

# Lógica clausal

José A. Alonso

Dpto. de Ciencias de la Computación e Inteligencia Artificial

UNIVERSIDAD DE SEVILLA

# Sistemas lógicos

- Elementos de un sistema lógico:
  - Sintaxis: alfabeto, expresiones
  - Semántica:  $S \models F$
  - Cálculo:  $S \vdash F$
- Propiedades de los sistemas lógicos:
  - Adecuación:  $S \vdash F \implies S \models F$
  - Completitud:  $S \models F \implies S \vdash F$
  - Decidibilidad

# Lógica clausal proposicional: Sintaxis

- Cláusulas proposicionales:

- Definición:  $A_1; \dots; A_n :- B_1, \dots, B_m$

- Cabeza y cuerpo de una cláusula

- Ejemplo: `par; impar :- numero, natural`

- Fórmulas equivalentes:

$$B_1 \wedge \dots \wedge B_m \rightarrow A_1 \vee \dots \vee A_n$$

$$\neg B_1 \vee \dots \vee \neg B_m \vee A_1 \vee \dots \vee A_n$$

$$L_1 \vee \dots \vee L_p \text{ con } L_i \text{ literales}$$

- Gramática de la lógica clausal proposicional:

`cláusula ::= cabeza[: - cuerpo]`

`cabeza ::= [átomo[;átomo]*]`

`cuerpo ::= [átomo[,átomo]*]`

`átomo ::= <palabra comenzando con minúscula>`

- Programa clausal proposicional:

`hombre; mujer :- persona.`

`persona :- hombre.`

`persona :- mujer.`

- Fórmula equivalente a un programa:

`(persona -> hombre | mujer)`

`& (hombre -> persona)`

`& (mujer -> persona)`

- Hechos:  $A_1 :-$  es equivalente a  $A_1$

# Lógica clausal proposicional: Semántica

- Base de Herbrand de un programa  $B(P)$

```
hombre; mujer :- persona.  
persona :- hombre.  
persona :- mujer.
```

$B(P) = \{\text{persona, hombre, mujer}\}$

- Interpretación de Herbrand:  $I \subseteq B(P)$

```
I1 = {persona, mujer}  
I2 = {mujer}
```

- Modelo de una cláusula:

- Def.:  $I \models C$  si la cabeza de  $C$  es válida en  $I$  o el cuerpo de  $C$  no es válido en  $I$

- Ejemplos:

```
I1  $\models$  persona :- mujer  
I2  $\not\models$  persona :- mujer
```

- Modelo de un programa:

- $I \models P$  si es modelo de todas sus cláusulas

- Ejemplo:

```
I1  $\models$  P  
I2  $\not\models$  P
```

# Lógica clausal proposicional: Semántica

- Modelos de un programa:

P1:

```
hombre; mujer :- persona.  
persona :- hombre.  
persona :- mujer.
```

Modelos(P1) =  $\{\{\},$   
                  {mujer, persona},  
                  {hombre, persona},  
                  {hombre, mujer, persona}\}

P2:

```
hombre; mujer :- persona.  
persona :- hombre.  
persona :- mujer.  
mujer.
```

Modelos(P2) =  $\{\{mujer, persona\},$   
                  {hombre, mujer, persona}\}

P3:

```
hombre; mujer :- persona.  
persona.
```

Modelos(P3) =  $\{\{mujer, persona\},$   
                  {hombre, persona},  
                  {hombre, mujer, persona}\}

# Lógica clausal proposicional: Semántica

- Modelos minimales de un programa:

P1:

hombre; mujer :- persona.

persona :- hombre.

persona :- mujer.

Modelos\_minimales(P1) = {{}}

P2:

hombre; mujer :- persona.

persona :- hombre.

persona :- mujer.

mujer.

Modelos\_minimales(P2) = {{mujer, persona}}

P3:

hombre; mujer :- persona.

persona.

