Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Complejidad computacional de lógicas híbridas sub-booleanas

Daniel Koile
dkoile@dc.uba.ar

Director
Lic. Daniel Gorín
FCEyN UBA

Co-Director
Dr. Carlos Areces
INRIA Lorraine

Tesis para acceder al grado de
Licenciado en Ciencias de la Computación

22 de diciembre de 2008

Resumen

Las lógicas híbridas son una familia de lógicas modales que incorporan algunas nociones de identidad. Se caracterizan por incluir constantes (nominales) y, generalmente, algunos operadores híbridos, como @ ó $\downarrow$. El primero, llamado operador de satisfacción, permite indicar explícitamente el individuo sobre el cual se predica. El segundo se puede ver como un mecanismo de asignar nombres a individuos de manera dinámica.

En esta tesis investigamos el efecto que tiene, en términos de complejidad computacional, el agregado de nominales y operadores híbridos a una lógica modal (proceso conocido como *hibridización*), con el foco puesto en el operador $\downarrow$. Es sabido que este es un operador sumamente expresivo: mientras la lógica modal básica es PSPACE-completa, al hibridizarla únicamente con este operador obtenemos una lógica indecidible. Nuestro objetivo es estudiar hibridizaciones de lógicas modales sub-booleanas (i.e. que no incluyan un conjunto adecuado de operadores booleanos) para así delinear de manera más precisa la expresividad de este operador.

Para ello extendemos con operadores híbridos a NL, un cálculo presentado por Lambek en la década del 6́0, similar a un sistema de tipos simple para el cálculo-lambda, utilizado en lingüística computacional. Este cálculo puede ser visto alternativamente como una lógica modal sin estructura booleana. El problema de derivabilidad (o, alternativamente, entailment lógico) en este cálculo puede ser resuelto en tiempo polinomial. NL es parte de una familia de lógicas llamadas Categorial Type Logics (CTL).

Utilizando un sistema de inferencia basado en tableaux, y traducciones de lógicas conocidas, damos resultados de complejidad para las hibridizaciones de NL que llamamos hCTL(@), hCTL($@, \downarrow$) y hCTL($\downarrow$). En particular, podemos ver que esta última también es indecidible. Este es un resultado sorprendente, teniendo en cuenta la bajísima expresividad de la lógica hibridizada.

Agradecimientos

Primero quiero agradecer a Daniel y a Carlos por toda la voluntad, tiempo, y paciencia que pusieron para hacer este trabajo posible.

También me gustaría agradecer al jurado, por haberse dedicado en la lectura y corrección de esta tesis.

Además, quiero agradecerle a mi familia y amigos, que sin su soporte incondicional, sobretodo en la última etapa, no hubiese podido llegar a este punto.

Índice general

Capítulo 1

Las lógicas modales

1.1. Punto de partida

Aunque podríamos encontrarnos con *lógicas modales* en el trabajo de Aristóteles [Bochenski, 1961], no es sino hasta el año 1918 que podemos ver un claro origen de las lógicas modales contemporáneas en tanto disciplina matemática con un sistema axiomático formal, en el articulo *Survey of Symbolic Logic* de C.I. Lewis [Lewis, 1918].

Lewis considera, a partir de [Whitehead and Russell, 1925], las implicaciones $\varphi \Rightarrow \psi$ en las que la veracidad de la afirmación está dada por el contexto en el cual se está hablando (*contingencia*), y en las que es una *verdad universal*.

En 1932 [Lewis and Langford, 1932] Lewis introduce el *operador modal* $\Diamond$ que representa *posibilidad*, con el que define la implicación estricta $\mapsto$ de la siguiente manera: $\varphi \mapsto \psi \equiv \neg\Diamond(\varphi \wedge \neg\psi)$. Este operador se puede leer como: Es *lógicamente inconcebible* que φ sea verdadero y ψ falso a la vez.

Se dice que $\Diamond$ es un operador modal porque indica un *modo* de verdad. Cuando $\Diamond$ toma el modo de *posibilidad*, lo podemos usar para distinguir lo que *es verdadero* (o falso) de lo que *podría ser verdadero* (o falso). Más aún, si definimos $\Box\varphi \equiv \neg\Diamond\neg\varphi$, el operador $\Box$ estaría capturando la noción de *necesidad*. Como vemos hay una relación muy estrecha entre *necesidad* y *posibilidad*, y cuando dos operadores cumplen esta propiedad se dice que son *duales*, tal y como ocurre en la lógica de primer orden entre el $\forall$ y el $\exists$. Curiosamente el operador $\Box$ no apareció publicado hasta el año 1946 (en un paper de R. Barcan).

