mesa(c), sobre(d, e), sobre_mesa(e)
- Definiciones:
  - bajo(x, y) se verifica si el bloque x está debajo del bloque y
    $\forall x \ \forall y$ [bajo$(x, y) \leftrightarrow$ sobre(y, x)]

- encima(x, y) se verifica si el bloque x está encima del bloque y pudiendo haber otros bloques entre ellos
  $\forall x\ \forall y\ [\ \text{encima}(x, y) \leftrightarrow$
  $\text{sobre}(x, y) \vee \exists z\ [\text{sobre}(x, z) \wedge \text{encima}(z, y)]]$
- libre(x) se verifica si el bloque x no tiene bloques encima
  $\forall x\ [\text{libre}(x) \leftrightarrow \neg\exists y\ \ \text{sobre}(y, x)]$
- pila(x, y, z) se verifica si el bloque x está sobre el y, el y sobre el z y el z sobre la mesa
  $\forall x\ \forall y\ \forall z\ [\ \text{pila}(x, y, z) \leftrightarrow$
  $\text{sobre}(x, y) \wedge \text{sobre}(y, z) \wedge \text{sobre\_mesa}(z)]$

- Propiedades:
  - Si z, y, z es una pila entonces y no está libre
    $\forall x\ \forall y\ \forall z\ [\text{pila}(x, y, z) \rightarrow \neg\, \text{libre}(y)]$

Representación del mundo de los bloques con funciones e igualdad

- Simbolización:
  - es_bloque(x) se verifica si x es un bloque.
  - superior(x) es el bloque que está sobre el bloque x.
- Situación del ejemplo:
  - es_bloque(a), es_bloque(b), es_bloque(c), es_bloque(d), es_bloque(e)
  - $\text{superior}(b) = a, \text{superior}(c) = b, \text{superior}(e) = d$
- Definiciones:
  - sobre_mesa(x) se verifica si el bloque x está sobre la mesa
    $\forall x\ [\text{sobre\_mesa}(x) \leftrightarrow \text{es\_bloque}(x) \wedge \neg\exists y\ \ \text{superior}(y) = x]$
  - libre(x) se verifica si el bloque x no tiene bloques encima
    $\forall x\ [\text{libre}(x) \leftrightarrow \neg\exists y\ \ \text{superior}(x) = y]$
  - tope(x) es el bloque libre que está encima de x
    $\forall x\ [\ (\text{libre}(x) \rightarrow \text{tope}(x) = x) \wedge$
    $(\neg\, \text{libre}(x) \rightarrow \text{tope}(x) = \text{tope}(\text{superior}(x)))]$

3.1.3. Representación de conocimiento astronómico

- *La Tierra es un planeta:*
  $\text{planeta}(\text{Tierra})$
- *La Luna no es un planeta:*
  $\neg\,\text{planeta}(\text{Luna})$
- *La Luna es un satélite:*
  $\text{satélite}(\text{Luna})$
- *La Tierra gira alrededor del Sol:*
  $\text{gira}(\text{Tierra}, \text{Sol})$
- *Todo planeta es un satélite:*
  $\forall x\ [\text{planeta}(x) \to \text{satélite}(x)]$
- *Todo planeta gira alrededor del Sol:*
  $\forall x\ [\text{planeta}(x) \to \text{gira}(x, \text{Sol})]$
- *Algún planeta gira alrededor de la Luna:*
  $\exists x\ [\text{planeta}(x) \land \text{gira}(x, \text{Luna})]$
- *Hay por lo menos un satélite:*
  $\exists x\ \text{satélite}(x)$
- *Ningún planeta es un satélite:*
  $\neg\exists x\ [\text{planeta}(x) \land \text{satélite}(x)]$
- *Ningún objeto celeste gira alrededor de sí mismo:*
  $\neg\exists x\ \ \text{gira}(x, x)$
- *Alrededor de los satélites no giran objetos:*
  $\forall x\ [\text{satélite}(x) \to \neg\exists y\ \ \text{gira}(y, x)]$
- *Hay exactamente un satélite:*
  $\exists x\ [\text{satélite}(x) \land \forall y\ [\text{satélite}(y) \to x = y]]$
- *La Luna es un satélite de la Tierra:*
  $\text{satélite}(\text{Luna}, \text{Tierra})$
- *Todo planeta tiene un satélite:*
  $\forall x\ [\text{planeta}(x) \to \exists y\ \text{satélite}(y, x)]$
- *La Tierra no tiene satélites:*
  $\neg\exists x\ \text{satélite}(x, \text{Tierra})$

- *Algún planeta no tiene satélites:*
  $\exists x\ [\text{planeta}(x) \wedge \neg\exists y\ \text{satélite}(y,x)]$
- *Sólo los planetas tienen satélites:*
  $\forall x\ [\exists y\ \text{satélite}(y,x) \to \text{planeta}(x)]$
- *Todo satélite es satélite de algún planeta:*
  $\forall x\ [\text{satélite}(x) \to \exists y\ (\text{planeta}(y) \wedge \text{satélite}(x,y))]$
- *La Luna no gira alrededor de dos planetas diferentes:*
  $\neg\exists x\ \exists y\ [\ \text{planeta}(x) \wedge \text{planeta}(y) \wedge$
  $\quad \text{gira}(\text{Luna},x) \wedge \text{gira}(\text{Luna},y) \wedge x \neq y]$
- *Hay exactamente dos planetas:*
  $\exists x\ \exists y\ [\ \text{planeta}(x) \wedge \text{planeta}(y) \wedge x \neq y \wedge$
  $\quad \forall z\ [\text{planeta}(z) \to z = x \vee z = y]]$

3.2. Sintaxis de la lógica de primer orden

3.2.1. Lenguaje de primer orden

Lenguaje de primer orden

- Símbolos lógicos:
  - Variables: $x, y, z, \dots, x_1, x_2, \dots$.
  - Conectivas: $\neg, \wedge, \vee, \to, \leftrightarrow$.
  - Cuantificadores: $\forall, \exists$.
  - Símbolo de igualdad: $=$.
- Símbolos propios:
  - Símbolos de constantes: $a, b, c, \dots, a_1, a_2, \dots$.
  - Símbolos de predicado (con aridad): $P, Q, R, \dots, P_1, P_2, \dots$.
  - Símbolos de función (con aridad): $f, g, h, \dots, f_1, f_2, \dots$.
- Símbolos auxiliares: "(", ")", ",".
- Notación:
  - $L, L_1, L_2, \dots$ representan lenguajes de primer orden.
  - Var representa el conjunto de las variables.
- Los símbolos de predicados de aridad mayor que 1 se llaman de relaciones.

Ejemplos de lenguajes de primer orden

- Lenguaje del mundo de los bloques:
  - Símbolos de constantes: a, b, c, d, e
  - Símbolos de predicado (y de relación):
    – de aridad 1: sobre_mesa, libre, es_bloque
    – de aridad 2: sobre, bajo, encima
    – de aridad 3: pila
  - Símbolos de función (de aridad 1): superior, tope
- Lenguaje de la aritmética:
  - Símbolos de constantes: $0, 1$
  - Símbolos de función:
    – monaria: s (siguiente)
    – binarias: $+, \cdot$
  - Símbolo de predicado binario: $<$

3.2.2. Términos y fórmulas de primer orden

Términos

- Def. de término de un lenguaje de primer orden L:
  - Las variables son términos de L.
  - Las constantes de L son términos de L.
  - Si f es un símbolo de función n–aria de L y $t_1, \dots, t_n$ son términos de L, entonces $f(t_1, \dots, t_n)$ es un término de L.
- Ejemplos:
  - En el lenguaje de la aritmética,
    - $+(\cdot(x, 1), s(y))$ es un término, que se suele escribir como $(x \cdot 1) + s(y)$
    - $+(\cdot(x, <), s(y))$ no es un término
  - En el lenguaje del mundo de los bloques,
    - superior(superior(c)) es un término.
    - libre(superior(c)) no es un término.
- Notación:
  - $s, t, t_1, t_2, \dots$ representan términos.
  - Térm(L) representa el conjunto de los términos de L

Fórmulas atómicas

- Def. de fórmula atómica de un lenguaje de primer orden L:
  - Si t_1 y t_2 son términos de L, entonces $t_1 = t_2$ es una fórmula atómica de L.
  - Si P es un símbolo de relación n–aria de L y $t_1, \dots, t_n$ son términos de L, entonces $P(t_1, \dots, t_n)$ es una fórmula atómica de L.
- Ejemplos:
  - En el lenguaje de la aritmética,
    - $< (\cdot(x, 1), s(y))$ es una fórmula atómica que se suele escribir como $x \cdot 1 < s(y)$
    - $+(x, y) = \cdot(x, y)$ es una fórmula atómica que se suele escribir como $x + y = x \cdot y$
  - En el lenguaje del mundo de los bloques,
    - $\text{libre}(\text{superior}(c))$ es una fórmula atómica.
    - $\text{tope}(c) = \text{superior}(b)$ es una fórmula atómica.
- Notación:
  - $A, B, A_1, A_2, \dots$ representan fórmulas atómicas.
  - $\text{Atóm}(L)$ representa el conjunto de las fórmulas atómicas de L.