Modelos\_minimales(P3) = {{mujer, persona},  
{hombre, persona}}

- Programas definidos:

- Cláusulas definidas:  $A1 \text{ :- } B1, \dots, Bm$

- Programa definido

- Los programas definidos tienen un menor modelo

- Consecuencia lógica de un programa:

- Def.:  $P \models C$  si todo modelo de  $P$  es modelo de  $C$

- Ejemplo:  $P2 \models \text{persona}$



# Lógica clausal proposicional: Cálculo

- **Deducibilidad:**
  - Def.:  $C$  es deducible a partir de  $P$  si existe una deducción de  $C$  a partir de  $P$
  - Notación:  $P \vdash C$
- **Adecuación de resolución:**  $P \vdash C \implies P \models C$
- **Incompletitud de resolución:**  
 $P \models C \not\implies P \vdash C$ 
  - Ejemplo:  $P = \{ q. \}$  y  $C = q;r :-$ .
- **Cláusula vacía:**
  - Definición:  $:-$
  - Notación:  $\square$
  - $\square$  es falsa en cualquier interpretación
- **Completitud refutacional de la resolución:**  
 $P \models C \implies P \cup \{\neg C\} \vdash \square$

# Lógica clausal relacional: Sintaxis

- Necesidad de ampliación:

- Javier defiende a todos sus jugadores.
- Finidi es un jugador de Javier.
- Por tanto, Javier defiende a Finidi.

- Gramática de la lógica clausal relacional:

```
cláusula ::= cabeza[:- cuerpo]
cabeza   ::= [átomo[;átomo]*]
cuerpo   ::= [átomo[,átomo]*]
átomo    ::= predicado[(término[,término]*)]
predicado ::= <palabra comenzando con minúscula>
término  ::= constante | variable
constante ::= <palabra comenzando con minúscula>
variable ::= <palabra comenzando con mayúscula>
expresión ::= término | átomo
```

- Ejemplo de cláusula relacional:

```
defiende(javier,J) :- jugador_de(J,javier)
```

- Aridad de un predicado: defiende/2, entre/3

- Literales y programas relacionales.

- Ejemplo de programa relacional:

```
defiende(javier,J) :- jugador_de(J,javier)
jugador_de(finidi,javier).
```

- Expresiones básicas: sin variables

# Lógica clausal relacional: Semántica

- **Universo de Herbrand:**

- $UH(P)$  = conjunto de términos básicos de  $P$

- Ejemplo:

$UH(P) = \{\text{javier}, \text{finidi}\}$

- **Base de Herbrand:**

- $BH(P)$  = conjunto de átomos contruidos con predicados de  $P$  y términos básicos de  $P$

- Ejemplo: La BH del programa anterior es:

$\{\text{defiende}(\text{javier}, \text{javier}), \text{defiende}(\text{javier}, \text{finidi}),$   
 $\text{defiende}(\text{finidi}, \text{javier}), \text{defiende}(\text{finidi}, \text{finidi}),$   
 $\text{jugador\_de}(\text{javier}, \text{javier}), \text{jugador\_de}(\text{javier}, \text{finidi}),$   
 $\text{jugador\_de}(\text{finidi}, \text{javier}), \text{jugador\_de}(\text{finidi}, \text{finidi})\}$

- **Interpretación de Herbrand:**

- $I$  interpretación de Herbrand de  $P$  si  $I \subseteq BH(P)$

- Ejemplo:

$\{\text{defiende}(\text{javier}, \text{finidi}), \text{jugador\_de}(\text{finidi}, \text{javier})\}$

