Una Introducción a las Lógicas Híbridas

Carlos Areces

carlos.areces@gmail.com

FaMAF - UNC
Córdoba - Argentina

28 de Octubre, Rosario, Argentina

Estructura de la Charla

Lógicas Modales

Sintaxis: Lógica Proposicional $+$ *modalidades*

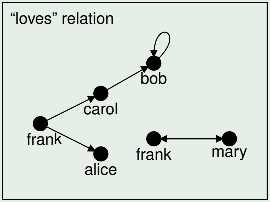
Semántica: Interpretada en términos de estructuras relacionales (Grafos)

Lógicas Modales

Sintaxis: Lógica Proposicional + *modalidades*

Semántica: Interpretada en términos de estructuras relacionales (Grafos)

Example



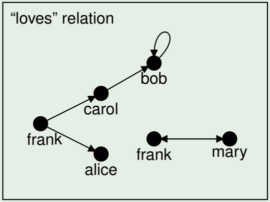
Query: "Alguien ama a un solitario/a?"
(es cierta esto en algún punto del modelo?)

Lógicas Modales

Sintaxis: Lógica Proposicional + *modalidades*

Semántica: Interpretada en términos de estructuras relacionales (Grafos)

Example



Query: "Alguien ama a un solitario/a?" (es cierta esto en algún punto del modelo?)

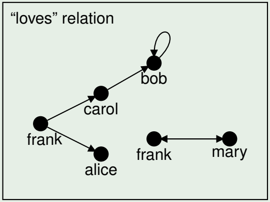
'The Modal Way': $\varphi := \langle loves \rangle [loves] \bot$

Lógicas Modales

Sintaxis: Lógica Proposicional + *modalidades*

Semántica: Interpretada en términos de estructuras relacionales (Grafos)

Example



Query: "Alguien ama a un solitario/a?"
(es cierta esto en algún punto del modelo?)

'The Modal Way': $\varphi := \langle \mathit{loves} \rangle [\mathit{loves}] \bot$

- ML puede ser pensada como un fragmento de LPO
- Las lógicas modales son (usualmente) decidibles!
  - SAT de la lógica modal básica es PSpace-complete

Lógicas Modales

Sintaxis

Dado PROP un conjunto de símbolos de proposición y REL un conjunto de símbolos de relación:

$$\text{FORM} := p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle R\rangle\varphi \mid [R]\varphi,$$

donde $p \in \text{PROP}$, $R \in \text{REL}$, $\varphi, \varphi_1, \varphi_2 \in \text{FORM}$.

Lógicas Modales

Sintaxis

Dado PROP un conjunto de símbolos de proposición y REL un conjunto de símbolos de relación:

$$\text{FORM} := p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle R \rangle \varphi \mid [R]\varphi,$$

donde $p \in$ PROP, $R \in$ REL, $\varphi, \varphi_1, \varphi_2 \in$ FORM.

Como dijimos, son formulas booleanas con algunos operadores unarios raros mezclados.

Lógicas Modales

Modelos

Dado $\langle \mathrm{PROP}, \mathrm{REL} \rangle$ un modelo modal (modelo de Kripke) es una estructura $\mathcal{M} = \langle W, \{R_i^{\mathcal{M}} \mid R_i \in \mathrm{REL}\}, V \rangle$ donde

- W es un conjunto no vacío
- $R_i^{\mathcal{M}} \subseteq W \times W$
- $V : \mathrm{PROP} \to 2^W$

Lógicas Modales

Modelos

Dado $\langle \mathrm{PROP}, \mathrm{REL} \rangle$ un modelo modal (modelo de Kripke) es una estructura $\mathcal{M} = \langle W, \{R_i^{\mathcal{M}} \mid R_i \in \mathrm{REL}\}, V \rangle$ donde

- W es un conjunto no vacío
- $R_i^{\mathcal{M}} \subseteq W \times W$
- $V : \mathrm{PROP} \rightarrow 2^W$

En otras palabras, es un grafo dirigido decorado (juro que sí es!) W es el conjunto de nodos, cada R_i es un conjunto de ejes (pueden pensar que son ejes de distinto color en el grafo), y V asigna differentes etiquetas a cada nodo.