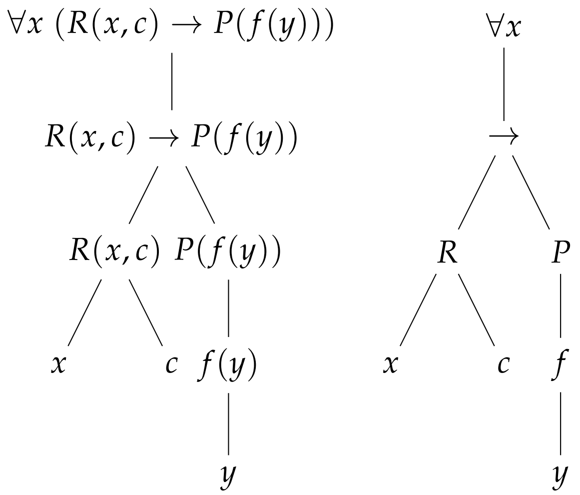
Fórmulas

- Definición de las fórmulas de L:
  - Las fórmulas atómicas de L son fórmulas de L.
  - Si F y G son fórmulas de L, entonces $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \to G)$ y $(F \leftrightarrow G)$ son fórmulas de L.
  - Si F es una fórmula de L, entonces $\forall x\ F$ y $\exists x\ F$ son fórmulas de L.
- Ejemplos:
  - En el lenguaje de la aritmética,
    - $\forall x\ \exists y\ < (x, y)$ es una fórmula que se escribe como $\forall x\ \exists y\ x < y$
    - $\forall x\ \exists y\ + (x, y)$ no es una fórmula.
  - En el lenguaje del mundo de los bloques,
    - $\forall x\ (\text{tope}(x) = x \leftrightarrow \text{libre}(x))$ es una fórmula.

- Notación:
  - $F, G, H, F_1, F_2, \ldots$ representan fórmulas.
  - $\text{Fórm}(L)$ representa el conjunto de las fórmulas de L.

3.2.3. Subfórmulas

Árboles de análisis (o de formación)



Subfórmulas

- Def: El conjunto $\text{Subf}(F)$ de las subfórmulas de una fórmula F se define recursivamente por:

$$\text{Subf}(F) = \begin{cases} \{F\}, & \text{si } F \text{ es una fórmula atómica;} \\ \{F\} \cup \text{Subf}(G), & \text{si } F = \neg G; \\ \{F\} \cup \text{Subf}(G) \cup \text{Subf}(H), & \text{si } F = G * H; \\ \{F\} \cup \text{Subf}(G), & \text{si } F = \forall x\ G; \\ \{F\} \cup \text{Subf}(G), & \text{si } F = \exists x\ G \end{cases}$$

- Ejemplo:

$$\begin{aligned} \text{Subf}(\forall x\ (R(x,c) \to P(f(y)))) = \{\ & \forall x\ (R(x,c) \to P(f(y))), \\ & (R(x,c) \to P(f(y))), \\ & R(x,c), \\ & P(f(y))\} \end{aligned}$$

Criterios de reducción de paréntesis

- Pueden eliminarse los paréntesis externos.
  $F \wedge G$ es una abreviatura de $(F \wedge G)$

- Precedencia de asociación de conectivas y cuantificadores: $\forall, \exists, \neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
  $\forall x\ P(x) \rightarrow Q(x)$ es una abreviatura de $(\forall x\ P(x)) \rightarrow Q(x)$

- Cuando una conectiva se usa repetidamente, se asocia por la derecha.
  $F \vee G \vee H$ es una abreviatura de $(F \vee (G \vee H))$
  $F \wedge G \wedge H \rightarrow \neg F \vee G$ es una abreviatura de $((F \wedge (G \wedge H)) \rightarrow (\neg F \vee G))$

- Los símbolos binarios pueden escribirse en notación infija.
  $x + y$ es una abreviatura de $+(x, y)$
  $x < y$ es una abreviatura de $< (x, y)$

3.2.4. Variables libres y ligadas

Conjuntos de variables

- Def.: El conjunto de las variables del término t es

$$V(t) = \begin{cases} \emptyset, & \text{si } t \text{ es una constante;} \\ \{x\}, & \text{si } t \text{ es una variable } x; \\ V(t_1) \cup \cdots \cup V(t_n), & \text{si } t \text{ es } f(t_1, \ldots, t_n) \end{cases}$$

- Def.: El conjunto de las variables de la fórmula F es

$$V(F) = \begin{cases} V(t_1) \cup V(t_2), & \text{si } F \text{ es } t_1 = t_2; \\ V(t_1) \cup \cdots \cup V(t_n), & \text{si } F \text{ es } P(t_1, \ldots, t_n); \\ V(G), & \text{si } F \text{ es } \neg G; \\ V(G) \cup V(H), & \text{si } F \text{ es } G * H; \\ V(G), & \text{si } F \text{ es } \forall x\ G; \\ V(G), & \text{si } F \text{ es } \exists x\ G \end{cases}$$

- Ejemplos:
  - El conjunto de las variables de $\forall x\ (R(x, c) \rightarrow P(f(y)))$ es $\{x, y\}$.
  - El conjunto de las variables de $\forall x\ (R(a, c) \rightarrow P(f(y)))$ es $\{y\}$.

Apariciones libres y ligadas

- Def.: Una aparición (u ocurrencia) de la variable x en la fórmula F es ligada si es en una subfórmula de F de la forma $\forall x\ G$ ó $\exists x\ G$.

- Def.: Una aparición (u ocurrencia) de la variable x en la fórmula F es libre si no es ligada.

- Ejemplo: Las apariciones ligadas son las subrayadas:
  $\forall x\ (P(\underline{x}) \to R(\underline{x},y)) \to (\exists y\ P(\underline{y}) \to R(z,x))$
  $\exists x\ R(\underline{x},y) \lor \forall y\ P(\underline{y})$
  $\forall x\ (P(\underline{x}) \to \exists y\ R(\underline{x},\underline{y}))$
  $P(x) \to R(x,y)$

Variables libres y ligadas

- La variable x es libre en F si tiene una aparición libre en F.

- La variable x es ligada en F si tiene una aparición ligada en F.

- El conjunto de las variables libres de una fórmula F es:

$$\mathrm{VL}(F) = \begin{cases} \mathrm{V}(t_1) \cup \mathrm{V}(t_2), & \text{si } F \text{ es } t_1 = t_2; \\ \mathrm{V}(t_1) \cup \dots \cup \mathrm{V}(t_n), & \text{si } F \text{ es } P(t_1,\dots,t_n); \\ \mathrm{VL}(G), & \text{si } F \text{ es } \neg G; \\ \mathrm{VL}(G) \cup \mathrm{VL}(H), & \text{si } F \text{ es } G * H; \\ \mathrm{VL}(G) \setminus \{x\}, & \text{si } F \text{ es } \forall x\ G; \\ \mathrm{VL}(G) \setminus \{x\}, & \text{si } F \text{ es } \exists x\ G \end{cases}$$

- Ejemplo:

Fórmula	Ligadas	Libres
$\forall x\ (P(x) \to R(x,y)) \to (\exists y\ P(y) \to R(x,z))$	x,y	x,y,z
$\forall x\ (P(x) \to \exists y\ R(x,y))$	x,y	
$\forall z\ (P(x) \to R(x,y))$		x,y

Fórmulas cerradas y abiertas

- Fórmula cerradas:
  - Def.: Una fórmula cerrada (o sentencia) es una fórmula sin variables libres.
  - Ejemplos: $\forall x\ (P(x) \to \exists y\ R(x,y))$ es cerrada.
    $\exists x\ R(x,y) \lor \forall y\ P(y)$ no es cerrada.