Así como interpretamos el operador $\Diamond$ como posibilidad, podríamos haberlo interpretado como *deseo* y pensar el significado de $\Diamond\mathsf{llueve}$ como "desearía que estuviera lloviendo". Pero si queremos capturar efectivamente un significado determinado, como posibilidad, tenemos que dar un paso más y ver que las inferencias hechas con este sistema tienen sentido. Inferir que $\Diamond\mathsf{llueve}$ dado que llueve es correcto si $\Diamond$ representa *posibilidad*, pero es incorrecto si el significado de $\Diamond$ es deseo: Si llueve, claramente es posible que llueva, pero que llueva no implica "desearía que estuviera lloviendo". Formalmente, la fórmula $\mathsf{llueve} \Rightarrow \Diamond\mathsf{llueve}$ es válida en un caso y no en el otro. Para llegar a esto, Lewis analiza en sus trabajos distintos *sistemas axiomáticos*, que son sistemas formales que consisten en axiomas y reglas de deducción, en los que se pueden obtener teoremas por medio de pruebas formales. A partir de estos estudios surgió el interés de capturar sintácticamente otros modos tales como *obligación*, *creencia*, *conocimiento*, *demostrabilidad*... que hasta este momento sólo eran conceptos de la intuición.

Uno de los estudios particularmente interesante en el área fue el de representar y razonar sobre conceptos temporales. Éstas *lógicas temporales* (*temporal logics* o *tense logics*) fueron introducidas por Arthur Prior [Prior, 1957; Prior, 1967] al analizar los argumentos del lógico estoico Diodorus Chronos, quien definió que una proposición es *posible* cuando es verdadera o lo será. Motivado por esto, Prior concibió la idea de utilizar un sistema lógico con operadores temporales análogos a los de la lógica modal, y así introdujo los siguientes conectivos:

Conectivo	Interpretación
F	En algún instante en el futuro
P	En algún instante en el pasado
G	Siempre ocurrirá que
H	Siempre ha ocurrido que

Los operadores F y P son modalidades como $\Diamond$, siendo G y H sus duales (es decir, para una fórmula φ, $G\varphi$ puede definirse como $\neg F\neg\varphi$ y $H\varphi$ como $\neg P\neg\varphi$). A éstos se les suele llamar comúnmente *operadores temporales* (o *tense operators*) y a la lógica se la suele llamar K_t.

Una limitación que podemos apreciar en las lógicas modales que hemos mencionado hasta el momento es el hecho de que no podamos referirnos a objetos particulares. En cierto sentido, las lógicas modales que hemos visto, pueden pensarse como extensiones de la lógica proposicional, sin una estructura de términos como la que usamos en lógica de primer orden para referirnos a objetos particulares del dominio. Por ejemplo, en K_t no podemos referirnos a instantes precisos en el tiempo, no podemos hablar de "ayer" o "el jueves próximo", sino que esta lógica razona sobre momentos indeterminados del futuro o del pasado. Es ya en los trabajos de Prior que se menciona este problema de expresividad de las lógicas modales clásicas, y se propone también una solución. Veremos una de las maneras de enriquecer el poder expresivo de la lógica modal en el Capítulo 2, cuando introduzcamos las lógicas híbridas.

Otra particularidad de K_t es que introduce más de una modalidad (F y P), las cuales no pueden definirse una en función de la otra. Cuando esto ocurre, decimos que la lógica es *multi-modal*. Definiremos ahora en forma genérica el lenguaje utilizado para notar las lógicas multi-modales.

Definición 1.1. Sean los siguientes conjuntos contables y disjuntos: $PROP$ de símbolos de proposición, y REL de símbolos de relación. Definimos las *fórmulas bien formadas* (FBF) de una lógica multi-modal como:

$$FBF ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid \langle r\rangle\varphi$$

con $p \in PROP$, $r \in REL$ y $\varphi, \psi \in FBF$. El operador $[r]$ se define como $[r]\varphi \equiv \neg\langle r\rangle\neg\varphi$. El resto de los operadores lógicos habituales se definen de la forma usual. En los casos en que REL sea un conjunto unitario, usaremos $\Diamond$ y $\Box$ para notar operadores modales, y los operadores F y P de Prior los representaremos como $\langle F\rangle$ y $\langle P\rangle$ respectivamente.

1.2. La semántica relacional

Discutimos hasta ahora las lógicas modales desde un punto de vista puramente sintáctico. Una de las características fundamentales de la lógica moderna es que la perspectiva sintáctica se complementa con una contrapartida semántica. Por ejemplo, si sólo contamos con un aparato sintáctico, la única forma válida de obtener teoremas es la aplicación recursiva de reglas de inferencia hasta llegar al teorema deseado. ¿Cómo probamos entonces que una fórmula *no* es un teorema? Tenemos que argumentar que no existe forma de derivarla en nuestro sistema de inferencia, lo que puede ser extremadamente complejo.

