Lógicas Modales

Lógicas modales vistas como fragmentos

Carlos Areces & Raul Fervari

1er Cuatrimestre 2017,
Córdoba, Argentina

Temario de hoy

- ¿Modelos de Kripke vs. modelos de primer orden?
- Traducciones a primer orden
- ¡*Hacer transferencia* es bueno!
- Traducciones más *refinadas*
- ¿Primer orden es el límite?

Bibliografía Relevante

- Capítulo 2 del "Modal Logic," Blackburn, de Rijke & Venema. Buscar la parte sobre la 'Standard Translation' (Seccion 2.4).
- "Tree-Based Heuristics in Modal Theorem Proving," Areces, Gennari, Heguiabehere and de Rijke.
- "Unsorted Functional Translations," Areces and Gorín.

Modelos de Kripke vs. Modelos de primer orden

Modelos de Kripke

- Un dominio no vacío W
- Una o más $R_i \subseteq W \times W$
- Una función de valuación $V : \mathsf{PROP} \to 2^W$

Modelos de primer orden

- Un dominio no vacío $\mathcal{D}$
- Por cada símbolo de predicado P de aridad n, un

$$P^{\mathcal{I}} \subseteq \underbrace{\mathcal{D} \times \ldots \times \mathcal{D}}_{n}$$

- Pero para cada $p \in \mathsf{PROP}$, $V(p) \subseteq W$
(es decir, V codifica una relación unaria por cada proposición p)
- Entonces podemos pensar a un modelo de Kripke como un modelo de primer orden

Correspondencia de modelos

- Formalmente, un modelo de Kripke

$$\mathcal{M} = \langle W, \{R_i\}_{i \in \mathsf{REL}}, V\rangle$$

definido sobre una signatura $\mathcal{S} = \langle \mathsf{PROP}, \mathsf{REL}\rangle$ *se corresponde* con un modelo de primer orden

$$\mathcal{I}^{\mathcal{M}} = \langle W, \cdot^{\mathcal{I}^{\mathcal{M}}}\rangle$$

cuyo vocabulario es $P^{\mathcal{S}} = \{P_i \mid i \in \mathsf{PROP} \cup \mathsf{REL}\}$ (P_i es unario si $i \in \mathsf{PROP}$; binario si $i \in \mathsf{REL}$) donde

$$P_i^{\mathcal{I}^{\mathcal{M}}} = \begin{cases} V(i) & \text{si } i \in \mathsf{PROP} \\ R_i & \text{si } i \in \mathsf{REL} \end{cases}$$

- A $P^{\mathcal{S}}$ se lo llama *lenguaje de correspondencia de primer orden*

¿Correspondencia entre fórmulas?

- *Correspondencia entre modelos:* la lógica modal básica y la de primer orden (en el lenguaje de correspondencia) operan sobre los mismos objetos semánticos
- Pero, ¿se puede comparar la forma en que operan ambas lógicas?
- *Correspondencia entre fórmulas:*
  - pensemos en una correspondencia como una traducción de fórmulas, de una lógica a otra
  - una fórmula y su traducción tienen que ser *equivalentes*
  - si existe, nos indica que las operaciones de la lógica origen son reproducibles en la lógica a la que se traduce

¿Correspondencia entre fórmulas? De modal a primer orden

- Queremos ver si podemos reproducir las operaciones de la lógica modal básica en lógica de primer orden
- Con lo cual, lo más razonable es...
  - Mirar las condiciones semánticas de la lógica modal básica
  - ¡Y tratar de reproducirlas en primer orden!

¿Correspondencia entre fórmulas? De modal a primer orden

$$\begin{array}{lll}
\mathcal{M}, w \models p & \text{sii} & w \in V(p) \\
\mathcal{M}, w \models \neg\varphi & \text{sii} & \mathcal{M}, w \not\models \varphi \\
\mathcal{M}, w \models \varphi \land \psi & \text{sii} & \mathcal{M}, w \models \varphi \text{ y } \mathcal{M}, w \models \psi \\
\mathcal{M}, w \models \langle R\rangle\varphi & \text{sii} & \exists w' \in W \text{ tq } R(w, w') \text{ y } \mathcal{M}, w' \models \varphi
\end{array}$$

$$\begin{array}{lll}
\mathsf{ST}_x(p) & \equiv & P_p(x) \\
\mathsf{ST}_x(\neg\varphi) & \equiv & \neg\mathsf{ST}_x(\varphi) \\
\mathsf{ST}_x(\varphi \land \psi) & \equiv & \mathsf{ST}_x(\varphi) \land \mathsf{ST}_x(\psi) \\
\mathsf{ST}_x(\langle R\rangle\varphi) & \equiv & \exists y.(R(x, y) \land \mathsf{ST}_y(\varphi))
\end{array}$$