- Fórmulas abiertas:
  - Def.: Una fórmula abierta es una fórmula con variables libres.
  - Ejemplos: $\forall x\ (P(x) \to \exists y\ R(x,y))$ no es abierta.
    $\exists x\ R(x,y) \lor \forall y\ P(y)$ es abierta.

3.3. Semántica de la lógica de primer orden

3.3.1. Estructuras, asignaciones e interpretaciones

Estructuras, asignaciones e interpretaciones

- Una estructura del lenguaje L es un par $\mathcal{I} = (U, I)$ tal que:
  - U es un conjunto no vacío, denominado universo de la estructura;
  - I es una función con dominio el conjunto de símbolos propios de L tal que
    - si c es una constante de L, entonces $I(c) \in U$;
    - si f es un símbolo de función n–aria de L, entonces $I(f) : U^n \to U$;
    - si P es un símbolo de relación 0–aria de L, entonces $I(P) \in \{1, 0\}$;
    - si R es un símbolo de relación n–aria ($n > 0$) de L, entonces $I(R) \subseteq U^n$;
- Una asignación A en una estructura (U, I) es una función $A : \text{Var} \to U$ que hace corresponder a cada variable del alfabeto un elemento del universo de la estructura.
- Una interpretación de L es un par $(\mathcal{I}, A)$ formado por una estructura $\mathcal{I}$ de L y una asignación A en $\mathcal{I}$.
- Notación: A veces se usa para los valores de verdad V y F en lugar de 1 y 0.

Ejemplos de estructuras

Sea L el lenguaje de la aritmética cuyos símbolos propios son:
constante: 0;
símbolo de función monaria: s;
símbolo de función binaria: $+$ y
símbolo de relación binaria: $\leq$

- Primera estructura de L:
  $U_1 = \mathbb{N}$
  $I_1(0) = 0$
  $I_1(s) = \{(n, n+1) : n \in \mathbb{N}\}$ (sucesor)
  $I_1(+) = \{(a, b, a+b) : a, b \in \mathbb{N}\}$ (suma)
  $I_1(\leq) = \{(n, m) : n, m \in \mathbb{N}, n \leq m\}$ (menor o igual)
- Segunda estructura de L:
  $U_2 = \{0, 1\}^*$ (cadenas de 0 y 1)
  $I_2(0) = \epsilon$ (cadena vacía)
  $I_2(s) = \{(w, w1) : w \in \{0, 1\}^*\}$ (siguiente)

$I_2(+) = \{(w_1, w_2, w_1w_2) : w_1, w_2 \in \{0,1\}^*\}$ (concatenación)
$I_2(\leq) = \{(w_1, w_2) : w_1, w_2 \in \{0,1\}^*, w_1$ es prefijo de $w_2\}$ (prefijo)

- Tercera estructura de L:
  $U_3 = \{abierto, cerrado\}$
  $I_3(0) = cerrado$
  $I_3(s) = \{(abierto, cerrado), (cerrado, abierto)\}$
  $I_3(+) = \{ (abierto, abierto, abierto), (abierto, cerrado, abierto),$
  $(cerrado, abierto, abierto), (cerrado, cerrado, cerrado)\}$
  $I_3(\leq) = \{ (abierto, abierto), (cerrado, abierto), (cerrado, cerrado)\}$

e	$I_3(s)(e)$
abierto	*cerrado*
cerrado	*abierto*

$I_3(+)$	*abierto*	*cerrado*
abierto	*abierto*	*abierto*
cerrado	*abierto*	*cerrado*

$I_3(\leq)$	*abierto*	*cerrado*
abierto	1	0
cerrado	1	1

3.3.2. Evaluación de términos y fórmulas

Ejemplo de evaluación de términos

- Sean L el lenguaje de la página 41 y t el término $s(x + s(0))$.
  - Si $\mathcal{I}$ es la primera estructura y $A(x) = 3$, entonces
    $$\begin{aligned} \mathcal{I}_A(t) &= \mathcal{I}_A(s(x+s(0))) &&= s^I(3 +^I s^I(0^I)) = \\ &= s^I(3 +^I s^I(0)) &&= s^I(3 +^I 1) = \\ &= s^I(4) &&= 5 \end{aligned}$$
  - Si $\mathcal{I}$ es la segunda estructura y $A(x) = 10$, entonces
    $$\begin{aligned} \mathcal{I}_A(t) &= \mathcal{I}_A(s(x+s(0))) &&= s^I(10 +^I s^I(0^I)) = \\ &= s^I(10 +^I s^I(\epsilon)) &&= s^I(10 +^I 1) = \\ &= s^I(101) &&= 1011 \end{aligned}$$
  - Si $\mathcal{I}$ es la tercera estructura y $A(x) = abierto$, entonces
    $$\begin{aligned} \mathcal{I}_A(t) &= \mathcal{I}_A(s(x+s(0))) &&= s^I(abierto +^I s^I(0^I)) = \\ &= s^I(abierto +^I s^I(cerrado)) &&= s^I(abierto +^I abierto) = \\ &= s^I(abierto) &&= cerrado \end{aligned}$$

Evaluación de términos

- Def.: Dada una estructura $\mathcal{I} = (U, I)$ de L y una asignación A en $\mathcal{I}$, se define la función de evaluación de términos $\mathcal{I}_A : \text{Térm}(L) \to U$ por

$$\mathcal{I}_A(t) = \begin{cases} I(c), & \text{si } t \text{ es una constante } c; \\ A(x), & \text{si } t \text{ es una variable } x; \\ I(f)(\mathcal{I}_A(t_1), \dots, \mathcal{I}_A(t_n)), & \text{si } t \text{ es } f(t_1, \dots, t_n) \end{cases}$$

- $\mathcal{I}_A(t)$ se lee "el valor de t en $\mathcal{I}$ respecto de A".

- Ejemplo: Sean L el lenguaje de la página 41, t el término $s(+(x, s(0)))$, $\mathcal{I}$ la primera estructura y $A(x) = 3$.

$$\begin{aligned} \mathcal{I}_A(t) &= \mathcal{I}_A(s(+(x, s(0)))) & &= I(s)(\mathcal{I}_A(+(x, s(0)))) = \\ &= I(s)(I(+)(\mathcal{I}_A(x), \mathcal{I}_A(s(0)))) & &= I(s)(I(+)(A(x), \mathcal{I}_A(s(0)))) = \\ &= I(s)(I(+)(3, I(s)(\mathcal{I}_A(0)))) & &= I(s)(I(+)(3, I(s)(I(0)))) = \\ &= I(s)(I(+)(3, I(s)(0))) & &= I(s)(I(+)(3, 1)) = \\ &= I(s)(4) & &= 5 \end{aligned}$$

Evaluación de fórmulas

- Def.: Dada una estructura $\mathcal{I} = (U, I)$ de L y una asignación A sobre $\mathcal{I}$, se define la función de evaluación de fórmulas $\mathcal{I}_A : \text{Fórm}(L) \to \mathbb{B}$ por
  - Si F es $t_1 = t_2$, $\mathcal{I}_A(F) = H_=(\mathcal{I}_A(t_1), I_A(t_2))$
  - Si F es $P(t_1, \dots, t_n)$, $\mathcal{I}_A(F) = H_{I(P)}(\mathcal{I}_A(t_1), \dots, \mathcal{I}_A(t_n))$
  - Si F es $\neg G$, $\mathcal{I}_A(F) = H_\neg(\mathcal{I}_A(G))$
  - Si F es $G * H$, $\mathcal{I}_A(F) = H_*(\mathcal{I}_A(G), \mathcal{I}_A(H))$
  - Si F es $\forall x\ G$, $\mathcal{I}_A(F) = \begin{cases} 1, & \text{si para todo } u \in U \text{ se tiene } \\ & \mathcal{I}_{A[x/u]}(G) = 1; \\ 0, & \text{en caso contrario} \end{cases}$
  - Si F es $\exists x\ G$, $\mathcal{I}_A(F) = \begin{cases} 1, & \text{si existe algún } u \in U \text{ tal que } \\ & \mathcal{I}_{A[x/u]}(G) = 1; \\ 0, & \text{en caso contrario} \end{cases}$

- $\mathcal{I}_A(F)$ se lee "el valor de F en $\mathcal{I}$ respecto de A".

