

Temas de “Lógica informática” (2003–04)

José A. Alonso Jiménez
Andrés Cordon Franco




Grupo de Lógica Computacional
Dpto. de Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla
Sevilla, 30 de Junio de 2004

Esta obra está bajo una licencia Reconocimiento–NoComercial–CompartirIgual 2.5 Spain de Creative Commons.

Se permite:

- copiar, distribuir y comunicar públicamente la obra
- hacer obras derivadas

Bajo las condiciones siguientes:

-  **Reconocimiento.** Debe reconocer los créditos de la obra de la manera especificada por el autor.
-  **No comercial.** No puede utilizar esta obra para fines comerciales.
-  **Compartir bajo la misma licencia.** Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.
 - Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
 - alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor-

Esto es un resumen del texto legal (la licencia completa). Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-nc-sa/2.5/es/> o envíe una carta a Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Índice general

1. Sintaxis y semántica de la lógica proposicional	5
2. Equivalencias y formas normales	19
3. Tableros semánticos	29
4. Lógica clausal. Resolución proposicional	37
5. Deducción natural proposicional	47
6. Sintaxis y semántica de la lógica de primer orden	65
7. Deducción natural en lógica de primer orden	99
8. Formas normales. Cláusulas	113
9. Modelos de Herbrand	127
10. Resolución en lógica de primer orden	139

Capítulo 1

Sintaxis y semántica de la lógica proposicional

Tema 1: Sintaxis y semántica de la lógica proposicional

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

Lógica

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

Argumentos y formalización

- 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:

Símbolo	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 .
- $p \wedge \neg q \rightarrow r, \neg r, p \models q$.

Sintaxis proposicional: Fórmulas proposicionales

- 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.
 - * Si F y G son fórmulas, entonces $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$, $(F \leftrightarrow G)$ son fórmulas.
- Ejemplos:
 - * Fórmulas: p , $(p \vee \neg q)$, $\neg(p \vee p)$, $((p \rightarrow q) \vee (q \rightarrow p))$
 - * No fórmulas: (p) , $p \vee \neg q$, $(p \vee \wedge q)$

Sintaxis proposicional: Omisión de paréntesis

- 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: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.

$F \wedge G \rightarrow \neg F \vee G$ es una abreviatura de $((F \wedge G) \rightarrow (\neg F \vee G))$

- 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 $((F \wedge (G \wedge H)) \rightarrow (\neg F \vee G))$

Sintaxis proposicional: Subfórmulas

- Subfórmulas:

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

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

- Ejemplos:

* $\text{Subf}(p) = \{p\}$

* $\text{Subf}(q) = \{q\}$

* $\text{Subf}(\neg q) = \{\neg q, q\}$

* $\text{Subf}(\neg q \vee p) = \{\neg q \vee p, \neg q, q, p\}$

* $\text{Subf}(p \rightarrow \neg q \vee p) = \{p \rightarrow \neg q \vee p, p, \neg q \vee p, \neg q, q\}$

Semántica proposicional: valores y funciones de verdad

- Valores de verdad (\mathbb{B}):

- 1: verdadero
- 0: falso

- Funciones de verdad:

- $H_{\neg} : \{0, 1\} \rightarrow \{0, 1\}$ t.q. $H_{\neg}(i) = \begin{cases} 1, & \text{si } i = 0; \\ 0, & \text{si } i = 1. \end{cases}$

- $H_{\wedge} : \{0, 1\}^2 \rightarrow \{0, 1\}$ t.q. $H_{\wedge}(i, j) = \begin{cases} 1, & \text{si } i = j = 1; \\ 0, & \text{en otro caso.} \end{cases}$

- $H_{\vee} : \{0, 1\}^2 \rightarrow \{0, 1\}$ t.q. $H_{\vee}(i, j) = \begin{cases} 0, & \text{si } i = j = 0; \\ 1, & \text{en otro caso.} \end{cases}$

- $H_{\rightarrow} : \{0, 1\}^2 \rightarrow \{0, 1\}$ t.q. $H_{\rightarrow}(i, j) = \begin{cases} 0, & \text{si } i = 1, j = 0; \\ 1, & \text{en otro caso.} \end{cases}$

- $H_{\leftrightarrow} : \{0, 1\}^2 \rightarrow \{0, 1\}$ t.q. $H_{\leftrightarrow}(i, j) = \begin{cases} 1, & \text{si } i = j; \\ 0, & \text{en otro caso.} \end{cases}$

Semántica proposicional: valoración de fórmulas

- Funciones de verdad mediante tablas de verdad:

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

- Valoración de verdad:

- Def.: Una valoración de verdad es una aplicación $v : VP \rightarrow \mathbb{B}$.

- Prop: Para cada valoración de verdad v existe una única aplicación $\hat{v} : \text{PROP} \rightarrow \mathbb{B}$ tal que:

$$\hat{v}(F) = \begin{cases} v(F), & \text{si } F \text{ es una variable;} \\ H_{\neg}(\hat{v}(G)), & \text{si } F = \neg G; \\ H_{*}(\hat{v}(G), \hat{v}(H)), & \text{si } F = G * H \end{cases}$$

Se dice que $\hat{v}(F)$ es el valor de verdad de F respecto de v .

Semántica proposicional: valoración de fórmulas

- Ejemplo: Sea $F = (p \vee q) \wedge (\neg q \vee r)$

- valor de F en una valoración v_1 tal que $v_1(p) = v_1(r) = 1, v_1(q) = 0$

$$\begin{array}{l} (p \vee q) \wedge (\neg q \vee r) \\ (1 \vee 0) \wedge (\neg 0 \vee 1) \\ 1 \quad \wedge \quad (1 \vee 1) \\ 1 \quad \wedge \quad 1 \\ 1 \end{array}$$

- valor de F en una valoración v_2 tal que $v_2(r) = 1, v_2(p) = v_2(q) = 0$

$$\begin{array}{l} (p \vee q) \wedge (\neg q \vee r) \\ (0 \vee 0) \wedge (\neg 0 \vee 1) \\ 0 \quad \wedge \quad (1 \vee 1) \\ 0 \quad \wedge \quad 1 \\ 0 \end{array}$$

- Prop.: Sea F una fórmula y v, v' dos valoraciones. Si $v(p) = v'(p)$ para todas las variables proposicionales de F , entonces $\hat{v}(F) = \hat{v}'(F)$.
- Notación: Se escribe $v(F)$ en lugar de $\hat{v}(F)$.

Semántica proposicional: modelos y satisfacibilidad

- Modelo de una fórmula

- Def.: v es modelo de F si $v(F) = 1$.

- Notación: $v \models F$.

- Ejemplo (continuación del anterior):

- si $v_1(p) = v_1(r) = 1, v_1(q) = 0$, entonces $v_1 \models (p \vee q) \wedge (\neg q \vee r)$
- si $v_2(r) = 1, v_2(p) = v_2(q) = 0$, entonces $v_2 \not\models (p \vee q) \wedge (\neg q \vee r)$

- Fórmulas satisfacibles e insatisfacibles

- Def.: F es satisfacible si F tiene algún modelo.

- Ejemplo: $(p \rightarrow q) \wedge (q \rightarrow r)$ es satisfacible

$$v(p) = v(q) = v(r) = 0$$

- Def.: F es insatisfacible si F no tiene ningún modelo.

- Ejemplo: $p \wedge \neg p$ es insatisfacible

p	$\neg p$	$p \wedge \neg p$
1	0	0
0	1	0

Semántica proposicional: tautologías y contradicciones

- Tautologías y contradicciones:

- Def.: F es una tautología si toda valoración es modelo de F .
Se representa por $\models F$.
- Def.: F es una contradicción si ninguna valoración es modelo de F .
- Def.: F es contingente si no es tautología ni contradicción.
- Ejemplos:
 1. $(p \rightarrow q) \vee (q \rightarrow p)$ es una tautología.
 2. $(p \rightarrow q) \wedge \neg(p \rightarrow q)$ es una contradicción.
 3. $p \rightarrow q$ es contingente.

p	q	$p \rightarrow q$	$q \rightarrow p$	$(p \rightarrow q) \vee (q \rightarrow p)$	$\neg(p \rightarrow q)$	$(p \rightarrow q) \wedge \neg(p \rightarrow q)$
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

Semántica proposicional: Clasificaciones de fórmulas

- Clasificaciones de fórmulas:

Todas las fórmulas		
<i>Tautologías</i>	<i>Contingentes</i>	<i>Contradicciones</i>
Verdadera en todas las valoraciones (ej. $p \vee \neg p$)	Verdadera en algunas valoraciones y falsa en otras (ej. $p \rightarrow q$)	Falsa en todas las valoraciones (ej. $p \wedge \neg p$)
<i>Satisfacibles</i>		<i>Insatisfacibles</i>
Todas las fórmulas		

Semántica proposicional: Algoritmos para SAT y TAUT

- Algoritmos de decisión para SAT y TAUT:

- Tablas de verdad para $\not\equiv (p \leftrightarrow q) \vee (q \leftrightarrow p)$

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

- Método de Quine para $\not\equiv (p \leftrightarrow q) \vee (q \leftrightarrow p)$

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

Semántica proposicional: selección de tautologías

- Selección de tautologías

- 1. $F \rightarrow F$ (ley de identidad).
- 2. $F \vee \neg F$ (ley del tercio excluso).
- 3. $\neg(F \wedge \neg F)$ (principio de no contradicción).
- 4. $(\neg F \rightarrow F) \rightarrow F$ (ley de Clavius).
- 5. $\neg F \rightarrow (F \rightarrow G)$ (ley de Duns Scoto).
- 6. $((F \rightarrow G) \rightarrow F) \rightarrow F$ (ley de Peirce).
- 7. $(F \rightarrow G) \wedge F \rightarrow G$ (modus ponens).
- 8. $(F \rightarrow G) \wedge \neg G \rightarrow \neg F$ (modus tollens).

Semántica proposicional: Modelo de conjuntos de fórmulas

• Notación:

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

• Modelo de un conjunto de fórmulas:

- Def.: v es modelo de S si para toda $F \in S$ se tiene que $v \models F$.

- Representación: $v \models S$.

- Ejemplo: Sea $S = \{(p \vee q) \wedge (\neg q \vee r), q \rightarrow r\}$

La valoración v_1 tal que $v_1(p) = 1, v_1(q) = 0, v_1(r) = 1$ es modelo de S ($v_1 \models S$).

$$\{(p \vee q) \wedge (\neg q \vee r), q \rightarrow r\}$$

$$1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1$$

La valoración v_2 tal que $v_2(p) = 0, v_2(q) = 1, v_2(r) = 0$ no es modelo de S ($v_2 \not\models S$)

$$\{(p \vee q) \wedge (\neg q \vee r), q \rightarrow r\}$$

$$0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0$$

Semántica proposicional: Consistencia

• Conjunto consistente de fórmulas:

- Def.: S es consistente si S tiene algún modelo.

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

• Ejemplos:

- $\{(p \vee q) \wedge (\neg q \vee r), p \rightarrow r\}$ es consistente (con modelos v_4, v_6, v_8)

- $\{(p \vee q) \wedge (\neg q \vee r), p \rightarrow r, \neg r\}$ es inconsistente

	p	q	r	$(p \vee q)$	$(\neg q \vee r)$	$(p \vee q) \wedge (\neg q \vee r)$	$p \rightarrow r$	$\neg r$
v_1	0	0	0	0	1	0	1	1
v_2	0	0	1	0	1	0	1	0
v_3	0	1	0	1	0	0	1	1
v_4	0	1	1	1	1	1	1	0
v_5	1	0	0	1	1	1	0	1
v_6	1	0	1	1	1	1	1	0
v_7	1	1	0	1	0	0	0	1
v_8	1	1	1	1	1	1	1	0

Semántica proposicional: Consecuencia lógica

- **Consecuencia lógica:**

- Def.: F es consecuencia de S si todos los modelos de S son modelos de F .
- Representación: $S \models F$.

- Ejemplo: $\{p \rightarrow q, q \rightarrow r\} \models p \rightarrow r$

	p	q	r	$p \rightarrow q$	$q \rightarrow r$	$p \rightarrow r$
v_1	0	0	0	1	1	1
v_2	0	0	1	1	1	1
v_3	0	1	0	1	0	1
v_4	0	1	1	1	1	1
v_5	1	0	0	0	1	0
v_6	1	0	1	0	1	1
v_7	1	1	0	1	0	0
v_8	1	1	1	1	1	1

- Ejemplo: $\{p\} \not\models p \wedge q$

p	q	$p \wedge q$
1	1	1
1	0	0
0	1	0
0	0	0

Semántica: 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. $\models F_1 \wedge \dots \wedge F_n \rightarrow G$
3. $\neg(F_1 \wedge \dots \wedge F_n \rightarrow G)$ es insatisfacible
4. $\{F_1, \dots, F_n, \neg G\}$ es inconsistente

Semántica proposicional: argumentaciones

• Ejemplo de argumentación:

- Problema de los animales: Se sabe que
 1. Los animales con pelo 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:

$$\{ \text{tiene_pelos} \vee \text{da_leche} \rightarrow \text{es_mamífero},$$

$$\text{es_mamífero} \wedge (\text{tiene_pezuñas} \vee \text{rumia}) \rightarrow \text{es_ungulado},$$

$$\text{es_ungulado} \wedge \text{tiene_cuello_largo} \rightarrow \text{es_jirafa},$$

$$\text{es_ungulado} \wedge \text{tiene_rayas_negras} \rightarrow \text{es_cebra},$$

$$\text{tiene_pelos} \wedge \text{tiene_pezuñas} \wedge \text{tiene_rayas_negras} \}$$

$$\models \text{es_cebra}$$

Problemas lógicos

• El problema de los veraces y los 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:

$$F_1 = a \leftrightarrow (b \wedge c \leftrightarrow c), F_2 = b \leftrightarrow (a \wedge c \rightarrow b \wedge c \wedge \neg a) \text{ y } F_3 = c \leftrightarrow (\neg b \leftrightarrow a \vee b)$$
• Modelos de $\{F_1, F_2, F_3\}$:

Si v es modelo de $\{F_1, F_2, F_3\}$, entonces $v(a) = 1, v(b) = 1, v(c) = 0$

- Conclusión: A y B son veraces y C es mentiroso.

Bibliografía

- 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).
- M. Ben-Ari, *Mathematical logic for computer science (2nd ed.)*. (Springer, 2001)
Cap. 1 (Introduction) y 2 (Propositional calculus: formulas, models, tableaux).
- J.A. Díez *Iniciación a la Lógica*, (Ariel, 2002)
Cap. 2 (El lenguaje de la lógica proposicional) y 3 (Semántica formal. Consecuencia lógica).
- M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems*. (Cambridge University Press, 2000)
Cap. 1 (Propositional logic).

Bibliografía

- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)
Cap. 1 (La sintaxis de la Lógica) y Cap. 2 (La semántica de la Lógica).

Capítulo 2

Equivalencias y formas normales

Tema 2: Equivalencias y formas normales

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial

UNIVERSIDAD DE SEVILLA

Equivalencia lógica

- **Fórmulas equivalentes**

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

Equivalencia lógica: propiedades

- Relación entre equivalencia y bicondicional:
 - $F \equiv G \text{ sys} \models F \leftrightarrow G$
- 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:

$$\begin{aligned} F &= \neg(p \wedge q) \rightarrow \neg(p \wedge \neg\neg r) \\ G &= \neg(p \wedge q) \\ G' &= \neg p \vee \neg q \\ F' &= (\neg p \vee \neg q) \rightarrow \neg(p \wedge \neg\neg r) \end{aligned}$$

Formas normales: Forma normal conjuntiva

- Átomos y literales:
 - Def.: Un átomo es un variable proposicional (p.e. p, q, \dots).
 - Def.: Un literal es un átomo o su negación (p.e. $p, \neg p, q, \neg q, \dots$).
 - Notación: L, L_1, L_2, \dots representarán literales.
- Forma normal conjuntiva:
 - Def.: Una fórmula está en forma normal conjuntiva (FNC) si es una conjunción de disyunciones de literales; es decir, es de la forma

$$(L_{1,1} \vee \dots \vee L_{1,n_1}) \wedge \dots \wedge (L_{m,1} \vee \dots \vee L_{m,n_m}).$$
 - Ejemplos: $(\neg p \vee q) \wedge (\neg q \vee p)$ está en FNC
 $(\neg p \vee q) \wedge (q \rightarrow p)$ no está en FNC
 - Def.: Una fórmula G es una forma normal conjuntiva (FNC) de la fórmula F si G está en forma normal conjuntiva y es equivalente a F .
 - Ejemplo: Una FNC de $\neg(p \wedge (q \rightarrow r))$ es $(\neg p \vee q) \wedge (\neg p \vee \neg r)$.

Formas normales: Cálculo de forma normal conjuntiva

- Algoritmo de cálculo de forma normal conjuntiva:

- Algoritmo: Aplicando a una fórmula F los siguientes pasos se obtiene una forma normal conjuntiva de F :

1. Eliminar los bicondicionales usando la equivalencia

$$A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A) \quad (1)$$

2. Eliminar los condicionales usando la equivalencia

$$A \rightarrow B \equiv \neg A \vee B \quad (2)$$

3. Interiorizar las negaciones usando las equivalencias

$$\neg(A \wedge B) \equiv \neg A \vee \neg B \quad (3)$$

$$\neg(A \vee B) \equiv \neg A \wedge \neg B \quad (4)$$

$$\neg\neg A \equiv A \quad (5)$$

4. Interiorizar las disyunciones usando las equivalencias

$$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C) \quad (6)$$

$$(A \wedge B) \vee C \equiv (A \vee C) \wedge (B \vee C) \quad (7)$$

Formas normales: Cálculo de forma normal conjuntiva

- Ejemplo de cálculo de una FNC de $\neg(p \wedge (q \rightarrow r))$:

$$\begin{aligned} & \neg(p \wedge (q \rightarrow r)) \\ \equiv & \neg(p \wedge (\neg q \vee r)) && [\text{por (2)}] \\ \equiv & \neg p \vee \neg(\neg q \vee r) && [\text{por (3)}] \\ \equiv & \neg p \vee (\neg\neg q \wedge \neg r) && [\text{por (4)}] \\ \equiv & \neg p \vee (q \wedge \neg r) && [\text{por (5)}] \\ \equiv & (\neg p \vee q) \wedge (\neg p \vee \neg r) && [\text{por (6)}] \end{aligned}$$

- Ejemplo de cálculo de una FNC de $(p \rightarrow q) \vee (q \rightarrow p)$:

$$\begin{aligned} & (p \rightarrow q) \vee (q \rightarrow p) \\ \equiv & (\neg p \vee q) \vee (\neg q \vee p) && [\text{por (2)}] \\ \equiv & \neg p \vee q \vee \neg q \vee p \end{aligned}$$

Formas normales: Cálculo de forma normal conjuntiva

- Ejemplo de cálculo de una FNC de $(p \leftrightarrow q) \rightarrow r$:

$$\begin{aligned}
 & (p \leftrightarrow q) \rightarrow r \\
 \equiv & (p \rightarrow q) \wedge (q \rightarrow p) \rightarrow r && \text{[por (1)]} \\
 \equiv & \neg((\neg p \vee q) \wedge (\neg q \vee p)) \vee r && \text{[por (2)]} \\
 \equiv & (\neg(\neg p \vee q) \vee \neg(\neg q \vee p)) \vee r && \text{[por (3)]} \\
 \equiv & ((\neg\neg p \wedge \neg q) \vee (\neg\neg q \wedge \neg p)) \vee r && \text{[por (4)]} \\
 \equiv & ((p \wedge \neg q) \vee (q \wedge \neg p)) \vee r && \text{[por (5)]} \\
 \equiv & (((p \wedge \neg q) \vee q) \wedge ((p \wedge \neg q) \vee \neg p)) \vee r && \text{[por (6)]} \\
 \equiv & (((p \vee q) \wedge (\neg q \vee q)) \wedge ((p \vee \neg p) \wedge (\neg q \vee \neg p))) \vee r && \text{[por (7)]} \\
 \equiv & (((p \vee q) \wedge (\neg q \vee q)) \vee r) \wedge (((p \vee \neg p) \wedge (\neg q \vee \neg p)) \vee r) && \text{[por (7)]} \\
 \equiv & (((p \vee q) \vee r) \wedge ((\neg q \vee q) \vee r)) \wedge (((p \vee \neg p) \vee r) \wedge ((\neg q \vee \neg p) \vee r)) && \text{[por (7)]} \\
 \equiv & (p \vee q \vee r) \wedge (\neg q \vee q \vee r) \wedge (p \vee \neg p \vee r) \wedge (\neg q \vee \neg p \vee r) \\
 \equiv & (p \vee q \vee r) \wedge (\neg q \vee \neg p \vee r)
 \end{aligned}$$

Formas normales: Forma normal disyuntiva

- Forma normal disyuntiva:

- Def.: Una fórmula está en forma normal disyuntiva (FND) si es una disyunción de conjunciones de literales; es decir, es de la forma

$$(L_{1,1} \wedge \dots \wedge L_{1,n_1}) \vee \dots \vee (L_{m,1} \wedge \dots \wedge L_{m,n_m}).$$

- Ejemplos: $(\neg p \wedge q) \vee (\neg q \wedge p)$ está en FNC
 $(\neg p \wedge q) \vee (q \rightarrow p)$ no está en FNC
- Def.: Una fórmula G es una forma normal disyuntiva (FND) de la fórmula F si G está en forma normal disyuntiva y es equivalente a F .
- Ejemplo: Una FND de $\neg(p \wedge (q \rightarrow r))$ es $\neg p \vee (q \wedge \neg r)$.

Formas normales: Cálculo de forma normal disyuntiva

- Algoritmo de cálculo de forma normal disyuntiva:

- Algoritmo: Aplicando a una fórmula F los siguientes pasos se obtiene una forma normal disyuntiva de F :

1. Eliminar los bicondicionales usando la equivalencia

$$A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A) \quad (1)$$

2. Eliminar los condicionales usando la equivalencia

$$A \rightarrow B \equiv \neg A \vee B \quad (2)$$

3. Interiorizar las negaciones usando las equivalencias

$$\neg(A \wedge B) \equiv \neg A \vee \neg B \quad (3)$$

$$\neg(A \vee B) \equiv \neg A \wedge \neg B \quad (4)$$

$$\neg\neg A \equiv A \quad (5)$$

4. Interiorizar las conjunciones usando las equivalencias

$$A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C) \quad (6)$$

$$(A \vee B) \wedge C \equiv (A \wedge C) \vee (B \wedge C) \quad (7)$$

Formas normales: Cálculo de forma normal disyuntiva

- Ejemplo de cálculo de una FND de $\neg(p \wedge (q \rightarrow r))$:

$$\begin{aligned} & \neg(p \wedge (q \rightarrow r)) \\ \equiv & \neg(p \wedge (\neg q \vee r)) \quad [\text{por (2)}] \\ \equiv & \neg p \vee \neg(\neg q \vee r) \quad [\text{por (3)}] \\ \equiv & \neg p \vee (\neg\neg q \wedge \neg r) \quad [\text{por (4)}] \\ \equiv & \neg p \vee (q \wedge \neg r) \quad [\text{por (5)}] \end{aligned}$$

- Ejemplo de cálculo de una FND de $\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q))$:

$$\begin{aligned} & \neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q)) \\ \equiv & \neg(\neg(\neg p \vee \neg q) \vee \neg(p \wedge q)) \quad [\text{por (2)}] \\ \equiv & \neg\neg(\neg p \vee \neg q) \wedge \neg\neg(p \wedge q) \quad [\text{por (4)}] \\ \equiv & (\neg p \vee \neg q) \wedge (p \wedge q) \quad [\text{por (5)}] \\ \equiv & (\neg p \wedge (p \wedge q)) \vee (\neg q \wedge (p \wedge q)) \quad [\text{por (7)}] \\ \equiv & (\neg p \wedge p \wedge q) \vee (\neg q \wedge p \wedge q) \end{aligned}$$

Decisión de tautologicidad mediante FNC

- Literales complementarios:
 - El complementario de un literal L es $L^c = \begin{cases} \neg p & \text{si } L = p; \\ p & \text{si } L = \neg p \end{cases}$
- Propiedades de reducción de tautologías
 - $F_1 \wedge \dots \wedge F_n$ es una tautología syss F_1, \dots, F_n lo son.
 - $L_1 \vee \dots \vee L_n$ es una tautología syss $\{L_1, \dots, L_n\}$ contiene algún par de literales complementarios (i.e. existen i, j tales que $L_i = L_j^c$)
- Decisión de tautologías mediante FNC
 - Entrada: Una fórmula F .
 - Procedimiento:
 1. Calcular una FNC de F .
 2. Decidir si cada una de las conjunciones de la FNC tiene algún par de literales complementarios.

Decisión de tautologicidad mediante FNC

- Ejemplos de decisión de tautologías mediante FNC
 - $\neg(p \wedge (q \rightarrow r))$ no es tautología

$$\text{FNC}(\neg(p \wedge (q \rightarrow r))) = (\neg p \vee q) \wedge (\neg p \vee \neg r)$$
 Contramodelos de $\neg(p \wedge (q \rightarrow r))$:
 - v_1 tal que $v_1(p) = 1$ y $v_1(q) = 0$
 - v_2 tal que $v_2(p) = 1$ y $v_2(r) = 1$
 - $(p \rightarrow q) \vee (q \rightarrow p)$ es tautología:

$$\text{FNC}((p \rightarrow q) \vee (q \rightarrow p)) = \neg p \vee q \vee \neg q \vee p$$
 - $(p \leftrightarrow q) \rightarrow r$ no es tautología:

$$\text{FNC}((p \leftrightarrow q) \rightarrow r) = (p \vee q \vee r) \wedge (\neg q \vee \neg p \vee r)$$
 Contramodelos de $(p \leftrightarrow q) \rightarrow r$:
 - v_1 tal que $v_1(p) = 0, v_1(q) = 0$ y $v_1(r) = 0$
 - v_2 tal que $v_2(p) = 1, v_2(q) = 1$ y $v_2(r) = 0$

Decisión de satisfacibilidad mediante FND

- **Propiedades de reducción de satisfacibilidad**
 - $F_1 \vee \dots \vee F_n$ es satisfacible syss alguna de las fórmulas F_1, \dots, F_n lo es.
 - $L_1 \wedge \dots \wedge L_n$ es satisfacible syss $\{L_1, \dots, L_n\}$ no contiene ningún par de literales complementarios.
- **Decisión de satisfacibilidad mediante FND**
 - **Entrada:** Una fórmula F .
 - **Procedimiento:**
 1. Calcular una FND de F .
 2. Decidir si alguna de las disyunciones de la FND no tiene un par de literales complementarios.

Decisión de satisfacibilidad mediante FND

- **Ejemplos de decisión de satisfacibilidad mediante FND**
 - $\neg(p \wedge (q \rightarrow r))$ es satisfacible:

$$\text{FND}(\neg(p \wedge (q \rightarrow r))) = \neg p \vee (q \wedge \neg r)$$
 Modelos de $\neg(p \wedge (q \rightarrow r))$:
 - v_1 tal que $v_1(p) = 0$
 - v_2 tal que $v_2(q) = 1$ y $v_2(r) = 0$
 - $\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q))$ es insatisfacible:

$$\text{FND}(\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q))) = (\neg p \wedge p \wedge q) \vee (\neg q \wedge p \wedge q)$$

Bibliografía

- C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal*. (Ariel, 2000)
Cap. 8 (Equivalencia lógica) y 10 (Formas normales).
- M. Ben–Ari, *Mathematical logic for computer science (2nd ed.)*. (Springer, 2001)
Cap. 2 (Propositional calculus: formulas, models, tableaux).
- J.A. Díez *Iniciación a la Lógica*, (Ariel, 2002)
Cap. 3 (Semántica formal. Consecuencia lógica).
- M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems*. (Cambridge University Press, 2000)
Cap. 1 (Propositional logic).
- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)
Cap. 4.4 (Formas normales).