Lógicas Modales

Semántica

Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \mathrm{REL}\}, V\rangle$ y $w \in W$:

$$\mathcal{M}, w \models p \quad \text{sii} \quad w \in V(p) \text{ for } p \in \mathrm{PROP}$$

Lógicas Modales

Semántica

Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \text{REL}\}, V\rangle$ y $w \in W$:

$$
\begin{array}{rcl}
\mathcal{M}, w \models p & \text{sii} & w \in V(p) \text{ for } p \in \text{PROP} \\
\mathcal{M}, w \models \neg\varphi & \text{sii} & \mathcal{M}, w \not\models \varphi \\
\mathcal{M}, w \models \varphi \wedge \psi & \text{sii} & \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi
\end{array}
$$

Lógicas Modales

Semántica

Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \text{REL}\}, V\rangle$ y $w \in W$:

$$
\begin{array}{rcl}
\mathcal{M}, w \models p & \text{sii} & w \in V(p) \text{ for } p \in \text{PROP} \\
\mathcal{M}, w \models \neg\varphi & \text{sii} & \mathcal{M}, w \not\models \varphi \\
\mathcal{M}, w \models \varphi \wedge \psi & \text{sii} & \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi \\
\mathcal{M}, w \models \langle R\rangle\varphi & \text{sii} & \exists w' \in W \text{ s.t. } R^{\mathcal{M}}(w, w') \text{ and } \mathcal{M}, w' \models \varphi \\
\mathcal{M}, w \models [R]\varphi & \text{sii} & \forall w' \in W,\ R^{\mathcal{M}}(w, w') \text{ implies } \mathcal{M}, w' \models \varphi
\end{array}
$$

Lógicas Modales

Semántica

Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \text{REL}\}, V\rangle$ y $w \in W$:

$$
\begin{array}{rcl}
\mathcal{M}, w \models p & \text{sii} & w \in V(p) \text{ for } p \in \text{PROP} \\
\mathcal{M}, w \models \neg\varphi & \text{sii} & \mathcal{M}, w \not\models \varphi \\
\mathcal{M}, w \models \varphi \wedge \psi & \text{sii} & \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi \\
\mathcal{M}, w \models \langle R\rangle\varphi & \text{sii} & \exists w' \in W \text{ s.t. } R^{\mathcal{M}}(w, w') \text{ and } \mathcal{M}, w' \models \varphi \\
\mathcal{M}, w \models [R]\varphi & \text{sii} & \forall w' \in W,\ R^{\mathcal{M}}(w, w') \text{ implies } \mathcal{M}, w' \models \varphi
\end{array}
$$

$\langle R\rangle\varphi$: Algún R-sucesor satisface φ.

$[R]\varphi$: Todos los R-sucesores satisfacen φ.

Fragmentos de LPO: La Traducción Standard

Traducción Standard

La *traducción Standard ST* lleva fórmulas modales a fórmulas de LPO (en la signatura adecuada):

$$ST_x(p_i) \quad := \quad P_i(x)$$

Fragmentos de LPO: La Traducción Standard

Traducción Standard

La *traducción Standard ST* lleva fórmulas modales a fórmulas de LPO (en la signatura adecuada):

$$\begin{array}{rcl} ST_x(p_i) & := & P_i(x) \\ ST_x(\neg\varphi) & := & \neg ST_x(\varphi) \\ ST_x(\varphi_1 \vee \varphi_2) & := & ST_x(\varphi_1) \vee ST_x(\varphi_2) \end{array}$$