Conceptos auxilares para la evaluación de fórmulas

- La función de verdad de la igualdad en U es la función $H_= : U^2 \to \mathbb{B}$ definida por

$$H_=(u_1, u_2) = \begin{cases} 1, & \text{si } u_1 = u_2; \\ 0, & \text{en caso contrario} \end{cases}$$

- Función de verdad de una relación: Si R es una relación n–aria en U (i.e. $R \subseteq U^n$), entonces la función de verdad de R es la función $H_R : U^n \to \mathbb{B}$ definida por
$$H_R(u_1,\dots,u_n) = \begin{cases} 1, & \text{si } (u_1,\dots,u_n) \in R; \\ 0, & \text{en caso contrario} \end{cases}$$
- Variante de una asignación: Sea A una asignación en la estructura (U, I) y $u \in U$. Mediante $A[x/u]$ se representa la asignación definida por
$$A[x/u](y) = \begin{cases} u, & \text{si } y \text{ es } x; \\ A(y) & \text{si } y \text{ es distinta de } x \end{cases}$$

Ejemplo de evaluación de fórmula

Evaluación de $\forall x\ \exists y\ P(x,y)$ en la estructura $\mathcal{I} = (U,I)$ tal que $U = \{1,2\}$ e $I(P) = \{(1,1),(2,2)\}$

$\mathcal{I}_A(\forall x\ \exists y\ P(x,y)) = \mathsf{V} \Leftrightarrow \mathcal{I}_{A[x/1]}(\exists y\ P(x,y)) = \mathsf{V}$ y
$\mathcal{I}_{A[x/2]}(\exists y\ P(x,y)) = \mathsf{V}$

$\mathcal{I}_{A[x/1]}(\exists y\ P(x,y)) = \mathsf{V} \Leftrightarrow \mathcal{I}_{A[x/1,y/1]}P(x,y) = \mathsf{V}$ ó
$\mathcal{I}_{A[x/1,y/2]}P(x,y) = \mathsf{V}$

$\mathcal{I}_{A[x/1,y/1]}P(x,y) = P^I(1,1) = \mathsf{V}$
Luego, $\mathcal{I}_{A[x/1]}(\exists y\ P(x,y)) = \mathsf{V}$.

$\mathcal{I}_{A[x/2]}(\exists y\ P(x,y)) = \mathsf{V} \Leftrightarrow \mathcal{I}_{A[x/2,y/1]}P(x,y) = \mathsf{V}$ ó
$\mathcal{I}_{A[x/2,y/2]}P(x,y) = \mathsf{V}$

$\mathcal{I}_{A[x/2,y/2]}P(x,y) = P^I(2,2) = \mathsf{V}$
Luego, $\mathcal{I}_{A[x/2]}(\exists y\ P(x,y)) = \mathsf{V}$.

Por tanto, $\mathcal{I}_A(\forall x\ \exists y\ P(x,y)) = \mathsf{V}$

Ejemplo de evaluación de fórmulas

Evaluación de $\forall x\ g(g(x)) = x$ en la estructura $\mathcal{I} = (U,I)$ tal que $U = \{1,2\}$ e $I(g) = \{(1,2),(2,1)\}$.

$$\begin{aligned}\mathcal{I}_A(\forall x\ g(g(x)) = x) = \mathsf{V} \Leftrightarrow\ & \mathcal{I}_{A[x/1]} g(g(x)) = x = \mathsf{V} \text{ y}\\ & \mathcal{I}_{A[x/2]} g(g(x)) = x = \mathsf{V}\end{aligned}$$

$$\begin{aligned}\mathcal{I}_{A[x/1]}(g(g(x)) = x) &= (g^I(g^I(1)) = 1)\\ &= (g^I(2) = 1)\\ &= (1 = 1)\\ &= \mathsf{V}\end{aligned}$$

$$\begin{aligned}\mathcal{I}_{A[x/2]}(g(g(x)) = x) &= (g^I(g^I(2)) = 2)\\ &= (g^I(1) = 2)\\ &= (2 = 2)\\ &= \mathsf{V}\end{aligned}$$

Por tanto, $\mathcal{I}_A(\forall x\ g(g(x)) = x) = \mathsf{V}$.

Dependencias en la evaluación de fórmulas

- Ejemplo de dependencia del universo: Sea G la fórmula $\forall x\ \exists y\ R(y, x)$, entonces
  - $\mathcal{I}_A(G) = \mathsf{V}$, siendo $\mathcal{I} = (\mathbb{Z}, I), I(R) = <$ y A una asignación en $\mathcal{I}$.
  - $\mathcal{I}_A(G) = \mathsf{F}$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = <$ y A una asignación en $\mathcal{I}$.
- Ejemplo de dependencia de la estructura: Sea G la fórmula $\exists x\ \forall y\ R(x, y)$, entonces
  - $\mathcal{I}_A(G) = \mathsf{V}$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \leq$ y A una asignación en $\mathcal{I}$.
  - $\mathcal{I}_A(G) = \mathsf{F}$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \geq$ y A una asignación en $\mathcal{I}$.
- Ejemplo de dependencia de la asignación: Sea G la fórmula $\forall y\ R(x, y)$, entonces
  - $\mathcal{I}_A(G) = \mathsf{V}$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \leq$ y A una asignación en $\mathcal{I}$ tal que $A(x) = 0$.
  - $\mathcal{I}_A(G) = \mathsf{F}$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \leq$ y A una asignación en $\mathcal{I}$ tal que $A(x) = 5$.

Evaluación y variables libres

- Sea t un término de L e $\mathcal{I}$ una estructura de L.
  - Si A y B son dos asignaciones en $\mathcal{I}$ que coinciden sobre las variables de t, entonces $\mathcal{I}_A(t) = \mathcal{I}_B(t)$.
  - Si t no tiene variables, entonces $\mathcal{I}_A(t) = \mathcal{I}_B(t)$ para cualesquiera asignaciones A y B en $\mathcal{I}$. Se suele escribir simplemente $\mathcal{I}(t)$.

- Sea F una fórmula de L e $\mathcal{I}$ una estructura de L.
  - Si A y B son dos asignaciones en $\mathcal{I}$ que coinciden sobre las variables libres de F, entonces $\mathcal{I}_A(F) = \mathcal{I}_B(F)$.
  - Si F es cerrada, entonces $\mathcal{I}_A(F) = \mathcal{I}_B(F)$ para cualesquiera asignaciones A y B en $\mathcal{I}$. Se suele escribir simplemente $\mathcal{I}(F)$.

3.3.3. Modelo, satisfacibilidad y validez de fórmulas

Modelo de una fórmula

- Sean F una fórmula de L e $\mathcal{I}$ una estructura de L.
  - $(\mathcal{I}, A)$ es una realización de F si A es una asignación en $\mathcal{I}$ tal que $\mathcal{I}_A(F) = 1$.
    Se representa por $\mathcal{I}_A \models F$.
  - $\mathcal{I}$ es un modelo de F si, para todo asignación A en $\mathcal{I}$, $\mathcal{I}_A(F) = 1$.
    Se representa por $\mathcal{I} \models F$.
- Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(f) = +$ e $I(g) = *$.
  - Si A es una asignación en $\mathcal{I}$ tal que $A(x) = A(y) = 2$. Entonces
    $\mathcal{I}_A \models f(x,y) = g(x,y)$,
  - Si B es una asignación en $\mathcal{I}$ tal que $B(x) = 1, B(y) = 2$. Entonces
    $\mathcal{I}_B \not\models f(x,y) = g(x,y)$,
  - $\mathcal{I} \not\models f(x,y) = g(x,y)$
  - $\mathcal{I} \models f(x,y) = f(y,x)$

Satisfacibilidad y validez

- Def.: Sea F una fórmula de L.
  - F es válida si toda estructura de L es modelo de F,
    (i.e. para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$ se tiene que $\mathcal{I}_A(F) = 1$).
    Se representa por $\models F$.
  - F es satisfacible si tiene alguna realización
    (i.e. existe alguna estructura $\mathcal{I}$ de L y alguna asignación A en I tales que $\mathcal{I}_A(F) = 1$).
  - F es insatisfacible si no tiene ninguna realización
    (i.e. para toda estructura $\mathcal{I}$ de L y toda asignación A en I se tiene que $\mathcal{I}_A(F) = 0$).