Capítulo 3

Tableros semánticos

Tema 3: Tableros semánticos

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial

UNIVERSIDAD DE SEVILLA

Demostración por tableros semánticos

- Demostración de fórmula tautológica:

- Demostración:

$\neg p \vee \neg q \rightarrow \neg(p \wedge q)$ es una tautología

syss $\{\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q))\}$ es inconsistente

syss $\{\neg p \vee \neg q, \neg\neg(p \wedge q)\}$ es inconsistente

syss $\{\neg p \vee \neg q, p \wedge q\}$ es inconsistente

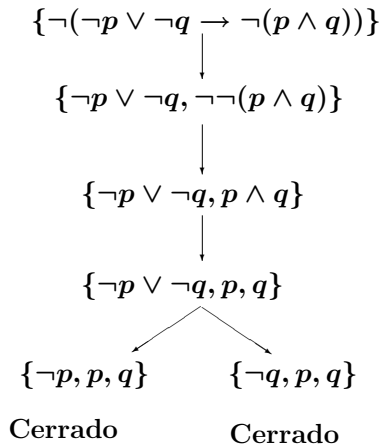
syss $\{p, q, \neg p \vee \neg q\}$ es inconsistente

syss $\{p, q, \neg p\}$ es inconsistente y

$\{p, q, \neg q\}$ es inconsistente

Demostración por tableros semánticos

- Tablero semántico cerrado:



Refutación por tableros semánticos

- Refutación de fórmula no tautológica:

- Refutación:

$\neg p \vee \neg q \rightarrow \neg(p \wedge r)$ es una tautología
 syss $\{\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge r))\}$ es inconsistente
 syss $\{\neg p \vee \neg q, \neg\neg(p \wedge r)\}$ es inconsistente
 syss $\{\neg p \vee \neg q, p \wedge r\}$ es inconsistente
 syss $\{p, r, \neg p \vee \neg q\}$ es inconsistente
 syss $\{p, r, \neg p\}$ es inconsistente y
 $\{p, r, \neg q\}$ es inconsistente

- Contramodelos de $\neg p \vee \neg q \rightarrow \neg(p \wedge r)$:

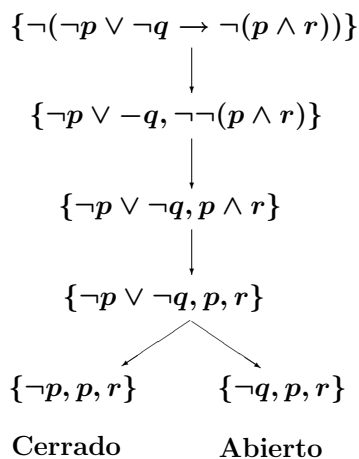
Las valoraciones v tales que $v(p) = 1$, $v(q) = 0$ y $v(r) = 1$.

- Una forma normal disyuntiva de $\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge r))$:

$$p \vee r \vee \neg q$$

Refutación por tableros semánticos

- Tablero semántico:



Notación uniforme: Literales y dobles negaciones

- Literales
 - Un literal es un átomo o la negación de un átomo (p.e. $p, \neg p, q, \neg q, \dots$).
- Dobles negaciones
 - F es una doble negación si es de la forma $\neg\neg G$.
 - Ley de doble negación: Si F es $\neg\neg G$, entonces $F \equiv G$.

Notación uniforme: fórmulas alfa y beta

• Fórmulas alfa

- Las fórmulas alfa, junto con sus componentes, son

F	F_1	F_2
$A_1 \wedge A_2$	A_1	A_2
$\neg(A_1 \rightarrow A_2)$	A_1	$\neg A_2$
$\neg(A_1 \vee A_2)$	$\neg A_1$	$\neg A_2$
$A_1 \leftrightarrow A_2$	$A_1 \rightarrow A_2$	$A_2 \rightarrow A_1$

- Si F es una fórmula alfa y sus componentes son F_1 y F_2 , entonces $F \equiv F_1 \wedge F_2$.

• Fórmulas beta

- Las fórmulas beta, junto con sus componentes, son

F	F_1	F_2
$B_1 \vee B_2$	B_1	B_2
$B_1 \rightarrow B_2$	$\neg B_1$	B_2
$\neg(B_1 \wedge B_2)$	$\neg B_1$	$\neg B_2$
$\neg(B_1 \leftrightarrow B_2)$	$\neg(B_1 \rightarrow B_2)$	$\neg(B_2 \rightarrow B_1)$

- Si F es una fórmula beta y sus componentes son F_1 y F_2 , entonces $F \equiv F_1 \vee F_2$.

Completación de tableros

- Procedimiento de completación de tableros:

Un tablero de un conjunto de fórmulas S es un árbol construido mediante las reglas:

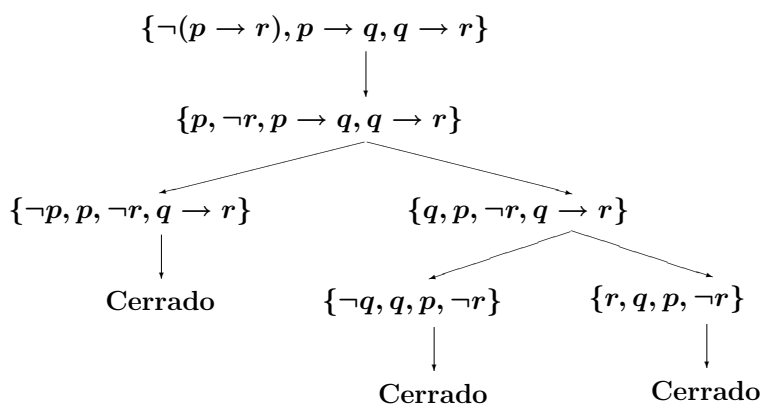
- * El árbol cuyo único nodo tiene como etiqueta S es un tablero de S .
- * Sea \mathcal{T} un tablero de S y S_1 la etiqueta de una hoja de \mathcal{T} .
 1. Si S_1 cerrado (es decir, es un conjunto de literales que contiene una fórmula y su negación), entonces el árbol obtenido añadiendo como hijo de S_1 el nodo etiquetado con cerrado es un tablero de S .
 2. Si S_1 es abierto (es decir, es un conjunto de literales que no contiene una fórmula y su negación), entonces el árbol obtenido añadiendo como hijo de S_1 el nodo etiquetado con abierto es un tablero de S .
 3. Si S_1 contiene una doble negación $\neg\neg F$, entonces el árbol obtenido añadiendo como hijo de S_1 el nodo etiquetado con $(S_1 \setminus \{\neg\neg F\}) \cup \{F\}$ es un tablero de S .
 4. Si S_1 contiene una fórmula alfa F de componentes F_1 y F_2 , entonces el árbol obtenido añadiendo como hijo de S_1 el nodo etiquetado con $(S_1 \setminus \{F\}) \cup \{F_1, F_2\}$ es un tablero de S .
 5. Si S_1 contiene una fórmula beta F de componentes F_1 y F_2 , entonces el árbol obtenido añadiendo como hijos de S_1 los nodos etiquetados con $(S_1 \setminus \{F\}) \cup \{F_1\}$ y $(S_1 \setminus \{F\}) \cup \{F_2\}$ es un tablero de S .

Teorema por tableros

- Def.: Un tablero completo de S es un tablero de S al que no se le puede aplicar ninguna de las reglas de expansión; es decir, todas sus hojas son abiertas o cerradas.
- Def.: Un tablero es cerrado si todas sus hojas están etiquetadas con cerrado.
- Def.: Una fórmula F es un teorema (mediante tableros semánticos) si tiene una prueba mediante tableros; es decir, si $\{\neg F\}$ tiene un tablero completo cerrado. Se representa por $\vdash_{Tab} F$.
- Ejemplos:
 - $\vdash_{Tab} \neg p \vee \neg q \rightarrow \neg(p \wedge q)$
 - $\not\vdash_{Tab} \neg p \vee \neg q \rightarrow \neg(p \wedge r)$
- Teor.: El cálculo de tableros semánticos es adecuado y completo; es decir,
 - Adecuado: $\vdash_{Tab} F \implies \models F$
 - Completo: $\models F \implies \vdash_{Tab} F$
- Tableros y formas normales disyuntivas: Si $\{L_{1,1}, \dots, L_{1,n_1}\}, \dots, \{L_{m,1}, \dots, L_{m,n_m}\}$ son los nodos abiertos del tablero completo de F , entonces una forma normal disyuntiva de F es $(L_{1,1} \wedge \dots \wedge L_{1,n_1}) \vee \dots \vee (L_{m,1} \wedge \dots \wedge L_{m,n_m})$.

Deducción por tableros

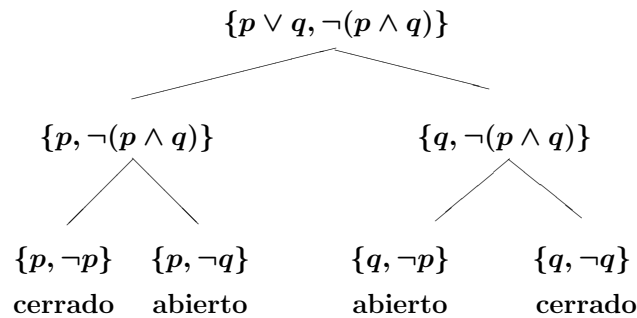
- Def.: La fórmula F es deducible (mediante tableros semánticos) a partir del conjunto de fórmulas S si existe una prueba mediante tableros de F a partir de S ; es decir, existe un tablero completo cerrado de $S \cup \{\neg F\}$. Se representa por $S \vdash_{Tab} F$.
- Ejemplo: $\{p \rightarrow q, q \rightarrow r\} \vdash_{Tab} p \rightarrow r$



- Teor.: $S \vdash_{Tab} F$ syss $S \models F$.

Deducción por tableros

- Ejemplo: $\{p \vee q\} \not\vdash_{Tab} p \wedge q$



- Contramodelos de $\{p \vee q\} \not\vdash_{Tab} p \wedge q$
 - las valoraciones v_1 tales que $v_1(p) = 1$ y $v_1(q) = 0$
 - las valoraciones v_2 tales que $v_2(p) = 0$ y $v_2(q) = 1$

Bibliografía

- Ben-Ari, M. *Mathematical Logic for Computer Science (2nd ed.)* (Springer, 2001)
 - Cap. 2: Propositional calculus: formulas, models, tableaux
- Fitting, M. *First-Order Logic and Automated Theorem Proving (2nd ed.)* (Springer, 1995)
 - Cap. 3: Semantic tableaux and resolution
- Hortalá, M.T.; Leach, J. y Rodríguez, M. *Matemática discreta y lógica matemática* (Ed. Complutense, 1998)
 - Cap. 7.9: Tableaux semánticos par la lógica de proposiciones
- Nerode, A. y Shore, R.A. *Logic for Applications* (Springer, 1997)
 - Cap. 1.4: Tableau proofs in propositional calculus
- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)
 - Cap. 4.3: Métodos de las tablas semánticas

Capítulo 4

Lógica clausal. Resolución proposicional

Tema 4: Lógica clausal. Resolución

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial

UNIVERSIDAD DE SEVILLA

Lógica clausal: sintaxis

- Sintaxis de la lógica clausal

- Un átomo es una variable proposicional.
Variables sobre átomos: $p, q, r, \dots, p_1, p_2, \dots$
- Un literal es un átomo (p) o la negación de un átomo ($\neg p$).
Variables sobre literales: L, L_1, L_2, \dots
- Una cláusula es un conjunto finito de literales.
Variables sobre cláusulas: C, C_1, C_2, \dots
- La cláusula vacía es el conjunto vacío de literales.
La cláusula vacía se representa por \square .
- Conjuntos finitos de cláusulas.
Variables sobre conjuntos finitos de cláusulas: S, S_1, S_2, \dots

Lógica clausal: semántica

- Semántica de la lógica clausal

- Def.: Una valoración de verdad es una aplicación $v : VP \rightarrow \mathbb{B}$.
- Def.: El valor de un literal positivo p en una valoración v es $v(p)$.
- Def.: El valor de un literal negativo $\neg p$ en una valoración v es

$$v(\neg p) = \begin{cases} 1, & \text{si } v(p) = 0; \\ 0, & \text{si } v(p) = 1. \end{cases}$$

- Def.: El valor de una cláusula C en una valoración v es

$$v(C) = \begin{cases} 1, & \text{si existe un } L \in C \text{ tal que } v(L) = 1; \\ 0, & \text{en caso contrario.} \end{cases}$$

- Def.: El valor de un conjunto de cláusulas S en una valoración v es

$$v(S) = \begin{cases} 1, & \text{si para toda } C \in S, v(C) = 1 \\ 0, & \text{en caso contrario.} \end{cases}$$

- Prop.: En cualquier valoración v , $v(\square) = 0$.

Cláusulas y fórmulas

- Equivalencias entre cláusulas y fórmulas

- Def.: Una cláusula C y una fórmula F son equivalentes si $v(C) = v(F)$ para cualquier valoración v .
- Def.: Un conjunto de cláusulas S y una fórmula F son equivalentes si $v(S) = v(F)$ para cualquier valoración v .
- Def.: Un conjunto de cláusulas S y un conjunto de fórmulas $\{F_1, \dots, F_n\}$ son equivalentes si, para cualquier valoración v , $v(S) = 1$ si y sólo si v es un modelo de $\{F_1, \dots, F_n\}$.

- De cláusulas a fórmulas

- Prop.: La cláusula $\{L_1, L_2, \dots, L_n\}$ es equivalente a la fórmula $L_1 \vee L_2 \vee \dots \vee L_n$.
- Prop.: El conjunto de cláusulas $\{\{L_{1,1}, \dots, L_{1,n_1}\}, \dots, \{L_{m,1}, \dots, L_{m,n_m}\}\}$ es equivalente a la fórmula $(L_{1,1} \vee \dots \vee L_{1,n_1}) \wedge \dots \wedge (L_{m,1} \vee \dots \vee L_{m,n_m})$.

Cláusulas y fórmulas

- De fórmulas a cláusulas (forma clausal)
 - Def.: Una forma clausal de una fórmula F es un conjunto de cláusulas equivalente a F .
 - Prop.: Si $(L_{1,1} \vee \dots \vee L_{1,n_1}) \wedge \dots \wedge (L_{m,1} \vee \dots \vee L_{m,n_m})$ es una forma normal conjuntiva de la fórmula F . Entonces, una forma clausal de F es $\{\{L_{1,1}, \dots, L_{1,n_1}\}, \dots, \{L_{m,1}, \dots, L_{m,n_m}\}\}$.
 - Ejemplos:
 - * Una forma clausal de $\neg(p \wedge (q \rightarrow r))$ es $\{\{\neg p, q\}, \{\neg p, \neg r\}\}$.
 - * Una forma clausal de $p \rightarrow q$ es $\{\{\neg p, q\}\}$.
 - * La cláusula $\{\{\neg p, q\}, \{r\}\}$ es una forma clausal de las fórmulas $(p \rightarrow q) \wedge r$ y $\neg \neg r \wedge (\neg q \rightarrow \neg p)$.
 - Def.: Una forma clausal de un conjunto de fórmulas S es un conjunto de cláusulas equivalente a S .
 - Prop.: Si S_1, \dots, S_n son formas clausales de F_1, \dots, F_n , entonces $S_1 \cup \dots \cup S_n$ es una forma clausal de $\{F_1, \dots, F_n\}$.

Modelos, consistencia y consecuencia

- Modelos, consistencia y consecuencia
 - Def.: Una valoración v es modelo de un conjunto de cláusulas S si $v(S) = 1$.
 - Ej.: La valoración v tal que $v(p) = v(q) = 1$ es un modelo de $\{\{\neg p, q\}, \{p, \neg q\}\}$.
 - Def.: Un conjunto de cláusulas es consistente si tiene modelos e inconsistente, en caso contrario.
 - Ejemplos:
 - * $\{\{\neg p, q\}, \{p, \neg q\}\}$ es consistente.
 - * $\{\{\neg p, q\}, \{p, \neg q\}, \{p, q\}, \{\neg p, \neg q\}\}$ es inconsistente.
 - Si $\square \in S$, entonces S es inconsistente.
 - Def.: $S \models C$ si para todo modelo v de S , $v(C) = 1$.

Reducción de consecuencia a inconsistencia de cláusulas

- Reducción de consecuencia a inconsistencia de cláusulas:
 - Prop: Sean S_1, \dots, S_n formas clausales de las fórmulas F_1, \dots, F_n .
 - * $\{F_1, \dots, F_n\}$ es consistente syss $S_1 \cup \dots \cup S_n$ es consistente.
 - * Si S es una forma clausal de $\neg G$, entonces son equivalentes
 1. $\{F_1, \dots, F_n\} \models G$.
 2. $\{F_1, \dots, F_n, \neg G\}$ es inconsistente.
 3. $S_1 \cup \dots \cup S_n \cup S$ es inconsistente.
 - Ejemplo: $\{p \rightarrow q, q \rightarrow r\} \models p \rightarrow r$ syss $\{\neg p, q\}, \{\neg q, r\}, \{p\}, \{\neg r\}$ es inconsistente.

Regla de resolución

- Reglas de inferencia:
 - Reglas habituales:

Modus Ponens:	$\frac{p \rightarrow q, \quad p}{q}$	$\frac{\{\neg p, q\}, \quad \{p\}}{\{q\}}$
Modus Tollens:	$\frac{p \rightarrow q, \quad \neg q}{\neg p}$	$\frac{\{\neg p, q\}, \quad \{\neg q\}}{\{\neg p\}}$
Encadenamiento:	$\frac{p \rightarrow q, \quad q \rightarrow r}{p \rightarrow r}$	$\frac{\{\neg p, q\}, \quad \{\neg q, r\}}{\{\neg p, r\}}$
 - Regla de resolución proposicional:

$$\frac{\{p_1, \dots, r, \dots, p_m\}, \quad \{q_1, \dots, \neg r, \dots, q_n\}}{\{p_1, \dots, p_m, q_1, \dots, q_n\}}$$

Regla de resolución

- **Resolventes**

- Def.: Sean C_1 una cláusula, L un literal de C_1 y C_2 una cláusula que contiene el complementario de L . La resolvente de C_1 y C_2 respecto de L es

$$\text{Res}_L(C_1, C_2) = (C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\})$$

- Ejemplos: $\text{Res}_q(\{p, q\}, \{\neg q, r\}) = \{p, r\}$
 $\text{Res}_q(\{q, \neg p\}, \{p, \neg q\}) = \{p, \neg p\}$
 $\text{Res}_p(\{q, \neg p\}, \{p, \neg q\}) = \{q, \neg q\}$
 $\text{Res}_p(\{q, \neg p\}, \{q, p\}) = \{q\}$
 $\text{Res}_p(\{p\}, \{\neg p\}) = \square$

- **Resolventes de dos cláusulas:**

- Def.: $\text{Res}(C_1, C_2)$ es el conjunto de las resolventes entre C_1 y C_2

- Ejemplos: $\text{Res}(\{\neg p, q\}, \{p, \neg q\}) = \{\{p, \neg p\}, \{q, \neg q\}\}$
 $\text{Res}(\{\neg p, q\}, \{p, q\}) = \{\{q\}\}$
 $\text{Res}(\{\neg p, q\}, \{q, r\}) = \emptyset$

- Nota: $\square \notin \text{Res}(\{p, q\}, \{\neg p, \neg q\})$

Demostraciones por resolución

- **Ejemplo de refutación por resolución:**

- Refutación de $\{\{p, q\}, \{\neg p, q\}, \{p, \neg q\}, \{\neg p, \neg q\}\}$:

- 1 $\{p, q\}$ Hipótesis
- 2 $\{\neg p, q\}$ Hipótesis
- 3 $\{p, \neg q\}$ Hipótesis
- 4 $\{\neg p, \neg q\}$ Hipótesis
- 5 $\{q\}$ Resolvente de 1 y 2
- 6 $\{\neg q\}$ Resolvente de 3 y 4
- 7 \square Resolvente de 5 y 6

Demostraciones por resolución

- **Definiciones**

- Sea S un conjunto de cláusulas.
- La sucesión (C_1, \dots, C_n) es una demostración por resolución de la cláusula C a partir de S si $C = C_n$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
 - * $C_i \in S$;
 - * existen $j, k < i$ tales que C_i es una resolvente de C_j y C_k
- La cláusula C es demostrable por resolución a partir de S si existe una demostración por resolución de C a partir de S .
- Una refutación por resolución de S es una demostración por resolución de la cláusula vacía a partir de S .
- Se dice que S es refutable por resolución si existe una refutación por resolución a partir de S .

Demostraciones por resolución

- **Demostraciones por resolución**

- Def.: Sean S_1, \dots, S_n formas clausales de las fórmulas F_1, \dots, F_n y S una forma clausal de $\neg F$
Una demostración por resolución de F a partir de $\{F_1, \dots, F_n\}$ es una refutación por resolución de $S_1 \cup \dots \cup S_n \cup S$.
- Def.: La fórmula F es demostrable por resolución a partir de $\{F_1, \dots, F_n\}$ si existe una demostración por resolución de F a partir de $\{F_1, \dots, F_n\}$.
Se representa por $\{F_1, \dots, F_n\} \vdash_{Res} F$.
- Ejemplo: Demostración por resolución de $p \wedge q$ a partir de $\{p \vee q, p \leftrightarrow q\}$

1	$\{p, q\}$	Hipótesis
2	$\{\neg p, q\}$	Hipótesis
3	$\{p, \neg q\}$	Hipótesis
4	$\{\neg p, \neg q\}$	Hipótesis
5	$\{q\}$	Resolvente de 1 y 2
6	$\{\neg q\}$	Resolvente de 3 y 4
7	\square	Resolvente de 5 y 6

Adecuación y completitud de la resolución

- Propiedades:

- Si C es una resolvente de C_1 y C_2 , entonces $\{C_1, C_2\} \models C$.
- Si $\square \in S$, entonces S es inconsistente.
- Si el conjunto de cláusulas S es refutable, entonces S es inconsistente.
- Teor.: El cálculo de resolución es adecuado y completo; es decir,

$$\begin{array}{l} \text{Adecuado: } S \vdash_{Res} F \quad \implies \quad S \models F \\ \text{Completo: } S \models F \quad \implies \quad S \vdash_{Res} F \end{array}$$

Argumentación y resolución

- Problema de los animales: Se sabe que
 1. Los animales con pelo y los 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:

$$\left\{ \begin{array}{l} \text{tiene_pelos} \vee \text{da_leche} \rightarrow \text{es_mamífero}, \\ \text{es_mamífero} \wedge (\text{tiene_pezuñas} \vee \text{rumia}) \rightarrow \text{es_ungulado}, \\ \text{es_ungulado} \wedge \text{tiene_cuello_largo} \rightarrow \text{es_jirafa}, \\ \text{es_ungulado} \wedge \text{tiene_rayas_negras} \rightarrow \text{es_cebra}, \\ \text{tiene_pelos} \wedge \text{tiene_pezuñas} \wedge \text{tiene_rayas_negras} \end{array} \right\} \quad \vdash_{Res} \text{es_cebra}$$

Argumentación y resolución

- Resolución:

1	{ \neg tiene_pelos, es_mamífero}	Hipótesis
2	{ \neg da_leche, es_mamífero}	Hipótesis
3	{ \neg es_mamífero, \neg tiene_pezuñas, es_ungulado}	Hipótesis
4	{ \neg es_mamífero, \neg rumia, es_ungulado}	Hipótesis
5	{ \neg es_ungulado, \neg tiene_cuello_largo, es_jirafa}	Hipótesis
6	{ \neg es_ungulado, \neg tiene_rayas_negras, es_cebra}	Hipótesis
7	{tiene_pelos}	Hipótesis
8	{tiene_pezuñas}	Hipótesis
9	{tiene_rayas_negras}	Hipótesis
10	{ \neg es_cebra}	Hipótesis
11	{es_mamifero}	Resolvente de 1 y 7
12	{ \neg tiene_pezuñas, es_ungulado}	Resolvente de 11 y 3
13	{es_ungulado}	Resolvente de 12 y 8
14	{ \neg tiene_rayas_negras, es_cebra}	Resolvente de 13 y 6
15	{es_cebra}	Resolvente de 14 y 9
16	□	Resolvente de 15 y 10

Bibliografía

- M. Ben–Ari, *Mathematical logic for computer science (2nd ed.)*. (Springer, 2001)
 - Cap. 4: Propositional calculus: resolution and BDDs
- C.–L. Chang y R.C.–T. Lee *Symbolic Logic and Mechanical Theorem Proving* (Academic Press, 1973)
 - Cap. 5.2: The resolution principle for the propositional logic
- N.J. Nilsson *Inteligencia artificial (Una nueva síntesis)* (McGraw–Hill, 2001)
 - Cap. 14: La resolución en el cálculo proposicional
- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)
 - Cap. 5.7: El principio de resolución en lógica proposicional
- U. Schöning *Logic for Computer Scientists* (Birkäuser, 1989)
 - Cap. 1.5: Resolution

Capítulo 5

Deducción natural proposicional

Tema 5: Otros sistemas proposicionales

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

Sistemas proposicionales

- **Sistemas estudiados:**
 - Tablas de verdad.
 - Método de Quine.
 - Formas normales.
 - Tableros semánticos.
 - Resolución.
- **Otros sistemas proposicionales:**
 - Sistemas axiomáticos.
 - Cálculo de secuentes.
 - Deducción natural (DN).