- **Sustitución:**

- Def.: Aplicación de las variables en los términos

- Ej.:  $s = \{J/\text{finidi}\}$

# Lógica clausal relacional: Semántica

- Aplicación de una sustitución a una cláusula:
  - $Cs$ : sustituir cada variable  $X$  de  $C$  por  $s(X)$
  - Ejemplo:  
 $C = \text{defiende}(\text{javier}, J1) :- \text{jugador\_de}(J, \text{javier})$   
 $s = \{J/\text{finidi}\}$   
 $Cs = \text{defiende}(\text{javier}, J1) :- \text{jugador\_de}(\text{finidi}, \text{javier})$
- Instancias:
  - $C$  es instancia de  $D$  si existe  $s$  t.q.  $C = Ds$
  - Instancia básicas
  - Ejemplo: Las instancias básicas de  $P$  son:  
 $\text{defiende}(\text{javier}, \text{javier}) :- \text{jugador\_de}(\text{javier}, \text{javier}).$   
 $\text{defiende}(\text{javier}, \text{finidi}) :- \text{jugador\_de}(\text{finidi}, \text{javier})$   
 $\text{jugador\_de}(\text{finidi}, \text{javier}).$
- Cláusulas relacionales básicas y cláusulas proposicionales
- Modelos:
  - $I \models C$  si lo es de todas sus instancias básicas
  - $I \models P$  si lo es de todas sus cláusulas
  - Ejemplo de modelo de  $P$ :  
 $\{\text{defiende}(\text{javier}, \text{finidi}), \text{jugador\_de}(\text{finidi}, \text{javier})\}$

# Lógica clausal relacional: Cálculo

- Aproximación ingenua: Aplicar resolución proposicional a las instancias básicas
- Ejemplo de derivación con resolución y unificación:

P :

```
defiende(javier,J) :- jugador_de(J,javier).
jugador_de(X,Y) :- juega_en(X,E), entrena(Y,E).
juega_en(finidi,betis).
entrena(javier,betis).
```

C = defiende(javier,finidi)

Derivación:

```
:- defiende(javier,finidi)
| defiende(javier,J1) :- jugador_de(J1,javier)
| {J1/finidi}
:- jugador_de(finidi,javier)
| jugador_de(X2,Y2) :- juega_en(X2,E2), entrena(Y2,E2)
| {X2/finidi, Y2/javier}
:- juega_en(finidi,E2), entrena(javier,E2)
| juega_en(finidi,betis)
| {E2/betis}
:- entrena(javier,betis)
| entrena(javier,betis)
| {}
:-
```

# Lógica clausal relacional: Cálculo

- Ejemplo de cálculo de respuesta:

P :

```
defiende(javier,J) :- jugador_de(J,javier).
jugador_de(X,Y) :- juega_en(X,E), entrena(Y,E).
juega_en(finidi,betis).
entrena(javier,betis).
```

C = defiende(javier,J).

Derivación:

```
:- defiende(javier,J0)
| defiende(javier,J1) :- jugador_de(J1,javier)
| {J1/J0}
:- jugador_de(J0,javier)
| jugador_de(X2,Y2) :- juega_en(X2,E2), entrena(Y2,E2)
| {X2/J0, Y2/javier}
:- juega_en(J0,E2), entrena(javier,E2)
| juega_en(finidi,betis)
| {J=0/finidi, E2/betis}
:- entrena(javier,betis)
| entrena(javier,betis)
| {}
:-
```

Respuesta = {J/finidi}

- Propiedades del cálculo clausal relacional:

- Adecuación:  $P \vdash C \implies P \models C$
- Completitud refutacional:  $P \models C \implies P \cup \{\neg C\} \vdash \square$
- Decidibilidad

# Lógica clausal completa: Sintaxis

- Necesidad de ampliación:

Representación de “Cada persona tiene un padre”:

- En lógica clausal relacional:

```
es_padre_de(adan,cain).  
es_padre_de(adan,abel).  
...
```

- En lógica clausal completa:

```
es_padre_de(el_padre_de(X),X).
```

- Gramática de la lógica clausal completa:

```
cláusula ::= cabeza[:- cuerpo]  
cabeza ::= [átomo[;átomo]*]  
cuerpo ::= [átomo[,átomo]*]  
átomo ::= predicado[(término[,término]*)]  
predicado ::= <palabra comenzando con minúscula>  
término ::= variable | functor[(término[,término]*)]  
functor ::= <palabra comenzando con minúscula>  
variable ::= <palabra comenzando con mayúscula>  
expresión ::= término | átomo
```