Fragmentos de LPO: La Traducción Standard

Traducción Standard

La *traducción Standard ST* lleva fórmulas modales a fórmulas de LPO (en la signatura adecuada):

$$
\begin{array}{rcl}
ST_x(p_i) & := & P_i(x) \\
ST_x(\neg\varphi) & := & \neg ST_x(\varphi) \\
ST_x(\varphi_1 \vee \varphi_2) & := & ST_x(\varphi_1) \vee ST_x(\varphi_2) \\
ST_x(\langle r_i \rangle \varphi) & := & \exists y.(R_i(x,y) \wedge ST_y(\varphi)) \\
ST_x([r_i]\varphi) & := & \forall y.(R_i(x,y) \rightarrow ST_y(\varphi))
\end{array}
$$

(donde y es una variable *nueva*)

Fragmentos de LPO: La Traducción Standard

Traducción Standard

La *traducción Standard ST* lleva fórmulas modales a fórmulas de LPO (en la signatura adecuada):

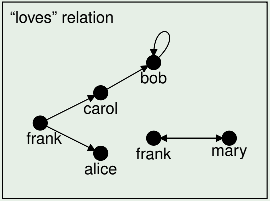
$$
\begin{array}{rcl}
ST_x(p_i) & := & P_i(x) \\
ST_x(\neg\varphi) & := & \neg ST_x(\varphi) \\
ST_x(\varphi_1 \vee \varphi_2) & := & ST_x(\varphi_1) \vee ST_x(\varphi_2) \\
ST_x(\langle r_i \rangle \varphi) & := & \exists y.(R_i(x,y) \wedge ST_y(\varphi)) \\
ST_x([r_i]\varphi) & := & \forall y.(R_i(x,y) \rightarrow ST_y(\varphi))
\end{array}
$$

(donde y es una variable *nueva*)

Para todo φ y todo $\mathcal{M}, w$: $\mathcal{M}, w \models \varphi$ sii $\mathcal{M} \models ST_x(\varphi)[x := w]$

Lógicas Modales

Example

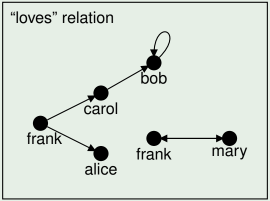


Query: "Alguien ama a algun solitario/a?"

En Modal: $\varphi := \langle loves \rangle [loves] \bot$

Lógicas Modales

Example



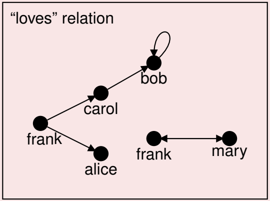
Query: “Alguien ama a algun solitario/a?”

En Modal: $\varphi := \langle loves \rangle [loves] \bot$

$$ST_x(\varphi) := \exists y.(loves(x, y) \land \forall z.(loves(y, z) \rightarrow \bot))$$

Los Límites de la Expresividad Modal

Ciertas propiedades no pueden expresarse en el lenguaje modal básico. . .



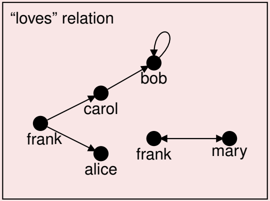
Query: "Ama Frank a Alice?"

Query: "Hay alguien que se quiera a sí mismo?"

Query: "Hay dos personas que se quieran entre ellas?"

Los Límites de la Expresividad Modal

Ciertas propiedades no pueden expresarse en el lenguaje modal básico…



Query: "Ama Frank a Alice?"

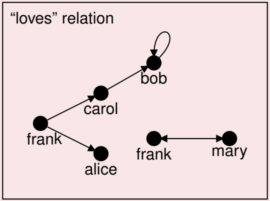
Query: "Hay alguien que se quiera a sí mismo?"

Query: "Hay dos personas que se quieran entre ellas?"

- Que hace falta?
  1. *constantes*
  2. *igualdades*

Los Límites de la Expresividad Modal

Ciertas propiedades no pueden expresarse en el lenguaje modal básico. . .