DN: Reglas de la conjunción

• Reglas de la conjunción:

• Regla de introducción de la conjunción: $\frac{F \quad G}{F \wedge G} \wedge i$

• Reglas de eliminación de la conjunción: $\frac{F \wedge G}{F} \wedge e_1$ $\frac{F \wedge G}{G} \wedge e_2$

• Ejemplo: $p \wedge q, r \vdash q \wedge r$:

1 :	$p \wedge q, r$	premises
2 :	q	$\wedge e_2$ 1.1
3 :	$q \wedge r$	$\wedge i$ 2,1.2

• Adecuación de las reglas de la conjunción:

* $\wedge i : \{F, G\} \models F \wedge G$

* $\wedge e_1 : F \wedge G \models F$

* $\wedge e_2 : F \wedge G \models G$

DN: Reglas de la doble negación

• Reglas de la doble negación

• Regla de eliminación de la doble negación: $\frac{\neg\neg F}{F} \neg\neg e$

• Regla de introducción de la doble negación: $\frac{F}{\neg\neg F} \neg\neg i$

• Ejemplo: $p, \neg\neg(q \wedge r) \vdash \neg\neg p \wedge r$:

1 :	$p, \neg\neg(q \wedge r)$	premises
2 :	$\neg\neg p$	$\neg\neg i$ 1.1
3 :	$q \wedge r$	$\neg\neg e$ 1.2
4 :	r	$\wedge e_2$ 3
5 :	$\neg\neg p \wedge r$	$\wedge i$ 2,4

• Adecuación de las reglas de la doble negación:

* $\neg\neg e : \{\neg\neg F\} \models F$

* $\neg\neg i : \{F\} \models \neg\neg F$

DN: Regla de eliminación del condicional

- Regla de eliminación del condicional:

- Regla de eliminación del condicional: $\frac{F \quad F \rightarrow G}{G} \rightarrow_e$

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

1 : $\neg p \wedge q, \neg p \wedge q \rightarrow r \vee \neg p$ 2 : $r \vee \neg p$	premises \rightarrow_e 1.1,1.2
---	-------------------------------------

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

1 : $p, p \rightarrow q, p \rightarrow (q \rightarrow r)$ 2 : q 3 : $q \rightarrow r$ 4 : r	premises \rightarrow_e 1.1,1.2 \rightarrow_e 1.1,1.3 \rightarrow_e 2,3
--	---

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

DN: Regla derivada de modus tollens (MT)

- Regla derivada de modus tollens (MT)

- Regla derivada de modus tollens (MT): $\frac{F \rightarrow G \quad \neg G}{\neg F} MT$

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

1 : $p \rightarrow (q \rightarrow r), p, \neg r$ 2 : $q \rightarrow r$ 3 : $\neg q$	premises \rightarrow_e 1.2,1.1 MT 1.3,2
---	---

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

1 : $\neg p \rightarrow q, \neg q$ 2 : $\neg \neg p$ 3 : p	premises MT 1.2,1.1 $\neg \neg_e$ 2
--	---

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

DN: Regla de introducción del condicional

- Regla de introducción del condicional

- Regla de introducción del condicional:

$$\frac{\begin{array}{|c|} \hline F \\ \vdots \\ G \\ \hline \end{array}}{F \rightarrow G} \rightarrow i$$

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

1 :	$p \rightarrow q$	premise
2 :	$\neg q$	assumption
3 :	$\neg p$	MT 2,1
4 :	$\neg q \rightarrow \neg p$	$\rightarrow i$ 2–3

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

DN: Regla de introducción del condicional

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

1 :	$\neg q \rightarrow \neg p$	premise
2 :	p	assumption
3 :	$\neg \neg p$	$\neg \neg i$ 2
4 :	$\neg \neg q$	MT 3,1
5 :	$p \rightarrow \neg \neg q$	$\rightarrow i$ 2–4

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

1 :	p	assumption
2 :	$p \rightarrow p$	$\rightarrow i$ 1–1

DN: Regla de introducción del condicional

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

1 :	$q \rightarrow r$	assumption
2 :	$\neg q \rightarrow \neg p$	assumption
3 :	p	assumption
4 :	$\neg \neg p$	$\neg \neg i$ 3
5 :	$\neg \neg q$	MT 4,2
6 :	q	$\neg \neg e$ 5
7 :	r	$\rightarrow e$ 6,1
8 :	$p \rightarrow r$	$\rightarrow i$ 3-7
9 :	$(\neg q \rightarrow \neg p) \rightarrow (p \rightarrow r)$	$\rightarrow i$ 2-8
10 :	$(q \rightarrow r) \rightarrow ((\neg q \rightarrow \neg p) \rightarrow (p \rightarrow r))$	$\rightarrow i$ 1-9

DN: Regla de introducción del condicional

- Ejemplo: $p \wedge q \rightarrow r \vdash p \rightarrow (q \rightarrow r)$:

1 :	$p \wedge q \rightarrow r$	premise
2 :	p	assumption
3 :	q	assumption
4 :	$p \wedge q$	$\wedge i$ 2,3
5 :	r	$\rightarrow e$ 4,1
6 :	$q \rightarrow r$	$\rightarrow i$ 3-5
7 :	$p \rightarrow (q \rightarrow r)$	$\rightarrow i$ 2-6

DN: Regla de introducción del condicional

- Ejemplo: $p \rightarrow (q \rightarrow r) \vdash (p \wedge q) \rightarrow r$:

1 :	$p \rightarrow (q \rightarrow r)$	premise
2 :	$p \wedge q$	assumption
3 :	p	$\wedge e1$ 2
4 :	$q \rightarrow r$	$\rightarrow e$ 3,1
5 :	q	$\wedge e2$ 2
6 :	r	$\rightarrow e$ 5,4
7 :	$(p \wedge q) \rightarrow r$	$\rightarrow i$ 2–6

DN: Regla de introducción del condicional

- Ejemplo: $p \rightarrow q \vdash p \wedge r \rightarrow q \wedge r$:

1 :	$p \rightarrow q$	premise
2 :	$p \wedge r$	assumption
3 :	p	$\wedge e1$ 2
4 :	r	$\wedge e2$ 2
5 :	q	$\rightarrow e$ 3,1
6 :	$q \wedge r$	$\wedge i$ 5,4
7 :	$p \wedge r \rightarrow q \wedge r$	$\rightarrow i$ 2–6

DN: Reglas de la disyunción

- Reglas de la disyunción:

- Reglas de introducción de la disyunción: $\frac{F}{F \vee G} \vee i_1$ $\frac{G}{F \vee G} \vee i_2$

- Regla de eliminación de la disyunción:

$$\frac{F \vee G \quad \boxed{\begin{array}{c} F \\ \vdots \\ H \end{array}} \quad \boxed{\begin{array}{c} G \\ \vdots \\ H \end{array}}}{H} \vee e$$

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

1 :	$p \vee q$	premise
2 :	p	assumption
3 :	$q \vee p$	$\vee i_2$ 2
4 :	q	assumption
5 :	$q \vee p$	$\vee i_1$ 4
6 :	$q \vee p$	$\vee e$ 1,2-3,4-5

DN: Reglas de la disyunción

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

1 :	$q \rightarrow r$	premise
2 :	$p \vee q$	assumption
3 :	p	assumption
4 :	$p \vee r$	$\vee i_1$ 3
5 :	q	assumption
6 :	r	$\rightarrow e$ 5,1
7 :	$p \vee r$	$\vee i_2$ 6
8 :	$p \vee r$	$\vee e$ 2,3-4,5-7
9 :	$p \vee q \rightarrow p \vee r$	$\rightarrow i$ 2-8

DN: Reglas de la disyunción

- Ejemplo: $(p \vee q) \vee r \vdash p \vee (q \vee r)$:

1 :	$(p \vee q) \vee r$	premise
2 :	$p \vee q$	assumption
3 :	p	assumption
4 :	$p \vee (q \vee r)$	$\vee i$ 3
5 :	q	assumption
6 :	$q \vee r$	$\vee i$ 5
7 :	$p \vee (q \vee r)$	$\vee i$ 6
8 :	$p \vee (q \vee r)$	$\vee e$ 2,3–4,5–7
9 :	r	assumption
10 :	$q \vee r$	$\vee i$ 9
11 :	$p \vee (q \vee r)$	$\vee i$ 10
12 :	$p \vee (q \vee r)$	$\vee e$ 1,2–8,9–11

DN: Reglas de la disyunción

- Ejemplo (distributiva): $p \wedge (q \vee r) \vdash (p \wedge q) \vee (p \wedge r)$:

1 :	$p \wedge (q \vee r)$	premise
2 :	p	$\wedge e$ 1
3 :	$q \vee r$	$\wedge e$ 1
4 :	q	assumption
5 :	$p \wedge q$	$\wedge i$ 2,4
6 :	$(p \wedge q) \vee (p \wedge r)$	$\vee i$ 5
7 :	r	assumption
8 :	$p \wedge r$	$\wedge i$ 2,7
9 :	$(p \wedge q) \vee (p \wedge r)$	$\vee i$ 8
10 :	$(p \wedge q) \vee (p \wedge r)$	$\vee e$ 3,4–6,7–9

DN: Regla de copia

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

1 :	p	assumption
2 :	q	assumption
3 :	p	hyp 1
4 :	q \rightarrow p	\rightarrow i 2-3
5 :	p \rightarrow (q \rightarrow p)	\rightarrow i 1-4

DN: Reglas de la negación

- Extensiones de la lógica para usar falso:
 - Extensión de la sintaxis: \perp es una fórmula proposicional.
 - Extensión de la semántica: $v(\perp) = 0$ en cualquier valoración.
- Reglas de la negación:

- Regla de eliminación de lo falso: $\frac{\perp}{F} \perp e$
- Regla de eliminación de la negación: $\frac{F \quad \neg F}{\perp} \neg e$
- Adecuación de las reglas de la negación:

* $\perp \models F$

* $\{F, \neg F\} \models \perp$

DN: Reglas de la negación

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

1 :	$\neg p \vee q$	premise
2 :	$\neg p$	assumption
3 :	p	assumption
4 :	\perp	$\neg e$ 2,3
5 :	q	$\perp e$ 4
6 :	$p \rightarrow q$	$\rightarrow i$ 3–5
7 :	q	assumption
8 :	p	assumption
9 :	q	hyp 7
10 :	$p \rightarrow q$	$\rightarrow i$ 8–9
11 :	$p \rightarrow q$	$\vee e$ 1,2–6,7–10

DN: Reglas de la negación

- Regla de introducción de la negación:

$$\frac{\boxed{F} \vdash \perp}{\neg F} \neg i$$

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

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

1 :	$p \rightarrow q, p \rightarrow \neg q$	premises
2 :	p	assumption
3 :	q	$\rightarrow e$ 2,1.1
4 :	$\neg q$	$\rightarrow e$ 2,1.2
5 :	\perp	$\neg e$ 4,3
6 :	$\neg p$	$\neg i$ 2–5

DN: Reglas de la negación

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

1 :	$p \rightarrow \neg p$	premise
2 :	p	assumption
3 :	$\neg p$	$\rightarrow e$ 2,1
4 :	\perp	$\neg e$ 3,2
5 :	$\neg p$	$\neg i$ 2-4

DN: Reglas de la negación

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

1 :	$p \wedge \neg q \rightarrow r, \neg r, p$	premises
2 :	$\neg q$	assumption
3 :	$p \wedge \neg q$	$\wedge i$ 1.3,2
4 :	r	$\rightarrow e$ 3,1.1
5 :	\perp	$\neg e$ 1.2,4
6 :	$\neg \neg q$	$\neg i$ 2-5
7 :	q	$\neg \neg e$ 6

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

1 :	$p \rightarrow (q \rightarrow r), p, \neg r$	premises
2 :	$q \rightarrow r$	$\rightarrow e$ 1.2,1.1
3 :	$\neg q$	MT 1.3,2

DN: Reglas del bicondicional

• Regla de introducción del bicondicional: $\frac{F \rightarrow G \quad G \rightarrow F}{F \leftrightarrow G} \leftrightarrow i$

• Ejemplo: $p \wedge q \leftrightarrow q \wedge p$:

1 :	$p \wedge q$	assumption
2 :	p	$\wedge e1$ 1
3 :	q	$\wedge e2$ 1
4 :	$q \wedge p$	$\wedge i$ 3,2
5 :	$p \wedge q \rightarrow q \wedge p$	$\rightarrow i$ 1–4
6 :	$q \wedge p$	assumption
7 :	q	$\wedge e1$ 6
8 :	p	$\wedge e2$ 6
9 :	$p \wedge q$	$\wedge i$ 8,7
10 :	$q \wedge p \rightarrow p \wedge q$	$\rightarrow i$ 6–9
11 :	$p \wedge q \leftrightarrow q \wedge p$	$\leftrightarrow i$ 5,10

DN: Reglas del bicondicional

• Reglas de eliminación del bicondicional: $\frac{F \leftrightarrow G}{F \rightarrow G} \leftrightarrow e1$ $\frac{F \leftrightarrow G}{G \rightarrow F} \leftrightarrow e2$

• Ejemplo: $p \leftrightarrow q, p \vee q \vdash p \wedge q$:

1 :	$p \leftrightarrow q, p \vee q$	premises
2 :	p	assumption
3 :	$p \rightarrow q$	$\leftrightarrow e1$ 1.1
4 :	q	$\rightarrow e$ 2,3
5 :	$p \wedge q$	$\wedge i$ 2,4
6 :	q	assumption
7 :	$q \rightarrow p$	$\leftrightarrow e2$ 1.1
8 :	p	$\rightarrow e$ 6,7
9 :	$p \wedge q$	$\wedge i$ 8,6
10 :	$p \wedge q$	$\vee e$ 1.2,2–5,6–9

DN: Reglas derivadas: modus tollens

- Regla derivada de modus tollens (MT): $\frac{F \rightarrow G \quad \neg G}{\neg F} MT$

1 :	$F \rightarrow G, \neg G$	premises
2 :	F	assumption
3 :	G	$\rightarrow e$ 2,1.1
4 :	\perp	$\neg e$ 1,2,3
5 :	$\neg F$	$\neg i$ 2-4

DN: Reglas derivadas: introducción de doble negación

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

1 :	F	premise
2 :	$\neg F$	assumption
3 :	\perp	$\neg e$ 2,1
4 :	$\neg\neg F$	$\neg i$ 2-3

DN: Reglas derivadas: reducción al absurdo (RAA)

- Regla de reducción al absurdo:

$$\frac{\begin{array}{c} \neg F \\ \vdots \\ \perp \end{array}}{F} \text{ RAA}$$

1 :	$\neg F \rightarrow \perp$	premise
2 :	$\neg F$	assumption
3 :	\perp	$\rightarrow e$ 2,1
4 :	$\neg\neg F$	$\neg i$ 2–3
5 :	F	$\neg\neg e$ 4

DN: Reglas derivadas: ley del tercio excluido (LEM)

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

1 :	$\neg(F \vee \neg F)$	assumption
2 :	F	assumption
3 :	$F \vee \neg F$	$\vee i$ 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 :	$\neg\neg(F \vee \neg F)$	$\neg i$ 1–7
9 :	$F \vee \neg F$	$\neg\neg e$ 8

DN: Reglas derivadas: ley del tercio excluido (LEM)

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

1 :	$p \rightarrow q$	premise
2 :	$p \vee \neg p$	LEM
3 :	p	assumption
4 :	q	$\rightarrow e$ 3,1
5 :	$\neg p \vee q$	$\vee i_2$ 4
6 :	$\neg p$	assumption
7 :	$\neg p \vee q$	$\vee i_1$ 6
8 :	$\neg p \vee q$	$\vee e$ 2,3–5,6–7

DN: Reglas de deducción natural

- Reglas de deducción natural:

	Introducción	Eliminación
\wedge	$\frac{F \quad G}{F \wedge G} \wedge i$	$\frac{F \wedge G}{F} \wedge e_1 \quad \frac{F \wedge G}{G} \wedge e_2$
\vee	$\frac{F}{F \vee G} \vee i_1 \quad \frac{G}{F \vee G} \vee i_2$	$\frac{F \vee G \quad \begin{array}{ c } \hline F \\ \hline \vdots \\ \hline H \\ \hline \end{array} \quad \begin{array}{ c } \hline G \\ \hline \vdots \\ \hline H \\ \hline \end{array}}{H} \vee e$
\rightarrow	$\frac{\begin{array}{ c } \hline F \\ \hline \vdots \\ \hline G \\ \hline \end{array}}{F \rightarrow G} \rightarrow i$	$\frac{F \quad F \rightarrow G}{G} \rightarrow e$

DN: Reglas de deducción natural

- Reglas de deducción natural:

	Introducción	Eliminación
\neg	$\frac{\boxed{\begin{array}{c} F \\ \vdots \\ \perp \end{array}}}{\neg F} \neg i$	$\frac{F \quad \neg F}{\perp} \neg e$
\perp		$\frac{\perp}{F} \perp e$
$\neg\neg$		$\frac{\neg\neg F}{F} \neg\neg e$
\leftrightarrow	$\frac{F \rightarrow G \quad G \rightarrow F}{F \leftrightarrow G} \leftrightarrow i$	$\frac{F \leftrightarrow G}{F \rightarrow G} \leftrightarrow e_1 \quad \frac{F \leftrightarrow G}{G \rightarrow F} \leftrightarrow e_2$

- Adecuación y completitud del cálculo de deducción natural.

Bibliografía

- C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal*. (Ariel, 2000) Cap. 16: Cálculo deductivo.
- R. Bornat *Using ItL Jape with X* (Department of Computer Science, QMW, 1998)
- J.A. Díez *Iniciación a la Lógica*, (Ariel, 2002) Cap. 4: Cálculo deductivo. Deducibilidad.
- M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems*. (Cambridge University Press, 2000) Cap. 1: Propositional logic.
- Fitting, M. *First-Order Logic and Automated Theorem Proving (2nd ed.)* (Springer, 1995) Cap. 4.2: Natural deduction.
- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003) Cap. 3.6: El método de la Deducción Natural

Capítulo 6

Sintaxis y semántica de la lógica de primer orden

Tema 6: Sintaxis y semántica de la lógica de primer orden

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

Limitación expresiva de la lógica proposicional

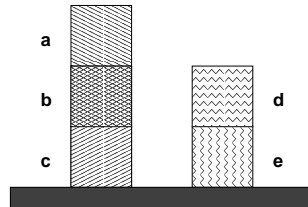
- Ejemplo 1: *Si Sevilla es vecina de Cádiz, entonces Cádiz es vecina de Sevilla. Sevilla es vecina de Cádiz. Por tanto, Cádiz es vecina de Sevilla*
- Representación en lógica proposicional:

$$\{SvC \rightarrow CvS, SvC\} \models CvS$$
- 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:

$$\{(\forall x)(\forall y)[vecina(x, y) \rightarrow vecina(y, x)], vecina(Sevilla, Cadiz)\} \\ \models vecina(Cadiz, Sevilla)$$

Potencia expresiva de la lógica de primer orden

- Mundo de los bloques



- $sobre(x, y)$ 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:

$sobre(a, b), sobre(b, c), sobre_mesa(c), sobre(d, e), sobre_mesa(e)$

Potencia expresiva de la lógica de primer orden

- $bajo(x, y)$ 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

$$(\forall x)(\forall y)[encima(x, y) \leftrightarrow sobre(x, y) \vee (\exists z)[sobre(z, x) \wedge encima(z, y)]]$$
- $libre(x)$ se verifica si el bloque x no tiene bloques encima

$$(\forall x)[libre(x) \leftrightarrow \neg(\exists y)sobre(y, x)]$$
- $pila(x, y, z)$ se verifica si el bloque x está sobre el y , el y sobre el z y el z sobre la mesa

$$(\forall x)(\forall y)(\forall z)[pila(x, y, z) \leftrightarrow sobre(x, y) \wedge sobre(y, z) \wedge sobre_mesa(z)]$$
- Prop.: Si z, y, z es una pila entonces y no está libre

$$(\forall x)(\forall y)(\forall z)[pila(x, y, z) \rightarrow \neg libre(y)]$$

Potencia expresiva de la lógica de primer orden

- Representación con funciones e igualdad

- $es_bloque(x)$ se verifica si x es un bloque
- $superior(x)$ es el bloque que está sobre el bloque x
- Situación del ejemplo:

$$es_bloque(a), es_bloque(b), es_bloque(c), es_bloque(d), es_bloque(e)$$

$$superior(b) = a, superior(c) = b, superior(e) = d$$

- $sobre_mesa(x)$ se verifica si el bloque x está sobre la mesa

$$(\forall x)[sobre_mesa(x) \leftrightarrow es_bloque(x) \wedge \neg(\exists y)superior(y) = x]$$

- $libre(x)$ se verifica si el bloque x no tiene bloques encima

$$(\forall x)[libre(x) \leftrightarrow \neg(\exists y)superior(x) = y]$$

- $tope(x)$ es el bloque libre que está encima de x

$$(\forall x)[(libre(x) \rightarrow tope(x) = x) \wedge (\neg libre(x) \rightarrow tope(x) = tope(superior(x)))]$$

Potencia expresiva de la lógica de primer orden

- Ejemplos de formalización:

- *La Tierra es un planeta:* planeta(Tierra)
- *La Luna no es un planeta:* \neg planeta(Luna)
- *La Luna es un satélite:* satélite(Luna)
- *La Tierra gira alrededor del Sol:* gira(Tierra, Sol)
- *Todo planeta es un satélite:* $(\forall x)[planeta(x) \rightarrow satélite(x)]$
- *Todo planeta gira alrededor del Sol:* $(\forall x)[planeta(x) \rightarrow gira(x, Sol)]$
- *Algún planeta gira alrededor de la Luna:* $(\exists x)[planeta(x) \wedge gira(x, Luna)]$
- *Hay por lo menos un satélite:* $(\exists x)satélite(x)$
- *Ningún planeta es un satélite:* $\neg(\exists x)[planeta(x) \wedge satélite(x)]$
- *Ningún objeto celeste gira alrededor de sí mismo:* $\neg(\exists x)gira(x, x)$

Potencia expresiva de la lógica de primer orden

- *Alrededor de los satélites no giran objetos:* $(\forall x)[\text{satélite}(x) \rightarrow \neg(\exists y)\text{gira}(y, x)]$
- *Hay exactamente un satélite:* $(\exists x)[\text{satélite}(x) \wedge (\forall y)[\text{satélite}(y) \rightarrow x = y]]$
- *La Luna es un satélite de la Tierra:* $\text{satélite}(\text{Luna}, \text{Tierra})$
[Notar la sobrecarga de la relación satélite]
- *Todo planeta tiene un satélite:* $(\forall x)[\text{planeta}(x) \rightarrow (\exists y)\text{satélite}(y, x)]$
- *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:* $(\forall x)[(\exists y)\text{satélite}(y, x) \rightarrow \text{planeta}(x)]$
- *Todo satélite es satélite de algún planeta:*
 $(\forall x)[\text{satélite}(x) \rightarrow (\exists y)(\text{planeta}(y) \wedge \text{satélite}(x, y))]$
- *La Luna no gira alrededor de dos planetas diferentes:*
 $\neg(\exists x)(\exists y)[\text{planeta}(x) \wedge \text{planeta}(y) \wedge \text{gira}(\text{Luna}, x) \wedge \text{gira}(\text{Luna}, y) \wedge x \neq y]$
- *Hay exactamente dos planetas:*
 $(\exists x)(\exists y)[\text{planeta}(x) \wedge \text{planeta}(y) \wedge x \neq y \wedge (\forall z)[\text{planeta}(z) \rightarrow (z = x \vee z = y)]]$

Lenguaje de primer orden

- **Lenguaje de primer orden:**
 - Símbolos lógicos:
 - Variables: $x, y, z, \dots, x_1, x_2, \dots$
 - Conectivas: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
 - 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: $<$

Sintaxis: términos

- 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 .
 - Ejemplo: En el lenguaje de la aritmética,
 1. $+(\cdot(x, 1), s(y))$ es un término, que se suele escribir como $(x \cdot 1) + s(y)$
 2. $+(\cdot(x, <), s(y))$ no es un término
 - Notación:
 - s, t, t_1, t_2, \dots representan términos.
 - $\text{Térm}(L)$ representa el conjunto de los términos de L

Sintaxis: fórmulas atómicas

- **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 .

- Ejemplo: En el lenguaje de la aritmética,

1. $<(\cdot(x, 1), s(y))$ es una fórmula atómica que se suele escribir como $x \cdot 1 < s(y)$
2. $+(x, y) = \cdot(x, y)$ es una fórmula atómica que se suele escribir como $x + y = x \cdot y$

- Notación:

- A, B, A_1, A_2, \dots representan fórmulas atómicas.
 - $\text{Átom}(L)$ representa el conjunto de las fórmulas atómicas de L

Sintaxis: fórmulas

- **Fórmulas:**

- Def. de las fórmulas de L :

- Las fórmulas atómicas de L son fórmulas de L .
 - Si F y G son fórmulas de L , entonces $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$ 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 .

- Ejemplo: En el lenguaje de la aritmética,

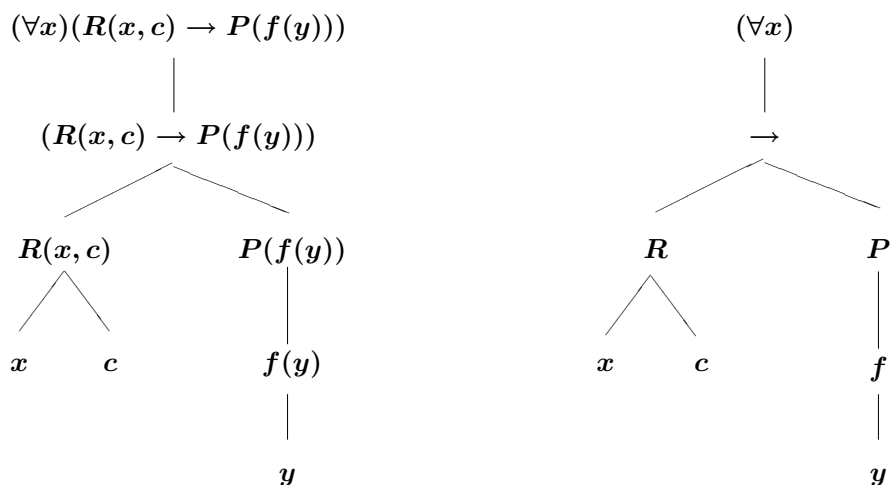
1. $(\forall x)(\exists y) <(x, y)$ es una fórmula que se suele escribir como $(\forall x)(\exists y)x < y$
2. $(\forall x)(\exists y) +(x, y)$ no es una fórmula.

- Notación:

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

Sintaxis: fórmulas

- Árboles de análisis (o de formación) y esquemáticos



Sintaxis: subfórmulas

- Subfórmulas:

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

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

- Ejemplo:

$$\text{Subf}((\forall x)(R(x, c) \rightarrow P(f(y)))) = \{ (\forall x)(R(x, c) \rightarrow P(f(y))), \\ (R(x, c) \rightarrow P(f(y))), \\ R(x, c), \\ P(f(y)) \}$$

Sintaxis: omisión de paréntesis

- 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 $<(x, y)$

Sintaxis: conjuntos de variables

- Conjuntos de variables:

- Def.: El conjunto de las variables de un término t se define recursivamente por:

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

- Def.: El conjunto de las variables de una fórmula F se define recursivamente por:

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

- 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 $\{y\}$.

Sintaxis: apariciones libres y ligadas

- 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:

$$(\forall x)(P(\underline{x}) \rightarrow R(\underline{x}, y)) \rightarrow ((\exists y)P(\underline{y}) \rightarrow R(z, x))$$

$$(\exists x)R(\underline{x}, y) \vee (\forall y)P(\underline{y})$$

$$(\forall x)(P(\underline{x}) \rightarrow (\exists y)R(\underline{x}, \underline{y}))$$

$$P(x) \rightarrow R(x, y)$$

Sintaxis: variables libres y ligadas

- Variables libres y ligadas:

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

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

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

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

- Ejemplo:

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

Sintaxis: fórmulas cerradas y básicas

- **Fórmula cerradas:**
 - Def.: Una fórmula cerrada (o sentencia) es una fórmula sin variables libres.
 - Ejemplos: $(\forall x)(P(x) \rightarrow (\exists y)R(x, y))$ es cerrada.
 $(\exists x)R(x, y) \vee (\forall y)P(y)$ no es cerrada.
- **Fórmulas básicas:**
 - Def.: Una fórmula básica es una fórmula sin variables.
 - Ejemplos: $P(a) \rightarrow R(a, b)$ es básica.
 $(\forall x)(P(x) \rightarrow (\exists y)R(x, y))$ no es básica.