- Ejemplo de programa:

```
suma(0,X,X).  
suma(s(X),Y,s(Z)) :- suma(X,Y,Z).
```

# Lógica clausal completa: Semántica

- **Universo de Herbrand:**

- $UH(P)$  = conjunto de términos básicos de  $P$

- Ejemplo:

$$UH(P) = \{0, s(0), s(s(0)), \dots\}$$

- **Base de Herbrand:**

- $BH(P)$  = conjunto de átomos contruidos con predicados de  $P$  y términos básicos de  $P$

- Ejemplo: La BH del programa anterior es:

$$\{\text{suma}(0,0,0), \text{suma}(0,0,s(0)), \text{suma}(0,0,s(s(0))), \dots, \\ \text{suma}(0,s(0),0), \text{suma}(0,s(0),s(0)), \dots\}$$

- $I$  interpretación de Herbrand de  $P$  si  $I \subseteq BH(P)$

- **Modelos:**

- $I \models C$  si lo es de todas sus instancias básicas

- $I \models P$  si lo es de todas sus cláusulas

- Ejemplo de modelo de  $P$ :

$$\{\text{suma}(0,0,0), \text{suma}(s(0),0,s(0)), \\ \text{suma}(s(s(0)),0,s(s(0))), \dots, \\ \text{suma}(0,s(0),s(0)), \text{suma}(s(0),s(0),s(s(0))), \dots\}$$

# Lógica clausal completa: Cálculo

- Ejemplo de derivación:

P:  $\text{suma}(0, X, X).$   
 $\text{suma}(s(X), Y, s(Z)) \text{ :- suma}(X, Y, Z).$

C:  $\text{suma}(s(0), s(0), X)$

Derivación:

```
:- suma(s(0), s(0), X0)
|   suma(s(X1), Y1, s(Z1)) :- suma(X1, Y1, Z1)
|   {X1/0, Y1/s(0), X0/s(Z1)}
:- suma(0, s(0), Z1)
|   suma(0, X2, X2)
|   {X2/s(0), Z1/s(0)}
:-
```

Respuesta =  $\{X/s(s(0))\}$

- Propiedades del cálculo clausal completo:

- Adecuación:  $P \vdash C \implies P \models C$
- Completitud refutacional:  $P \models C \implies P \cup \{\neg C\} \vdash \square$
- Semi-Decidibilidad

# Lógica clausal definida

- Cláusulas definidas:  $A1 :- B1, \dots, Bm$
- Ventajas de la restricción:
  - Eficiencia del cálculo
  - Existencia de menor modelo
- Inconvenientes de la restricción:
  - Menor potencia expresiva
  - Tratamiento de la negación
- Ejemplo de cláusulas indefinidas a pseudo-definidas

P1:

```
casado(X); soltero(X) :- hombre(X), mayor_de_30(X).  
hombre(juan).  
mayor_de_30(juan).
```

Modelos\_minimales(P1) =

```
{{hombre(juan), mayor_de_30(juan), casado(juan)},  
{hombre(juan), mayor_de_30(juan), soltero(juan)}}
```

# Lógica clausal definida

P2:

```
casado(X) :-  
    hombre(X),  
    mayor_de_30(X),  
    not(soltero(X)).  
hombre(juan).  
mayor_de_30(juan).
```

Menor\_modelo(P2) =  
{hombre(juan), mayor\_de\_30(juan), casado(juan)}

P3:

```
soltero(X) :-  
    hombre(X),  
    mayor_de_30(X),  
    not(casado(X)).  
hombre(juan).  
mayor_de_30(juan).
```

Menor\_modelo(P3) =  
{hombre(juan), mayor\_de\_30(juan), soltero(juan)}

## Bibliografía

- Flach, P. *Simply Logical (Intelligent Reasoning by Example)* (John Wiley, 1994)
  - Cap. 2: “Clausal logic and resolution: theoretical backgrounds”.