- Ejemplos:
  - $\exists x\ P(x) \lor \forall x\ \neg P(x)$ es válida.
  - $\exists x\ P(x) \land \exists x\ \neg P(x)$ es satisfacible, pero no es válida.
  - $\forall x\ P(x) \land \exists x\ \neg P(x)$ es insatisfacible.

- F es válida syss $\neg F$ es insatisfacible.
  F es válida
  $\iff$ para toda estructura $\mathcal{I}$ y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$
  $\iff$ para toda estructura $\mathcal{I}$ y toda asignación A se tiene que $\mathcal{I}_A(\neg F) = 0$
  $\iff$ $\neg F$ es insatisfacible.

- Si F es válida, entonces F es satisfacible.
  F es válida
  $\implies$ para toda estructura $\mathcal{I}$ y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$
  $\implies$ existe una estructura $\mathcal{I}$ y una asignación A tales que $\mathcal{I}_A(F) = 1$
  $\implies$ F es satisfacible.

- F es satisfacible $\not\Longrightarrow$ $\neg F$ es insatisfacible.
  $\forall x\ P(x)$ y $\neg\forall x\ P(x)$ son satisfacibles.

- Sea F una fórmula de L y $x_1, \dots, x_n$ las variables libres de F.
  - F es válida syss $\forall x_1\ \dots \forall x_n\ F$ es válida.
    [$\forall x_1\ \dots \forall x_n\ F$ es el cierre universal de F].
  - F es satisfacible syss $\exists x_1\ \dots \exists x_n\ F$ es satisfacible.
    [$\exists x_1\ \dots \exists x_n\ F$ es el cierre existencial de F].

3.3.4. Modelo y consistencia de conjuntos de fórmulas

Modelo de un conjunto de fórmulas

- Notación: $S, S_1, S_2, \dots$ representarán conjuntos de fórmulas.

- Def.: Sean S un conjunto de fórmulas de L, $\mathcal{I}$ una estructura de L y A una asignación en $\mathcal{I}$.
  - $(\mathcal{I}, A)$ es una realización de S si A es una asignación en $\mathcal{I}$ tal que para toda $F \in S$ se tiene que $\mathcal{I}_A(F) = 1$. Se representa por $\mathcal{I}_A \models S$.
  - $\mathcal{I}$ es un modelo de S si para toda $F \in S$ se tiene que $\mathcal{I} \models F$
    (i.e. para toda $F \in S$ y toda asignación A en $\mathcal{I}$ se tiene $\mathcal{I}_A(F) = 1$). Se representa por $\mathcal{I} \models S$.

- Ejemplo: Sea $S = \{\forall y\ R(x,y),\ \forall y\ f(x,y) = y\}$.
  - $(\mathcal{I}, A)$ con $\mathcal{I} = (\mathbb{N}, I), R^I = \leq, f^I = +, A(x) = 0$ es realización de S.
  - $(\mathcal{I}, A)$ con $\mathcal{I} = (\mathbb{N}, I), R^I = <, f^I = +, A(x) = 0$ no es realización de S.
- Ejemplo: Sea $S = \{R(e,y),\ f(e,y) = y\}$.
  - $\mathcal{I} = (\mathbb{N}, I)$ con $R^I = \leq, f^I = +, e^I = 0$ es modelo de S.
  - $\mathcal{I} = (\mathbb{N}, I)$ con $R^I = <, f^I = +, e^I = 0$ no es modelo de S.

Consistencia de un conjunto de fórmulas

- Def.: Sea S un conjunto de fórmulas de L.
  - S es consistente si S tiene alguna realización
    (i.e. existe alguna estructura $\mathcal{I}$ de L y alguna asignación A en $\mathcal{I}$ tales que, para toda $F \in S$, $I_A(F) = 1$).
  - S es inconsistente si S no tiene ninguna realización
    (i.e. para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$, existe alguna $F \in S$, tal que $I_A(F) = 0$).
- Ejemplos:
  - $S = \{\forall y\ R(x,y),\ \forall y\ f(x,y) = y\}$ es consistente.
    $(\mathcal{I}, A)$ con $\mathcal{I} = (\mathbb{N}, I), R^I = \leq, f^I = +, A(x) = 0$ es realización de S.
  - $S = \{P(x) \to Q(x),\ \forall y\ P(y),\ \neg Q(x)\}$ es inconsistente.
- Prop.: Sea S un conjunto de fórmulas *cerradas* de L. Entonces S es consistente syss S tiene algún modelo.

3.3.5. Consecuencia lógica

Consecuencia lógica

- Def.: Sean F una fórmula de L y S un conjunto de fórmulas de L.
  - F es consecuencia lógica de S si todas las realizaciones de S lo son de F.
    (i.e. para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$,
    si $\mathcal{I}_A \models S$ entonces $\mathcal{I}_A \models F$).
    Se representa por $S \models F$.
  - Se escribe $G \models F$ en lugar de $\{G\} \models F$.
  - Se escribe $G \not\models F$ en lugar de $\{G\} \not\models F$.

- Ejemplos:
  - $\forall x\ P(x) \models P(y)$
  - $P(y) \not\models \forall x\ P(x)$
    $(\mathcal{I}, A)$ con $\mathcal{I} = (U, I), U = \{1, 2\}, P^I = \{1\}, A(y) = 1$.
  - $\{\forall x\ (P(x) \to Q(x)),\ P(c)\} \models Q(c)$
  - $\{\forall x\ (P(x) \to Q(x)),\ Q(c)\} \not\models P(c)$
    $(\mathcal{I}, A)$ con $\mathcal{I} = (U, I), U = \{1, 2\}, c^I = 1, P^I = \{2\}, Q^I = \{1, 2\}$.
  - $\{\forall x\ (P(x) \to Q(x)),\ \neg Q(c)\} \models \neg P(c)$
  - $\{P(c),\ \neg P(d)\} \models c \neq d$

Consecuencia lógica e inconsistencia

- $S \models F$ syss $S \cup \{\neg F\}$ es inconsistente.
  $S \models F$
  $\iff$ para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$,
  si, para todo $G \in S, \mathcal{I}_A(G) = 1$ entonces $\mathcal{I}_A(F) = 1$.
  $\iff$ para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$,
  si, para todo $G \in S, \mathcal{I}_A(G) = 1$ entonces $\mathcal{I}_A(\neg F) = 0$.
  $\iff$ para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$,
  existe alguna $H \in S \cup \{\neg F\}$ tal que $\mathcal{I}_A(H) = 0$.
  $\iff$ $S \cup \{\neg F\}$ es inconsistente.
- Sean F una fórmula *cerrada* de L y S un conjunto de fórmulas *cerradas* de L. Entonces, son equivalentes
  - F es consecuencia lógica de S
  - todos los modelos de S lo son de F.

3.3.6. Equivalencia lógica

- Def.: Sean F y G fórmulas de L. F y G son equivalentes si para toda estructura $\mathcal{I}$ de L y toda asignación A en $\mathcal{I}$, $\mathcal{I}_A(F) = \mathcal{I}_A(G)$.
  Se representa por $F \equiv G$.
- Ejemplos:
  - $P(x) \not\equiv P(y)$.
    $\mathcal{I} = (\{1, 2\}, I)$ con $P^I = \{1\}$ y $A(x) = 1, A(y) = 2$.
  - $\forall x\ P(x) \equiv \forall y\ P(y)$.

- $\forall x\ (P(x) \land Q(x)) \equiv \forall x\ P(x) \land \forall x\ Q(x)$.
- $\exists x\ (P(x) \land Q(x)) \not\equiv \exists x\ P(x) \land \exists x\ Q(x)$.
  $\mathcal{I} = (\{1,2\}, I)$ con $P^I = \{1\}$ y $Q^I = \{2\}$.