Query: "Ama Frank a Alice?"

Query: "Hay alguien que se quiera a sí mismo?"

Query: "Hay dos personas que se quieran entre ellas?"

- Que hace falta?
  1. *constantes*
  2. *igualdades*
- Estas limitaciones motival el trabajo en Lógicas Híbridas

La Lógica Híbrida $\mathcal{H}(@)$

La Receta Básica

lógica modal básica

La Lógica Híbrida $\mathcal{H}(@)$

La Receta Básica

lógica modal básica
\+ *nominales* $\rightarrow$ un nuevo sort de átomos

La Lógica Híbrida $\mathcal{H}(@)$

La Receta Básica

lógica modal básica

+	*nominales*	$\rightarrow$	un nuevo sort de átomos
+	@	$\rightarrow$	el operador ‘at’

La Lógica Híbrida $\mathcal{H}(@)$

La Receta Básica

lógica modal básica			
+	*nominales*	$\rightarrow$	un nuevo sort de átomos
+	$@$	$\rightarrow$	el operador 'at'
	$\mathcal{H}(@)$	$\rightarrow$	la lógica híbrida básica

La Lógica Híbrida $\mathcal{H}(@)$

La Receta Básica

	lógica modal básica		
+	*nominales*	$\rightarrow$	un nuevo sort de átomos
+	@	$\rightarrow$	el operador 'at'
	$\mathcal{H}(@)$	$\rightarrow$	la lógica híbrida básica

- Los nominales denotan elementos (nodos) en el modelo
- $@_i\varphi$ es verdadera sii φ es cierta en el elemento denotado por i

La Lógica Híbrida $\mathcal{H}(@)$

Sintaxis: $\mathrm{FORM} := p \mid i \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle R\rangle\varphi \mid [R]\varphi \mid @_i\varphi$,
$i \in \mathrm{NOM}$

La Lógica Híbrida $\mathcal{H}(@)$

Sintaxis: FORM $:= p \mid i \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle R\rangle\varphi \mid [R]\varphi \mid @_i\varphi$, $i \in$ NOM

Semántica: Restringir la valuación V de modo que $V(i)$ es un conjunto unitario para $i \in$ NOM.
Definimos

$$\begin{array}{rcl} \mathcal{M}, w \models i & \text{sii} & w \in V(i) \text{ (sii } V(i) = \{w\}) \\ \mathcal{M}, w \models @_i\varphi & \text{sii} & \mathcal{M}, w' \models \varphi \text{ for } w' \in V(i) \end{array}$$

La Lógica Híbrida $\mathcal{H}(@)$

Sintaxis: FORM $:= p \mid i \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle R \rangle \varphi \mid [R]\varphi \mid @_i\varphi$, $i \in$ NOM

Semántica: Restringir la valuación V de modo que $V(i)$ es un conjunto unitario para $i \in$ NOM.
Definimos

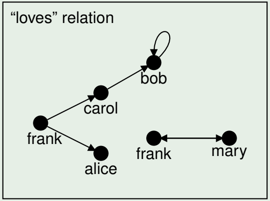
$$\begin{array}{rcl} \mathcal{M}, w \models i & \text{sii} & w \in V(i) \text{ (sii } V(i) = \{w\}) \\ \mathcal{M}, w \models @_i\varphi & \text{sii} & \mathcal{M}, w' \models \varphi \text{ for } w' \in V(i) \end{array}$$

Extensión de ST_x a $\mathcal{H}(@)$

$$\begin{array}{lcl} ST_x(i) & := & x = i \\ ST_x(@_i\varphi) & := & \exists x.(x = i \wedge ST_x(\varphi)) \end{array}$$

Poder Expresivo de $\mathcal{H}(@)$

Example

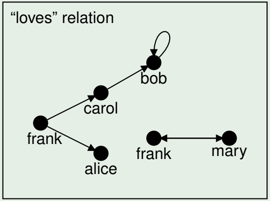


Query: "Hay dos personas que se quieren entre ellas?"