Sintaxis: sustituciones

- **Sustituciones (de un lenguaje):**
 - Def.: Una sustitución σ (de L) es una aplicación $\sigma : \text{Var} \rightarrow \text{Térm}(L)$.
 - Ejemplo: La aplicación σ de Var en los términos de la aritmética tal que $\sigma(x) = s(0)$, $\sigma(y) = x + y$ y $\sigma(z) = z$ para $z \in \text{Var} \setminus \{x, y\}$ es una sustitución.
 - 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: La sustitución del ejemplo anterior se representa por $[x/s(0), y/x + y]$
 - Notación: $\sigma, \sigma_1, \sigma_2, \dots$ representarán sustituciones.

Sintaxis: Aplicación de sustituciones a términos

- **Aplicación de sustituciones a términos:**

- Def.: $t[x_1/t_1, \dots, x_n/t_n]$ 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 $\sigma : \text{Térm}(L) \rightarrow \text{Térm}(L)$ definida por

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

- 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 .
 - $h(a, x, w)\sigma = h(a\sigma, x\sigma, w\sigma) = h(a, f(y, a), w)$
 - $f(x, y)\sigma = f(x\sigma, y\sigma) = f(f(y, a), z)$
 - $h(a, f(x, y), w)\sigma = h(a\sigma, f(x, y)\sigma, w\sigma) = h(a, f(f(y, a), z), w)$

Sintaxis: Composición de sustituciones

- **Composición de sustituciones:**

- Diferencia entre sustituciones simultáneas y consecutivas:
 - $g(x, z)[x/g(z, b), z/a] = g(g(z, b), a)$
 - $g(x, z)[x/g(z, b)][z/a] = g(g(z, b), z)[z/a] = g(g(a, b), a)$
- Cálculo de la composición: Si $\sigma_1 = [x/f(z, a), y/w]$ y $\sigma_2 = [x/b, z/g(w)]$, entonces
 - $x\sigma_1\sigma_2 = (x\sigma_1)\sigma_2 = f(z, a)\sigma_2 = f(z\sigma_2, a\sigma_2) = f(g(w), a)$
 - $y\sigma_1\sigma_2 = (y\sigma_1)\sigma_2 = w\sigma_2 = w$
 - $z\sigma_1\sigma_2 = (z\sigma_1)\sigma_2 = z\sigma_2 = g(w)$
 - $w\sigma_1\sigma_2 = (w\sigma_1)\sigma_2 = w\sigma_2 = w$

Por tanto, $\sigma_1\sigma_2 = [x/f(g(w), a), y/w, z/g(w)]$. Comprobación:

$$\begin{aligned} h(y, x)\sigma_1\sigma_2 &= (h(y, x)\sigma_1)\sigma_2 = h(w, f(z, a))\sigma_2 = h(w\sigma_2, f(z, a)\sigma_2) = h(w, f(g(w), a)) \\ h(y, x)\sigma_1\sigma_2 &= h(y, x)[x/f(g(w), a), y/w, z/g(w)] = h(w, f(g(w), a)) \end{aligned}$$

Sintaxis: Aplicación de sustituciones a fórmulas

- **Aplicación de sustituciones a fórmulas:**

- Def.: $F[x_1/t_1, \dots, x_n/t_n]$ 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 $\sigma : \text{Fórm}(L) \rightarrow \text{Fórm}(L)$ definida por

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

donde σ_x es la sustitución definida por

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

Sintaxis: Aplicación de sustituciones a fórmulas

- Ejemplos: Si $\sigma = [x/f(y), y/b]$, entonces

1. $((\forall x)(Q(x) \rightarrow R(x, y)))\sigma = (\forall x)((Q(x) \rightarrow R(x, y))\sigma_x)$
 $= (\forall x)(Q(x)\sigma_x \rightarrow R(x, y)\sigma_x)$
 $= (\forall x)(Q(x) \rightarrow R(x, b))$
2. $(Q(x) \rightarrow (\forall x)R(x, y))\sigma = Q(x)\sigma \rightarrow ((\forall x)R(x, y))\sigma$
 $= Q(f(y)) \rightarrow (\forall x)(R(x, y)\sigma_x)$
 $= Q(f(y)) \rightarrow (\forall x)R(x, b)$
3. $((\forall x)(Q(x) \rightarrow (\forall y)R(x, y)))\sigma = (\forall x)((Q(x) \rightarrow (\forall y)R(x, y))\sigma_x)$
 $= (\forall x)(Q(x)\sigma_x \rightarrow ((\forall y)R(x, y))\sigma_x)$
 $= (\forall x)(Q(x) \rightarrow (\forall y)(R(x, y)\sigma_{xy}))$
 $= (\forall x)(Q(x) \rightarrow (\forall y)R(x, y))$

Sintaxis: Sustituciones libres

- 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 $(\forall x)(P(x) \rightarrow Q(x, f(y)))$
 $(\forall x)(P(x) \rightarrow Q(x, f(y)))[y/g(y)] = (\forall x)(P(x) \rightarrow Q(x, f(g(y))))$
- $[y/g(x)]$ no es libre para $(\forall x)(P(x) \rightarrow Q(x, f(y)))$
 $(\forall x)(P(x) \rightarrow Q(x, f(y)))[y/g(x)] = (\forall x)(P(x) \rightarrow Q(x, f(g(x))))$

- Convenio: Al escribir $F\sigma$ supondremos que σ es libre para F .

Semántica: 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 cuyo dominio es el conjunto de símbolos propios de L y tal que
 - si c es una constante de L , entonces $I(c) \in U$
(i.e. $I(c)$ es un elemento de U);
 - si f es un símbolo de función n -aria ($n > 0$) de L , entonces $I(f) : U^n \rightarrow U$
(i.e. $I(f)$ es una función n -aria en U);
 - si R es un símbolo de relación n -aria ($n > 0$) de L , entonces $I(R) \subseteq U^n$
(i.e. $I(R)$ es una relación n -aria en U).
- Una asignación A en una estructura (U, I) es una función $A : \text{Var} \rightarrow U$ 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.

Semántica: Estructuras

- Ejemplos: 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)
 $I_2(\leq) = \{(w_1, w_2) : w_1, w_2 \in \{0, 1\}^*, w_1 \text{ es prefijo de } w_2\}$ (prefijo)

Semántica: Estructuras

- Ejemplos (cont.):

- Tercera estructura de L :

$U_3 = \{\text{abierto}, \text{cerrado}\}$
 $I_3(0) = \text{cerrado}$
 $I_3(s) = \{(\text{abierto}, \text{cerrado}),$
 $\quad (\text{cerrado}, \text{abierto})\}$
 $I_3(+)$ = $\{(\text{abierto}, \text{abierto}, \text{abierto}),$
 $\quad (\text{abierto}, \text{cerrado}, \text{abierto}),$
 $\quad (\text{cerrado}, \text{abierto}, \text{abierto}),$
 $\quad (\text{cerrado}, \text{cerrado}, \text{cerrado})\}$
 $I_3(\leq) = \{(\text{abierto}, \text{abierto}),$
 $\quad (\text{cerrado}, \text{abierto}),$
 $\quad (\text{cerrado}, \text{cerrado})\}$

e	$I_3(s)(e)$	$I_3(+)$	abierto	cerrado	$I_3(\leq)$	abierto	cerrado
abierto	cerrado	abierto	abierto	abierto	abierto	1	0
cerrado	abierto	cerrado	abierto	cerrado	cerrado	1	1

Semántica: Evaluación de términos

• 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 $\mathcal{I}_A : \text{Térm}(L) \rightarrow U$ 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 ”.
- Ejemplos: Sean L el lenguaje de la página ?? y t el término $s(+ (x, s(0)))$.
 - Si \mathcal{I} es la primera estructura y $A(x) = 3$, entonces

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

Semántica: Evaluación de términos

• Ejemplos (cont.)

- Si \mathcal{I} es la segunda estructura y $A(x) = 10$, entonces

$$\begin{aligned} \mathcal{I}_A(t) &= \mathcal{I}_A(s(+ (x, s(0)))) &= I(s)(\mathcal{I}_A(+ (x, s(0)))) &= \\ &= I(s)(I(+)(\mathcal{I}_A(x), \mathcal{I}_A(s(0)))) &= I(s)(I(+)(A(x), \mathcal{I}_A(s(0)))) &= \\ &= I(s)(I(+)(10, I(s)(\mathcal{I}_A(0)))) &= I(s)(I(+)(10, I(s)(I(0)))) &= \\ &= I(s)(I(+)(10, I(s)(\epsilon))) &= I(s)(I(+)(10, 1)) &= \\ &= I(s)(101) &= 10111 \end{aligned}$$

- Si \mathcal{I} es la tercera estructura y $A(x) = \text{abierto}$, entonces

$$\begin{aligned} \mathcal{I}_A(t) &= \mathcal{I}_A(s(+ (x, s(0)))) &= I(s)(\mathcal{I}_A(+ (x, s(0)))) &= \\ &= I(s)(I(+)(\mathcal{I}_A(x), \mathcal{I}_A(s(0)))) &= I(s)(I(+)(A(x), \mathcal{I}_A(s(0)))) &= \\ &= I(s)(I(+)(\text{abierto}, I(s)(\mathcal{I}_A(0)))) &= I(s)(I(+)(\text{abierto}, I(s)(I(0)))) &= \\ &= I(s)(I(+)(\text{abierto}, I(s)(\text{cerrado}))) &= I(s)(I(+)(\text{abierto}, \text{abierto})) &= \\ &= I(s)(\text{abierto}) &= \text{cerrado} \end{aligned}$$

Semántica: Evaluación de términos

- Ejemplo anterior con notación reducida e infija:

Sean L el lenguaje de la página ?? 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 $A(x) = \text{abierto}$, entonces

$$\begin{aligned}\mathcal{I}_A(t) &= \mathcal{I}_A(s(x + s(0))) = s^I(\text{abierto} +^I s^I(0^I)) = \\ &= s^I(\text{abierto} +^I s^I(\text{cerrado})) = s^I(\text{abierto} +^I \text{abierto}) = \\ &= s^I(\text{abierto}) = \text{cerrado}\end{aligned}$$

Semántica: Evaluación de fórmulas

- Variante de una asignación:

- Def.: Sea A una asignación en la estructura (U, I) y $u \in U$. Mediante $A[x/u]$ se representa la asignación definida por

$$A[x/u](y) = \begin{cases} u, & \text{si } y \text{ es } x; \\ A(y) & \text{si } y \text{ es distinta de } x \end{cases}$$

- Función de verdad de una relación:

- Def.: 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 $H_R : U^n \rightarrow \mathbb{B}$ definida por

$$H_R(u_1, \dots, u_n) = \begin{cases} 1, & \text{si } (u_1, \dots, u_n) \in R; \\ 0, & \text{en caso contrario} \end{cases}$$

- Función de verdad de la igualdad:

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

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

Semántica: Evaluación de fórmulas

- 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 $\mathcal{I}_A : \text{Fórm}(L) \rightarrow \mathbb{B}$ por

- Si F es $t_1 = t_2$, $\mathcal{I}_A(F) = H_{=}(\mathcal{I}_A(t_1), \mathcal{I}_A(t_2))$

- 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 ”.

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\exists y)P(x, y)$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(P) = \{(1, 1), (2, 2)\}$ y $A(x) = 1$

- En notación completa:

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

$$\begin{aligned} \mathcal{I}_{A[y/1]}P(x, y) &= H_{I(P)}(\mathcal{I}_{A[y/1]}(x), \mathcal{I}_{A[y/1]}(y)) \\ &= H_{I(P)}(A[y/1](x), A[y/1](y)) \\ &= H_{I(P)}(1, 1) \\ &= V \end{aligned}$$

$$\text{Luego, } \mathcal{I}_A((\exists y)P(x, y)) = V.$$

- En notación reducida:

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

$$\begin{aligned} \mathcal{I}_{A[y/1]}P(x, y) &= P^I(A[y/1](x), A[y/1](y)) \\ &= P^I(1, 1) \\ &= V \end{aligned}$$

$$\text{Luego, } \mathcal{I}_A((\exists y)P(x, y)) = V.$$

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\forall x)(\exists y)P(x, y)$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$ e $I(P) = \{(1, 1), (2, 2)\}$

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

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

$$\begin{aligned} \mathcal{I}_{A[x/1, y/1]}P(x, y) &= H_{I(P)}(\mathcal{I}_{A[x/1, y/1]}(x), \mathcal{I}_{A[x/1, y/1]}(y)) \\ &= H_{I(P)}(A[x/1, y/1](x), A[x/1, y/1](y)) \\ &= H_{I(P)}(1, 1) \\ &= \mathbf{V} \end{aligned}$$

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

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

$$\begin{aligned} \mathcal{I}_{A[x/2, y/2]}P(x, y) &= H_{I(P)}(\mathcal{I}_{A[x/2, y/2]}(x), \mathcal{I}_{A[x/2, y/2]}(y)) \\ &= H_{I(P)}(A[x/2, y/2](x), A[x/2, y/2](y)) \\ &= H_{I(P)}(2, 2) \\ &= \mathbf{V} \end{aligned}$$

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

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

Semántica: Evaluación de fórmulas

- Ejemplo anterior en notación reducida:
Evaluación de $(\forall x)(\exists y)P(x, y)$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$ e $I(P) = \{(1, 1), (2, 2)\}$

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

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

$$\begin{aligned} \mathcal{I}_{A[x/1, y/1]}P(x, y) &= P^I(1, 1) \\ &= \mathbf{V} \end{aligned}$$

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

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

$$\begin{aligned} \mathcal{I}_{A[x/2, y/2]}P(x, y) &= P^I(2, 2) \\ &= \mathbf{V} \end{aligned}$$

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

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

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\forall x)(P(x) \rightarrow Q(g(x), a))$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

$$\mathcal{I}_A((\forall x)(P(x) \rightarrow Q(g(x), a))) = \mathbf{V} \Leftrightarrow \mathcal{I}_{A[x/1]}(P(x) \rightarrow Q(g(x), a)) = \mathbf{V} \text{ y } \mathcal{I}_{A[x/2]}(P(x) \rightarrow Q(g(x), a)) = \mathbf{V}$$

$$\begin{aligned} & \mathcal{I}_{A[x/1]}(P(x) \rightarrow Q(g(x), a)) = \\ & = H_{\rightarrow}(\mathcal{I}_{A[x/1]}(P(x)), \mathcal{I}_{A[x/1]}(Q(g(x), a))) = \\ & = H_{\rightarrow}(H_{I(P)}(\mathcal{I}_{A[x/1]}(x)), H_{I(Q)}(\mathcal{I}_{A[x/1]}(g(x)), \mathcal{I}_{A[x/1]}(a))) = \\ & = H_{\rightarrow}(H_{I(P)}(A[x/1](x)), H_{I(Q)}(I(g)(\mathcal{I}_{A[x/1]}(x)), I(a))) = \\ & = H_{\rightarrow}(H_{I(P)}(1), H_{I(Q)}(I(g)(A[x/1](x)), 1)) = \\ & = H_{\rightarrow}(\mathbf{F}, H_{I(Q)}(I(g)(1), 1)) = \\ & = H_{\rightarrow}(\mathbf{F}, H_{I(Q)}(2, 1)) = \\ & = H_{\rightarrow}(\mathbf{F}, \mathbf{F}) = \\ & = \mathbf{V} \end{aligned}$$

Semántica: Evaluación de fórmulas

- Ejemplo (cont.)

$$\begin{aligned} & \mathcal{I}_{A[x/2]}(P(x) \rightarrow Q(g(x), a)) = \\ & = H_{\rightarrow}(\mathcal{I}_{A[x/2]}(P(x)), \mathcal{I}_{A[x/2]}(Q(g(x), a))) = \\ & = H_{\rightarrow}(H_{I(P)}(\mathcal{I}_{A[x/2]}(x)), H_{I(Q)}(\mathcal{I}_{A[x/2]}(g(x)), \mathcal{I}_{A[x/2]}(a))) = \\ & = H_{\rightarrow}(H_{I(P)}(A[x/2](x)), H_{I(Q)}(I(g)(\mathcal{I}_{A[x/2]}(x)), I(a))) = \\ & = H_{\rightarrow}(H_{I(P)}(2), H_{I(Q)}(I(g)(A[x/2](x)), 1)) = \\ & = H_{\rightarrow}(\mathbf{V}, H_{I(Q)}(I(g)(2), 1)) = \\ & = H_{\rightarrow}(\mathbf{V}, H_{I(Q)}(1, 1)) = \\ & = H_{\rightarrow}(\mathbf{V}, \mathbf{V}) = \\ & = \mathbf{V} \end{aligned}$$

Por tanto, $\mathcal{I}_A((\forall x)(P(x) \rightarrow Q(g(x), a))) = \mathbf{V}$.

Semántica: Evaluación de fórmulas

- Ejemplo anterior con notación reducida:

Evaluación de $(\forall x)(P(x) \rightarrow Q(g(x), a))$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

$$\mathcal{I}_A((\forall x)(P(x) \rightarrow Q(g(x), a))) = V \Leftrightarrow \mathcal{I}_{A[x/1]}(P(x) \rightarrow Q(g(x), a)) = V \text{ y} \\ \mathcal{I}_{A[x/2]}(P(x) \rightarrow Q(g(x), a)) = V$$

$$\begin{aligned} \mathcal{I}_{A[x/1]}(P(x) \rightarrow Q(g(x), a)) &= P^I(1) \rightarrow Q^I(g^I(1), a^I) \\ &= F \rightarrow Q^I(2, 1) \\ &= F \rightarrow F \\ &= V \end{aligned}$$

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

Por tanto, $\mathcal{I}_A((\forall x)(P(x) \rightarrow Q(g(x), a))) = V$.

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\exists x)(P(g(x)) \wedge Q(x, g(a)))$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

$$\mathcal{I}_A((\exists x)(P(g(x)) \wedge Q(x, g(a)))) = V \Leftrightarrow \mathcal{I}_{A[x/1]}(P(g(x)) \wedge Q(x, g(a))) = V \text{ ó} \\ \mathcal{I}_{A[x/2]}(P(g(x)) \wedge Q(x, g(a))) = V$$

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

Por tanto, $\mathcal{I}_A((\exists x)(P(g(x)) \wedge Q(x, g(a)))) = V$.

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\exists x)(P(x) \wedge Q(x, a))$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

$$\mathcal{I}_A((\exists x)(P(x) \wedge Q(x, a))) = V \Leftrightarrow \begin{array}{l} \mathcal{I}_{A[x/1]}(P(x) \wedge Q(x, a)) = V \text{ ó} \\ \mathcal{I}_{A[x/2]}(P(x) \wedge Q(x, a)) = V \end{array}$$

$$\begin{aligned} \mathcal{I}_{A[x/1]}(P(x) \wedge Q(x, a)) &= P^I(1) \wedge Q^I(1, a^I) \\ &= F \wedge Q^I(1, 1) \\ &= F \wedge V \\ &= F \end{aligned}$$

$$\begin{aligned} \mathcal{I}_{A[x/2]}(P(x) \wedge Q(x, a)) &= P^I(2) \wedge Q^I(2, a^I) \\ &= V \wedge Q^I(2, 1) \\ &= V \wedge F \\ &= F \end{aligned}$$

Por tanto, $\mathcal{I}_A((\exists x)(P(x) \wedge Q(x, a))) = F$.

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\forall x)(\exists y)(P(x) \wedge Q(x, y))$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

$$\mathcal{I}_A((\forall x)(\exists y)(P(x) \wedge Q(x, y))) = V \Leftrightarrow \begin{array}{l} \mathcal{I}_{A[x/1]}(\exists y)(P(x) \wedge Q(x, y)) = V \text{ y} \\ \mathcal{I}_{A[x/2]}(\exists y)(P(x) \wedge Q(x, y)) = V \end{array}$$

$$\mathcal{I}_{A[x/1]}(\exists y)(P(x) \wedge Q(x, y)) = V \Leftrightarrow \begin{array}{l} \mathcal{I}_{A[x/1, y/1]}(P(x) \wedge Q(x, y)) = V \text{ ó} \\ \mathcal{I}_{A[x/1, y/2]}(P(x) \wedge Q(x, y)) = V \end{array}$$

$$\begin{aligned} \mathcal{I}_{A[x/1, y/1]}(P(x) \wedge Q(x, y)) &= P^I(1) \wedge Q^I(1, 1) \\ &= F \wedge V \\ &= F \end{aligned}$$

$$\begin{aligned} \mathcal{I}_{A[x/1, y/2]}(P(x) \wedge Q(x, y)) &= P^I(1) \wedge Q^I(1, 2) \\ &= F \wedge V \\ &= F \end{aligned}$$

Luego, $\mathcal{I}_{A[x/1]}(\exists y)(P(x) \wedge Q(x, y)) = F$

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

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\forall x)g(x) = x$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

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

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

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

Semántica: Evaluación de fórmulas

- Ejemplo: Evaluación de $(\forall x)g(g(x)) = x$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

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

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

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

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

Semántica: Evaluación de fórmulas

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

Semántica: Evaluación de fórmulas

- Evaluación de fórmulas en las estructuras de las páginas ??-??:

Fórmula	\mathcal{I}_1	\mathcal{I}_2	\mathcal{I}_3
$(\forall x)0 \leq x$	V	V	V
$(\forall x)x \leq s(x)$	V	V	F
$(\exists x)s(x) = 0$	F	F	V
$(\exists x)s(x) = x$	F	F	F

Semántica: Evaluación variables libres

- Evaluación y variables libres:
 - Sea t un término de L , 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 de t , entonces $\mathcal{I}_A(t) = \mathcal{I}_B(t)$.
 - 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 t no tiene variables, entonces $\mathcal{I}_A(F) = \mathcal{I}_B(F)$ para cualesquiera asignaciones A y B en \mathcal{I} . Se suele escribir simplemente $\mathcal{I}(t)$.
 - 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 $\mathcal{I}(F)$.
 - Si las variables libres de F son x_1, \dots, x_n , entonces son equivalentes
 - $\mathcal{I}_A(F) = 1$, para toda asignación A en \mathcal{I} .
 - $\mathcal{I}((\forall x_1) \dots (\forall x_n) F) = 1$.

Semántica: Realización de una fórmula

- Realización de una fórmula:
 - Def.: Sean F una fórmula de L , \mathcal{I} una estructura de L y A una asignación en \mathcal{I} .
 - (\mathcal{I}, A) es una realización de F si $\mathcal{I}_A(F) = 1$.
Se representa por $\mathcal{I}_A \models F$.
 - (\mathcal{I}, A) no es una realización de F si $\mathcal{I}_A(F) = 0$.
Se representa por $\mathcal{I}_A \not\models F$.
 - F se verifica en \mathcal{I} respecto de A si $\mathcal{I}_A \models F$.
 - F no se verifica en \mathcal{I} respecto de A si $\mathcal{I}_A \not\models F$.
 - Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(R) = \leq$.
 - Si A es una asignación en \mathcal{I} tal que $A(x) = 0$, entonces

$$\mathcal{I}_A \models (\forall y)R(x, y),$$
 - Si A es una asignación en \mathcal{I} tal que $A(x) = 5$, entonces

$$\mathcal{I}_A \not\models (\forall y)R(x, y),$$

Semántica: Satisfacibilidad en una estructura

- Satisfacibilidad en una estructura
 - Def.: Sean F una fórmula de L e \mathcal{I} una estructura de L .
 - F es satisfacible en \mathcal{I} si existe alguna asignación A en I tal que $\mathcal{I}_A \models F$.
 - F es insatisfacible en \mathcal{I} si no existe ninguna asignación A en I tal que $\mathcal{I}_A \models F$.
 - Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(R) = \leq$.
 - $(\forall y)R(x, y)$ es satisfacible en \mathcal{I} .
 $\mathcal{I}_A \models (\forall y)R(x, y)$, con $A(x) = 0$.
 - $(\forall x)R(x, y)$ es insatisfacible en \mathcal{I} .
 No existe $n \in \mathbb{N}$ tal que para todo $m \in \mathbb{N}$, se tenga $m \leq n$.
 - Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(R) = \geq$.
 - $(\forall y)R(x, y)$ es insatisfacible en \mathcal{I} .
 No existe $m \in \mathbb{N}$ tal que para todo $n \in \mathbb{N}$, se tenga $m \geq n$.
 - $(\forall x)R(x, y)$ es satisfacible en \mathcal{I} .
 $\mathcal{I}_A \models (\forall y)R(x, y)$, con $A(x) = 0$.

Semántica: Validez en una estructura

- Validez en una estructura
 - Def.: Sean F una fórmula de L e \mathcal{I} una estructura de L .
 - F es válida en \mathcal{I} si, para toda asignación A en \mathcal{I} , $\mathcal{I}_A \models F$.
 Se representa por $\mathcal{I} \models F$.
 - F no es válida en \mathcal{I} si, para alguna asignación A en \mathcal{I} , $\mathcal{I}_A \not\models F$.
 Se representa por $\mathcal{I} \not\models F$.
 - Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(R) = <$.
 - $\mathcal{I} \models (\exists y)R(x, y)$.
 Si A es una asignación en \mathcal{I} , entonces $\mathcal{I}_A \models (\exists y)R(x, y)$
 $I_{A[y/A(x)+1]}(R(x, y)) = V$
 - $\mathcal{I} \not\models (\forall y)R(x, y)$.
 Sea A una asignación en \mathcal{I} tal que $A(x) = 5$. Entonces $\mathcal{I}_A \not\models (\forall y)R(x, y)$
 $I_{A[y/3]}(R(x, y)) = F$

Semántica: Satisfacibilidad y validez en una estructura

- Satisfacibilidad y validez en una estructura para sentencias
 - Sea F una sentencia de L e \mathcal{I} una estructura de L .
 - F es válida en \mathcal{I} syss F es satisfacible en \mathcal{I} .
 - Se cumple una, y sólo una, de las siguientes condiciones
 1. F es válida en \mathcal{I} .
 2. $\neg F$ es válida en \mathcal{I} .

- Cierres cuantificacionales:
 - Sea F una fórmula de L , \mathcal{I} una estructura de L y $\{x_1, \dots, x_n\}$ el conjunto de las variables libres de F .
 - F es válida en \mathcal{I} syss $(\forall x_1) \dots (\forall x_n) F$ es válida en \mathcal{I}
 - F es satisfacible en \mathcal{I} syss $(\exists x_1) \dots (\exists x_n) F$ es satisfacible en \mathcal{I}

Semántica: Modelo de una fórmula

- Modelo de una fórmula:
 - Def.: Sean F una fórmula de L e \mathcal{I} una estructura de L .
 - \mathcal{I} es un modelo de F si $\mathcal{I} \models F$.
 - \mathcal{I} no es un modelo de F si $\mathcal{I} \not\models F$.
 - Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(R) = <$. Entonces

$$\mathcal{I} \models (\exists y)R(x, y). \quad \mathcal{I} \not\models (\forall y)R(x, y).$$
 - Ejemplos: Sea F la fórmula $(\forall x)f(x, e) = x$. Las siguientes estructuras son modelos de F .
 - (U, I) con $U = \mathbb{N}$, $I(e) = 0$ e $I(f)$ como la suma.
 - (U, I) con $U = \{0, 1\}^*$, $I(e) = \epsilon$ e $I(f)$ la concatenación.
 - (U, I) con $U = \mathbb{B}$, $I(e) = 1$ e $I(f) = H_\wedge$
 Las siguientes estructuras no son modelo de F
 - (U, I) con $U = \mathbb{N}$, $I(e) = 5$ e $I(f)$ como la suma.
 - (U, I) con $U = \mathbb{N}$, $I(e) = 0$ e $I(f)$ como el producto.

Semántica: Satisfacibilidad de una fórmula

- Satisfacibilidad de una fórmula:

- Def.: Sea F una fórmula de L .