ST, la traducción estándar a primer orden

- $\mathsf{ST}_x(\varphi)$ le hace corresponder, a cada fórmula φ, una fórmula de primer orden con exactamente una variable libre x
- Esta variable libre da cuenta del punto de evaluación en la semántica modal (recordar la "perspectiva interna")

Teorema
Para toda fórmula φ de la lógica modal básica, todo modelo $\mathcal{M}$, todo w en el dominio de $\mathcal{M}$ y toda asignación g,

$$\mathcal{M}, w \models \varphi \text{ sii } \mathcal{M}, g[x \mapsto w] \models \mathsf{ST}_x(\varphi)$$

Demostración.
Fácil, por inducción en φ. Es basicamente lo que ya hicimos... □

¿Correspondencia entre fórmulas? De primer orden a modal

- ST es inyectiva, pero no sobreyectiva
  - Notar que todo cuantificador viene con una *guarda*
- ¿Podemos encontrar una traducción de primer orden a lógica modal básica?
  - Miremos las cláusulas semánticas:

$$\begin{array}{lll}
\mathcal{I}, g \models P(x_1, \ldots, x_n) & \text{sii} & (g(x_1), \ldots, g(x_n)) \in P^{\mathcal{I}} \\
\mathcal{I}, g \models \neg\varphi & \text{sii} & \mathcal{I}, g \not\models \varphi \\
\mathcal{I}, g \models \varphi \wedge \psi & \text{sii} & \mathcal{I}, g \models \varphi \text{ y } \mathcal{I}, g \models \psi \\
\mathcal{I}, g \models \exists x.\varphi & \text{sii} & \text{existe } w \text{ tal que } \mathcal{I}, g[x \mapsto w] \models \varphi
\end{array}$$

  - Que no se nos ocurra un traducción... ¡no significa que no exista!
  - Pero si no existe, ¿cómo lo probamos?
- Veremos varias formas de responder estas preguntas... (luego)

Transfiriendo resultados de primer orden

- La traducción estándar nos permite importar con facilidad muchos resultados conocidos de lógica de primer orden
- ¡Excelente relación costo-beneficio!
- Vamos a ver dos ejemplos:
  1. Compacidad
  2. Löwenheim-Skolem

(¿Qué es compacidad? ← notar que se abre un gran paréntesis

Teorema (Compacidad de la lógica de primer orden)

a) *Si $\Gamma \models \varphi$, entonces para algún $\Gamma_0 \subseteq \Gamma$ finito, $\Gamma_0 \models \varphi$.*
b) *Si todo subconjunto finito Γ_0 de Γ es satisfacible, Γ lo es.*
c) *Si Γ es insatisfacible, algún subconjunto finito de Γ lo es.*

- Está bueno porque:
  - Todo *razonamiento* en una lógica con compacidad involucra finitas premisas
  - Es una herramienta para probar resultados de existencia (no constructiva) de modelos...
  - ... y resultados de no-existencia también

¡Compacidad en acción!

- Consideremos las siguientes fórmulas:

$$\begin{aligned}
\mathsf{AlMenos}_2 &:= \exists x_1, x_2 \,.\, x_1 \neq x_2\\
\mathsf{AlMenos}_3 &:= \exists x_1, x_2, x_3 \,.\, x_1 \neq x_2 \wedge x_1 \neq x_3 \wedge x_2 \neq x_3\\
&\vdots\\
\mathsf{AlMenos}_n &:= \exists x_1, \ldots, x_n \,.\, \bigwedge_{i \neq j} x_i \neq x_j
\end{aligned}$$

- ¿Qué propiedad debe tener un $\mathcal{I}$ para que valga $\mathcal{I} \models \mathsf{AlMenos}_n$?
- ¿Y para que valga $\mathcal{I} \models \mathsf{AlMenos}_n \wedge \neg\mathsf{AlMenos}_{n+1}$?