En $\mathcal{H}(@)$: $\varphi :=$
$@_f\langle loves\rangle m \wedge @_m\langle loves\rangle f \wedge @_f\neg m$

Poder Expresivo de $\mathcal{H}(@)$

Example



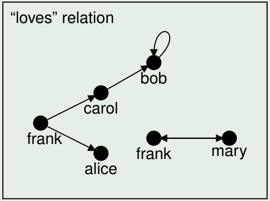
Query: "Hay dos personas que se quieren entre ellas?"

En $\mathcal{H}(@)$: $\varphi :=$
$@_f\langle loves\rangle m \wedge @_m\langle loves\rangle f \wedge @_f\neg m$

$$ST_x(\varphi) := loves(f, m) \wedge loves(m, f) \wedge \neg(f = m)$$

Poder Expresivo de $\mathcal{H}(@)$

Example



Query: "Hay dos personas que se quieren entre ellas?"

En $\mathcal{H}(@)$: $\varphi :=$
$@_f\langle loves\rangle m \wedge @_m\langle loves\rangle f \wedge @_f\neg m$

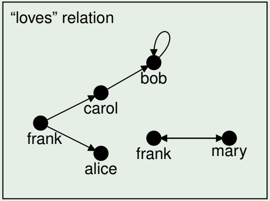
$$ST_x(\varphi) := loves(f,m) \wedge loves(m,f) \wedge \neg(f = m)$$

$$\mid \; @_i i$$

Para $i, j, k \in$ NOM:

Poder Expresivo de $\mathcal{H}(@)$

Example



Query: "Hay dos personas que se quieren entre ellas?"

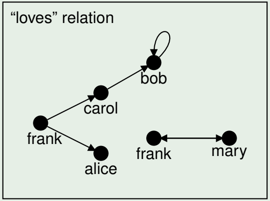
En $\mathcal{H}(@)$: $\varphi :=$ $@_f\langle loves\rangle m \wedge @_m\langle loves\rangle f \wedge @_f\neg m$

$$ST_x(\varphi) := loves(f, m) \wedge loves(m, f) \wedge \neg(f = m)$$

Para $i, j, k \in$ NOM: $\left| \begin{array}{l} @_i i \\ @_i j \rightarrow @_j i \end{array} \right.$

Poder Expresivo de $\mathcal{H}(@)$

Example



Query: “Hay dos personas que se quieren entre ellas?”

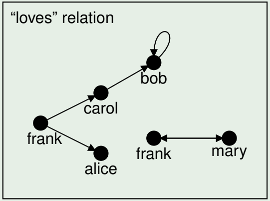
En $\mathcal{H}(@)$: $\varphi :=$
$@_f\langle loves\rangle m \wedge @_m\langle loves\rangle f \wedge @_f\neg m$

$$ST_x(\varphi) := loves(f, m) \wedge loves(m, f) \wedge \neg(f = m)$$

$$\text{Para } i, j, k \in \text{NOM:} \left| \begin{array}{l} @_i i \\ @_i j \rightarrow @_j i \\ @_i j \wedge @_j k \rightarrow @_i k \end{array} \right.$$

Poder Expresivo de $\mathcal{H}(@)$

Example



Query: "Hay dos personas que se quieren entre ellas?"

En $\mathcal{H}(@)$: $\varphi :=$
$@_f\langle loves\rangle m \wedge @_m\langle loves\rangle f \wedge @_f\neg m$

$$ST_x(\varphi) := loves(f, m) \wedge loves(m, f) \wedge \neg(f = m)$$

$$\text{Para } i, j, k \in \text{NOM:} \left| \begin{array}{l} @_i i \\ @_i j \rightarrow @_j i \\ @_i j \wedge @_j k \rightarrow @_i k \\ @_i j \rightarrow (@_i \varphi \leftrightarrow @_j \varphi) \end{array} \right.$$

Algunas Propiedades de $\mathcal{H}(@)$

- Complejidad: Todavía PSpace-complete

Algunas Propiedades de $\mathcal{H}(@)$

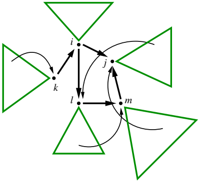
- Complejidad: Todavía PSpace-complete
  - Pero $K_t + 1$ nominal es ExpTime-complete!