Un avance importante en la lógica en general, y en la lógica modal en particular fue el uso de técnicas algebraicas para estudiar los sistemas modales: podemos agregar un operador para interpretar $\Diamond$ sobre un álgebra Booleana y así obtener las llamadas *álgebras modales*. Lo que se intentaba a partir de la década del sesenta, era *interpretar* mediante estructuras matemáticas los sistemas modales y así darles una *semántica*, lo cual simplifica enormemente la forma de trabajar con las lógicas. Estudios particularmente importantes fueron los de Samuel Kripke [Kripke, 1959; Kripke, 1963a; Kripke, 1963b], que introdujo los llamados *modelos de Kripke*, las estructuras que comúnmente se utilizan como modelos de las lógicas modales. Hoy en día, desde el punto de vista de las Ciencias de la Computación, un modelo de Kripke no es más que un multigrafo dirigido con nodos etiquetados por conjuntos de símbolos proposicionales:

Definición 1.2 (Modelo de Kripke)**.** Un modelo de Kripke se define como una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R \in REL}, V \rangle$, en donde

$$\begin{array}{rl} W & \text{es un conjunto no vacío} \\ R^{\mathcal{M}} \subseteq W \times W & \text{es una relación binaria } \forall R \in REL \\ V(p) \subseteq W & \forall p \in \mathsf{PROP}. \end{array}$$

Comúnmente se llama *mundo*, *nodo* o *estado* a cada $w \in W$, *relaciones de accesibilidad* a cada $R^{\mathcal{M}}$, y *valuación* a la función V.

Kripke se inspiró en el planteo de Leibniz, quien definió (aunque no literalmente) que lo *verdadero* es lo que ocurre necesariamente cuando su negación implica una contradicción, y que hay tantos mundos posibles como cosas puedan concebirse sin contradicción.

Intuitivamente podemos ver a cada nodo de un modelo de Kripke como un mundo posible y a las relaciones de accesibilidad como las distintas modalidades: imaginemos que estamos en un contexto, uno de estos mundos posibles, supongamos que se llama a, y digamos que queremos ver si $\langle R \rangle\mathsf{llueve}$ en este mundo: interpretando al $\langle R \rangle$ como posibilidad querríamos ver si este mundo en el que estamos parados está relacionado con algún otro mundo en el que llueve. La decisión de si en un mundo vale o no una proposición está dada por la valuación, entonces a es uno de los mundos en que llueve cuando $a \in V(\mathsf{llueve})$. Consideremos un ejemplo concreto. Tomemos el modelo

$$\mathcal{M} = \langle \{a, b, c\}, \{R^{\mathcal{M}}(a, b), R^{\mathcal{M}}(a, c)\}, V(\mathsf{llueve}) = \{b\} \rangle$$

evaluamos ahora la fórmula $\langle R \rangle\mathsf{llueve}$ desde el mundo posible a. $\langle R \rangle\mathsf{llueve}$ será efectivamente cierta en a ya que puedo moverme por la relación de accesibilidad a un mundo en el que llueve. Decimos entonces que el modelo $\mathcal{M}$ en el mundo a, *satisface* la fórmula $\langle R \rangle\mathsf{llueve}$, y lo notamos $\mathcal{M}, a \models \langle R \rangle\mathsf{llueve}$. Por otro lado, también puede no llover, ya que desde a puedo acceder a c. En otras palabras, no es *necesario* que llueva, $\mathcal{M}, a \not\models [R]\mathsf{llueve}$. Veamos ahora la definición formal de la relación $\models$ que se define dado un modelo $\mathcal{M}$, un elemento w en el dominio de $\mathcal{M}$ y una fórmula φ.

Definición 1.3 (Interpretación de una fórmula)**.** Dado un lenguaje modal definido sobre $PROP$ y REL, y una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{\in REL}, V \rangle$, $w \in W$, la relación de satisfacción $\models$ se define recursivamente como:

$$\begin{array}{lll} \mathcal{M}, w \models p & \text{sii} & w \in V(p),\ p \in PROP \\ \mathcal{M}, w \models \neg\varphi & \text{sii} & \mathcal{M}, w \not\models \varphi,\ \varphi \in FBF \\ \mathcal{M}, w \models \varphi \wedge \psi & \text{sii} & \mathcal{M}, w \models \varphi \text{ y } \mathcal{M}, w \models \psi,\ \varphi, \psi \in FBF \\ \mathcal{M}, w \models \langle R \rangle\varphi & \text{sii} & \exists w' \in W \ / \ (w, w') \in R^{\mathcal{M}} \text{ y } \mathcal{M}, w' \models \varphi,\ R \in REL. \end{array}$$

Cuando para todo $w \in W$ se cumpla $\mathcal{M}, w \models \varphi$ diremos simplemente $\mathcal{M} \models \varphi$, en cuyo caso se dice que la fórmula φ es *válida* en el modelo $\mathcal{M}$. A esta lógica modal se la llama *lógica modal básica* (notada comúnmente K o LMB).

1.3. La Lógica Modal en la actualidad

Actualmente las lógicas modales se han convertido en herramientas ampliamente utilizadas, principalmente en diversas áreas de la Lingüística y la Computación; con aplicaciones tales como especificación y verificación de sistemas [Manna and Pnueli, 1992], representación de conocimiento [Areces and de Rijke, 2001], y *planning* en inteligencia artificial [Bacchus and Kabanza, 2000]. Quizás se deba a su capacidad de balancear poder expresivo con complejidad computacional (ver Sección 1.4), y a su versatilidad en tanto que se puede hacer una suerte de ingeniería de lógicas dependiendo de las aplicaciones y las propiedades buscadas.

