

# Negación en ASP

Jorge Baier

Departamento de Ciencia de la Computación  
Pontificia Universidad Católica de Chile

Santiago, Chile



- Comprender el uso de negaciones en reglas de Clingo



## Definición

Una *regla* en programación en lógica es un objeto de la forma:

$$Head \leftarrow Pos, not(Neg),$$

donde *Head*, *Pos* y *Neg* son conjuntos de átomos.

Ejemplo:  $\{p, q\} \leftarrow \{r, s\}, not(\{t\}), \{t\} \leftarrow \{\}, \{\} \leftarrow not(\{r, s\})$ .

En el sistema *clingo* estas reglas, respectivamente, se anotan así:

```
p;q :- r,s,not t.  
t.  
:- not r, not s.
```

Para declarar que un átomo no está en el modelo, se usa la regla:

```
:- p.
```



# Ejemplo

```
planta(tomate; durazno; uva; oregano).  
regador(r1; r2; r3).  
riega(r1, durazno).  
riega(r1, uva).  
riega(r2, tomate).  
riega(r3, oregano).  
riega(r2, oregano).  
regada(P) :- planta(P), riega(R, P), encendido(R).  
seca(P) :- planta(P), not regada(P).  
1 {encendido(r1); encendido(r2); encendido(r3)} 1.  
:- seca(oregano).
```



# Ejemplo

Consideremos ahora este programa

$p :- \text{not } q.$

$q :- \text{not } p.$

Aunque puede parecer contradictorio, este programa tiene dos modelos  $\{p\}$  y  $\{q\}$ . Intuitivamente esto ocurre porque, si  $q$  no está en el modelo, entonces  $\text{not } q$  es verdadero, lo que, por la primera regla, implica que  $p$  está en el modelo. Luego, si  $p$  está en el modelo,  $\text{not } p$  es falso, por lo que, según la segunda regla,  $q$  no está en el modelo. De manera simétrica, si  $q$  está en el modelo,  $p$  no está.



# Ejemplo

Por otro lado, el programa

$$p \text{ :- not } p.$$

Es efectivamente una contradicción, por lo que es insatisfacible, es decir, no tiene modelo.



## Definición (Reducción)

La *reducción* un programa  $\Pi$  relativa a un conjunto  $X$ , denotada por  $\Pi^X$  es la que resulta de hacer:

- 1  $\Pi^X := \Pi$
- 2 **Borrar** toda regla  $Head \leftarrow Pos \cup not(Neg)$  de  $\Pi^X$  cuando  $Neg \cap X \neq \emptyset$ .
- 3 **Reemplazar** cada regla  $Head \leftarrow Pos \cup not(Neg)$  en  $\Pi^X$  por  $Head \leftarrow Pos$  cuando  $Neg \cap X = \emptyset$ .



## Definición (Reducción)

La *reducción* un programa  $\Pi$  relativa a un conjunto  $X$ , denotada por  $\Pi^X$  es la que resulta de hacer:

- 1  $\Pi^X := \Pi$
- 2 **Borrar** toda regla  $Head \leftarrow Pos \cup not(Neg)$  de  $\Pi^X$  cuando  $Neg \cap X \neq \emptyset$ .
- 3 **Reemplazar** cada regla  $Head \leftarrow Pos \cup not(Neg)$  en  $\Pi^X$  por  $Head \leftarrow Pos$  cuando  $Neg \cap X = \emptyset$ .

## Definición (Modelo de un programa con negación)

$X$  es un modelo de un programa con negación  $\Pi$  ssi  $X$  es un modelo para  $\Pi^X$ .





# Ejemplo

Las definiciones anteriores se entienden mejor aplicadas a un ejemplo. Asumimos que tenemos el siguiente programa  $\Pi$ :

```
p.  
q.  
t :- not r.  
s :- not q.
```

Y el siguiente conjunto  $X = \{p, q, t\}$



# Ejemplo

La reducción  $\Pi^X$  de  $\Pi$  relativa a  $X$  se calcula como:

- 1 Inicialmente,  
 $\Pi^X := \Pi := \{p\}, \{q\}, \{t\} \leftarrow \text{not}(\{r\}), \{s\} \leftarrow \text{not}(\{q\})$
- 2 Borramos la regla  $\{s\} \leftarrow \text{not}(\{q\})$  de  $\Pi^X$  ya que  $\{q\} \cap X \neq \emptyset$ .
- 3 Reemplazamos la regla  $\{t\} \leftarrow \text{not}(\{r\})$  en  $\Pi^X$  por  $\{t\} \leftarrow \{\}$  porque  $\{r\} \cap X = \emptyset$ .

De esta manera, la reducción  $\Pi^X$  corresponde a:

p.  
q.  
t.

Y como  $X = \{p, q, t\}$  es un modelo de  $\Pi^X$ , entonces también es un modelo de  $\Pi$



- Comprender el uso de negaciones en reglas de Clingo