¡Compacidad en acción! (cont.)

Problema: ¿Existirá φ tal que $\mathcal{I} \models \varphi$ sii $\mathcal{I}$ es un modelo finito?

- Si existe, deberíamos poder construirla...
- Pero si no existe, ¿cómo lo mostramos? ¡Compacidad al rescate!
  1. Supongamos que existe φ como la pedida, y sea
  $$\Gamma := \{\varphi\} \cup \bigcup_{i=2}^{\infty} \{\mathsf{AlMenos}_i\}$$
  2. Todo Γ_0, subconjunto finito de Γ, es satisfacible...
  3. Y por compacidad, si todo Γ_0 es satisfacible, Γ debe serlo también
  4. ¡Pero Γ no lo es!
  5. Llegamos a un absurdo que viene de suponer que existe tal φ

Momento de reflexión) $\leftarrow$ ¡y cerramos el paréntesis!

- Acabamos de mostrar con un ejemplo que hay cosas que en primer orden no se pueden expresar
- Podemos usar lógicas de orden más alto...
- ...al costo de perder buenas propiedades meta-lógicas (e.g., compacidad) con lo cual son difíciles de usar
- *Ufa... ¿Entonces? ¿Estamos fregados?*

- "Ni". No existe LA lógica. Hay compromisos entre *expresividad* y *comportamiento meta-lógico* de acuerdo a cada necesidad

Transferimos compacidad

Teorema (Compacidad de la lógica modal básica)
Si todo subconjunto finito de Γ es satisfacible, Γ lo es.

Demostración.

1. Definamos, para todo conjunto de fórmulas modales Δ,
$$\mathsf{ST}_x(\Delta) := \{\mathsf{ST}_x(\varphi) \mid \varphi \in \Delta\}$$
2. Sea Γ_0 un subconjunto finito de Γ; sabemos que $\mathsf{ST}_x(\Gamma_0)$ va a ser satisfacible sii Γ_0 lo es
3. Entonces, si todo Γ_0 es satisfacible, todo $\mathsf{ST}_x(\Gamma_0)$ lo es; y por compacidad de primer orden, también $\mathsf{ST}_x(\Gamma)$
4. Pero entonces Γ tiene que ser satisfacible también $\square$

(El turno de Löwenheim-Skolem ← otro paréntesis…

Teorema (Löwenheim-Skolem)

Si Γ es un conjunto satisfacible de fórmulas de primer orden, entonces Γ es satisfacible en algún modelo numerable (finito o infinito)

(Cardinales infinitos for dummies)

- Hay *tantos* números naturales como impares
- Hay *tantos* números naturales como racionales
- Pero hay *más* números reales que naturales (diagonal de Cantor)
- O sea que hay "infinitos más grandes que otros" (aunque hay que medirlos con cuidado)

El turno de Löwenheim-Skolem) ← y cerramos el último…

Teorema (Löwenheim-Skolem)

Si Γ es un conjunto satisfacible de fórmulas de primer orden, entonces Γ es satisfacible en algún modelo numerable (finito o infinito)

Corolario directo del Teorema de Löwenheim-Skolem:

No existe ninguna fórmula φ que cumpla $\mathcal{I} \models \varphi$ sii $\mathcal{I}$ es infinito no numerable

Trasferimos Löwenheim-Skolem

Teorema (Löwenheim-Skolem para la lógica modal básica)

Si Γ es un conjunto satisfacible de fórmulas de la lógica modal básica, entonces Γ es satisfacible en algún modelo numerable (finito o infinito)

Demostración.