- F es satisfacible si tiene alguna realización
(i.e. existe una estructura \mathcal{I} y una asignación A en \mathcal{I} tales que $\mathcal{I}_A(F) = 1$).
 - F es insatisfacible si no tiene ninguna realización
(i.e. para toda estructura \mathcal{I} y toda asignación A se tiene que $\mathcal{I}_A(F) = 0$).

- Ejemplos:

- $(\forall y)R(x, y)$ es satisfacible
 $\mathcal{I}_A((\forall y)R(x, y)) = 1$, siendo $\mathcal{I} = (\mathbb{N}, I)$, $I(R) = \leq$ y $A(x) = 0$.
 - $(\exists x)P(x) \wedge (\forall x)\neg P(x)$ es insatisfacible.

Semántica: Satisfacibilidad y modelo

- Propiedades:

- Sea F una fórmula cerrada. Son equivalentes:

- F es satisfacible.
 - F tiene modelo.

- Si F es insatisfacible, entonces no tiene ningún modelo.

- Existen fórmulas satisfacibles que tienen realizaciones, pero no tienen modelos.

Por ejemplo, sea F la fórmula $x \neq y$.

La fórmula F es satisfacible

$$\mathcal{I}_A(F) = 1, \text{ siendo } \mathcal{I} = (\{p, q\}, I), A(x) = p, A(y) = q$$

La fórmula F no tiene modelo

Sea \mathcal{I} una estructura. Existe una asignación A en \mathcal{I} tal que $A(x) = A(y)$.
Luego, $\mathcal{I}_A(F) = 0$ y $\mathcal{I} \not\models F$.

Semántica: Validez de una fórmula

- Validez de una fórmula:
 - 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} y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$). Se representa por $\models F$.
 - F no es válida si alguna estructura de L no es modelo de F
(i.e. existe alguna estructura \mathcal{I} y alguna asignación A tales que $\mathcal{I}_A(F) = 0$). Se representa por $\not\models F$.
 - Ejemplos:
 - $(\exists x)P(x) \vee (\forall x)\neg P(x)$ es válida.
 - $(\forall y)R(x, y)$ no es válida.
 $\mathcal{I}_A((\forall y)R(x, y)) = 0$, siendo $\mathcal{I} = (\mathbb{N}, I)$, $I(R) = \leq$ y $A(x) = 5$.
 - $(\exists x)(P(x) \rightarrow (\forall y)P(y))$ es válida.

Semántica: Satisfacibilidad y validez

- Relaciones entre satisfacibilidad y validez:
 - Prop.: 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 $\not\implies \neg F$ es insatisfacible.
 - $(\forall x)P(x)$ es satisfacible.
modelo $\mathcal{I} = (U, I)$ con $U = \{1, 2\}$ e $I(P) = \{a\}$
 - $\neg(\forall x)P(x)$ es satisfacible.
modelo $\mathcal{I} = (U, I)$ con $U = \{1, 2\}$ e $I(P) = \{a\}$

Semántica: Realización de un conjunto de fórmulas

- **Notación:**

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

- **Realización de un conjunto 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 para toda $F \in S$ se tiene que $\mathcal{I}_A(F) = 1$.
Se representa por $\mathcal{I}_A \models S$.

- (\mathcal{I}, A) no es una realización de S si para alguna $F \in S$ se tiene que $\mathcal{I}_A(F) = 0$.
Se representa por $\mathcal{I}_A \not\models S$.

- **Ejemplos:** Sea $S = \{(\forall y)R(x, y), (\forall y)f(x, y) = y\}$.

- (\mathcal{I}, A) con $\mathcal{I} = (U, I), U = \mathbb{N}, R^I = \leq, f^I = +, A(x) = 0$ es realización de S .

- (\mathcal{I}, A) con $\mathcal{I} = (U, I), U = \mathbb{N}, R^I = <, f^I = +, A(x) = 0$ no es realización de S .

- (\mathcal{I}, A) con $\mathcal{I} = (U, I), U = \mathbb{N}, R^I = \leq, f^I = *, A(x) = 0$ no es realización de S .

Semántica: Consistencia de un conjunto de fórmulas

- **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 $F \in S$, $\mathcal{I}_A(F) = 1$).

- 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 $\mathcal{I}_A(F) = 0$).

- **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 .

- $S = \{P(x) \rightarrow Q(x), (\exists y)P(y), \neg Q(x)\}$ es inconsistente.

Semántica: Modelo de un conjunto de fórmulas

• Modelo de un conjunto de fórmulas:

- Def.: Sean S un conjunto de fórmulas de L e \mathcal{I} una estructura de L .
 - \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 $\mathcal{I} \models S$.
 - \mathcal{I} no es un modelo de S si para alguna $F \in S$ se tiene que $\mathcal{I} \not\models F$
(i.e. para alguna $F \in S$ y alguna asignación A en \mathcal{I} se tiene $\mathcal{I}_A(F) = 0$).
Se representa por $\mathcal{I} \not\models S$.
- Ejemplos: 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 .
 - $\mathcal{I} = (\mathbb{N}, I)$ con $R^I = \leq, f^I = *, e^I = 0$ no es modelo de S .
 - $\mathcal{I} = (\{0, 1\}^*, I)$ con $R^I = \text{prefijo}, f^I = \text{concatenación y } e^I = \epsilon$ es modelo de S .

Semántica: Consistencia y modelo

• Propiedades:

- Sea S un conjunto de fórmulas cerradas. Son equivalentes:
 - S es consistente.
 - S tiene modelo.
- Si S es inconsistente, entonces no tiene ningún modelo.
- Existen conjuntos de fórmulas consistentes que tienen realizaciones, pero no tienen modelos.

Por ejemplo, sea $S = \{x \neq y\}$.

El conjunto S es consistente

$$\mathcal{I}_A \models S, \text{ siendo } \mathcal{I} = (\{p, q\}, I), A(x) = p, A(y) = q$$

El conjunto S no tiene modelo

Sea \mathcal{I} una estructura. Existe una asignación A en \mathcal{I} tal que $A(x) = A(y)$.
Luego, $\mathcal{I}_A(x \neq y) = 0$ y $\mathcal{I} \not\models S$.

Semántica: 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 $\mathcal{I}_A \models F$).
(i.e. 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 $\mathcal{I}_A(F) = 1$).
Se representa por $S \models F$.
 - F no es consecuencia lógica de S si alguna realización de S no lo es de F .
(i.e. para alguna estructura \mathcal{I} de L y alguna asignación A en \mathcal{I} se tiene que
 $\mathcal{I}_A \models S$ y $\mathcal{I}_A \not\models F$).
(i.e. para alguna estructura \mathcal{I} de L y alguna asignación A en \mathcal{I} se tiene que,
para todo $G \in S$, $\mathcal{I}_A(G) = 1$ y $\mathcal{I}_A(F) = 0$).
Se representa por $S \not\models F$.
 - Se escribe $G \models F$ en lugar de $\{G\} \models F$.
 - Se escribe $G \not\models F$ en lugar de $\{G\} \not\models F$.

Semántica: Consecuencia lógica

- Ejemplos:

- $(\forall x)P(x) \models P(y)$
- $P(y) \not\models (\forall x)P(x)$
 (\mathcal{I}, A) con $\mathcal{I} = (U, I), U = \{1, 2\}, P^I = \{1\}, A(y) = 1$.
- $(\forall x)P(x) \models (\exists y)P(y)$
- $(\exists x)P(x) \not\models (\forall y)P(y)$
 $\mathcal{I} = (U, I)$ con $U = \{1, 2\}, P^I = \{1\}$
 $\mathcal{I} = (U, I)$ con $U = \mathbb{N}$ y $P^I = \{n \in \mathbb{N} : n \text{ es par}\}$
- $(\exists x)(\forall y)Q(x, y) \models (\forall y)(\exists x)Q(x, y)$
- $(\forall y)(\exists x)Q(x, y) \not\models (\exists x)(\forall y)Q(x, y)$
 $\mathcal{I} = (U, I)$ con $U = \{1, 2\}, Q^I = \{(1, 1), (2, 2)\}$
 $\mathcal{I} = (U, I)$ con $U = \mathbb{N}, Q^I = <$

Semántica: Consecuencia lógica

- Ejemplos:

- $\{P(x) \rightarrow Q(x), P(c)\} \models Q(c)$
- $\{P(x) \rightarrow Q(x), Q(c)\} \not\models P(c)$
- $\{P(x) \rightarrow Q(x), \neg Q(c)\} \models \neg P(c)$
- $\{P(c), \neg P(d)\} \models c \neq d$

- Ejemplos: Se consideran las fórmulas

$$F_1 : (\forall x)R(x, x),$$

$$F_2 : (\forall x)(\forall y)(R(x, y) \rightarrow R(y, x)).$$

$$F_3 : (\forall x)(\forall y)(R(x, y) \wedge R(y, x) \rightarrow x = y),$$

- $\{F_2, F_3\} \not\models F_1$

Contraejemplo: $\mathcal{I} = (U, I)$ con $U = \{a, b\}$ y $R^I = \{(a, a)\}$

- $\{F_1, F_2\} \not\models F_3$

Contraejemplo: $\mathcal{I} = (U, I)$ con $U = \{a, b\}$ y $R^I = U^2$

- $\{F_1, F_3\} \models F_2$

Semántica: Consecuencia lógica

- Propiedades:

- $S \models F$ 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 $\mathcal{I}_A(\neg H) = 0$.

$\iff S \cup \{\neg F\}$ es inconsistente.

- Sean F una fórmula cerrada de L y S un conjunto de fórmulas cerradas de L .
Entonces, F es consecuencia lógica de S syss todos los modelos de S lo son de F .

Semántica: Equivalencia lógica

- 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:

- $P(x) \not\equiv P(y)$.

$$\mathcal{I} = (\{1, 2\}, I) \text{ con } P^I = \{1\} \text{ y } A(x) = 1, A(y) = 2.$$

- $(\forall x)P(x) \equiv (\forall y)P(y)$.

- $(\forall x)(P(x) \wedge Q(x)) \equiv (\forall x)P(x) \wedge (\forall x)Q(x)$.

- $(\exists x)(P(x) \wedge Q(x)) \equiv (\exists x)P(x) \wedge (\exists x)Q(x)$.

$$\mathcal{I} = (\{1, 2\}, I) \text{ con } P^I = \{1\} \text{ y } Q^I = \{2\}.$$

- Propiedades: Sean F y G fórmulas cerradas de L .

- $F \equiv G \text{ syss } \models F \leftrightarrow G$.

- $F \equiv G \text{ syss } F \models G \text{ y } G \models F$.

Bibliografía

- C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal*. (Ariel, 2000) pp. 195–259 y 323–326.
- M.L. Bonet *Apuntes de LPO*. (Univ. Politécnica de Cataluña, 2003) pp. 17–26.
- C.-L. Chang y R.C.-T. Lee *Symbolic logic and mechanical theorem proving* (Academic Press, 1973) pp. 26–35.
- J. Dingel *Propositional and predicate logic: a review*. (2000) pp. 21–27.
- J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional*. (UNED, 2003) pp. 64–87.
- J.H. Gallier *Logic for computer science (foundations of automatic theorem Proving)* (June 2003) pp. 146–186.
- M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems*. (Cambridge University Press, 2000) pp. 90–109 y 128–140.
- 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.
- L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 22–29.

Capítulo 7

Deducción natural en lógica de primer orden

Tema 7: Cálculo deductivo de primer orden

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

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 1: $P(c), (\forall x)(P(x) \rightarrow \neg Q(x)) \vdash \neg Q(c)$

1 :	actual y , $P(y)$, $\forall x.(P(x) \rightarrow \neg Q(x))$	premises
2 :	$P(y) \rightarrow \neg Q(y)$	$\forall e$ 1.3,1.1
3 :	$\neg Q(y)$	$\rightarrow e$ 2,1.2

- Nota: $(\forall x)(\exists y)(x < y) \not\vdash (\exists y)(y < y)$.

Reglas del cuantificador universal

- Regla de introducción del cuantificador universal

- Regla de introducción del cuantificador universal:

$$\frac{\begin{array}{c} x_0 \\ \vdots \\ F[x/x_0] \end{array}}{(\forall x)F} \forall i$$

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

- Nota: Analogía con $\wedge i$.
- Ejemplo 2: $(\forall x)(P(x) \rightarrow Q(x)), (\forall x)P(x) \vdash (\forall x)Q(x)$

1 :	$\forall x.(P(x) \rightarrow Q(x))$	$, \forall x.P(x)$	premises
2 :	actual i		assumption
3 :	$P(i) \rightarrow Q(i)$		$\forall e$ 1.1,2
4 :	$P(i)$		$\forall e$ 1.2,2
5 :	$Q(i)$		$\rightarrow e$ 3,4
6 :	$\forall x.Q(x)$		$\forall i$ 2–5

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

1 :	actual j , $\forall x.P(x)$	premises
2 :	$P(j)$	$\forall e$ 1.2,1.1
3 :	$\exists x.P(x)$	$\exists i$ 2,1.1

Reglas del cuantificador existencial

- Regla de eliminación del cuantificador existencial

- Regla de eliminación del cuantificador existencial:

$$\frac{(\exists x)F \quad \boxed{\begin{array}{c} x_0 \quad F[x/x_0] \\ \vdots \\ G \end{array}}}{G} \exists e$$

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

- Nota: Analogía con $\forall e$.

- Ejemplo 4: $(\forall x)(P(x) \rightarrow Q(x)), (\exists x)P(x) \vdash (\exists x)Q(x)$

1 :	$\forall x.(P(x) \rightarrow Q(x))$, $\exists x.P(x)$	premises
2 :	actual i , $P(i)$		assumptions
3 :	$P(i) \rightarrow Q(i)$		$\forall e$ 1.1,2.1
4 :	$Q(i)$		$\rightarrow e$ 3,2.2
5 :	$\exists x.Q(x)$		$\exists i$ 4,2.1
6 :	$\exists x.Q(x)$		$\exists e$ 1.2,2-5

Reglas del cuantificador existencial

- Ejemplo 5: $(\forall x)(Q(x) \rightarrow R(x)), (\exists x)(P(x) \wedge Q(x)) \vdash (\exists x)(P(x) \wedge R(x))$

1 :	$\forall x.(Q(x) \rightarrow R(x))$, $\exists x.(P(x) \wedge Q(x))$	premises
2 :	actual i , $P(i) \wedge Q(i)$		assumptions
3 :	$Q(i) \rightarrow R(i)$		$\forall e$ 1.1,2.1
4 :	$Q(i)$		$\wedge e$ 2.2
5 :	$R(i)$		$\rightarrow e$ 3,4
6 :	$P(i)$		$\wedge e$ 2.2
7 :	$P(i) \wedge R(i)$		$\wedge i$ 6,5
8 :	$\exists x.(P(x) \wedge R(x))$		$\exists i$ 7,2.1
9 :	$\exists x.(P(x) \wedge R(x))$		$\exists e$ 1.2,2-8

Reglas del cuantificador existencial

- Ejemplo 6: $(\exists x)P(x), (\forall x)(\forall y)(P(x) \rightarrow Q(y)) \vdash (\forall y)Q(y)$

1 :	$\exists x.P(x) , \forall x.\forall y.(P(x)\rightarrow Q(y))$	premises
2 :	actual i	assumption
3 :	actual $i1$, $P(i1)$	assumptions
4 :	$\forall y.(P(i1)\rightarrow Q(y))$	$\forall e$ 1.2,3.1
5 :	$P(i1)\rightarrow Q(i)$	$\forall e$ 4,2
6 :	$Q(i)$	$\rightarrow e$ 5,3.2
7 :	$Q(i)$	$\exists e$ 1.1,3–6
8 :	$\forall y.Q(y)$	$\forall i$ 2–7

Equivalencias

- **Equivalencias:**

- Sean F y G fórmulas.

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

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

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

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

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

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

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

Equivalencias

- Sean F y G fórmulas.

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

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

- Sean F y G fórmulas.

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

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

Equivalencias

- Equivalencia 1(a): $\neg(\forall x)F \vdash (\exists x)\neg F$

1 :	$\neg\forall x.P(x)$	premise
2 :	$\neg\exists x.\neg P(x)$	assumption
3 :	actual i	assumption
4 :	$\neg P(i)$	assumption
5 :	$\exists x.\neg P(x)$	$\exists i$ 4,3
6 :	\perp	$\neg e$ 5,2
7 :	$P(i)$	RAA 4-6
8 :	$\forall x.P(x)$	$\forall i$ 3-7
9 :	\perp	$\neg e$ 8,1
10 :	$\exists x.\neg P(x)$	RAA 2-9

Equivalencias

- Equivalencia 1(a): $(\exists x)\neg F \vdash \neg(\forall x)F$

1 :	$\exists x.\neg P(x)$	premise
2 :	$\neg\neg\forall x.P(x)$	assumption
3 :	$\forall x.P(x)$	$\neg\neg$ e 2
4 :	actual i, $\neg P(i)$	assumptions
5 :	$P(i)$	\forall e 3,4.1
6 :	\perp	\neg e 5,4.2
7 :	\perp	\exists e 1,4–6
8 :	$\neg\forall x.P(x)$	RAA 2–7

Equivalencias

- Equivalencia 1(a): $\neg(\forall x)F \equiv (\exists x)\neg F$

1 :	$\neg\forall x.P(x)$	assumption
2 :	$\exists x.\neg P(x)$	Conjecture $\neg\forall x.P(x) \vdash \exists x.\neg P(x)$ 1
3 :	$\neg\forall x.P(x) \rightarrow \exists x.\neg P(x)$	\rightarrow i 1–2
4 :	$\exists x.\neg P(x)$	assumption
5 :	$\neg\forall x.P(x)$	Theorem $\exists x.\neg P(x) \vdash \neg\forall x.P(x)$ 4
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

Equivalencias

- Equivalencia 3(a): $(\forall x)(F \wedge G) \vdash (\forall x)F \wedge (\forall x)G$

1 :	$\forall x.(P(x) \wedge Q(x))$	premise
2 :	actual i1	assumption
3 :	$P(i1) \wedge Q(i1)$	$\forall e$ 1,2
4 :	$P(i1)$	$\wedge e$ 3
5 :	$\forall x.P(x)$	$\forall i$ 2–4
6 :	actual i	assumption
7 :	$P(i) \wedge Q(i)$	$\forall e$ 1,6
8 :	$Q(i)$	$\wedge e$ 7
9 :	$\forall x.Q(x)$	$\forall i$ 6–8
10 :	$\forall x.P(x) \wedge \forall x.Q(x)$	$\wedge i$ 5,9

Equivalencias

- Equivalencia 3(a): $(\forall x)F \wedge (\forall x)G \vdash (\forall x)(F \wedge G)$

1 :	$\forall x.P(x) \wedge \forall x.Q(x)$	premise
2 :	actual i	assumption
3 :	$\forall x.P(x)$	$\wedge e$ 1
4 :	$P(i)$	$\forall e$ 3,2
5 :	$\forall x.Q(x)$	$\wedge e$ 1
6 :	$Q(i)$	$\forall e$ 5,2
7 :	$P(i) \wedge Q(i)$	$\wedge i$ 4,6
8 :	$\forall x.(P(x) \wedge Q(x))$	$\forall i$ 2–7

Equivalencias

- Equivalencia 3(a): $(\forall x)(F \wedge G) \equiv (\forall x)F \wedge (\forall x)G$

1 : $\forall x.(P(x) \wedge Q(x))$	assumption
2 : $\forall x.P(x) \wedge \forall x.Q(x)$	Theorem $\forall x.(P(x) \wedge Q(x)) \vdash \forall x.P(x) \wedge \forall x.Q(x)$ 1
3 : $\forall x.(P(x) \wedge Q(x)) \rightarrow \forall x.P(x) \wedge \forall x.Q(x)$	\rightarrow i 1–2
4 : $\forall x.P(x) \wedge \forall x.Q(x)$	assumption
5 : $\forall x.(P(x) \wedge Q(x))$	Theorem $\forall x.P(x) \wedge \forall x.Q(x) \vdash \forall x.(P(x) \wedge Q(x))$ 4
6 : $\forall x.P(x) \wedge \forall x.Q(x) \rightarrow \forall x.(P(x) \wedge Q(x))$	\rightarrow i 4–5
7 : $\forall x.(P(x) \wedge Q(x)) \leftrightarrow \forall x.P(x) \wedge \forall x.Q(x)$	\leftrightarrow i 3,6

Equivalencias

- Equivalencia 3(b): $(\exists x)F \vee (\exists x)G \vdash (\exists x)(F \vee G)$

1 : $\exists x.P(x) \vee \exists x.Q(x)$	premise
2 : $\exists x.P(x)$	assumption
3 : actual i, P(i)	assumptions
4 : P(i) \vee Q(i)	\vee i1 3.2
5 : $\exists x.(P(x) \vee Q(x))$	\exists i 4,3.1
6 : $\exists x.(P(x) \vee Q(x))$	\exists e 2,3–5
7 : $\exists x.Q(x)$	assumption
8 : actual i1, Q(i1)	assumptions
9 : P(i1) \vee Q(i1)	\vee i2 8.2
10 : $\exists x.(P(x) \vee Q(x))$	\exists i 9,8.1
11 : $\exists x.(P(x) \vee Q(x))$	\exists e 7,8–10
12 : $\exists x.(P(x) \vee Q(x))$	\vee e 1,2–6,7–11

Equivalencias

- Equivalencia 3(b): $(\exists x)(F \vee G) \vdash (\exists x)F \vee (\exists x)G$

1 :	$\exists x.(P(x) \vee Q(x))$	premise
2 :	actual i , $P(i) \vee Q(i)$	assumptions
3 :	$P(i)$	assumption
4 :	$\exists x.P(x)$	$\exists i$ 3,2.1
5 :	$\exists x.P(x) \vee \exists x.Q(x)$	\vee intro 4
6 :	$Q(i)$	assumption
7 :	$\exists x.Q(x)$	$\exists i$ 6,2.1
8 :	$\exists x.P(x) \vee \exists x.Q(x)$	\vee intro 7
9 :	$\exists x.P(x) \vee \exists x.Q(x)$	$\vee e$ 2.2,3–5,6–8
10 :	$\exists x.P(x) \vee \exists x.Q(x)$	$\exists e$ 1,2–9

Equivalencias

- Equivalencia 3(b): $(\exists x)F \vee (\exists x)G \equiv (\exists x)(F \vee G)$

1 :	$\exists x.P(x) \vee \exists x.Q(x)$	assumption
2 :	$\exists x.(P(x) \vee Q(x))$	Theorem $\exists x.P(x) \vee \exists x.Q(x) \vdash \exists x.(P(x) \vee Q(x))$ 1
3 :	$\exists x.P(x) \vee \exists x.Q(x) \rightarrow \exists x.(P(x) \vee Q(x))$	$\rightarrow i$ 1–2
4 :	$\exists x.(P(x) \vee Q(x))$	assumption
5 :	$\exists x.P(x) \vee \exists x.Q(x)$	Conjecture $\exists x.(P(x) \vee Q(x)) \vdash \exists x.P(x) \vee \exists x.Q(x)$ 4
6 :	$\exists x.(P(x) \vee Q(x)) \rightarrow \exists x.P(x) \vee \exists x.Q(x)$	$\rightarrow i$ 4–5
7 :	$\exists x.P(x) \vee \exists x.Q(x) \leftrightarrow \exists x.(P(x) \vee Q(x))$	$\leftrightarrow i$ 3,6

Equivalencias

- Equivalencia 4(b): $(\exists x)(\exists y)F \vdash (\exists y)(\exists x)F$

1 :	$\exists x.\exists y.P(x,y)$	premise
2 :	actual i , $\exists y.P(i,y)$	assumptions
3 :	actual $i1$, $P(i,i1)$	assumptions
4 :	$\exists x.P(x,i1)$	$\exists i$ 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

Equivalencias

- Equivalencia 4(b): $(\exists x)(\exists y)F \equiv (\exists y)(\exists x)F$

1 :	$\exists x.\exists y.P(x,y)$	assumption
2 :	$\exists y.\exists x.P(x,y)$	Conjecture $\exists x.\exists y.P(x,y) \vdash \exists y.\exists x.P(x,y)$ 1
3 :	$\exists x.\exists y.P(x,y) \rightarrow \exists y.\exists x.P(x,y)$	$\rightarrow i$ 1–2
4 :	$\exists y.\exists x.P(x,y)$	assumption
5 :	$\exists x.\exists y.P(x,y)$	Conjecture $\exists x.\exists y.P(x,y) \vdash \exists y.\exists x.P(x,y)$ 4
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

Reglas 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 $(x + 1 > 1) \rightarrow (x + 1 > 0)$ premisa
- 3 $(1 + x > 1) \rightarrow (1 + x > 0) = e 1,2$

- 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 $t_1 = t_3 = e 2,1$

Reglas de la igualdad

- Regla de introducción de la igualdad:

$$\overline{t = t} = i$$

- 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

- C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal*. (Ariel, 2000) pp. 259–287.
- R. Bornat *Using ItL Jape with X* (Department of Computer Science, QMW, 1998)
- J. Dingel *Propositional and predicate logic: a review*. (2000) pp. 28–33.
- J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional*. (UNED, 2003) pp. 88–94.
- M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems*. (Cambridge University Press, 2000) pp. 109-127.

Capítulo 8

Formas normales. Cláusulas

Tema 8: Formas normales. Cláusulas

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

Equivalencias

- Equivalencia lógica
 - Prop.: $F \equiv G$ 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:

$$\begin{aligned} F &= (\forall x)P(x) \rightarrow (\exists x)Q(x) \\ G &= (\forall x)P(x) \\ G' &= (\forall y)P(y) \\ F' &= (\forall y)P(y) \rightarrow (\exists x)Q(x) \end{aligned}$$

Forma rectificada

- **Fórmula en forma rectificada:**

- Def.: F está en forma rectificada si ninguna variable aparece libre y ligada y cada cuantificador se refiere a una variable diferente.

- Ejemplos: $(\forall x)P(x) \rightarrow (\forall y)Q(z, y)$ está en forma rectificada
 $(\forall x)P(x) \rightarrow (\forall y)Q(x, y)$ no está en forma rectificada
 $(\forall x)P(x) \rightarrow (\forall x)Q(z, x)$ no está en forma rectificada

- Prop.: Para toda fórmula F existe una fórmula equivalente G en forma rectificada.

- Lema del renombramiento: Si y no aparece libre en F , entonces

$$(\forall x)F \equiv (\forall y)F[x/y]$$

$$(\exists x)F \equiv (\exists y)F[x/y].$$

- Ejemplos de rectificación:

$$(\forall x)P(x) \rightarrow (\forall x)Q(z, x) \equiv (\forall x)P(x) \rightarrow (\forall u)Q(z, u)$$

$$(\forall x)P(x) \rightarrow (\forall y)Q(x, y) \equiv (\forall z)P(z) \rightarrow (\forall y)Q(x, y)$$

Forma normal prenexa

- **Fórmula en forma normal prenexa**

- Def.: La fórmula F está en forma normal prenexa (FNP) si es de la forma $(Q_1x_1) \dots (Q_nx_n)G$, donde $Q_i \in \{\forall, \exists\}$, $n \geq 0$ y G no tiene cuantificadores.

$(Q_1x_1) \dots (Q_nx_n)$ se llama el prefijo de F y G se llama la matriz de F .