Podemos mencionar como ejemplo la lógica *PDL* (Propositional Dynamic Logic) [Fischer and Ladner, 1979], con infinitas modalidades y usada para modelar el comportamiento de programas.

En esta lógica para cada programa (no determinístico) π vamos a tener una modalidad $\langle\pi\rangle$, en la que la interpretación de $\langle\pi\rangle\varphi$ va a ser: "alguna ejecución que termina de π desde el estado actual nos lleva a un estado donde vale φ". PDL hace explícita la estructura inductiva de los programas, y permite estudiar la forma en la que los programas se pueden componer, iterar, etc. formando nuevos programas.

Otras lógicas que vale la pena mencionar son las lógicas para la descripción (*description logics* o DL). Las DL [Baader *et al.*, 2003] son una familia de lógicas utilizadas principalmente para construir sistemas de representación de conocimiento (*knowledge representation systems* o KRS). Estos sistemas están formados por una base de conocimiento (*knowledge base*) y una serie de servicios de razonamiento (*reasoning services*). Una base de conocimiento la podemos dividir en el conjunto de conceptos, relaciones y reglas que regulan el universo que estamos modelando y en una serie de hechos concretos acerca del mismo.

Con lógicas como éstas se vuelve cada vez más evidente que la complejidad computacional y eficiencia algorítmica juega un papel más que importante en la practicidad de las aplicaciones.

1.4. Sobre los problemas y métodos de inferencia

Ahora bien, sabemos que hay distintos problemas que las lógicas modales ayudan a modelar (o representar) de una u otra manera, pero una vez que modelamos el problema ¿qué es lo que podemos querer hacer con una lógica modal?.

Hay diversas tareas de inferencia que son de interés, por ejemplo podemos querer ver si dada una fórmula y un modelo, la fórmula es *válida* en el modelo. A éste se lo llama un problema de chequeo de modelos (o *model checking*) [Manna and Pnueli, 1992]. Podemos querer también encontrar, dado un modelo y un elemento, una fórmula que lo describa univocamente. Este es un problema de *generación de descripciones* [Areces *et al.*, 2008]. También podríamos querer saber si una fórmula es verdadera en todo mundo de todo modelo (i.e., si la fórmula es universalmente *válida*), a este problema se le llama problema de *validez*. Un problema relacionado al de la validez es el de la *satisfacibilidad*, que requiere determinar dada una fórmula si existe un modelo y un mundo que la satisfagan. Estos últimos dos problemas se dicen *duales* ya que una fórmula φ es válida si y sólo si $\neg\varphi$ es insatisfacible.

Otro problema que nos va a interesar particularmente en esta tesis, es el de *entailment semántico*, o *entailment*, que es una noción semántica de consecuencia. En este problema, dados un conjunto de fórmulas Γ y otra fórmula φ, vamos a querer saber si en todo modelo y todo mundo donde valen todas las fórmulas de Γ, también vale φ. Se lo nota $\Gamma \models \varphi$, y un ejemplo que podemos ver en la lógica modal es el siguiente:

$$\Box(a \wedge b) \wedge \Diamond c \models \Diamond a \wedge \Diamond b$$

ya que si el mundo inicial tiene un sucesor (en el que vale c), y en todos los sucesores valen a y b a la vez, sabremos en particular que ese mundo inicial tiene un sucesor en el que vale a y otro (quizás el mismo) en que vale b.

Para intentar resolver estos problemas, existen distintos métodos de inferencia. Usualmente requerimos que estos métodos sean al menos *correctos*, es decir, que en el caso de responder a un problema de inferencia, retornen la respuesta adecuada. En muchos casos queremos que el método sea también *completo*, es decir, que dé un resultado para cualquier instancia arbitraria del problema. Usualmente el símbolo $\vdash$ representa una derivación sintáctica en el sistema de inferencia. Por ejemplo, en el caso del problema de entailment, cuando el sistema de prueba es completo, las nociones de consecuencia semántica ($\models$) y sintáctica ($\vdash$) coinciden.

Si además estamos interesados en aplicaciones computacionales, usualmente querremos que estos métodos sean también decidibles, y si es posible de baja complejidad. Por ejemplo, el problema de satisfacibilidad de la lógica proposicional es completo para la clase de complejidad NP (problemas que pueden ser resueltos por una Máquina de Turing no determinística en tiempo polinomial respecto del tamaño de la entrada). Existen además métodos como el de Davis-Putnam [Davis and Putnam, 1960], Tableaux [D'Agostino *et al.*, 1999] o Resolución [Robinson, 1965] que son correctos y completos para problemas de complejidad NP. De hecho, el problema de SAT para LP (la lógica proposicional) fue el primer problema identificado como perteneciente a la clase de complejidad NP-complete.