Algunas Propiedades de $\mathcal{H}(@)$

- Complejidad: Todavía PSpace-complete
  - Pero $K_t + 1$ nominal es ExpTime-complete!
- Se pierde la "Tree Model Property"

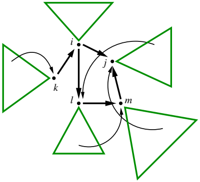
Algunas Propiedades de $\mathcal{H}(@)$

- Complejidad: Todavía PSpace-complete
  - Pero $K_t + 1$ nominal es ExpTime-complete!
- Se pierde la "Tree Model Property"
  - Pero preservamos una forest model property:



Algunas Propiedades de $\mathcal{H}(@)$

- Complejidad: Todavía PSpace-complete
  - Pero $K_t + 1$ nominal es ExpTime-complete!
- Se pierde la "Tree Model Property"
  - Pero preservamos una forest model property:



  - El μ-calculus híbrido con operador de pasado y modalidad universal es ExpTime-complete, y la demostración usa tree-automatas.

La Lógica Híbrida $\mathcal{H}(@, \downarrow)$

- Agregar un tercer conjunto VAR de símbolso atómicos (variables)

La Lógica Híbrida $\mathcal{H}(@,\downarrow)$

- Agregar un tercer conjunto VAR de símbolso atómicos (variables)
- Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \text{REL}\}, V\rangle$, sea $g : \text{VAR} \to W$ una asignación para $\mathcal{M}$. Interpretar las variables de VAR como:

$$\mathcal{M}, g, w \models x \text{ sii } g(x) = w$$

La Lógica Híbrida $\mathcal{H}(@,\downarrow)$

- Agregar un tercer conjunto VAR de símbolso atómicos (variables)
- Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \text{REL}\}, V\rangle$, sea $g : \text{VAR} \to W$ una asignación para $\mathcal{M}$. Interpretar las variables de VAR como:

$$\mathcal{M}, g, w \models x \text{ sii } g(x) = w$$

- Agregar binders. E.g. $\downarrow$, con la siguiente semántica:

$$\mathcal{M}, g, w \models \downarrow x.\varphi \text{ sii } \mathcal{M}, g^x_w, w \models \varphi$$

donde $g^x_w(x) = w$ y $g^x_w = g$ si no

La Lógica Híbrida $\mathcal{H}(@, \downarrow)$

- Agregar un tercer conjunto VAR de símbolso atómicos (variables)
- Dado un modelo $\mathcal{M} = \langle W, \{R^{\mathcal{M}} \mid R \in \text{REL}\}, V \rangle$, sea $g : \text{VAR} \to W$ una asignación para $\mathcal{M}$. Interpretar las variables de VAR como:

$$\mathcal{M}, g, w \models x \text{ sii } g(x) = w$$

- Agregar binders. E.g. $\downarrow$, con la siguiente semántica:

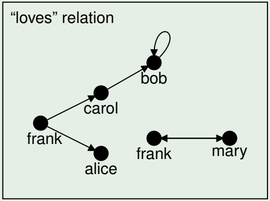
$$\mathcal{M}, g, w \models \downarrow x.\varphi \text{ sii } \mathcal{M}, g^x_w, w \models \varphi$$

donde $g^x_w(x) = w$ y $g^x_w = g$ si no

Extensión de ST_x a $\mathcal{H}(@, \downarrow)$

$$\begin{array}{lcl} ST_x(x_i) & := & x = x_i \\ ST_x(\downarrow x_i.\varphi) & := & \exists x_i.(x_i = x \wedge ST_x(\varphi)) \end{array}$$

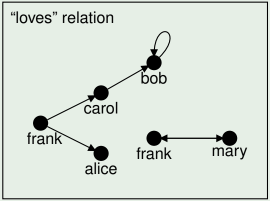
Example



Query: "Hay alguien que se quiera a sí mismo?"