- Ejemplos:

Fórmula	¿está en FNP?
$\neg(\exists x)[P(x) \rightarrow (\forall x)P(x)]$	no
$(\forall x)(\exists y)[P(x) \wedge \neg P(y)]$	sí
$(\forall x)P(x) \vee (\exists y)Q(y)$	no
$(\forall x)(\exists y)[P(x) \vee Q(y)]$	sí
$(\exists y)(\forall x)[P(x) \vee Q(y)]$	sí
$\neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\forall x)[Q(x) \rightarrow R(x)] \rightarrow (\forall x)[P(x) \rightarrow R(x)])$	no
$(\exists z)(\forall x)(\forall y)[((\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))) \wedge (P(z) \wedge \neg R(z))]$	sí

Forma normal prenexa: Cálculo de forma normal prenexa

- Algoritmo de cálculo de forma normal prenexa:

- Algoritmo: Aplicando a una fórmula los siguientes pasos se obtiene otra fórmula equivalente y que está en forma normal prenexa rectificada:

1. Rectificar la fórmula usando las equivalencias

$$(\forall x)F \equiv (\forall y)F[x/y] \quad (1)$$

$$(\exists x)F \equiv (\exists y)F[x/y] \quad (2)$$

donde y es una variable que no ocurre libre en F .

2. Eliminar los bicondicionales usando la equivalencia

$$F \leftrightarrow G \equiv (F \rightarrow G) \wedge (G \rightarrow F) \quad (3)$$

3. Eliminar los condicionales usando la equivalencia

$$F \rightarrow G \equiv \neg F \vee G \quad (4)$$

Forma normal prenexa: Cálculo de forma normal prenexa

- Algoritmo (cont.)

4. Interiorizar las negaciones usando las equivalencias

$$\neg(F \wedge G) \equiv \neg F \vee \neg G \quad (5)$$

$$\neg(F \vee G) \equiv \neg F \wedge \neg G \quad (6)$$

$$\neg\neg F \equiv F \quad (7)$$

$$\neg(\forall x)F \equiv (\exists x)\neg F \quad (8)$$

$$\neg(\exists x)F \equiv (\forall x)\neg F \quad (9)$$

5. Exteriorizar los cuantificadores usando las equivalencias

$$(\forall x)F \wedge G \equiv (\forall x)(F \wedge G) \quad \text{con } x \text{ no libre en } G. \quad (11)$$

$$(\forall x)F \vee G \equiv (\forall x)(F \vee G) \quad \text{con } x \text{ no libre en } G. \quad (12)$$

$$(\exists x)F \wedge G \equiv (\exists x)(F \wedge G) \quad \text{con } x \text{ no libre en } G. \quad (13)$$

$$(\exists x)F \vee G \equiv (\exists x)(F \vee G) \quad \text{con } x \text{ no libre en } G. \quad (14)$$

$$G \wedge (\forall x)F \equiv (\forall x)(G \wedge F) \quad \text{con } x \text{ no libre en } G. \quad (15)$$

$$G \vee (\forall x)F \equiv (\forall x)(G \vee F) \quad \text{con } x \text{ no libre en } G. \quad (16)$$

$$G \wedge (\exists x)F \equiv (\exists x)(G \wedge F) \quad \text{con } x \text{ no libre en } G. \quad (17)$$

$$G \vee (\exists x)F \equiv (\exists x)(G \vee F) \quad \text{con } x \text{ no libre en } G. \quad (18)$$

Forma normal prenexa: Cálculo de forma normal prenexa

- Ejemplo de cálculo de una forma normal prenexa de

$$\begin{aligned}
& \neg(\exists x)[P(x) \rightarrow (\forall x)P(x)] \\
\equiv & \neg(\exists x)[P(x) \rightarrow (\forall y)P(y)] \quad [\text{por (1)}] \\
\equiv & \neg(\exists x)[\neg P(x) \vee (\forall y)P(y)] \quad [\text{por (4)}] \\
\equiv & (\forall x)[\neg(\neg P(x) \vee (\forall y)P(y))] \quad [\text{por (9)}] \\
\equiv & (\forall x)[\neg\neg P(x) \wedge \neg(\forall y)P(y)] \quad [\text{por (6)}] \\
\equiv & (\forall x)[P(x) \wedge (\exists y)\neg P(y)] \quad [\text{por (7 y 8)}] \\
\equiv & (\forall x)(\exists y)[P(x) \wedge \neg P(y)] \quad [\text{por (17)}]
\end{aligned}$$

- Ejemplo de cálculo de una forma normal prenexa de

$$\begin{aligned}
& (\forall x)P(x) \vee (\exists y)Q(y) \\
\equiv & (\forall x)[P(x) \vee (\exists y)Q(y)] \quad [\text{por (12)}] \\
\equiv & (\forall x)(\exists y)[P(x) \vee Q(y)] \quad [\text{por (18)}]
\end{aligned}$$

- Ejemplo de cálculo de otra forma normal prenexa de

$$\begin{aligned}
& (\forall x)P(x) \vee (\exists y)Q(y) \\
\equiv & (\exists y)[(\forall x)P(x) \vee Q(y)] \quad [\text{por (18)}] \\
\equiv & (\exists y)(\forall x)[P(x) \vee Q(y)] \quad [\text{por (12)}]
\end{aligned}$$

Forma normal prenexa: Cálculo de forma normal prenexa

- Ejemplo de cálculo de una forma normal prenexa de

$$\begin{aligned}
& \neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\forall x)[Q(x) \rightarrow R(x)] \rightarrow (\forall x)[P(x) \rightarrow R(x)]) \\
\equiv & \neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\forall y)[Q(y) \rightarrow R(y)] \rightarrow (\forall z)[P(z) \rightarrow R(z)]) \quad [\text{por (1)}] \\
\equiv & \neg(\neg((\forall x)[\neg P(x) \vee Q(x)] \wedge (\forall y)[\neg Q(y) \vee R(y)]) \vee (\forall z)[\neg P(z) \vee R(z)]) \quad [\text{por (4)}] \\
\equiv & \neg\neg((\forall x)[\neg P(x) \vee Q(x)] \wedge (\forall y)[\neg Q(y) \vee R(y)]) \wedge \neg(\forall z)[\neg P(z) \vee R(z)] \quad [\text{por (6)}] \\
\equiv & ((\forall x)[\neg P(x) \vee Q(x)] \wedge (\forall y)[\neg Q(y) \vee R(y)]) \wedge (\exists z)[\neg(\neg P(z) \vee R(z))] \quad [\text{por (7, 8)}] \\
\equiv & ((\forall x)[\neg P(x) \vee Q(x)] \wedge (\forall y)[\neg Q(y) \vee R(y)]) \wedge (\exists z)[\neg\neg P(z) \wedge \neg R(z)] \quad [\text{por (6)}] \\
\equiv & ((\forall x)[\neg P(x) \vee Q(x)] \wedge (\forall y)[\neg Q(y) \vee R(y)]) \wedge (\exists z)[P(z) \wedge \neg R(z)] \quad [\text{por (7)}] \\
\equiv & (\exists z)[((\forall x)[\neg P(x) \vee Q(x)] \wedge (\forall y)[\neg Q(y) \vee R(y)]) \wedge (P(z) \wedge \neg R(z))] \quad [\text{por (17)}] \\
\equiv & (\exists z)[(\forall x)[(\neg P(x) \vee Q(x)) \wedge (\forall y)[\neg Q(y) \vee R(y)]] \wedge (P(z) \wedge \neg R(z))] \quad [\text{por (11)}] \\
\equiv & (\exists z)(\forall x)[((\neg P(x) \vee Q(x)) \wedge (\forall y)[\neg Q(y) \vee R(y)]) \wedge (P(z) \wedge \neg R(z))] \quad [\text{por (11)}] \\
\equiv & (\exists z)(\forall x)[(\forall y)[(\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))] \wedge (P(z) \wedge \neg R(z))] \quad [\text{por (15)}] \\
\equiv & (\exists z)(\forall x)(\forall y)[((\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))) \wedge (P(z) \wedge \neg R(z))] \quad [\text{por (11)}]
\end{aligned}$$

Forma normal prenexa conjuntiva

- **Fórmula en forma normal prenexa conjuntiva**

- Def.: La fórmula F está en forma normal prenexa conjuntiva (FNPC) si es de la forma $(Q_1x_1) \dots (Q_nx_n)G$, donde $Q_i \in \{\forall, \exists\}$, $n \geq 0$, G no tiene cuantificadores y G está en forma normal conjuntiva.

- **Algoritmo de cálculo de forma normal prenexa conjuntiva:**

- Algoritmo: Aplicando a una fórmula los siguientes pasos se obtiene otra fórmula equivalente y que está en forma normal prenexa conjuntiva rectificada:

1. Calcular una forma normal prenexa rectificada usando las equivalencias (1)–(18)
2. Interiorizar las disyunciones usando las equivalencias

$$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C) \quad (19)$$

$$(A \wedge B) \vee C \equiv (A \vee C) \wedge (B \vee C) \quad (20)$$

- Ejemplo de cálculo de una FNPC de $(\forall x)(\exists y)[P(x) \vee (Q(y) \wedge \neg R(y))]$:

$$\begin{aligned} & (\forall x)(\exists y)[P(x) \vee (Q(y) \wedge \neg R(y))] \\ \equiv & (\forall x)(\exists y)[(P(x) \vee Q(y)) \wedge (P(x) \vee \neg R(y))] \quad [\text{por (19)}] \end{aligned}$$

Forma de Skolem

- **Forma de Skolem:**

- Def.: La fórmula F está en forma de Skolem (FS) si es de la forma $(\forall x_1) \dots (\forall x_n)G$, donde $n \geq 0$ y G no tiene cuantificadores.

- Ejemplos: $(\forall x)(\exists y)P(x, y)$ no está en forma de Skolem
 $(\forall x)P(x, f(x))$ sí está en forma de Skolem
 $(\exists x)Q(x)$ no está en forma de Skolem
 $Q(a)$ sí está en forma de Skolem

- **Equisatisfacibilidad:**

- Def.: Las fórmulas F y G son equisatisfacible si:

F es satisfacible syss G es satisfacible.

Se representa por $F \equiv_{sat} G$

- Ejemplos: $(\exists x)Q(x) \equiv_{sat} Q(a)$
 $(\exists x)Q(x) \not\equiv_{sat} Q(a)$
 $(\forall x)(\exists y)P(x, y) \equiv_{sat} (\forall x)P(x, f(x))$
 $(\forall x)(\exists y)P(x, y) \not\equiv_{sat} (\forall x)P(x, f(x))$

Forma de Skolem

- Propiedades:

- Si a es una constante que no ocurre en F , entonces $(\exists x)F \equiv_{sat} F[x/a]$.
- Si g es un símbolo de función n -aria que no ocurre en F , entonces $(\forall x_1) \dots (\forall x_n)(\exists x)F \equiv_{sat} (\forall x_1) \dots (\forall x_n)F[x/g(x_1, \dots, x_n)]$.

- Algoritmo de cálculo de forma de Skolem:

- Algoritmo: Sea F una fórmula en forma normal prenexa rectificada, la forma de Skolem de F es

$$\text{Sko}(F) = \begin{cases} \text{Sko}(G[x/a]), & \text{si } F \text{ es } (\exists x)G \text{ y } a \text{ es una nueva constante} \\ & \text{que no ocurre en } F \text{ (constante de Skolem);} \\ \text{Sko}(G[x/f(x_1, \dots, x_n)]), & \text{si } F \text{ es } (\forall x_1) \dots (\forall x_n)(\exists x)G \text{ y } f \text{ es un símbolo} \\ & \text{de función } n\text{-aria que no ocurre en } F \\ & \text{(función de Skolem);} \\ F, & \text{si } F \text{ está en forma de Skolem} \end{cases}$$

- Propiedad: Si F es una fórmula en forma normal prenexa rectificada, entonces $\text{Sko}(F)$ está en forma de Skolem y $\text{Sko}(F) \equiv_{sat} F$.

Forma de Skolem: Cálculo de forma de Skolem

- Ejemplos de cálculo de forma de Skolem:

- Ejemplo 1:

$$\begin{aligned} & \text{Sko}((\exists x)(\forall y)(\forall z)(\exists u)(\forall v)(\exists w)P(x, y, z, u, v, w)) \\ &= \text{Sko}((\forall y)(\forall z)(\exists u)(\forall v)(\exists w)P(a, y, z, u, v, w)) \\ &= \text{Sko}((\forall y)(\forall z)(\forall v)(\exists w)P(a, y, z, f(y, z), v, w)) \\ &= \text{Sko}((\forall y)(\forall z)(\forall v)P(a, y, z, f(y, z), v, g(y, z, v))) \\ &= (\forall y)(\forall z)(\forall v)P(a, y, z, f(y, z), v, g(y, z, v)) \end{aligned}$$

- Ejemplo 2:

$$\begin{aligned} & \text{Sko}((\forall x)(\exists y)(\forall z)(\exists w)[\neg P(a, w) \vee Q(f(x), y)]) \\ &= \text{Sko}((\forall x)(\forall z)(\exists w)[\neg P(a, w) \vee Q(f(x), g_1(h(x)))] \\ &= \text{Sko}((\forall x)(\forall z)[\neg P(a, g_2(x, z)) \vee Q(f(x), g_1(h(x)))] \\ &= (\forall x)(\forall z)[\neg P(a, g_2(x, z)) \vee Q(f(x), g_1(h(x)))] \end{aligned}$$

Forma de Skolem: Cálculo de forma de Skolem

- Ejemplo de cálculo de una forma de Skolem de

$$\begin{aligned} & \neg(\exists x)[P(x) \rightarrow (\forall x)P(x)] \\ \equiv & (\forall x)(\exists y)[P(x) \wedge \neg P(y)] \quad [\text{por página ??}] \\ \equiv_{sat} & (\forall x)[P(x) \wedge \neg P(f(x))] \end{aligned}$$

- Ejemplo de cálculo de una forma de Skolem de

$$\begin{aligned} & (\forall x)P(x) \vee (\exists y)Q(y) \\ \equiv & (\forall x)(\exists y)[P(x) \vee Q(y)] \quad [\text{por página ??}] \\ \equiv_{sat} & (\forall x)[P(x) \vee Q(f(x))] \end{aligned}$$

- Ejemplo de cálculo de otra forma de Skolem de

$$\begin{aligned} & (\forall x)P(x) \vee (\exists y)Q(y) \\ \equiv & (\exists y)(\forall x)[P(x) \vee Q(y)] \quad [\text{por página ??}] \\ \equiv_{sat} & (\forall x)[P(x) \vee Q(a)] \end{aligned}$$

- Ejemplo de cálculo de una forma de Skolem de

$$\begin{aligned} & \neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\forall x)[Q(x) \rightarrow R(x)] \rightarrow (\forall x)[P(x) \rightarrow R(x)]) \\ \equiv & (\exists z)(\forall x)(\forall y)[((\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))) \wedge (P(z) \wedge \neg R(z))] \quad [\text{por p. ??}] \\ \equiv_{sat} & (\forall x)(\forall y)[((\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))) \wedge (P(a) \wedge \neg R(a))] \end{aligned}$$

Lógica clausal: sintaxis

- Sintaxis de la lógica clausal

- Un átomo es una fórmula atómica.

Variables sobre átomos: $A, B, C, \dots, A_1, A_2, \dots$

- Un literal es un átomo (A) o la negación de un átomo ($\neg A$).

Variables sobre literales: L, L_1, L_2, \dots

- Una cláusula es un conjunto finito de literales.

Variables sobre cláusulas: C, C_1, C_2, \dots

- La cláusula vacía es el conjunto vacío de literales.

La cláusula vacía se representa por \square .

- Conjuntos finitos de cláusulas.

Variables sobre conjuntos finitos de cláusulas: S, S_1, S_2, \dots

Lógica clausal: semántica

- **Fórmulas correspondientes:**

- Def.: La fórmula correspondiente a la cláusula $\{L_1, \dots, L_n\}$ es

$$(\forall x_1) \dots (\forall x_p)[L_1 \vee \dots \vee L_n],$$

donde x_1, \dots, x_p son las variables libres de $L_1 \vee \dots \vee L_n$.

- Def.: La fórmula correspondiente a la cláusula \square es \perp .

- Def.: La fórmula correspondiente al conjunto de cláusulas

$$\{\{L_1^1, \dots, L_{n_1}^1\}, \dots, \{L_1^m, \dots, L_{n_m}^m\}\}$$

$$(\forall x_1) \dots (\forall x_p)[(L_1^1 \vee \dots \vee L_{n_1}^1) \wedge \dots \wedge (L_1^m \vee \dots \vee L_{n_m}^m)],$$

donde x_1, \dots, x_p son las variables libres de $(L_1^1 \vee \dots \vee L_{n_1}^1) \wedge \dots \wedge (L_1^m \vee \dots \vee L_{n_m}^m)$.

- Def.: La fórmula correspondiente al conjunto de cláusulas \emptyset es \top .

- **Semántica:**

- Def.: En cualquier interpretación $\mathcal{I} = (U, I)$, $I(\top) = 1$ e $I(\perp) = 0$.

- Def.: Los conceptos semánticos relativos a las cláusulas y a los conjuntos de cláusulas son los de sus correspondientes fórmulas.

Forma clausal de una fórmula

- **Forma clausal de una fórmula**

- Def.: Una forma clausal de una fórmula F es un conjunto de cláusulas S tal que $F \equiv_{sat} S$.

- Algoritmo: Aplicando a la fórmula F los siguientes pasos se obtiene S que una forma clausal de F :

1. Sea $F_1 = (\exists y_1) \dots (\exists y_n)F$, donde y_1, \dots, y_n son las variables libres de F .

2. Sea F_2 una forma normal prenexa conjuntiva rectificada de F_1 calculada mediante el algoritmo de la página ??.

3. Sea $F_3 = \text{Sko}(F_2)$, que tiene la forma

$$(\forall x_1) \dots (\forall x_p)[(L_1^1 \vee \dots \vee L_{n_1}^1) \wedge \dots \wedge (L_1^m \vee \dots \vee L_{n_m}^m)],$$

4. Sea $S = \{\{L_1^1, \dots, L_{n_1}^1\}, \dots, \{L_1^m, \dots, L_{n_m}^m\}\}$.

- Prop.: $F \equiv_{sat} F_1 \equiv F_2 \equiv_{sat} F_3 \equiv S$.

Forma clausal de una fórmula: Ejemplos

- Ejemplo de cálculo de una forma clausal de

$$\begin{aligned} & \neg(\exists x)[P(x) \rightarrow (\forall x)P(x)] \\ \equiv_{sat} & (\forall x)[P(x) \wedge \neg P(f(x))] \quad [\text{pág. ??}] \\ \equiv & \{\{P(x)\}, \{\neg P(f(x))\}\} \end{aligned}$$

- Ejemplo de cálculo de una forma clausal de

$$\begin{aligned} & (\forall x)P(x) \vee (\exists y)Q(y) \\ \equiv_{sat} & (\forall x)[P(x) \vee Q(f(x))] \quad [\text{pág. ??}] \\ \equiv & \{\{P(x), Q(f(x))\}\} \end{aligned}$$

- Ejemplo de cálculo de otra forma clausal de

$$\begin{aligned} & (\forall x)P(x) \vee (\exists y)Q(y) \\ \equiv_{sat} & (\forall x)[P(x) \vee Q(a)] \quad [\text{pág. ??}] \\ \equiv & \{\{P(x), Q(a)\}\} \end{aligned}$$

- Ejemplo de cálculo de una forma clausal de

$$\begin{aligned} & \neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\forall x)[Q(x) \rightarrow R(x)] \rightarrow (\forall x)[P(x) \rightarrow R(x)]) \\ \equiv_{sat} & (\forall x)(\forall y)[((\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))) \wedge (P(a) \wedge \neg R(a))] \quad [\text{pág. ??}] \\ \equiv & \{\{\neg P(x), Q(x)\}, \{\neg Q(y), R(y)\}, \{P(a)\}, \{\neg R(a)\}\} \end{aligned}$$

Forma clausal de una fórmula: Ejemplos

- Ejemplo de cálculo de una forma clausal de

$$\begin{aligned} & \neg(\exists x)[P(x, z) \vee (\forall y)Q(x, f(y))] \vee (\forall y)P(g(x, y), z) \\ \equiv_{sat} & (\exists z)[\neg(\exists x)[P(x, z) \vee (\forall y)Q(x, f(y))] \vee (\forall y)P(g(x, y), z)] \quad [1] \\ \equiv & (\exists z)[\neg(\exists x)[P(x, z) \vee (\forall y)Q(x, f(y))] \vee (\forall w)P(g(x, w), z)] \quad [(1)] \\ \equiv & (\exists z)[(\forall x)[\neg(P(x, z) \vee (\forall y)Q(x, f(y)))] \vee (\forall w)P(g(x, w), z)] \quad [(9)] \\ \equiv & (\exists z)[(\forall x)[\neg P(x, z) \wedge \neg(\forall y)Q(x, f(y))] \vee (\forall w)P(g(x, w), z)] \quad [(6)] \\ \equiv & (\exists z)[(\forall x)[\neg P(x, z) \wedge (\exists y)\neg Q(x, f(y))] \vee (\forall w)P(g(x, w), z)] \quad [(8)] \\ \equiv & (\exists z)(\forall x)[(\neg P(x, z) \wedge (\exists y)\neg Q(x, f(y))) \vee (\forall w)P(g(x, w), z)] \quad [(12)] \\ \equiv & (\exists z)(\forall x)[(\exists y)[\neg P(x, z) \wedge \neg Q(x, f(y))] \vee (\forall w)P(g(x, w), z)] \quad [(17)] \\ \equiv & (\exists z)(\forall x)(\exists y)[(\neg P(x, z) \wedge \neg Q(x, f(y))) \vee (\forall w)P(g(x, w), z)] \quad [(14)] \\ \equiv & (\exists z)(\forall x)(\exists y)(\forall w)[(\neg P(x, z) \wedge \neg Q(x, f(y))) \vee P(g(x, w), z)] \quad [(16)] \\ \equiv & (\exists z)(\forall x)(\exists y)(\forall w)[(\neg P(x, z) \vee P(g(x, w), z)) \wedge (\neg Q(x, f(y)) \vee P(g(x, w), z))] \quad [(20)] \\ \equiv_{sat} & (\forall x)(\exists y)(\forall w)[(\neg P(x, a) \vee P(g(x, w), a)) \wedge (\neg Q(x, f(y)) \vee P(g(x, w), a))] \quad [\text{Sko}] \\ \equiv_{sat} & (\forall x)(\forall w)[(\neg P(x, a) \vee P(g(x, w), a)) \wedge (\neg Q(x, f(h(x))) \vee P(g(x, w), a))] \quad [\text{Sko}] \\ \equiv & \{\{\neg P(x, a), P(g(x, w), a)\}, \{\neg Q(x, f(h(x))), P(g(x, w), a)\}\} \end{aligned}$$

Forma clausal de una fórmula: Ejemplos

- Ejemplo de cálculo de una forma clausal de

$$\begin{aligned}
 & \neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\exists x)P(x) \rightarrow (\exists x)Q(x)) \\
 \equiv & \neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\exists y)P(y) \rightarrow (\exists z)Q(z)) \quad [(2)] \\
 \equiv & \neg(\neg((\forall x)[P(x) \rightarrow Q(x)] \wedge (\exists y)P(y)) \vee (\exists z)Q(z)) \quad [(4)] \\
 \equiv & \neg(\neg((\forall x)[\neg P(x) \vee Q(x)] \wedge (\exists y)P(y)) \vee (\exists z)Q(z)) \quad [(4)] \\
 \equiv & \neg\neg((\forall x)[\neg P(x) \vee Q(x)] \wedge (\exists y)P(y)) \wedge \neg(\exists z)Q(z) \quad [(6)] \\
 \equiv & ((\forall x)[\neg P(x) \vee Q(x)] \wedge (\exists y)P(y)) \wedge \neg(\exists z)Q(z) \quad [(7)] \\
 \equiv & ((\forall x)[\neg P(x) \vee Q(x)] \wedge (\exists y)P(y)) \wedge (\forall z)\neg Q(z) \quad [(9)] \\
 \equiv & (\exists y)[(\forall x)[\neg P(x) \vee Q(x)] \wedge P(y)] \wedge (\forall z)\neg Q(z) \quad [(17)] \\
 \equiv & (\exists y)[(\forall x)[\neg P(x) \vee Q(x)] \wedge P(y)] \wedge (\forall z)\neg Q(z) \quad [(13)] \\
 \equiv & (\exists y)[(\forall x)[(\neg P(x) \vee Q(x)) \wedge P(y)]] \wedge (\forall z)\neg Q(z) \quad [(11)] \\
 \equiv & (\exists y)(\forall x)[(\neg P(x) \vee Q(x)) \wedge P(y)] \wedge (\forall z)\neg Q(z) \quad [(11)] \\
 \equiv & (\exists y)(\forall x)(\forall z)[(\neg P(x) \vee Q(x)) \wedge P(y) \wedge \neg Q(z)] \quad [(15)] \\
 \equiv_{sat} & (\forall x)(\forall z)[(\neg P(x) \vee Q(x)) \wedge P(a) \wedge \neg Q(z)] \quad [(15)] \\
 \equiv & \{\{\neg P(x), Q(x)\}, \{P(a)\}, \{\neg Q(z)\}\}
 \end{aligned}$$

Forma clausal de un conjunto de fórmulas

- Equisatisfacibilidad de conjuntos de fórmulas:

- Def.: Los conjuntos de fórmulas S_1 y S_2 son equisatisfacible si:

S_1 es satisfacible syss S_2 es satisfacible.

Se representa por $S_1 \equiv_{sat} S_2$

- Forma clausal de un conjunto de fórmulas

- Def.: Una forma clausal de un conjunto de fórmulas S es un conjunto de cláusulas equisatisfacible con S .

- Prop.: Si S_1, \dots, S_n son formas clausales de F_1, \dots, F_n , entonces $S_1 \cup \dots \cup S_n$ es una forma clausal de $\{F_1, \dots, F_n\}$.

- Ejemplo: Una forma clausal de

$$\{(\forall x)[P(x) \rightarrow Q(x)], (\exists x)P(x), \neg(\exists x)Q(x)\}$$

es

$$\{\{\neg P(x), Q(x)\}, \{P(a)\}, \{\neg Q(z)\}\}.$$

Reducción de consecuencia a insatisfacibilidad de cláusulas

- Reducción de consecuencia a insatisfacibilidad de cláusulas:
 - Prop: Sean S_1, \dots, S_n formas clausales de las fórmulas F_1, \dots, F_n . Si S es una forma clausal de $\neg G$, entonces son equivalentes
 1. $\{F_1, \dots, F_n\} \models G$.
 2. $\{F_1, \dots, F_n, \neg G\}$ es insatisfacible.
 3. $S_1 \cup \dots \cup S_n \cup S$ es insatisfacible.
 - Ejemplos:
 - Ejemplo 1:

$$\{(\forall x)[P(x) \rightarrow Q(x)], (\exists x)P(x)\} \models (\exists x)Q(x)$$
 syss $\{\{\neg P(x), Q(x)\}, \{P(a)\}, \{\neg Q(z)\}\}$ es insatisfacible.
 - Ejemplo 2:

$$\{(\forall x)[P(x) \rightarrow Q(x)], (\forall x)[Q(x) \rightarrow R(x)]\} \models (\forall x)[P(x) \rightarrow R(x)]$$
 syss $\{\{\neg P(x), Q(x)\}, \{\neg Q(y), R(y)\}, \{P(a)\}, \{\neg R(a)\}\}$ es insatisfacible.