Teorema 1.1. *[Cook, 1971] El problema de satisfacibilidad para la lógica proposicional es NP-complete.*

El problema de satisfacibilidad para la lógica de primer orden, en cambio, es indecidible. Es decir, no existe un método correcto y completo que garantice una respuesta en un número finito de pasos. O sea, que no puede existir un método que dada una fórmula arbitraria de la lógica de primer orden nos devuelva el resultado correcto a la pregunta "¿existe un modelo donde la fórmula sea satisfecha?", estando seguros que vaya a terminar en algún momento.

Teorema 1.2. *[Church, 1936; Turing, 1936] El problema de satisfacibilidad para la lógica de primer orden es indecidible.*

En particular sabemos que la lógica modal básica (K) es un fragmento de la lógica de primer orden, pero es un fragmento *decidible* de la misma. Su complejidad computacional es PSPACE-complete.

Teorema 1.3. *[Ladner, 1977] El problema de satisfacibilidad para la lógica modal básica es PSPACE-complete.*

1.5. Las lógicas modales en esta tesis

Tomando como ejemplo a las lógicas clásicas, la proposicional tiene buenas propiedades computacionales lo que le da la capacidad de resolver problemas en ella algorítmicamente, aunque las aplicaciones que pueden dársele están bastante limitadas por su poder expresivo. La lógica de primer orden, por otro lado, tiene una mayor capacidad de expresión en desmedro de la decidibilidad. Gracias a las lógicas modales, presentadas en este capítulo, podemos tomar las características que nos sean convenientes de distintas lógicas para intentar obtener la lógica "justa" y el modelado buscado.

En esta tesis vamos a combinar varios operadores modales, y en particular propondremos un método de inferencia completo para la lógica resultante, basado en tableaux. Estudiaremos su complejidad computacional, y la de algunos fragmentos.

En el Capítulo 2 vamos a introducir las lógicas híbridas, extensiones de las lógicas modales que incluyen una noción de identidad. En particular, presentamos el operador híbrido $\downarrow$, que nos permite nombrar mundos del modelo. Suele ocurrir que una lógica que incluye $\downarrow$ sea indecidible.

Nos preguntamos si la pérdida de decidibilidad se debe a la interacción con la base booleana de las lógicas. Por ello, en el Capítulo 3 presentaremos CTL, un cálculo presentado por Lambek, similar a un sistema de tipos simple para el cálculo lambda, con aplicaciones lingüísticas, y que puede verse como una lógica modal sin operadores booleanos.

El objetivo de éste trabajo es entonces analizar la interacción de CTL con el operador híbrido $\downarrow$. Para esto introduciremos la lógica hCTL, o sea, hibridizaremos CTL en el Capítulo 4, y presentaremos un sistema de tableaux que demostraremos consistente y completo. A partir del mismo, en los Capítulos 5, 6 y 7 podremos estudiar la complejidad computacional del problema de entailment de la lógica hCTL, utilizando como principal herramienta las traducciones entre distintas lógicas.

Capítulo 2

Las lógicas híbridas

2.1. Motivación

Una de las limitaciones que habíamos mencionado al analizar la lógica temporal básica de Prior, es que no podíamos referirnos a instantes precisos en el futuro o en el pasado, no podíamos expresar conceptos como “ayer llovió” o “la semana que viene granizará”. De hecho esta limitación no es inherente a la lógica temporal. En general podemos pensar las lógicas modales como lenguajes que nos permiten describir un modelo desde una perspectiva interna: estamos *parados* en un contexto, en un mundo en particular del modelo, y podemos *movernos* por las relaciones de accesibilidad a otros mundos y seguir evaluando desde ahí. Pero no podemos ir a algún mundo específico en particular, ni podemos distinguir cuándo dos nodos son o no el mismo, ni tampoco podemos volver al mundo del que partimos con la certeza de que sea el mismo, y no uno suficientemente parecido (*indistinguible*).

Como respuesta a esta limitación expresiva, surgen las *lógicas híbridas*. Las lógicas híbridas son extensiones de las lógicas modales, capaces de referirse explícitamente a estados particulares del modelo. Esta característica los hace particularmente útiles en una variedad de aplicaciones, para las cuales son frecuentemente incluso el lenguaje que naturalmente surge como representación . Las lógicas híbridas usan fórmulas atómicas, llamadas *nominales*, para referirse a estados particulares de un modelo. Un nominal es verdadero en exactamente un estado de un modelo dado, a diferencia de las proposiciones que denotan conjuntos de estados. Ya Prior en sus trabajos utilizó la idea de nominales en lenguajes altamente expresivos, que incluían el cuantificador universal clásico $\forall$ [Prior, 1968].