- Propiedades: Sean F y G fórmulas cerradas de L.
  - $F \equiv G$ syss $\models F \leftrightarrow G$.
  - $F \equiv G$ syss $F \models G$ y $G \models F$.
- Propiedades básicas de la equivalencia lógica:
  - Reflexiva: $F \equiv F$
  - Simétrica: Si $F \equiv G$, entonces $G \equiv F$
  - Transitiva: Si $F \equiv G$ y $G \equiv H$, entonces $F \equiv H$
- Principio de sustitución de fórmulas equivalentes:
  - Prop.: Si en la fórmula F_1 se sustituye una de sus subfórmulas G_1 por una fórmula G_2 lógicamente equivalente a G_1, entonces la fórmula obtenida, F_2, es lógicamente equivalente a F_1.
  - Ejemplo: $$\begin{aligned} F_1 &= \forall x\ P(x) \to \exists x\ Q(x) \\ G_1 &= \forall x\ P(x) \\ G_2 &= \forall y\ P(y) \\ F_2 &= \forall y\ P(y) \to \exists x\ Q(x) \end{aligned}$$

Bibliografía

1. C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal.* (Ariel, 2000) pp. 195–259 y 323–326.
2. M.L. Bonet *Apuntes de LPO.* (Univ. Politécnica de Cataluña, 2003) pp. 17–26.
3. J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional.* (UNED, 2003) pp. 64–87.
4. J.H. Gallier *Logic for computer science (foundations of automatic theorem Proving)* (June 2003) pp. 146–186.
5. M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems.* (Cambridge University Press, 2000) pp. 90–109 y 128–140.
6. M. Ojeda e I. Pérez de Guzmán *Lógica para la computación (Vol. 2: Lógica de primer orden)* (Ágora, 1997) pp. 1–37 y 49–51.
7. L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 22–29.

Bibliografía complementaria

Tema 4

Deducción natural en lógica de primer orden

Contenido

4.1. Sustituciones

4.1.1. Definición de sustitución

- Def.: Una sustitución σ (de L) es una aplicación $\sigma : \text{Var} \to \text{Térm}(L)$.
- Notación: $[x_1/t_1, x_2/t_2, \dots, x_n/t_n]$ representa la sustitución σ definida por

$$\sigma(x) = \begin{cases} t_i, & \text{si } x \text{ es } x_i; \\ x, & \text{si } x \notin \{x_1, \dots, x_n\} \end{cases}$$

- Ejemplo: $[x/s(0), y/x+y]$ es la sustitución σ de Var en los términos de la aritmética definida por
  $\sigma(x) = s(0), \sigma(y) = x + y$ y $\sigma(z) = z$ para $z \in \text{Var} \setminus \{x, y\}$
- Notación: $\sigma, \sigma_1, \sigma_2, \ldots$ representarán sustituciones.

4.1.2. Aplicación de sustituciones a términos

- Def.: $t[x_1/t_1, \ldots, x_n/t_n]$ es el término obtenido sustituyendo en t las apariciones de x_i por t_i.
- Def.: La extensión de σ a términos es la aplicación $\sigma : \text{Térm}(L) \to \text{Térm}(L)$ definida por

$$t\sigma = \begin{cases} c, & \text{si } t \text{ es una constante } c; \\ \sigma(x), & \text{si } t \text{ es una variable } x; \\ f(t_1\sigma, \ldots, t_n\sigma), & \text{si } t \text{ es } f(t_1, \ldots, t_n) \end{cases}$$

- Ejemplo: Si $\sigma = [x/f(y,a), y/z]$, entonces
  - $a\sigma = a$, donde a es una constante.
  - $w\sigma = w$, donde w es una variable distinta de x e y.
  - $h(a,x,w)\sigma = h(a\sigma, x\sigma, w\sigma) = h(a, f(y,a), w)$
  - $f(x,y)\sigma = f(x\sigma, y\sigma) = f(f(y,a), z)$
  - $h(a, f(x,y), w)\sigma = h(a\sigma, f(x,y)\sigma, w\sigma) = h(a, f(f(y,a), z), w)$

4.1.3. Aplicación de sustituciones a fórmulas

- Def.: $F[x_1/t_1, \ldots, x_n/t_n]$ es la fórmula obtenida sustituyendo en F las apariciones libres de x_i por t_i.
- Def.: La extensión de σ a fórmulas es la aplicación $\sigma : \text{Fórm}(L) \to \text{Fórm}(L)$ definida por

$$F\sigma = \begin{cases} P(t_1\sigma, \ldots, t_n\sigma), & \text{si } F \text{ es la fórmula atómica } P(t_1, \ldots, t_n); \\ t_1\sigma = t_2\sigma, & \text{si } F \text{ es la fórmula } t_1 = t_2; \\ \neg(G\sigma), & \text{si } F \text{ es } \neg G; \\ G\sigma * H\sigma, & \text{si } F \text{ es } G * H; \\ (Qx)(G\sigma_x), & \text{si } F \text{ es } (Qx)G \text{ y } Q \in \{\forall, \exists\} \end{cases}$$

  donde σ_x es la sustitución definida por

$$\sigma_x(y) = \begin{cases} x, & \text{si } y \text{ es } x; \\ \sigma(y), & \text{si } y \text{ es distinta de } x \end{cases}.$$

- Ejemplos: Si $\sigma = [x/f(y), y/b]$, entonces
  - $$\begin{aligned} (\forall x\ (Q(x) \to R(x,y)))\sigma &= \forall x\ ((Q(x) \to R(x,y))\sigma_x) \\ &= \forall x\ (Q(x)\sigma_x \to R(x,y)\sigma_x) \\ &= \forall x\ (Q(x) \to R(x,b)) \end{aligned}$$
  - $$\begin{aligned} (Q(x) \to \forall x\ R(x,y))\sigma &= Q(x)\sigma \to (\forall x\ R(x,y))\sigma \\ &= Q(f(y)) \to \forall x\ (R(x,y)\sigma_x) \\ &= Q(f(y)) \to \forall x\ R(x,b) \end{aligned}$$
  - $$\begin{aligned} (\forall x\ (Q(x) \to \forall y\ R(x,y)))\sigma &= \forall x\ ((Q(x) \to \forall y\ R(x,y))\sigma_x) \\ &= \forall x\ (Q(x)\sigma_x \to (\forall y\ R(x,y))\sigma_x) \\ &= \forall x\ (Q(x) \to \forall y\ (R(x,y)\sigma_{xy})) \\ &= \forall x\ (Q(x) \to \forall y\ R(x,y)) \end{aligned}$$

4.1.4. Sustituciones libres

- Def.: Una sustitución se denomina libre para una fórmula cuando todas las apariciones de variables introducidas por la sustitución en esa fórmula resultan libres.
- Ejemplos:
  - $[y/x]$ no es libre para $\exists x\ (x < y)$
    $\exists x\ (x < y)[y/x] = \exists x\ (x < x)$
  - $[y/g(y)]$ es libre para $\forall x\ (P(x) \to Q(x, f(y)))$
    $\forall x\ (P(x) \to Q(x, f(y)))[y/g(y)]$
    $= \forall x\ (P(x) \to Q(x, f(g(y))))$
  - $[y/g(x)]$ no es libre para $\forall x\ (P(x) \to Q(x, f(y)))$
    $\forall x\ (P(x) \to Q(x, f(y)))[y/g(x)]$
    $= \forall x\ (P(x) \to Q(x, f(g(x))))$
- Convenio: Al escribir $F\sigma$ supondremos que σ es libre para F.

4.2. Reglas de deducción natural de cuantificadores

4.2.1. Reglas del cuantificador universal

Regla de eliminación del cuantificador universal

- Regla de eliminación del cuantificador universal:
  $$\frac{\forall x\ F}{F[x/t]}\ \forall e$$
  donde $[x/t]$ es libre para F.

- Nota: Analogía con $\wedge e_1$ y $\wedge e_2$.
- Ejemplo: $P(c), \forall x\ [P(x) \to \neg Q(x)] \vdash \neg Q(c)$

1	$P(c)$	premisa
2	$\forall x\ (P(x) \to \neg Q(x))$	premisa
3	$P(c) \to \neg Q(c)$	$\forall$e 2
4	$\neg Q(c)$	$\to$e 3, 1

- Nota: $\forall x\ \exists y\ (x < y) \nvdash \exists y\ (y < y)$.

Regla de introducción del cuantificador universal

$$\frac{\boxed{\begin{array}{l} x_0 \\ \quad\vdots \\ \quad F[x/x_0] \end{array}}}{\forall x\ F}\ \forall i$$

donde x_0 es una variable nueva, que no aparece fuera de la caja.