Bibliografía

- M.L. Bonet *Apuntes de LPO*. (Univ. Politécnica de Cataluña, 2003) pp. 26–31.
- C.L. Chang y R.C.T. Lee *Symbolic logic and mechanical theorem proving* (Academic Press, 1973) pp. 35–39 y 46–51.
- J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional*. (UNED, 2003) pp. 101–106.
- S. Hölldobler *Computational logic*. (U. de Dresden, 2004) pp. 62–67.
- A. Leitsch *The resolution calculus* (Springer-Verlag, 1997) pp. 11–22.
- R. Nieuwenhuis *Lógica de primer orden*. (U. Politécnica de Cataluña, 2003) pp. 16–17
- M. Ojeda e I. Pérez *Lógica para la computación (Vol. 2: Lógica de Primer Orden)* (Ágora, 1997) pp. 37–49
- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003) pp. 153–160.
- L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 43–47.
- U. Schöning *Logic for computer scientists* (Birkhäuser, 1989) pp. 51–61.

Bibliografía complementaria

- A. Leitsch *Resolution calculus and proof complexity* (Technische Universität Wien, Austria, 1994) pp. 8–12.

Capítulo 9

Modelos de Herbrand

Tema 9: Modelos de Herbrand

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

Reducción de la LPO básica a proposicional

- **Observación:**
 - En este tema sólo se consideran lenguajes de primer orden sin igualdad.
- **Reducción de la LPO básica a proposicional**
 - **Def.:** Una fórmula básica es una fórmula sin variables ni cuantificadores.
 - **Prop.:** Sea S un conjunto de fórmulas básicas. Son equivalentes:
 1. S es consistente en el sentido de la lógica de primer orden.
 2. S es consistente en el sentido de la lógica proposicional.

Reducción de la LPO básica a proposicional

• Ejemplos:

- $\{P(a) \vee P(b), \neg P(b) \vee P(c), P(a) \rightarrow P(c)\}$
es consistente en el sentido de la lógica de primer orden (con modelos $\mathcal{I}_4, \mathcal{I}_6, \mathcal{I}_8$).
- $\{P(a) \vee P(b), \neg P(b) \vee P(c), P(a) \rightarrow P(c), \neg P(c)\}$
es inconsistente en el sentido de la lógica de primer orden.

	P^I	$P(a) \vee P(b)$	$\neg P(b) \vee P(c)$	$P(a) \rightarrow P(c)$	$\neg P(c)$
\mathcal{I}_1	\emptyset	0	1	1	1
\mathcal{I}_2	$\{c^I\}$	0	1	1	0
\mathcal{I}_3	$\{b^I\}$	1	0	1	1
\mathcal{I}_4	$\{b^I, c^I\}$	1	1	1	0
\mathcal{I}_5	$\{a^I\}$	1	1	0	1
\mathcal{I}_6	$\{a^I, c^I\}$	1	1	1	0
\mathcal{I}_7	$\{a^I, b^I\}$	1	0	0	1
\mathcal{I}_8	$\{a^I, b^I, c^I\}$	1	1	1	0

Reducción de la LPO básica a proposicional

• Ejemplos:

- $\{P(a) \vee P(b), \neg P(b) \vee P(c), P(a) \rightarrow P(c)\}$
es consistente en el sentido proposicional (con modelos v_4, v_6, v_8).
- $\{P(a) \vee P(b), \neg P(b) \vee P(c), P(a) \rightarrow P(c), \neg P(c)\}$
es inconsistente en el sentido proposicional.

Se consideran los cambios $P(a)/p, P(b)/q, P(c)/r$

	p	q	r	$p \vee q$	$\neg q \vee r$	$p \rightarrow r$	$\neg r$
v_1	0	0	0	0	1	1	1
v_2	0	0	1	0	1	1	0
v_3	0	1	0	1	0	1	1
v_4	0	1	1	1	1	1	0
v_5	1	0	0	1	1	0	1
v_6	1	0	1	1	1	1	0
v_7	1	1	0	1	0	0	1
v_8	1	1	1	1	1	1	0

Interpretaciones de Herbrand

- **Notación:**

- L representa un lenguaje de primer orden sin igualdad.
- \mathcal{C} es el conjunto de constantes de L .
- \mathcal{F} es el conjunto de símbolos de función de L .
- \mathcal{R} es el conjunto de símbolos de relación de L .
- \mathcal{F}_n es el conjunto de símbolos de función n -aria de L .
- \mathcal{R}_n es el conjunto de símbolos de relación n -aria de L .
- f/n indica que f es un símbolo de función n -aria de L .
- P/n indica que f es un símbolo de relación n -aria de L .

Interpretaciones de Herbrand: Universo de Herbrand

- **Universo de Herbrand de L :**

- **Def.:** El universo de Herbrand de L es el conjunto de los términos básicos de L . Se representa por $\text{UH}(L)$.

- **Prop.:** $\text{UH}(L) = \bigcup_{i \geq 0} H_i(L)$, donde $H_i(L)$ es el nivel i del $\text{UH}(L)$ definido por

$$H_0(L) = \begin{cases} \mathcal{C}, & \text{si } \mathcal{C} \neq \emptyset; \\ \{a\}, & \text{en caso contrario. (} a \text{ es una nueva constante).} \end{cases}$$

$$H_{i+1}(L) = H_i(L) \cup \{f(t_1, \dots, t_n) : f \in \mathcal{F}_n \text{ y } t_1, \dots, t_n \in H_i(L)\}$$

- **Prop.:** $\text{UH}(L)$ es finito syss L no tiene símbolos de función.

- **Ejemplos:**

– Si $\mathcal{C} = \{a, b, c\}$ y $\mathcal{F} = \emptyset$, entonces

$$H_0(L) = \{a, b, c\}$$

$$H_1(L) = \{a, b, c\}$$

⋮

$$\text{UH}(L) = \{a, b, c\}$$

Interpretaciones de Herbrand: Universo de Herbrand

- Ejemplos:

- Si $\mathcal{C} = \emptyset$ y $\mathcal{F} = \{f/1\}$, entonces

$$H_0(L) = \{a\}$$

$$H_1(L) = \{a, f(a)\}$$

$$H_2(L) = \{a, f(a), f(f(a))\}$$

⋮

$$UH(L) = \{a, f(a), f(f(a)), \dots\} = \{f^i(a) : i \in \mathbb{N}\}$$

- Si $\mathcal{C} = \{a, b\}$ y $\mathcal{F} = \{f/1, g/1\}$, entonces

$$H_0(L) = \{a, b\}$$

$$H_1(L) = \{a, b, f(a), f(b), g(a), g(b)\}$$

$$H_2(L) = \{a, b, f(a), f(b), g(a), g(b), f(f(a)), f(f(b)), f(g(a)), f(g(b)), g(f(a)), g(f(b)), g(g(a)), g(g(b))\}$$

⋮

- Si $\mathcal{C} = \{a, b\}$ y $\mathcal{F} = \{f/2\}$, entonces

$$H_0(L) = \{a, b\}$$

$$H_1(L) = \{a, b, f(a, a), f(a, b), f(b, a), f(b, b)\}$$

$$H_2(L) = \{a, b, f(a, a), f(a, b), f(b, a), f(b, b), f(a, f(a, a)), f(a, f(a, b)), \dots\}$$

⋮

Interpretaciones de Herbrand: Base de Herbrand

- Base de Herbrand:

- Def.: La base de Herbrand de L es el conjunto de los átomos básicos de L . Se representa por $BH(L)$.

- Prop.: $BH(L) = \{P(t_1, \dots, t_n) : P \in \mathcal{R}_n \text{ y } t_1, \dots, t_n \in UH(L)\}$.

- Prop.: $BH(L)$ es finita syss L no tiene símbolos de función.

- Ejemplos:

- Si $\mathcal{C} = \{a, b, c\}$, $\mathcal{F} = \emptyset$ y $\mathcal{R} = \{P/1\}$, entonces

$$UH(L) = \{a, b, c\}$$

$$BH(L) = \{P(a), P(b), P(c)\}$$

- Si $\mathcal{C} = \{a\}$, $\mathcal{F} = \{f/1\}$ y $\mathcal{R} = \{P/1, Q/1, R/1\}$, entonces

$$UH(L) = \{a, f(a), f(f(a)), \dots\} = \{f^i(a) : i \in \mathbb{N}\}$$

$$BH(L) = \{P(a), Q(a), R(a), P(f(a)), Q(f(a)), R(f(a)), \dots\}$$

Interpretaciones de Herbrand

- **Interpretaciones de Herbrand:**

- **Def.:** Una interpretación $\mathcal{I} = (U, I)$ de L es de Herbrand si

- U es el universo de Herbrand de L ;
- $I(c) = c$, para cada constante c de L ;
- $I(f) = f$, para cada símbolo de función f de L .

- **Prop.:** Sea \mathcal{I} una interpretación de Herbrand de L . Si t es un término básico de L , entonces $\mathcal{I}(t) = t$.

- **Prop.:** Una interpretación de Herbrand queda determinada por un subconjunto de la base de Herbrand, el conjunto de átomos básicos verdaderos en esa interpretación.

- **Ejemplo:** Si $\mathcal{C} = \{a, b, c\}$, $\mathcal{F} = \emptyset$ y $\mathcal{R} = \{P/1\}$, entonces las interpretaciones de Herbrand de L son

n	1	2	3	4	5	6	7	8
$I_n(P)$	\emptyset	$\{c\}$	$\{b\}$	$\{b, c\}$	$\{a\}$	$\{a, c\}$	$\{a, b\}$	$\{a, b, c\}$
IH_n	\emptyset	$\{P(c)\}$	$\{P(b)\}$	$\{P(b), P(c)\}$	$\{P(a)\}$	$\{P(a), P(c)\}$	$\{P(a), P(b)\}$	$\{P(a), P(b), P(c)\}$

Modelos de Herbrand

- **Modelos de Herbrand:**

- **Nota:** Las definiciones de universo de Herbrand, base de Herbrand e interpretación de Herbrand definidas para un lenguaje se extienden a fórmulas y conjuntos de fórmulas considerando el lenguaje formado por los símbolos no lógicos que aparecen.

- **Def.:** Un modelo de Herbrand de una fórmula F es una interpretación de Herbrand de F que es modelo de F .

- **Def.:** Un modelo de Herbrand de un conjunto de fórmulas S es una interpretación de Herbrand de S que es modelo de S .

- **Ejemplo:** Los modelos de Herbrand de $\{P(a) \vee P(b), \neg P(b) \vee P(c), P(a) \rightarrow P(c)\}$ son $\{P(b), P(c)\}$, $\{P(a), P(c)\}$ y $\{P(a), P(b), P(c)\}$ (ver página ??).

- **Ejemplo:** Sea $S = \{(\forall x)(\forall y)[Q(b, x) \rightarrow P(a) \vee R(y)], P(b) \rightarrow \neg(\exists z)(\exists u)Q(z, u)\}$. Entonces, $\text{UH}(S) = \{a, b\}$

$$\text{BH}(S) = \{P(a), P(b), Q(a, a), Q(a, b), Q(b, a), Q(b, b), R(a), R(b)\}$$

Un modelo de Herbrand de S es $\{P(a)\}$.

Interpretación de Herbrand correspondiente

• Interpretación de Herbrand correspondiente a una interpretación:

- Sea $S = \{\{\neg Q(b, x), P(a), R(y)\}, \{\neg P(b), \neg Q(z, u)\}\}$ e $\mathcal{I} = (\{1, 2\}, I)$ con $a^I = 1, b^I = 2, P^I = \{1\}, Q^I = \{(1, 1), (2, 2)\}, R^I = \{2\}$. Entonces, $\mathcal{I} \models S$.

Cálculo de la interpretación de Herbrand \mathcal{I}^* correspondiente a \mathcal{I} :

$$\begin{aligned}
 \mathcal{I}^* &= (\text{UH}(S), I^*) \\
 \text{UH}(S) &= \{a, b\} \\
 \text{BH}(S) &= \{P(a), P(b), Q(a, a), Q(a, b), Q(b, a), Q(b, b), R(a), R(b)\} \\
 I^*(P(a)) &= P^I(a^I) = P^I(1) = \text{V} \\
 I^*(P(b)) &= P^I(b^I) = P^I(2) = \text{F} \\
 I^*(Q(a, a)) &= Q^I(a^I, a^I) = Q^I(1, 1) = \text{V} \\
 I^*(Q(a, b)) &= Q^I(a^I, b^I) = Q^I(1, 2) = \text{F} \\
 I^*(Q(b, a)) &= Q^I(b^I, a^I) = Q^I(2, 1) = \text{F} \\
 I^*(Q(b, b)) &= Q^I(b^I, b^I) = Q^I(2, 2) = \text{V} \\
 I^*(R(a)) &= R^I(a^I) = R^I(1) = \text{F} \\
 I^*(R(b)) &= R^I(b^I) = R^I(2) = \text{V}
 \end{aligned}$$

$$I^* = \{P(a), Q(a, a), Q(b, b), R(b)\}$$

$$\mathcal{I}^* \models S.$$

Interpretación de Herbrand correspondiente

• Interpretación de Herbrand correspondiente a una interpretación:

- Sea S el conjunto de cláusulas $\{\{P(a)\}, \{Q(y, f(a))\}\}$ e $\mathcal{I} = (\{1, 2\}, I)$ con $a^I = 1, f^I = \{(1, 2), (2, 1)\}, P^I = \{1\}, Q^I = \{(1, 2), (2, 2)\}$. Entonces, $\mathcal{I} \models S$.

Cálculo de la interpretación de Herbrand \mathcal{I}^* correspondiente a \mathcal{I} :

$$\begin{aligned}
 \mathcal{I}^* &= (\text{UH}(S), I^*) \\
 \text{UH}(S) &= \{a, f(a), f(f(a)), \dots\} = \{f^i(a) : i \in \mathbb{N}\} \\
 \text{BH}(S) &= \{P(f^n(a)) : n \in \mathbb{N}\} \cup \{Q(f^n(a), f^m(a)) : n, m \in \mathbb{N}\} \\
 I^*(P(a)) &= P^I(a^I) = P^I(1) = \text{V} \\
 I^*(P(f(a))) &= P^I(f^I(a^I)) = P^I(f^I(1)) = P^I(2) = \text{F} \\
 I^*(P(f(f(a)))) &= P^I(f^I(f^I(a^I))) = P^I(1) = \text{V} \\
 I^*(P(f^n(a))) &= \begin{cases} \text{V}, & \text{si } n \text{ es par;} \\ \text{F}, & \text{en caso contrario.} \end{cases} \\
 I^*(Q(f^n(a), f^m(a))) &= \begin{cases} \text{V}, & \text{si } m \text{ es impar;} \\ \text{F}, & \text{en caso contrario.} \end{cases}
 \end{aligned}$$

$$I^* = \{P(f^{2n}(a)) : n \in \mathbb{N}\} \cup \{Q(f^n(a), f^{2m+1}(a)) : n, m \in \mathbb{N}\}$$

$$\mathcal{I}^* \models S.$$

Consistencia mediante modelos de Herbrand

- **Consistencia mediante modelos de Herbrand:**
 - Prop.: Sea S un conjunto de fórmulas básicas. Son equivalentes:
 1. S es consistente.
 2. S tiene un modelo de Herbrand.
 - Prop.: Sea S un conjunto de cláusulas. Si \mathcal{I}^* es una interpretación de Herbrand correspondiente a un modelo \mathcal{I} de S , entonces \mathcal{I}^* es un modelo de S .
 - Prop.: Sea S un conjunto de cláusulas. Son equivalentes:
 1. S es consistente.
 2. S tiene un modelo de Herbrand.
 - Prop.: Sea S un conjunto de cláusulas. Son equivalentes:
 1. S es inconsistente.
 2. S no tiene ningún modelo de Herbrand.
 - Prop.: Existen conjuntos de fórmulas consistentes sin modelos de Herbrand.

Consistencia mediante modelos de Herbrand

- **Ejemplo de conjunto consistente sin modelos de Herbrand:**
 - Sea $S = \{(\exists x)P(x), \neg P(a)\}$. Entonces,
 - S es consistente.

$$\mathcal{I} \models S \text{ con } \mathcal{I} = (\{1, 2\}, I), a^I = 1 \text{ y } P^I = \{2\}.$$
 - S no tiene modelos de Herbrand

$$\text{UH}(S) = \{a\}$$

$$\text{BH}(S) = \{P(a)\}$$

Las interpretaciones de Herbrand de S son \emptyset y $\{P(a)\}$.

$$\emptyset \not\models S$$

$$\{P(a)\} \not\models S$$

Extensiones de Herbrand

- **Instancias básicas de una cláusula**

- Def.: Una sustitución σ (de L) es una aplicación $\sigma : \text{Var} \rightarrow \text{Térm}(L)$.
- Def.: Sea $C = \{L_1, \dots, L_n\}$ una cláusula de L y σ una sustitución de L . Entonces, $C\sigma = \{L_1\sigma, \dots, L_n\sigma\}$ es una instancia de C .
- Ejemplo: Sea $C = \{P(x, a), \neg P(x, f(y))\}$.
 $C[x/a, y/f(a)] = \{P(f(a), a), \neg P(f(a), f(f(a)))\}$
- Def.: $C\sigma$ es una instancia básica de C si todos los literales de $C\sigma$ son básicos.
- Ejemplo: Sea $C = \{P(x, a), \neg P(x, f(y))\}$.
 $\{P(f(a), a), \neg P(f(a), f(f(a)))\}$ es una instancia básica de C .
 $\{P(f(a), a), \neg P(f(f(a)), f(a))\}$ no es una instancia básica de C .
 $\{P(x, a), \neg P(f(f(a)), f(a))\}$ no es una instancia básica de C .

Extensiones de Herbrand

- **Extensiones de Herbrand**

- Def.: La extensión de Herbrand de un conjunto de cláusulas S es el conjunto de fórmulas

$$\text{EH}(S) = \{C\sigma : C \in S \text{ y, para toda variable } x \text{ en } C, \sigma(x) \in \text{UH}(S)\}.$$
- Prop.: $\text{EH}(L) = \bigcup_{i \geq 0} \text{EH}_i(L)$, donde $\text{EH}_i(L)$ es el nivel i de la $\text{EH}(L)$ definido por

$$\text{EH}_i(S) = \{C\sigma : C \in S \text{ y, para toda variable } x \text{ en } C, \sigma(x) \in \text{UH}_i(S)\}.$$
- Ejemplo: Sea $S = \{\{P(x)\}, \{\neg P(f(x))\}\}$ (p. 8.17). Entonces,

$$\begin{aligned} \text{EH}_0(S) &= \{\{P(a)\}, \{\neg P(f(a))\}\} \\ \text{EH}_1(S) &= \text{EH}_0(S) \cup \{\{P(f(a))\}, \{\neg P(f(f(a)))\}\} \\ \text{EH}_2(S) &= \text{EH}_1(S) \cup \{\{P(f(f(a)))\}, \{\neg P(f(f(f(a))))\}\} \end{aligned}$$
- Ejemplo: Sea $S = \{\{\neg P(x), Q(x)\}, \{P(a)\}, \{\neg Q(z)\}\}$ (p. 8.21).
 Entonces, $\text{EH}(S) = \{\{\neg P(a), Q(a)\}, \{P(a)\}, \{\neg Q(a)\}\}.$
- Ejemplo: Sea $S = \{\{\neg P(x), Q(x)\}, \{\neg Q(y), R(y)\}, \{P(a)\}, \{\neg R(a)\}\}$ (p. 8.21).
 Entonces, $\text{EH}(S) = \{\{\neg P(a), Q(a)\}, \{\neg Q(a), R(a)\}, \{P(a)\}, \{\neg R(a)\}\}.$

Teorema de Herbrand

- **Teorema de Herbrand:**
 - Teorema de Herbrand: Sea S un conjunto de cláusulas. Son equivalentes:
 1. S es consistente.
 2. $\text{EH}(S)$ es consistente (en el sentido proposicional).
 - Prop.: Sea S un conjunto de cláusulas. Entonces, son equivalentes
 1. S es inconsistente.
 2. $\text{EH}(S)$ tiene un subconjunto finito inconsistente (en el sentido proposicional).
 3. Para algún i , $\text{EH}_i(S)$ es inconsistente (en el sentido proposicional).

Teorema de Herbrand

- **Semidecisión de la consistencia mediante el teorema de Herbrand:**
 - Entrada: Un conjunto de cláusulas S .
 - Procedimiento:
 1. Hacer $i := 0$.
 2. Calcular $\text{EH}_i(S)$.
 3. Si $\text{EH}_i(S)$ es inconsistente (en el sentido proposicional), parar e indicar que S es inconsistente.
 4. Si $\text{EH}_i(S)$ es consistente (en el sentido proposicional), hacer $i := i + 1$ y volver al paso 2.

Teorema de Herbrand

- Ejemplo: $S = \{\{\neg P(x), Q(x)\}, \{P(a)\}, \{\neg Q(z)\}\}$ (p. ??) es inconsistente.

$\text{EH}_0(S) = \{\{\neg P(a), Q(a)\}, \{P(a)\}, \{\neg Q(a)\}\}$ es inconsistente.

- 1 $\{\neg P(a), Q(a)\}$
- 2 $\{P(a)\}$
- 3 $\{\neg Q(a)\}$
- 4 $\{Q(a)\}$ Res 1, 2
- 5 \square Res 3, 4

- Ejemplo: $S = \{\{\neg P(x), Q(x)\}, \{\neg Q(y), R(y)\}, \{P(a)\}, \{\neg R(a)\}\}$ es inconsistente.

$\text{EH}_0(S) = \{\{\neg P(a), Q(a)\}, \{\neg Q(a), R(a)\}, \{P(a)\}, \{\neg R(a)\}\}$.

- 1 $\{\neg P(a), Q(a)\}$
- 2 $\{\neg Q(a), R(a)\}$
- 3 $\{P(a)\}$
- 4 $\{\neg R(a)\}$
- 5 $\{Q(a)\}$ Res 1, 3
- 6 $\{R(a)\}$ Res 5, 2
- 7 \square Res 6, 4

Teorema de Herbrand

- Ejemplo: $S = \{\{P(x)\}, \{\neg P(f(x))\}\}$ es inconsistente (p. ??).

– $\text{EH}_0(S) = \{\{P(a)\}, \{\neg P(f(a))\}\}$ es consistente

$\mathcal{I} = \{P(a)\} \models \text{EH}_0(S)$

– $\text{EH}_1(S) = \text{EH}_0(S) \cup \{\{P(f(a))\}, \{\neg P(f(f(a)))\}\}$ es inconsistente.

- 1 $\{P(f(a))\}$
- 2 $\{\neg P(f(a))\}$
- 3 \square Res 1, 2

- Ejemplo: $S = \{\{\neg P(x), Q(f(x), x)\}, \{P(g(b))\}, \{\neg Q(y, z)\}\}$ es inconsistente.

– $S' = \{\{\neg P(g(b)), Q(f(g(b)), g(b))\}, \{P(g(b))\}, \{\neg Q(f(g(b)), g(b))\}\} \subset \text{EH}(S)$
es inconsistente.

- 1 $\{\neg P(g(b)), Q(f(g(b)), g(b))\}$
- 2 $\{P(g(b))\}$
- 3 $\{\neg Q(f(g(b)), g(b))\}$
- 4 $\{Q(f(g(b)), g(b))\}$ Res 1, 2
- 5 \square Res 3, 3

Bibliografía

- M.L. Bonet *Apuntes de LPO*. (Univ. Politécnica de Cataluña, 2003) pp. 31–34.
- C.L. Chang y R.C.T. Lee *Symbolic logic and mechanical theorem proving* (Academic Press, 1973) pp. 51–62.
- M. Ojeda e I. Pérez *Lógica para la computación (Vol. 2: Lógica de Primer Orden)* (Ágora, 1997) pp. 59–74.
- E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003) pp. 160–169.
- L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 47–50.
- U. Schöning *Logic for computer scientists* (Birkäuser, 1989) pp. 70–78.

Bibliografía complementaria

- A. Leitsch *Resolution calculus and proof complexity* (Technische Universität Wien, Austria, 1994) pp. 12–15.

Capítulo 10

Resolución en lógica de primer orden

Tema 10: Resolución

José A. Alonso Jiménez
Andrés Cordon Franco

Dpto. de Ciencias de la Computación e Inteligencia Artificial

UNIVERSIDAD DE SEVILLA

Resolución no restringida

- **Resolvente no restringida**

- **Def.:** La cláusula C es una resolvente no restringida de las cláusulas C_1 y C_2 si existen dos sustituciones σ_1 y σ_2 y un literal $L \in C_1\sigma_1$ tales que $L^c \in C_2\sigma_2$ y

$$C = (C_1\sigma_1 \setminus \{L\sigma_1\}) \cup (C_2\sigma_2 \setminus \{L^c\sigma_2\}).$$

- Sean
 - $C_1 = \{\neg R(x, f(y)), P(x, f(z)), Q(f(z))\},$
 - $C_2 = \{R(z, f(z)), R(f(y), u), Q(z)\},$
 - $\sigma_1 = [x/f(y), y/f(y), z/y],$
 - $C_1\sigma_1 = \{\neg R(f(y), f(f(y))), P(f(f(y)), f(y)), Q(f(y))\},$
 - $\sigma_2 = [z/f(y), u/f(f(y))],$
 - $C_2\sigma_2 = \{R(f(y), f(f(y))), R(f(y), f(f(y))), Q(f(y))\},$
 - $L = \neg R(f(y), f(f(y)))$

Entonces, una resolvente no restringida de C_1 y C_2 es

$$C = \{P(f(f(y)), f(y)), Q(f(y))\}$$

Resolución no restringida

- Ejemplos de refutación por resolución no restringida:

- Refutación no restringida de

$$S = \{\{-P(x), Q(f(x), x)\}, \{P(g(b))\}, \{\neg Q(y, z)\}\}$$

- | | | |
|---|-----------------------------|---------------------|
| 1 | $\{\neg P(x), Q(f(x), x)\}$ | Hipótesis |
| 2 | $\{P(g(b))\}$ | Hipótesis |
| 3 | $\{\neg Q(y, z)\}$ | Hipótesis |
| 4 | $\{\neg Q(f(g(b)), g(b))\}$ | Resolvente de 1 y 2 |
| 5 | \square | Resolvente de 3 y 4 |

Resolución no restringida

- Refutación no restringida de

$$\{\{R(c)\}, \{R(f(c))\}, \{\neg R(x), P(x)\}, \{\neg P(x), Q(x)\}, \{\neg Q(c), \neg Q(f(c)), \neg R(c)\}\}$$

- | | | |
|----|--|----------------------|
| 1 | $\{R(c)\}$ | Hipótesis |
| 2 | $\{R(f(c))\}$ | Hipótesis |
| 3 | $\{\neg R(x), P(x)\}$ | Hipótesis |
| 4 | $\{\neg P(x), Q(x)\}$ | Hipótesis |
| 5 | $\{\neg Q(c), \neg Q(f(c)), \neg R(c)\}$ | Hipótesis |
| 6 | $\{\neg R(x), Q(x)\}$ | Resolvente de 3 y 4 |
| 7 | $\{Q(c)\}$ | Resolvente de 1 y 6 |
| 8 | $\{Q(f(c))\}$ | Resolvente de 2 y 6 |
| 9 | $\{\neg Q(f(c)), \neg R(c)\}$ | Resolvente de 5 y 7 |
| 10 | $\{\neg R(c)\}$ | Resolvente de 8 y 9 |
| 10 | \square | Resolvente de 1 y 10 |

Resolución no restringida

• Definiciones

- Sea S un conjunto de cláusulas.
- La sucesión (C_1, \dots, C_n) es una demostración por resolución no restringida de la cláusula C a partir de S si $C = C_n$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
 - $C_i \in S$;
 - existen $j, k < i$ tales que C_i es una resolvente no restringida de C_j y C_k
- La cláusula C es demostrable por resolución no restringida a partir de S si existe una demostración por resolución de C a partir de S .
- Una refutación por resolución no restringida de S es una demostración por resolución de la cláusula vacía a partir de S .
- Se dice que S es refutable por resolución no restringida si existe una refutación por resolución a partir de S .