Usualmente, notamos los nominales con los símbolos i, j, k... Veámoslos ahora en acción. Consideremos la fórmula:

$$\varphi := \Diamond(i \wedge p) \wedge \Diamond(i \wedge q) \Rightarrow \Diamond(p \wedge q).$$

Si el antecedente de φ es satisfecho en un mundo w del modelo, eso significa que el mundo que tiene el nominal i debe ser accesible desde w, y tanto p como q deben valer en ese mundo, y por tanto φ es una fórmula válida. Ahora bien, si en lugar de i pusiésemos una proposición r cualquiera, φ dejaría de ser válida ya que del estado w podría acceder a dos estados distintos, uno donde $(r \wedge p)$ es cierta y otro donde vale $(r \wedge q)$, y por lo tanto el antecedente no asegura la existencia de un estado accesible donde ambos p y q sean ciertos.

2.2. La lógica híbrida básica

La *lógica híbrida básica* ofrece más que nominales: nos permite armar fórmulas del estilo $@_i\varphi$, siendo i un nominal. El operador @ se llama *operador de satisfacción*, y podemos interpretar intuitivamente $@_i\varphi$ como “el mundo en que vale i satisface φ”. Notemos que sin importar en qué mundo estemos “parados”, esta fórmula nos hace saltar directamente al único mundo en el que vale i. Es decir, sin importar el mundo del modelo en el que estemos, $@_i\varphi$ refiere directamente a la verdad o falsedad de φ en el mundo nombrado por i. A las fórmulas de la forma $@_i\varphi$ las llamamos (no muy originalmente) *fórmulas-@*.

Veamos entonces formalmente la sintaxis de la *lógica híbrida básica*, notada $\mathcal{H}(@)$, agregando nominales y el operador @ a la lógica modal básica:

Definición 2.1. Sean los siguientes conjuntos contables y disjuntos: *PROP* de símbolos de proposición, *NOM* de nominales, *REL* de símbolos de relación, y llamemos *átomos* al conjunto $AT = PROP \cup NOM$. Definimos las *fórmulas bien formadas* (*FBF*) de la lógica $\mathcal{H}(@)$ como:

$$FBF ::= a \mid \neg\varphi \mid \varphi \wedge \psi \mid \langle r\rangle\varphi \mid @_i\varphi.$$

con $a \in AT$, $r \in REL$, $i \in NOM$ y $\varphi, \psi \in FBF$. El resto de los operadores lógicos habituales se definen de la forma usual.

Como vemos en la precedente definición, pueden usarse nominales en los contextos en que se podían usar en LMB las proposiciones, aunque no a la inversa. Es mediante la definición de *modelo híbrido* que aseguramos que cada nominal denote exactamente un mundo.

Definición 2.2 (Modelo híbrido)**.** Un modelo híbrido es una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R\in REL}, V\rangle$ donde

$$\begin{array}{rl} W & \text{es un conjunto no vacío} \\ R^{\mathcal{M}} \subseteq W \times W & \text{es una relación binaria } \forall R \in REL \\ V(p) \subseteq W & \forall p \in PROP \\ V(i) = \{w\} \subseteq W & \forall i \in NOM. \end{array}$$

Como vemos, la diferencia entre las fórmulas atómicas que veníamos usando hasta el momento (proposiciones) y los nominales, es que para un nominal i, $V(i)$ es un conjunto unitario. La valuación fuerza que cada nominal valga en exactamente un punto del modelo.

Definición 2.3 (Interpretación de una fórmula de $\mathcal{H}(@)$)**.** Dado el lenguaje híbrido definido sobre *PROP*, *NOM*, y *REL*, y una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R\in REL}, V\rangle$, $w \in W$, la relación de satisfacibilidad $\models$ se define recursivamente como:

$$\begin{array}{lll} \mathcal{M}, w \models p & \text{sii} & w \in V(p),\ p \in PROP \\ \mathcal{M}, w \models i & \text{sii} & w \in V(i),\ i \in NOM \\ \mathcal{M}, w \models \neg\varphi & \text{sii} & \mathcal{M}, w \not\models \varphi,\ \varphi \in FBF \\ \mathcal{M}, w \models \varphi \wedge \psi & \text{sii} & \mathcal{M}, w \models \varphi \text{ y } \mathcal{M}, w \models \psi,\ \varphi, \psi \in FBF \\ \mathcal{M}, w \models \langle R\rangle\varphi & \text{sii} & \exists w' \in W \text{ tal que } (w, w') \in R^{\mathcal{M}} \text{ y } \mathcal{M}, w' \models \varphi,\ \varphi \in FBF,\ R \in REL \\ \mathcal{M}, w \models @_i\varphi & \text{sii} & \mathcal{M}, w' \models \varphi \text{ donde } V(i) = \{w'\},\ \varphi \in FBF. \end{array}$$

Es fácil ver que cualquier fórmula-@ verifica que si $M, w \models @_i\varphi$, entonces $M \models @_i\varphi$. Otra característica que vale la pena observar es que el operador @ es su propio dual: $@_i\varphi \equiv \neg @_i \neg\varphi$.