- Nota: Analogía con $\wedge$i.
- $\forall x\ [P(x) \to \neg Q(x)], \forall x\ P(x) \vdash \forall x\ \neg Q(x)$

1	$\forall x\ (P(x) \to \neg Q(x))$	premisa
2	$\forall x\ P(x)$	premisa
3	actual x_0	supuesto
4	$P(x_0) \to \neg Q(x_0)$	$\forall$e 1, 3
5	$P(x_0)$	$\forall$e 2, 3
6	$\neg Q(x_0)$	$\to$e 4, 5
7	$\forall x\ \neg Q(x)$	$\forall$i 3 − 6

4.2.2. Reglas del cuantificador existencial

Regla de introducción del cuantificador existencial

- Regla de introducción del cuantificador existencial:

$$\frac{F[x/t]}{\exists x\ F}\ \exists i$$

donde $[x/t]$ es libre para F.

- Nota: Analogía con $\vee i_1$ y $\vee i_2$.
- Ejemplo 3: $\forall x\ P(x) \vdash \exists x\ P(x)$

1	$\forall x\ P(x)$	premisa
2	$P(x_0)$	$\forall$e 1
3	$\exists x\ P(x)$	$\exists$i 2

Regla de eliminación del cuantificador existencial

$$\dfrac{\exists x\ F \quad \boxed{\begin{array}{ll} x_0 & F[x/x_0] \\ & \vdots \\ & G \end{array}}}{G}\ \exists e$$

donde x_0 es una variable nueva, que no aparece fuera de la caja.

- Nota: Analogía con $\vee$e.
- Ejemplo: $\forall x\ [P(x) \to Q(x)], \exists x\ P(x) \vdash \exists x\ Q(x)$

1	$\forall x\ (P(x) \to Q(x))$	premisa
2	$\exists x\ P(x)$	premisa
3	actual $x_0, P(x_0)$	supuesto
4	$P(x_0) \to Q(x_0)$	$\forall$e $1, 3$
5	$Q(x_0)$	$\to$e $4, 3$
6	$\exists x\ Q(x)$	$\exists$i 5
7	$\exists x\ Q(x)$	$\exists$e $2, 3-6$

4.2.3. Demostración de equivalencias por deducción natural

Equivalencias

- Sean F y G fórmulas.

  [1(a)] $\neg\forall x\ F \equiv \exists x\ \neg F$

  [1(b)] $\neg\exists x\ F \equiv \forall x\ \neg F$

- Sean F y G fórmulas y x una varible no libre en G.

  [2(a)] $\forall x\ F \wedge G \equiv \forall x\ (F \wedge G)$

  [2(b)] $\forall x\ F \vee G \equiv \forall x\ (F \vee G)$

  [2(c)] $\exists x\ F \wedge G \equiv \exists x\ (F \wedge G)$

[2(d)] $\exists x\ F \lor G \equiv \exists x\ (F \lor G)$

- Sean F y G fórmulas.

  [3(a)] $\forall x\ F \land \forall x\ G \equiv \forall x\ (F \land G)$

  [3(b)] $\exists x\ F \lor \exists x\ G \equiv \exists x\ (F \lor G)$

- Sean F y G fórmulas.

  [4(a)] $\forall x\ \forall y\ F \equiv \forall y\ \forall x\ F$

  [4(b)] $\exists x\ \exists y\ F \equiv \exists y\ \exists x\ F$

Equivalencia 1(a) $\rightarrow$

$\neg\forall x\ P(x) \vdash \exists x\ \neg P(x)$

1	$\neg\forall x\ P(x)$	premisa
2	$\neg\exists x\ \neg P(x)$	supuesto
3	actual x_0	supuesto
4	$\neg P(x_0)$	supuesto
5	$\exists x\ \neg P(x)$	$\exists$i $4, 3$
6	$\bot$	$\neg$e $2, 5$
7	$P(x_0)$	RAA $4-6$
8	$\forall x\ P(x)$	$\forall$i $3-7$
9	$\bot$	$\neg$e $1, 8$
10	$\exists x\ \neg P(x)$	RAA $2-9$

Equivalencia 1(a) $\leftarrow$

$\exists x\ \neg P(x) \vdash \neg\forall x\ P(x)$

1	$\exists x \; \neg P(x)$	premisa
2	$\neg\neg\forall x \; P(x)$	supuesto
3	actual $x_0, \neg P(x_0)$	supuesto
4	$\forall x \; P(x)$	$\neg\neg$e 2
5	$P(x_0)$	$\forall$e 4
6	$\perp$	$\neg$e $3, 5$
7	$\perp$	$\exists$e $1, \; 3-6$
8	$\neg\forall x \; P(x)$	RAA $2-7$

Equivalencia 1(a) $\leftrightarrow$

$\neg\forall x \; P(x) \equiv \exists x \; \neg P(x)$

1	$\neg\forall x \; P(x)$	supuesto
2	$\exists x \; \neg P(x)$	Lema $1(\mathrm{a}) \; \rightarrow$
3	$\neg\forall x \; P(x) \rightarrow \exists x \; \neg P(x)$	$\rightarrow$i $1-2$
4	$\exists x \; \neg P(x)$	supuesto
5	$\neg\forall x \; P(x)$	Lema $1(\mathrm{a}) \; \leftarrow$
6	$\exists x \; \neg P(x) \rightarrow \neg\forall x \; P(x)$	$\rightarrow$i $4-5$
7	$\neg\forall x \; P(x) \leftrightarrow \exists x \; \neg P(x)$	$\leftrightarrow$i $3, 6$

Equivalencia 3(a) $\rightarrow$

$\forall x \; (P(x) \wedge Q(x)) \vdash \forall x \; P(x) \wedge \forall x \; Q(x)$

1	$\forall x\ (P(x) \land Q(x))$	premisa
2	actual x_0	supuesto
3	$P(x_0) \land Q(x_0)$	$\forall$e $1, 2$
4	$P(x_0)$	$\land$e$_1$ 3
5	$\forall x\ P(x)$	$\forall$i $2 - 4$
6	actual x_1	supuesto
7	$P(x_1) \land Q(x_1)$	$\forall$e $1, 6$
8	$Q(x_1)$	$\land$e$_2$ 7
9	$\forall x\ Q(x)$	$\forall$i $6 - 8$
10	$\forall x\ P(x) \land \forall x\ Q(x)$	$\land$i $5, 9$

Equivalencia 3(a) $\leftarrow$

$\forall x\ P(x) \land \forall x\ Q(x) \vdash \forall x\ (P(x) \land Q(x))$

1	$\forall x\ P(x) \land \forall x\ Q(x)$	premisa
2	actual x_0	supuesto
3	$\forall x\ P(x)$	$\land$e$_1$ 1
4	$P(x_0)$	$\forall$e $3, 2$
5	$\forall x\ Q(x)$	$\land$e$_2$ 1
6	$Q(x_0)$	$\forall$e 5
7	$P(x_0) \land Q(x_0)$	$\land$i $4, 6$
8	$\forall x\ (P(x) \land Q(x))$	$\forall$i $2 - 7$

Equivalencia 3(a) $\leftrightarrow$

$\forall x\ (P(x) \land Q(x)) \equiv \forall x\ P(x) \land \forall x\ Q(x)$

1	$\forall x\ (P(x) \land Q(x))$	supuesto
2	$\forall x\ P(x) \land \forall x\ Q(x)$	Lema 3(a) $\rightarrow$
3	$\forall x\ (P(x) \land Q(x)) \rightarrow \forall x\ P(x) \land \forall x\ Q(x)$	$\rightarrow$i $1-2$
4	$\forall x\ P(x) \land \forall x\ Q(x)$	supuesto
5	$\forall x\ (P(x) \land Q(x))$	Lema 3(a) $\leftarrow$
6	$\forall x\ P(x) \land \forall x\ Q(x) \rightarrow \forall x\ (P(x) \land Q(x))$	$\rightarrow$i $4-5$
7	$\forall x\ (P(x) \land Q(x)) \leftrightarrow \forall x\ P(x) \land \forall x\ Q(x)$	$\leftrightarrow$i $3,6$

Equivalencia 3(b) $\rightarrow$

$\exists x\ P(x) \lor \exists x\ Q(x) \vdash \exists x\ (P(x) \lor Q(x))$

1	$\exists x\ P(x) \lor \exists x\ Q(x)$			premisa
2	$\exists x\ P(x)$	supuesto	$\exists x\ Q(x)$	supuesto
3	actual $x_0, P(x_0)$	supuesto	actual $x_1, Q(x_1)$	supuesto
4	$P(x_0) \lor Q(x_0)$	$\lor$i$_1$ 3	$P(x_1) \lor Q(x_1)$	$\lor$i$_2$ $3'$
5	$\exists x\ (P(x) \lor Q(x))$	$\exists$i $4,3$	$\exists x\ (P(x) \lor Q(x))$	$\exists$i $3',4'$
6	$\exists x\ (P(x) \lor Q(x))$	$\exists$e $2,3-5$	$\exists x\ (P(x) \lor Q(x))$	$\exists$e $2',3'-5'$
7	$\exists x\ (P(x) \lor Q(x))$			$\lor$1e $,2-6,2'-6'$