Análogo al caso de compacidad:

1. Si Γ es satisfacible, $\mathsf{ST}_x(\Gamma)$ lo es también
2. Entonces, por Löwenheim-Skolem para primer orden, existe $\mathcal{I}$ numerable y una valuación g, tal que $\mathcal{I}, g \models \mathsf{ST}_x(\Gamma)$
3. Luego, $\mathcal{I}, g(x) \models \Gamma$

$\square$

Otra aplicación de ST: Arme su demostrador de teoremas

- Un demostrador automático de teoremas es un programa que:
  - Recibe como entrada una fórmula
  - Dice si la fórmula es válida o no al terminar.
- Construir buenos demostradores de teoremas no es fácil
- Por suerte, desde hace muchos años hay gente haciendo sofisticados demostradores de teoremas para lógica de primer orden
- ¡Usando ST, tenemos *gratis* demostradores de teoremas para lógica modal!

$$\varphi \Longrightarrow \forall x.\mathsf{ST}_x(\varphi) \Longrightarrow \boxed{\begin{array}{c}\text{Demostrador de}\\ \text{primer order}\end{array}} \Longrightarrow \text{Respuesta}$$

Una ST más refinada...

- Miremos de nuevo la traducción estándar

$$\begin{array}{lcl}\mathsf{ST}_x(p) & \equiv & P_p(x)\\ \mathsf{ST}_x(\neg\varphi) & \equiv & \neg\mathsf{ST}_x(\varphi)\\ \mathsf{ST}_x(\varphi\wedge\psi) & \equiv & \mathsf{ST}_x(\varphi)\wedge\mathsf{ST}_x(\psi)\\ \mathsf{ST}_x(\langle m\rangle\varphi) & \equiv & \exists y\,.\,P_m(x,y)\wedge\mathsf{ST}_y(\varphi)\end{array}$$

- En ST_x, y representa una variable *nueva*. Con lo cual:

$$\begin{array}{lcl}\mathsf{ST}_y(\langle m\rangle\varphi) & \equiv & \exists z\,.\,P_m(y,z)\wedge\mathsf{ST}_z(\varphi)\\ \mathsf{ST}_z(\langle m\rangle\varphi) & \equiv & \exists w\,.\,P_m(y,w)\wedge\mathsf{ST}_w(\varphi)\\ & \vdots & \end{array}$$

- Pero, en $\mathsf{ST}_y(\varphi)$, x no vuelve a aparecer, ni libre ni ligada
- Con lo cual esto también funciona:

$$\mathsf{ST}_y(\langle m\rangle\varphi) \equiv \exists x\,.\,P_m(y,x)\wedge\mathsf{ST}_x(\varphi)$$

...¿para qué?

Vimos que ST puede escribirse para que use sólo dos variables. ¿Qué conclusiones sacamos?

1. Hay muchas traducciones posibles a primer orden
2. Mucho más importante: vamos a poder transferir un resultado de decidibilidad

Pero antes... ¿*decidibilidad* de qué?

Definición
Se dice que una lógica es *decidible* si el problema de determinar la validez (o satisfacibilidad) de sus fórmulas lo es

Teorema
La lógica de primer orden es indecidible

Demostración.
(Idea) Dada una máquina de Turing $\mathcal{T}$, se puede escribir una fórmula $\varphi_{\mathcal{T}}$ tal que

- Un modelo de $\varphi_{\mathcal{T}}$ represente una corrida de $\mathcal{T}$ que termina, con lo cual...
- $\varphi_{\mathcal{T}}$ sea satisfacible sii $\mathcal{T}$ termina

(Ver, e.g., 'Mathematical Logic', Ebbinghaus, Flum y Thomas) □

Más transferencia: decidibilidad de la lógica modal básica

Teorema

El fragmento formado por las fórmulas de primer orden en que sólo aparecen dos variables (FO2) es decidible

Demostración.

Acto de fé. La prueba original es de Scott, 1962 (‘A decision method for validity of sentences in two variables’) para FO2 sin igualdad. El resultado con igualdad es de Mortimer, 1975 (‘On languages with two variables’) □

Teorema

La lógica modal básica es decidible

Demostración.

Fácil: dada φ, traducimos con la ST que usa dos variables y usamos cualquier método de decisión para FO2 con $\forall x.\mathsf{ST}_x(\varphi)$ como entrada □

Decidibilidad y poder expresivo

- Teníamos una pregunta pendiente:

> ¿Existe una traducción de fórmulas de primer orden a fórmulas de la lógica modal básica?

- Y la acabamos de responder... ¡por la negativa!
- Porque si existiera tal traducción, primer orden sería decidible
- Usamos *decidibilidad* para hablar de *expresividad*
- No es “constructivo” (no nos dice qué no es expresable)
- Nueva pregunta:

> ¿Existe una traducción de fórmulas de FO2 a fórmulas de la lógica modal básica?