Con $\mathcal{H}(@)$ entonces podemos *saltar* por los distintos mundos nombrados del modelo, y referirnos a ellos directamente. Por ejemplo, interpretando un operador modal como *limita con*, podemos expresar “llueve en Grecia y en un país que limita con China está nublado” de la siguiente manera: $(@_{\mathsf{grecia}}\mathsf{llueve}) \wedge (@_{\mathsf{china}}\Diamond\mathsf{nublado})$, y está claro en este ejemplo que no importa si estamos enunciando la proposición en República Checa o en Argentina, la fórmula debe mantener su valor de verdad.

2.3. Un binder modal

Tanto en los trabajos de Prior como en trabajos subsiguientes, se utilizaron además de nominales, algún tipo de cuantificador. El agregado de operadores de cuantificación extendía el poder expresivo del lenguaje, pero al mismo tiempo incrementaban la complejidad de las distintas tareas de inferencia. En la búsqueda de lenguajes expresivos de menor complejidad, se investigaron binders (operadores que *ligan* variables a mundos o conjuntos de ellos) más débiles. Un operador particularmente interesante es el $\downarrow$ [Blackburn and Tzakova, 1999; Blackburn and Seligman, 1998], que nos permite nombrar los mundos por los que pasamos a medida que evaluamos, *ligándolo* a una *variable*.

Al agregar este nuevo operador a la sintaxis de la lógica $\mathcal{H}(@)$, se obtiene la lógica llamada $\mathcal{H}(@, \downarrow)$.

Definición 2.4. Sean los siguientes conjuntos contables y disjuntos: $PROP$ de símbolos de proposición, NOM de nominales, VAR de variables, REL de símbolos de relación. Llamemos también *átomos* al conjunto $AT = PROP \cup NOM \cup VAR$, y *símbolos de estado* al conjunto $SE = NOM \cup VAR$. Definimos las *fórmulas bien formadas* (FBF) de la lógica $\mathcal{H}(@, \downarrow)$ como:

$$FBF ::= a \mid \neg\varphi \mid \varphi \wedge \psi \mid \langle R\rangle\varphi \mid @_s\varphi \mid \downarrow x.\varphi$$

con $a \in AT$, $R \in REL$, $s \in SE$, $x \in VAR$ y $\varphi, \psi \in FBF$. El resto de los operadores lógicos habituales se definen de la forma usual. Como podemos observar, también se puede utilizar el operador @ en combinación con una variable, ya que en cada momento una variable puede hacer referencia a un estado a lo sumo. Las nociones de variables *libres* y *ligadas* están definidas análogamente a las de la lógica de primer orden, con $\downarrow$ como único binder.

Definición 2.5 (Interpretación de una fórmula de $\mathcal{H}(@, \downarrow)$)**.** Dado el lenguaje híbrido definido sobre $PROP$, NOM, VAR y REL, y una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R \in REL}, V\rangle$, $w \in W$. Una *asignación* g para $\mathcal{M}$ es una función $g : VAR \Rightarrow W$. Definimos $g[x/w]$ como $g[x/w](y) = g(y), \forall y \neq x$ y $g[x/w](x) = w$, $x, y \in VAR$. La relación de satisfacibilidad $\models$ la definimos entonces recursivamente como:

$$\begin{array}{lll}
\mathcal{M}, w, g \models p & \text{sii} & w \in V(p),\ p \in PROP \\
\mathcal{M}, w, g \models i & \text{sii} & w \in V(i),\ i \in NOM \\
\mathcal{M}, w, g \models x & \text{sii} & g(x) = w,\ x \in VAR \\
\mathcal{M}, w, g \models \neg\varphi & \text{sii} & \mathcal{M}, w, g \not\models \varphi,\ \varphi \in FBF \\
\mathcal{M}, w, g \models \varphi \wedge \psi & \text{sii} & \mathcal{M}, w, g \models \varphi \text{ y } \mathcal{M}, w \models \psi,\ \varphi, \psi \in FBF \\
\mathcal{M}, w, g \models \langle R\rangle\varphi & \text{sii} & \exists w' \in W \text{ tal que } (w, w') \in R^{\mathcal{M}} \text{ y } \mathcal{M}, w', g \models \varphi,\ R \in REL,\ \varphi \in FBF \\
\mathcal{M}, w, g \models @_i\varphi & \text{sii} & \mathcal{M}, w', g \models \varphi \text{ donde } V(i) = \{w'\},\ \varphi \in FBF \\
\mathcal{M}, w, g \models \downarrow x.\varphi & \text{sii} & \mathcal{M}, w, g[x/w] \models \varphi,\ \varphi \in FBF \\
\mathcal{M}, w, g \models @_x\varphi & \text{sii} & \mathcal{M}, g(x), g \models \varphi,\ \varphi \in FBF.
\end{array}$$

Cuando para todo $w \in W$ se cumpla $\mathcal{M}, w, g \models \varphi$ diremos que $\mathcal{M}, g \models \varphi$. Cuando además sea válida para toda valuación g, lo notaremos como $\mathcal{M} \models \varphi$. Podemos observar también, a partir de esta interpretación, que el operador $\downarrow$ es su propio dual: claramente, $\neg\downarrow x.\varphi \iff \downarrow x.\neg\varphi$ es universalmente válido.