Equivalencia 3(b) $\leftarrow$

$\exists x\ (P(x) \lor Q(x)) \vdash \exists x\ P(x) \lor \exists x\ Q(x)$

1	$\exists x\ (P(x) \lor Q(x))$	premisa
2	actual $x_0,\ P(x_0) \lor Q(x_0)$	supuesto
3	$P(x_0)$	supuesto
4	$\exists x\ P(x)$	$\exists$i $3,2$
5	$\exists x\ P(x) \lor \exists x\ Q(x)$	$\lor$i$_1$ 4
6	$Q(x_0)$	supuesto
7	$\exists x\ Q(x)$	$\exists$i $6,2$
8	$\exists x\ P(x) \lor \exists x\ Q(x)$	$\lor$i$_2$ 7
9	$\exists x\ P(x) \lor \exists x\ Q(x)$	$\lor$e $2,3-5,6-8$
10	$\exists x\ P(x) \lor \exists x\ Q(x)$	$\exists$e $1,2-9$

Equivalencia 3(b) $\leftrightarrow$

$\exists x\ P(x) \lor \exists x\ Q(x) \equiv \exists x\ (P(x) \lor Q(x))$

1	$\exists x\ P(x) \lor \exists x\ Q(x)$	supuesto
2	$\exists x\ (P(x) \lor Q(x))$	Lema 3(b) $\rightarrow$
3	$\exists x\ P(x) \lor \exists x\ Q(x) \rightarrow \exists x\ (P(x) \lor Q(x))$	$\rightarrow$i $1-2$
4	$\exists x\ (P(x) \lor Q(x))$	supuesto
5	$\exists x\ P(x) \lor \exists x\ Q(x)$	Lema 3(b) $\leftarrow$
6	$\exists x\ (P(x) \lor Q(x)) \rightarrow \exists x\ P(x) \lor \exists x\ Q(x)$	$\rightarrow$i $4-5$
7	$\exists x\ P(x) \lor \exists x\ Q(x) \leftrightarrow \exists x\ (P(x) \lor Q(x))$	$\leftrightarrow$i $3,6$

Equivalencia 4(b) $\rightarrow$

$\exists x\ \exists y\ P(x,y) \vdash \exists y\ \exists x\ P(x,y)$

1	$\exists x\ \exists y\ P(x,y)$	premisa
2	actual x_0, $\exists y\ P(x_0,y)$	supuesto
3	actual y_0, $P(x_0,y_0)$	supuesto
4	$\exists x\ P(x,y_0)$	$\exists$i 3,2,2,1
5	$\exists y\ \exists x\ P(x,y)$	$\exists$i $4,3,1$
6	$\exists y\ \exists x\ P(x,y)$	$\exists$e 2,2,$3-5$
7	$\exists y\ \exists x\ P(x,y)$	$\exists$e $1,2-6$

Equivalencia 4(b) $\leftrightarrow$

$\exists x\ \exists y\ P(x,y) \equiv \exists y\ \exists x\ P(x,y)$

1	$\exists x\ \exists y\ P(x,y)$	supuesto
2	$\exists y\ \exists x\ P(x,y)$	Lema 4(b) $\rightarrow$
3	$\exists x\ \exists y\ P(x,y) \rightarrow \exists y\ \exists x\ P(x,y)$	$\rightarrow$i $1-2$
4	$\exists y\ \exists x\ P(x,y)$	supuesto
5	$\exists x\ \exists y\ P(x,y)$	Lema 4(b) $\rightarrow$
6	$\exists y\ \exists x\ P(x,y) \rightarrow \exists x\ \exists y\ P(x,y)$	$\rightarrow$i $4-5$
7	$\exists x\ \exists y\ P(x,y) \leftrightarrow \exists y\ \exists x\ P(x,y)$	$\leftrightarrow$i $3,6$

4.3. Reglas de la igualdad

4.3.1. Regla de eliminación de la igualdad

- Regla de eliminación de la igualdad:

  $$\frac{t_1 = t_2 \quad F[x/t_1]}{F[x/t_2]} = e$$

  donde $[x/t_1]$ y $[x/t_2]$ son libres para F.

- Ejemplo:

1	$(x+1) = (1+x)$	premisa
2	$(x+1>1) \to (x+1>0)$	premisa
3	$(1+x>1) \to (1+x>0)$	=e $1,2$

- Ejemplo: $t_1 = t_2, t_2 = t_3 \vdash t_1 = t_3$

1	$t_1 = t_2$	premisa
2	$t_2 = t_3$	premisa
3	$t_1 = t_3$	=e $2,1$

4.3.2. Regla de introducción de la igualdad

- Regla de introducción de la igualdad:

  $$\frac{}{t = t} = i$$

- Ejemplo: $t_1 = t_2 \vdash t_2 = t_1$

1	$t_1 = t_2$	premisa
2	$t_1 = t_1$	=i
3	$t_2 = t_1$	=e $1,2$

Bibliografía

1. C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal.* (Ariel, 2000) pp. 259–287.
2. R. Bornat *Using ItL Jape with X* (Department of Computer Science, QMW, 1998)
3. J. Dingel *Propositional and predicate logic: a review.* (2000) pp. 28–33.
4. J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional.* (UNED, 2003) pp. 88–94.

5. M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems.* (Cambridge University Press, 2000) pp. 109-127.

Bibliografía

[1] J.A. Alonso y J. Borrego *Deducción automática (Vol. 1: Construcción lógica de sistemas lógicos).* (Ed. Kronos, 2002)

[2] L. Arenas *Lógica formal para informáticos.* (Ed. Díaz de Santos, 1996)

[3] C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal* (Ariel, 2000)

[4] M. Ben–Ari *Mathematical Logic for Computer Science (2nd ed.)* (Springer, 2001)

[5] R. Bornat *Proof and disproof in formal logic: an introduction for programmers.* (Department of Computer Science, QMW, 1998).

[6] I. Bratko *Prolog Programming for Artificial Intelligence.* (Pearson, 2001)

[7] K. Broda, S. Eisenbach, H. Khoshnevisan y S. Vickers *Reasoned Programming.* (Imperial College, 1994)

[8] C.–L. Chang y R.C.–T. Lee *Symbolic Logic and Mechanical Theorem Proving* (Academic Press, 1973).

[9] J. Cuena *Lógica Informática* (Alianza Ed., 1985)

[10] J.A. Díez *Iniciación a la Lógica* (Ed. Ariel, 2002)

[11] J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional.* (UNED, 2003)

[12] M. Fitting *First–Order Logic and Automated Theorem Proving (2nd ed.).* (Springer, 1996)

[13] J.H. Gallier *Logic for Computer Science (Foundations of Automatic Theorem Proving).* (Wiley, 1986)

[14] M. Genesereth *Computational Logic* (Stanford University, 2003)

[15] S. Hölldobler *Computational logic.* (U. de Dresden, 2004)

[16] Hortalá, M.T.; Leach, J. y Rogríguez, M. *Matemática discreta y lógica matemática* (Ed. Complutense, 1998)

[17] M. Huth y M. Ryan *Logic in Computer Science: Modelling and Reasoning about Systems* (Cambridge University Press, 2000)

[18] L. de Ledesma *Lógica para la Computación (Teorías de primer orden, resolución y elementos de programación lógica y Prolog)*. (RaMa, 2009).

[19] M. Manzano y A. Huertas *Lógica para principiantes* (Alianza editorial, 2004)

[20] A. Nerode y R.A. Shore *Logic for Applications*. (Springer, 1997)

[21] N.J. Nilsson *Inteligencia artificial (Una nueva síntesis)* (McGraw–Hill, 2001).

[22] M. Ojeda e I. Pérez de Guzmán *Lógica para la computación* (Ágora, 1997)

[23] E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)

[24] L. Paulson *Logic and proof* (U. Cambridge, 2011)

[25] U. Schöning *Logic for Computer Scientists*. (Birkäuser, 1989)