Extendiendo la traducción estándar

- Es fácil extender la traducción estándar a otros operadores modales
- ¡Y transferir resultados!

Ejemplo

1. $\mathcal{M}, w \models \mathsf{E}\varphi$ sii existe v tal que $\mathcal{M}, v \models \varphi$

$$\begin{aligned} \mathsf{ST}_x(\mathsf{E}\varphi) &\equiv \exists y.\mathsf{ST}_y(\varphi) \\ \mathsf{ST}_y(\mathsf{E}\varphi) &\equiv \exists y.\mathsf{ST}_y(\varphi) \end{aligned}$$

2. $\mathcal{M}, w \models \langle m\rangle^{-1}\varphi$ sii existe v tal que $R_m vw$ y $\mathcal{M}, v \models \varphi$

$$\begin{aligned} \mathsf{ST}_x(\langle m\rangle^{-1}\varphi) &\equiv \exists y.R_m(y,x) \wedge \mathsf{ST}_y(\varphi) \\ \mathsf{ST}_y(\langle m\rangle^{-1}\varphi) &\equiv \exists x.R_m(x,y) \wedge \mathsf{ST}_x(\varphi) \end{aligned}$$

Extendiendo la traducción estándar

Ejemplo (cont)

3. $\mathcal{M}, w \models \langle\pi\rangle\varphi$ sii existe v tal que $(w,v) \in \overline{\pi}$ y y $\mathcal{M}, v \models \varphi$

Donde

$$\begin{aligned} \overline{a} &:= R_a \\ \overline{\pi_1 \cup \pi_2} &:= \overline{\pi_1} \cup \overline{\pi_2} \\ \overline{\pi_1 ; \pi_2} &:= \overline{\pi_1} \circ \overline{\pi_2} \\ \overline{\pi^*} &:= \overline{\pi}^* \end{aligned}$$

- Nos alcanzaría con dar una traducción TR que cumpla

$$\mathcal{I}, g \models \mathsf{TR}_\pi(x,y) \quad \text{sii} \quad (g(x), g(y)) \in \overline{\pi}$$

- Porque en ese caso

$$\mathsf{ST}_x(\langle\pi\rangle\varphi) := \exists y\,.\,\mathsf{TR}_\pi(x,y) \wedge \mathsf{ST}_y(\varphi)$$

Traduciendo las relaciones de PDL

$$\begin{array}{lcl} \mathsf{TR}_a(x,y) & := & P_a(x,y) \\ \mathsf{TR}_{\pi_1 \cup \pi_2}(x,y) & := & \mathsf{TR}_{\pi_1}(x,y) \vee \mathsf{TR}_{\pi_2}(x,y) \\ \mathsf{TR}_{\pi_1;\pi_2}(x,y) & := & \exists z \,.\, \mathsf{TR}_{\pi_1}(x,z) \wedge \mathsf{TR}_{\pi_2}(z,y) \\ \mathsf{TR}_{\pi^*}(x,y) & := & \text{¿Nos rendimos?} \end{array}$$

El poder de la clausura (reflexivo-)transitiva

- Sea Γ el siguiente conjunto infinito de fórmulas

$$\begin{array}{r} \langle \pi^* \rangle \neg p \\ p \\ [\pi]\,p \\ [\pi][\pi]\,p \\ [\pi][\pi][\pi]\,p \\ \vdots \end{array}$$

- Todo Γ_0 finito, subconjunto de Γ, es satisfacible
- Pero Γ no lo es
- Con lo cual, PDL no tiene compacidad
- ¡Y por lo tanto no puede traducirse a primer orden!
- Corolario: algo importante que primer orden no puede expresar:
  ¡la clausura transitiva de una relación!

Para cerrar

En esta clase vimos…

- Que la lógica modal básica es un fragmento propio de primer orden
- Y que lo mismo se puede decir de muchas extensiones
- Que lo que perdemos en poder expresivo, lo ganamos en propiedades meta-lógicas
- Y que lo mismo sucede entre primer orden y lógicas más expresivas
- Que las lógicas modales no están confinadas a fragmentos de primer orden
  - (¡PDL es un fragmento *decidible* de lógica de segundo orden!)