Resolución no restringida

• Demostraciones por resolución

- Def.: Sean S_1, \dots, S_n formas clausales de las fórmulas F_1, \dots, F_n y S una forma clausal de $\neg F$. Una demostración por resolución no restringida de F a partir de $\{F_1, \dots, F_n\}$ es una refutación por resolución no restringida de $S_1 \cup \dots \cup S_n \cup S$.
- Def.: La fórmula F es demostrable por resolución no restringida a partir de $\{F_1, \dots, F_n\}$ si existe una demostración por resolución no restringida de F a partir de $\{F_1, \dots, F_n\}$.
Se representa por $\{F_1, \dots, F_n\} \vdash_{ResNoR} F$.
- Ejemplo: Demostración por resolución no restringida de $(\exists x)P(g(x))$ a partir de $\{(\forall x)[P(x) \rightarrow Q(f(x), x)], \neg(\exists y)(\exists z)Q(y, z)\}$
 - 1 $\{\neg P(x), Q(f(x), x)\}$ Hipótesis
 - 2 $\{\neg Q(y, z)\}$ Hipótesis
 - 3 $\{P(g(b))\}$, Hipótesis
 - 4 $\{\neg Q(f(b(b)), g(b))\}$ Resolvente de 1 y 3
 - 5 \square Resolvente de 2 y 4

Adecuación y completitud de la resolución

- **Propiedades:**

- Si C es una resolvente restringida de C_1 y C_2 , entonces $\{C_1, C_2\} \models C$.
- Si $\square \in S$, entonces S es inconsistente.
- Si el conjunto de cláusulas S es refutable por resolución no restringida, entonces S es inconsistente.
- Teor.: El cálculo de resolución no restringida es adecuado y completo; es decir,

$$\text{Adecuado: } S \vdash_{ResNoR} F \quad \Longrightarrow \quad S \models F$$

$$\text{Completo: } S \models F \quad \Longrightarrow \quad S \vdash_{ResNoR} F$$

Unificación: Unificadores

- **Unificador:**

- Def.: La sustitución σ es un unificador de los términos t_1 y t_2 si $t_1\sigma = t_2\sigma$.
- Def.: Los términos t_1 y t_2 son unificables si tienen algún unificador.
- Def.: t es una instancia común de t_1 y t_2 si existe una sustitución σ tal que $t = t_1\sigma = t_2\sigma$.
- Ejemplos:

t_1	t_2	Unificador	Instancia común
$f(x, g(z))$	$f(g(y), x)$	$[x/g(z), y/z]$	$f(g(z), g(z))$
$f(x, g(z))$	$f(g(y), x)$	$[x/g(y), z/y]$	$f(g(y), g(y))$
$f(x, g(z))$	$f(g(y), x)$	$[x/g(a), y/a]$	$f(g(a), g(a))$
$f(x, y)$	$f(y, x)$	$[x/a, y/a]$	$f(a, a)$
$f(x, y)$	$f(y, x)$	$[y/x]$	$f(x, x)$
$f(x, y)$	$g(a, b)$	No tiene	No tiene
$f(x, x)$	$f(a, b)$	No tiene	No tiene
$f(x)$	$f(g(x))$	No tiene	No tiene

- **Nota:** Las anteriores definiciones se extienden a conjuntos de términos y de literales.

Unificación: Composición de sustituciones

- Composición de sustituciones:

- Def.: La composición de las sustituciones σ_1 y σ_2 es la sustitución $\sigma_1\sigma_2$ definida por $x(\sigma_1\sigma_2) = (x\sigma_1)\sigma_2$, para toda variable x .

- Ejemplo: Si $\sigma_1 = [x/f(z, a), y/w]$ y $\sigma_2 = [x/b, z/g(w)]$, entonces

- $x\sigma_1\sigma_2 = (x\sigma_1)\sigma_2 = f(z, a)\sigma_2 = f(z\sigma_2, a\sigma_2) = f(g(w), a)$

- $y\sigma_1\sigma_2 = (y\sigma_1)\sigma_2 = w\sigma_2 = w$

- $z\sigma_1\sigma_2 = (z\sigma_1)\sigma_2 = z\sigma_2 = g(w)$

- $w\sigma_1\sigma_2 = (w\sigma_1)\sigma_2 = w\sigma_2 = w$

Por tanto, $\sigma_1\sigma_2 = [x/f(g(w), a), y/w, z/g(w)]$.

- Def.: La sustitución identidad es la sustitución ϵ tal que, para todo x , $x\epsilon = x$.

- Propiedades:

1. Asociativa: $\sigma_1(\sigma_2\sigma_3) = (\sigma_1\sigma_2)\sigma_3$

2. Neutro: $\sigma\epsilon = \epsilon\sigma = \sigma$.

Unificación: Comparación de sustituciones

- Comparación de sustituciones:

- Def.: La sustitución σ_1 es más general que la σ_2 si existe una sustitución σ_3 tal que $\sigma_2 = \sigma_1\sigma_3$. Se representa por $\sigma_2 \leq \sigma_1$.

- Def.: Las sustituciones σ_1 y σ_2 son equivalentes si $\sigma_1 \leq \sigma_2$ y $\sigma_2 \leq \sigma_1$. Se representa por $\sigma_1 \equiv \sigma_2$.

- Ejemplos: Sean $\sigma_1 = [x/g(z), y/z]$, $\sigma_2 = [x/g(y), z/y]$ y $\sigma_3 = [x/g(a), y/a]$. Entonces,

1. $\sigma_1 = \sigma_2[y/z]$

2. $\sigma_2 = \sigma_1[z/y]$

3. $\sigma_3 = \sigma_1[z/a]$

4. $\sigma_1 \equiv \sigma_2$

5. $\sigma_3 \leq \sigma_1$

- Ejemplo: $[x/a, y/a] \leq [y/x]$, ya que $[x/a, y/a] = [y/x][x/a, y/a]$.

Unificación: Unificador de máxima generalidad

- **Unificador de máxima generalidad:**
 - **Def.:** La sustitución σ es un unificador de máxima generalidad (UMG) de los términos t_1 y t_2 si
 - σ es un unificador de t_1 y t_2 .
 - σ es más general que cualquier unificador de t_1 y t_2 .
 - **Ejemplos:**
 1. $[x/g(z), y/z]$ es un UMG de $f(x, g(z))$ y $f(g(y), x)$.
 2. $[x/g(y), z/y]$ es un UMG de $f(x, g(z))$ y $f(g(y), x)$.
 3. $[x/g(a), y/a]$ no es un UMG de $f(x, g(z))$ y $f(g(y), x)$.
 - **Nota:** Las anterior definición se extienden a conjuntos de términos y de literales.

Unificación: Algoritmo de unificación

- **Notación de lista:**
 - (a_1, \dots, a_n) representa una lista cuyos elementos son a_1, \dots, a_n .
 - $(a|R)$ representa una lista cuyo primer elemento es a y resto es R .
 - $()$ representa la lista vacía.
- **Unificadores de listas de términos:**
 - **Def.:** σ es un unificador de $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ si $s_1\sigma = t_1\sigma, \dots, s_n\sigma = t_n\sigma$.
 - **Def.:** $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ son unificables si tienen algún unificador.
 - **Def.:** σ es un unificador de máxima generalidad (UMG) de $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ si σ es un unificador de $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ más general que cualquier otro.
- **Aplicación de una sustitución a una lista de ecuaciones:**
 - $(s_1 = t_1, \dots, s_n = t_n)\sigma = (s_1\sigma = t_1\sigma, \dots, s_n\sigma = t_n\sigma)$.
- **Algoritmo de unificación de listas de términos:**
 - **Entrada:** Una lista de ecuaciones $L = (s_1 = t_1, \dots, s_n = t_n)$ y una sustitución σ .
 - **Salida:** Un UMG de las listas $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$, si son unificables; “No unificables”, en caso contrario.

Unificación: Algoritmo de unificación

- Procedimiento $\text{unif}(L, \sigma)$:
 1. Si $L = ()$, entonces $\text{unif}(L, \sigma) = \sigma$.
 2. Si $L = (t = t|L')$, entonces $\text{unif}(L, \sigma) = \text{unif}(L', \sigma)$.
 3. Si $L = (f(t_1, \dots, t_m) = f(t'_1, \dots, t'_m)|L')$, entonces
 $\text{unif}(L, \sigma) = \text{unif}((t_1 = t'_1, \dots, t_m = t'_m|L'), \sigma)$.
 4. Si $L = (x = t|L')$ (ó $L = (t = x|L')$) y x no aparece en t , entonces
 $\text{unif}(L, \sigma) = \text{unif}(L'[x/t], \sigma[x/t])$.
 5. Si $L = (x = t|L')$ (ó $L = (t = x|L')$) y x aparece en t , entonces
 $\text{unif}(L, \sigma) = \text{"No unificables"}$.
 6. Si $L = (f(t_1, \dots, t_m) = g(t'_1, \dots, t'_m)|L')$, entonces
 $\text{unif}(L, \sigma) = \text{"No unificables"}$.
 7. Si $L = (f(t_1, \dots, t_m) = f(t'_1, \dots, t'_p)|L')$ y $m \neq p$, entonces
 $\text{unif}(L, \sigma) = \text{"No unificables"}$.

Unificación: Algoritmo de unificación

- Algoritmo de unificación de dos términos:
 - Entrada: Dos términos t_1 y t_2 .
 - Salida: Un UMG de t_1 y t_2 , si son unificables;
 "No unificables" , en caso contrario.
 - Procedimiento: $\text{unif}((t_1 = t_2), \epsilon)$.
 - Ejemplo 1: Unificar $f(x, g(z))$ y $f(g(y), x)$:

$$\begin{aligned} & \text{unif}((f(x, g(z)) = f(g(y), x)), \epsilon) \\ &= \text{unif}((x = g(y), g(z) = x), \epsilon) && \text{por 3} \\ &= \text{unif}((g(z) = x)[x/g(y)], \epsilon[x/g(y)]) && \text{por 4} \\ &= \text{unif}((g(z) = g(y)), [x/g(y)]) \\ &= \text{unif}((z = y), [x/g(y)]) && \text{por 3} \\ &= \text{unif}(), [x/g(y)][z/y] && \text{por 4} \\ &= \text{unif}(), [x/g(y), z/y] \\ &= [x/g(y), z/y] && \text{por 1} \end{aligned}$$

Unificación: Algoritmo de unificación

- Ejemplo 2: Unificar $f(x, b)$ y $f(a, y)$:

$$\begin{aligned}
 & \text{unif}((f(x, b) = f(a, y)), \epsilon) \\
 = & \text{unif}((x = a, b = y), \epsilon) && \text{por 3} \\
 = & \text{unif}((b = y)[x/a], \epsilon[x/a]) && \text{por 4} \\
 = & \text{unif}((b = y), [x/a]) \\
 = & \text{unif}(), [x/a][y/b] && \text{por 4} \\
 = & [x/a, y/b] && \text{por 1}
 \end{aligned}$$

- Ejemplo 3: Unificar $f(x, x)$ y $f(a, b)$:

$$\begin{aligned}
 & \text{unif}((f(x, x) = f(a, b)), \epsilon) \\
 = & \text{unif}((x = a, x = b), \epsilon) && \text{por 3} \\
 = & \text{unif}((x = b)[x/a], \epsilon[x/a]) && \text{por 4} \\
 = & \text{unif}((a = b), [x/a]) \\
 = & \text{“No unificable”} && \text{por 6}
 \end{aligned}$$

- Ejemplo 4: Unificar $f(x, g(y))$ y $f(y, x)$:

$$\begin{aligned}
 & \text{unif}((f(x, g(y)) = f(y, x)), \epsilon) \\
 = & \text{unif}((x = y, g(y) = x), \epsilon) && \text{por 3} \\
 = & \text{unif}((g(y) = x)[x/y], \epsilon[x/y]) && \text{por 4} \\
 = & \text{unif}((g(y) = y), [x/y]) \\
 = & \text{“No unificable”} && \text{por 5}
 \end{aligned}$$

Unificación: Algoritmo de unificación

- Ejemplo 5: Unificar $j(w, a, h(w))$ y $j(f(x, y), x, z)$

$$\begin{aligned}
 & \text{unif}((j(w, a, h(w)) = j(f(x, y), x, z)), \epsilon) \\
 = & \text{unif}((w = f(x, y), a = x, h(w) = z), \epsilon) && \text{por 3} \\
 = & \text{unif}((a = x, h(w) = z)[w/f(x, y)], \epsilon[w/f(x, y)]) && \text{por 4} \\
 = & \text{unif}((a = x, h(f(x, y)) = z), [w/f(x, y)]) \\
 = & \text{unif}((h(f(x, y)) = z)[x/a], [w/f(x, y)][x/a]) && \text{por 4} \\
 = & \text{unif}((h(f(a, y)) = z), [w/f(a, y), x/a]) \\
 = & \text{unif}(), [w/f(a, y), x/a][z/h(f(a, y))] && \text{por 4} \\
 = & [w/f(a, y), x/a, z/h(f(a, y))] && \text{por 1}
 \end{aligned}$$

- Ejemplo 6: Unificar $j(w, a, h(w))$ y $j(f(x, y), x, y)$

$$\begin{aligned}
 & \text{unif}((j(w, a, h(w)) = j(f(x, y), x, y)), \epsilon) \\
 = & \text{unif}((w = f(x, y), a = x, h(w) = y), \epsilon) && \text{por 3} \\
 = & \text{unif}((a = x, h(w) = y)[w/f(x, y)], \epsilon[w/f(x, y)]) && \text{por 4} \\
 = & \text{unif}((a = x, h(f(x, y)) = y), [w/f(x, y)]) \\
 = & \text{unif}((h(f(x, y)) = y)[x/a], [w/f(x, y)][x/a]) && \text{por 4} \\
 = & \text{unif}((h(f(a, y)) = y), [w/f(a, y), x/a]) \\
 = & \text{“No unificable”} && \text{por 5}
 \end{aligned}$$

Resolución: Separación de variables

- Separación de variables

- Def.: La sustitución $[x_1/t_1, \dots, x_n/t_n]$ es un renombramiento si todos los t_i son variables.
- Prop.: Si θ es un renombramiento, entonces $C \equiv C\theta$.
- Def.: Las cláusulas C_1 y C_2 están separadas si no tienen ninguna variable común.
- Def.: Una separación de las variables de C_1 y C_2 es un par de renombramientos θ_1, θ_2 tales que $C_1\theta_1$ y $C_2\theta_2$ están separadas.
- Ejemplo: Una separación de variables de $C_1 = \{P(x), Q(x, y)\}$ y $C_2 = \{R(f(x, y))\}$ es $(\theta_1 = [x/x_1, y/y_1], \theta_2 = [x/x_2, y/y_2])$.

Resolución: Resolvente binaria

- Resolución binaria:

- Def.: La cláusula C es una resolvente binaria de las cláusulas C_1 y C_2 si existen una separación de variables (θ_1, θ_2) de C_1 y C_2 , un literal $L_1 \in C_1$, un literal $L_2 \in C_2$ y un UMG σ de $L_1\theta_1$ y $L_2\theta_2$ tales que

$$C = (C_1\theta_1\sigma \setminus \{L_1\theta_1\sigma\}) \cup (C_2\theta_2\sigma \setminus \{L_2\theta_2\sigma\}).$$

- Ejemplo: Sean $C_1 = \{P(a, y), R(y)\}$,
 $C_2 = \{\neg P(x, f(x)), Q(g(x))\}$,
 $\theta_1 = \epsilon$,
 $\theta_2 = \epsilon$,
 $L_1 = P(a, y)$,
 $L_2 = \neg P(x, f(x))$,
 $\sigma = [x/a, y/f(a)]$

Entonces, $C = \{R(f(a)), Q(g(a))\}$ es una resolvente binaria de C_1 y C_2 .

Resolución: Resolvente binaria

- Ejemplo (Necesidad de separar variables):

$$\begin{aligned} \text{Sean } C_1 &= \{P(x), Q(x)\}, \\ C_2 &= \{\neg P(f(x)), R(x)\}, \\ \theta_1 &= [x/x_1], \\ \theta_2 &= [x/x_2], \\ L_1 &= P(x), \\ L_2 &= P(f(x)), \\ \sigma &= [x_1/f(x_2)] \end{aligned}$$

Entonces, $C = \{Q(f(x_2)), R(x_2)\}$ es una resolvente binaria de C_1 y C_2 .

Notas sobre la necesidad de separar variables:

- $P(x)$ y $P(f(x))$ no son unificables.
- $P(x_1)$ y $P(f(x_2))$ son unificables siendo $\sigma = [x_1/f(x_2)]$ un UMG.

Resolución: Factorización

- Factorización:

- Def.: La cláusula C es un factor de la cláusula D si existen dos literales L_1 y L_2 en D que son unificables y $C = D\sigma \setminus \{L_2\sigma\}$ donde σ es un UMG de L_1 y L_2 .

- Ejemplo: Sean $D = \{P(x, y), P(y, x), Q(a)\}$

$$L_1 = P(x, y)$$

$$L_2 = P(y, x)$$

$$\sigma = [y/x]$$

Entonces, $C = \{P(x, x), Q(a)\}$ es un factor de D .

Resolución

- Ejemplos de refutación por resolución:

- Refutación de $S = \{\{\neg P(x, f(x, y))\}, \{P(a, z), \neg Q(z, v)\}, \{Q(u, a)\}\}$
 - 1 $\{\neg P(x, f(x, y))\}$ Hipótesis
 - 2 $\{P(a, z), \neg Q(z, v)\}$ Hipótesis
 - 3 $\{Q(u, a)\}$ Hipótesis
 - 4 $\{\neg Q(f(a, y), v)\}$ Resolvente de 1 y 2 con $\sigma = [x/a, z/f(a, y)]$
 - 5 \square Resolvente de 3 y 4 con $\sigma = [u/f(a, y), v/a]$
- Refutación de $S = \{\{P(x)\}, \{\neg P(f(x))\}\}$
 - 1 $\{P(x)\}$ Hipótesis
 - 2 $\{\neg P(f(x))\}$ Hipótesis
 - 3 \square Resolvente de 1 y 2 con $\theta_1 = \epsilon, \theta_2 = [x/x'], \sigma = [x/f(x')]$
- Refutación de $S = \{\{P(x, y), P(y, x)\}, \{\neg P(u, v), \neg P(v, u)\}\}$
 - 1 $\{P(x, y), P(y, x)\}$ Hipótesis
 - 2 $\{\neg P(u, v), \neg P(v, u)\}$ Hipótesis
 - 3 $\{P(x, x)\}$ Factor de 1 con $[y/x]$
 - 4 $\{\neg P(u, u)\}$ Factor de 2 con $[v/u]$
 - 5 \square Resolvente de 3 y 4 con $[x/u]$

Resolución

- Definiciones

- Sea S un conjunto de cláusulas.
- La sucesión (C_1, \dots, C_n) es una demostración por resolución de la cláusula C a partir de S si $C = C_n$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
 - $C_i \in S$;
 - existen $j, k < i$ tales que C_i es una resolvente de C_j y C_k
 - existe $j < i$ tal que C_i es un factor de C_j
- La cláusula C es demostrable por resolución a partir de S si existe una demostración por resolución de C a partir de S .
- Una refutación por resolución de S es una demostración por resolución de la cláusula vacía a partir de S .
- Se dice que S es refutable por resolución si existe una refutación por resolución a partir de S .

Resolución

• Demostraciones por resolución

- Def.: Sean S_1, \dots, S_n formas clausales de las fórmulas F_1, \dots, F_n y S una forma clausal de $\neg F$. Una demostración por resolución de F a partir de $\{F_1, \dots, F_n\}$ es una refutación por resolución de $S_1 \cup \dots \cup S_n \cup S$.
- Def.: La fórmula F es demostrable por resolución no restringida a partir de $\{F_1, \dots, F_n\}$ si existe una demostración por resolución de F a partir de $\{F_1, \dots, F_n\}$. Se representa por $\{F_1, \dots, F_n\} \vdash_{Res} F$.
- Ejemplo: (tema 8 p. 21) $S = \{(\forall x)[P(x) \rightarrow Q(x)], (\exists x)P(x)\} \vdash_{Res} (\exists x)Q(x)$
 - 1 $\{\neg P(x), Q(x)\}$ Hipótesis
 - 2 $\{P(a)\}$ Hipótesis
 - 3 $\{\neg Q(z)\}$ Hipótesis
 - 4 $\{Q(a)\}$ Resolvente de 1 y 2 con $[x/a]$
 - 5 \square Resolvente de 3 y 4 con $[z/a]$

Resolución

- Ejemplo: (tema 8 p. 21)

$$S = \{(\forall x)[P(x) \rightarrow Q(x)], (\forall x)[Q(x) \rightarrow R(x)] \vdash_{Res} (\forall x)[P(x) \rightarrow R(x)]\}$$
 - 1 $\{\neg P(x), Q(x)\}$ Hipótesis
 - 2 $\{\neg Q(y), R(y)\}$ Hipótesis
 - 3 $\{P(a)\}$ Hipótesis
 - 4 $\{\neg R(a)\}$ Hipótesis
 - 5 $\{Q(a)\}$ Resolvente de 1 y 2 con $[x/a]$
 - 6 $\{R(a)\}$ Resolvente de 2 y 5 con $[y/a]$
 - 5 \square Resolvente de 6 y 4 con
- Ejemplo: (tema 6 p. 55) $\vdash_{Res} (\exists x)[P(x) \rightarrow (\forall y)P(y)]$
 - 1 $\{P(x)\}$ Hipótesis
 - 2 $\{\neg P(f(x))\}$ Hipótesis
 - 3 \square Resolvente de 1 y 2 con $\theta_2 = [x/x'], \sigma = [x/f(x')]$

Resolución

- Ejemplo: $\vdash_{Res} (\forall x)(\exists y)\neg(P(y, x) \leftrightarrow \neg P(y, y))$

– Forma clausal:

$$\begin{aligned}
 & \neg(\forall x)(\exists y)\neg(P(y, x) \leftrightarrow \neg P(y, y)) \\
 \equiv & \neg(\forall x)(\exists y)\neg((P(y, x) \rightarrow \neg P(y, y)) \wedge (\neg P(y, y) \rightarrow P(y, x))) \\
 \equiv & \neg(\forall x)(\exists y)\neg((\neg P(y, x) \vee \neg P(y, y)) \wedge (\neg\neg P(y, y) \vee P(y, x))) \\
 \equiv & \neg(\forall x)(\exists y)\neg((\neg P(y, x) \vee \neg P(y, y)) \wedge (P(y, y) \vee P(y, x))) \\
 \equiv & (\exists x)(\forall y)\neg((\neg P(y, x) \vee \neg P(y, y)) \wedge (P(y, y) \vee P(y, x))) \\
 \equiv & (\exists x)(\forall y)((\neg P(y, x) \vee \neg P(y, y)) \wedge (P(y, y) \vee P(y, x))) \\
 \equiv_{sat} & (\forall y)((\neg P(y, a) \vee \neg P(y, y)) \wedge (P(y, y) \vee P(y, a))) \\
 \equiv & \{\{\neg P(y, a), \neg P(y, y)\}, \{P(y, y), P(y, a)\}\}
 \end{aligned}$$

– Refutación:

- 1 $\{\neg P(y, a), \neg P(y, y)\}$ Hipótesis
- 2 $\{P(y, y), P(y, a)\}$ Hipótesis
- 3 $\{\neg P(a, a)\}$ Factor de 1 con $[y/a]$
- 4 $\{\neg P(a, a)\}$ Factor de 2 con $[y/a]$
- 5 \square Resolvente de 3 y 4

Resolución

- Ejemplo (Paradoja del barbero de Russell): En una isla pequeña hay sólo un barbero. El gobernador de la isla ha publicado la siguiente norma: “El barbero afeita a todas las personas que no se afeitan a sí misma y sólo a dichas personas”. Demostrar que la norma es inconsistente.

– Representación:

$$(\forall x)[afeita(b, x) \leftrightarrow \neg afeita(x, x)]$$

– Forma clausal:

$$\begin{aligned}
 & (\forall x)[afeita(b, x) \leftrightarrow \neg afeita(x, x)] \\
 \equiv & (\forall x)[(afeita(b, x) \rightarrow \neg afeita(x, x)) \wedge (\neg afeita(x, x) \rightarrow afeita(b, x))] \\
 \equiv & (\forall x)[(\neg afeita(b, x) \vee \neg afeita(x, x)) \wedge (\neg\neg afeita(x, x) \vee afeita(b, x))] \\
 \equiv & (\forall x)[(\neg afeita(b, x) \vee \neg afeita(x, x)) \wedge (afeita(x, x) \vee afeita(b, x))] \\
 \equiv & \{\{\neg afeita(b, x), \neg afeita(x, x)\}, \{afeita(x, x), afeita(b, x)\}\}
 \end{aligned}$$

– Refutación:

- 1 $\{\neg afeita(b, x), \neg afeita(x, x)\}$ Hipótesis
- 2 $\{afeita(x, x), afeita(b, x)\}$ Hipótesis
- 3 $\{\neg afeita(b, b)\}$ Factor de 1 con $[x/b]$
- 4 $\{\neg afeita(b, b)\}$ Factor de 2 con $[x/b]$
- 5 \square Resolvente de 3 y 4

Adecuación y completitud de la resolución

- **Propiedades:**

- Si C es una resolvente de C_1 y C_2 , entonces $\{C_1, C_2\} \models C$.
- Si C es un factor de C entonces $C \models D$.
- Si $\square \in S$, entonces S es inconsistente.
- Si el conjunto de cláusulas S es refutable por resolución, entonces S es inconsistente.
- Teor.: El cálculo de resolución es adecuado y completo; es decir,

$$\text{Adecuado: } S \vdash_{Res} F \quad \Longrightarrow \quad S \models F$$

$$\text{Completo: } S \models F \quad \Longrightarrow \quad S \vdash_{Res} F$$

Bibliografía

- M.L. Bonet *Apuntes de LPO*. (Univ. Politécnica de Cataluña, 2003) pp. 34–40.
- C.L. Chang y R.C.T. Lee *Symbolic logic and mechanical theorem proving* (Academic Press, 1973) pp. 70–99.
- J.I. García, P.A. García y J.M. Urbano *Fundamentos lógicos de la programación*. (Universidad de Granada, 2002) pp. 45–66
- M. Genesereth *Computational Logic (Chapter 9: Relational Resolution)* (Stanford University, 2003)
- S. Hölldobler *Computational logic*. (U. de Dresden, 2004) pp. 71–74.
- M. Ojeda e I. Pérez *Lógica para la computación (Vol. 2: Lógica de Primer Orden)* (Ágora, 1997) pp. 138–164.
- L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 50–61.
- U. Schöning *Logic for computer scientists* (Birkäuser, 1989) pp. 79–96.