En $\mathcal{H}(@, \downarrow)$: $\varphi := \downarrow x_i.\langle loves \rangle x_i$

Example

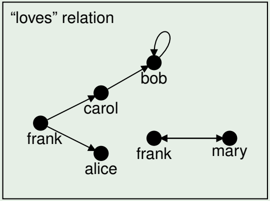


Query: "Hay alguien que se quiera a sí mismo?"

En $\mathcal{H}(@,\downarrow)$: $\varphi := \downarrow x_i.\langle loves\rangle x_i$

$$ST_x(\varphi) := \exists x_i.((x = x_i) \wedge \exists y.(loves(x,y) \wedge y = x_i)) = loves(x,x)$$

Example



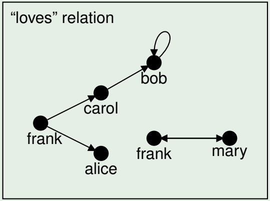
Query: "Hay alguien que se quiera a sí mismo?"

En $\mathcal{H}(@,\downarrow)$: $\varphi := \downarrow x_i.\langle loves\rangle x_i$

$$ST_x(\varphi) := \exists x_i.((x = x_i) \wedge \exists y.(loves(x,y) \wedge y = x_i)) = loves(x,x)$$

Para $x \in$ VAR: $\mid \downarrow x.x$

Example



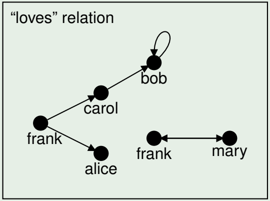
Query: "Hay alguien que se quiera a sí mismo?"

En $\mathcal{H}(@,\downarrow)$: $\varphi := \downarrow x_i.\langle loves\rangle x_i$

$$ST_x(\varphi) := \exists x_i.((x = x_i) \wedge \exists y.(loves(x,y) \wedge y = x_i)) = loves(x,x)$$

Para $x \in$ VAR: $\left| \begin{array}{l} \downarrow x.x \\ \neg\downarrow x.\varphi \leftrightarrow \downarrow x.\neg\varphi \end{array} \right.$

Example



Query: "Hay alguien que se quiera a sí mismo?"

En $\mathcal{H}(@,\downarrow)$: $\varphi := \downarrow x_i.\langle loves\rangle x_i$

$$ST_x(\varphi) := \exists x_i.((x = x_i) \wedge \exists y.(loves(x,y) \wedge y = x_i)) = loves(x,x)$$

Para $x \in$ VAR:

$$\left|\begin{array}{l} \downarrow x.x \\ \neg\downarrow x.\varphi \leftrightarrow \downarrow x.\neg\varphi \\ \downarrow x.(\langle R\rangle(\varphi \wedge x) \to \varphi) \end{array}\right.$$

Algunas Propiedades de $\mathcal{H}(@, \downarrow)$

- Una lógica con muy buen comportamiento teórico: simple axiomatización, buena teoría de modelos, etc.

Algunas Propiedades de $\mathcal{H}(@,\downarrow)$

- Una lógica con muy buen comportamiento teórico: simple axiomatización, buena teoría de modelos, etc.
- Muy Expresiva: Equivalente al “Bounded Fragment” de LPO

Algunas Propiedades de $\mathcal{H}(@, \downarrow)$

- Una lógica con muy buen comportamiento teórico: simple axiomatización, buena teoría de modelos, etc.
- Muy Expresiva: Equivalente al “Bounded Fragment” de LPO
- Ya la lógica modal básica $+ \downarrow$ es indecidible

Un Ejemplo Interesante

- Consideremos un sistema de alarma. Queremos verificar la propiedad de que siempre que la alarma suena, haya antes habido algún problema en el sistema.

Un Ejemplo Interesante

- Consideremos un sistema de alarma. Queremos verificar la propiedad de que siempre que la alarma suena, haya antes habido algún problema en el sistema.

$$[F](alarma \to \langle P \rangle problema)$$

Un Ejemplo Interesante

- Consideremos un sistema de alarma. Queremos verificar la propiedad de que siempre que la alarma suena, haya antes habido algún problema en el sistema.

$$[F](alarma \rightarrow \langle P\rangle problema)$$

- Agregemos un botón de reset (para apagar la alarma). Queremos verificar ahora que si la alarma suena, haya antes habido algún problema a partir del ultimo reset.

Un Ejemplo Interesante

- Consideremos un sistema de alarma. Queremos verificar la propiedad de que siempre que la alarma suena, haya antes habido algún problema en el sistema.

$$[F](alarma \to \langle P \rangle problema)$$

- Agregemos un botón de reset (para apagar la alarma). Queremos verificar ahora que si la alarma suena, haya antes habido algún problema a partir del ultimo reset.

$$[F](reset \to [F](alarma \to \langle P \rangle problema)) \ (?)$$

Un Ejemplo Interesante

- Consideremos un sistema de alarma. Queremos verificar la propiedad de que siempre que la alarma suena, haya antes habido algún problema en el sistema.

$$[F](alarma \to \langle P \rangle problema)$$

- Agregemos un botón de reset (para apagar la alarma). Queremos verificar ahora que si la alarma suena, haya antes habido algún problema a partir del ultimo reset.

$$[F](reset \to [F](alarma \to \langle P \rangle problema)) \text{ (?)}$$

- El sistema no puede olvidar la ocurrencia de un problema.

Un Ejemplo Interesante

- Consideremos un sistema de alarma. Queremos verificar la propiedad de que siempre que la alarma suena, haya antes habido algún problema en el sistema.

$$[F](alarma \rightarrow \langle P \rangle problema)$$

- Agregemos un botón de reset (para apagar la alarma). Queremos verificar ahora que si la alarma suena, haya antes habido algún problema a partir del ultimo reset.

$$[F](reset \rightarrow [F](alarma \rightarrow \langle P \rangle problema)) \ (?)$$

- El sistema no puede olvidar la ocurrencia de un problema.

$$[F]{\downarrow}x.(reset \rightarrow [F](alarma \rightarrow \langle P \rangle (problema \wedge \langle P \rangle x)))$$

Conclusiones

Lo que hicimos:

- Algoritmos eficientes de razonamiento para lógicas híbridas (tableaux, resolución, etc.)
    - Nuevas optimizaciones y estrategias
    - Implementaciones (hTab, HyLoRes)
- Extensiones (Memory Logics)
- Model Checking

Conclusiones

Lo que hicimos:

- Algoritmos eficientes de razonamiento para lógicas híbridas (tableaux, resolución, etc.)
  - Nuevas optimizaciones y estrategias
  - Implementaciones (hTab, HyLoRes)
- Extensiones (Memory Logics)
- Model Checking

Lo que falta hacer:

- Una perspectiva algebraica
- Testing en Aplicaciones

Conclusiones

Lo que hicimos:

- Algoritmos eficientes de razonamiento para lógicas híbridas (tableaux, resolución, etc.)
  - Nuevas optimizaciones y estrategias
  - Implementaciones (hTab, HyLoRes)
- Extensiones (Memory Logics)
- Model Checking

Lo que falta hacer:

- Una perspectiva algebraica
- Testing en Aplicaciones

Para mas información:

- Hybrid Logics Web Pages @ `http://hylo.loria.fr`
- InToHyLo Project @ `http://www.glyc.dc.uba.ar/intohylo`