Analicemos la fórmula $\varphi = \downarrow n.(\Diamond n)$: nos está diciendo que nos ligamos el estado actual a la variable n, luego nos movemos por la relación de accesibilidad a un estado, y en ese estado vale n, por lo tanto estamos diciendo que el estado inicial, desde el que partimos evaluando, está relacionado consigo mismo. Vemos entonces que φ expresa la reflexividad del estado. Si por ejemplo interpretamos a los mundos como números naturales, y a la modalidad $\Diamond$ como $\leq$, para el número n en el que evaluamos, $n \leq n$. En particular, esto debería ser así para todo número del modelo, sabemos entonces que $\forall V, \langle \mathbb{N}, \leq, V\rangle \models \downarrow n.(\Diamond n)$. De esta manera podríamos expresar también la irreflexibidad de un modelo $\mathcal{M}$, a través de la fórmula $\mathcal{M} \models \downarrow x.(\Box\neg x)$.

2.4. Algunos resultados en el área

Volviendo al tema de la complejidad computacional, como habíamos dicho en la Sección 1.4, el problema de satisfacibilidad (comúnmente denominado SAT) para K es PSPACE-complete.

Si bien $\mathcal{H}(@)$ tiene un mayor poder expresivo, incorporando una noción de igualdad, su problema de satisfacibilidad sigue siendo PSPACE-complete. La lógica temporal K_t también tiene una complejidad PSPACE-complete, pero si la enriquecemos con tan sólo un nominal, ya pasa a ser EXPTIME-complete. Como vemos, no suele ser tarea fácil predecir la complejidad de un problema de inferecia para una lógica dada. La lógica $\mathcal{H}(@, \downarrow)$ tiene un poder expresivo tan grande que su problema de validez se vuelve indecidible. Incluso sin el operador @, i.e., la lógica modal básica extendida con el operador $\downarrow$, $\mathcal{H}(\downarrow)$, es ya indecidible.

Teorema 2.1. *[Areces* et al.*, 1999; Blackburn and Seligman, 1995]*

- *El problema SAT para $H(@)$ es PSPACE-complete.*
- *El problema SAT para K_t con un nominal es EXPTIME-complete.*
- *El problema SAT para $\mathcal{H}(\downarrow)$ es indecidible.*

Aun si restringimos $\mathcal{H}(\downarrow)$ a fragmentos sin ocurrencias anidadas de $\downarrow$, el lenguaje sigue siendo indecidible [Marx, 2002]. Si restringimos la clase de modelos a estructuras lineales, entonces $\mathcal{H}(\downarrow)$ es decidible sólo si existe una única relación de accesibilidad, pero el lenguaje bimodal es ya indecidible [Schneider, 2007]. Hasta el momento casi no se conocen lógicas decidibles interesantes, que al hibridizarlas con $\downarrow$ no se vuelvan indecidibles.

Las lógicas híbridas además de agregar poder expresivo, también tuvieron efecto en el área de teoría de la prueba modal [Blackburn, 2000b]. Usando @ y nominales, es posible *internalizar* los cálculos de deducción vía etiquetas (*labelled deduction*), desarrollados para lógica modal [Blackburn, 2000a]. En el Capítulo 4 vamos a mostrar que podemos utilizar ideas similares al diseñar un cálculo de tableaux para lógicas categoriales híbridas.

Capítulo 3

Las lógicas de categorías

3.1. Motivación

Las gramáticas de categorías, gramáticas basadas en cálculo sintáctico, tienen sus orígenes en el cálculo de Adjukiewicz de 1935. Aunque no fue hasta los cincuentas que el algebrista J. Lambek introdujo el lenguaje básico de tipado de categorías (llamado CTL, Categorial Type Logics, o Lambek calculus).

Este lenguaje, que fue estudiado por distintas razones y en distintas áreas desde su nacimiento, proviene del análisis algebraico del sistema deductivo. Fueron Lambek y Szabo quienes luego miraron al sistema deductivo como un grafo con estructura adicional, o equivalentemente como una categoría con ecuaciones faltantes [Lambek, 1968; Lambek and Scott, 1986].

Podemos ver a este lenguaje como una manera de representar ecuacionalmente las categorías en el campo de la teoría de categorías algebraica, y parte del trabajo siguió por ese camino; uno de los precursores más relevantes fue Lawvere [Bell, 2005]. También puede ser vista la lógica de categorías como una *teoría de tipos*. Desde esta perspectiva, Lambek estableció su similitud con el sistema de tipos simple para el cálculo-λ [Lambek and Scott, 1986]. Con la llegada de la Lógica Lineal, se vio que este cálculo es también una lógica subestructural.

Este lenguaje es también usado en el área del Procesamiento del Lenguaje Natural, para razonar sobre distintos fenómenos lingüísticos, y describir relaciones estructurales complejas. Se lo usa como un sistema de inferencia sintáctica de categorías (entendiendo en este caso categorías como elemento lingüístico: sustantivo, verbo, etc.).

¿Pero en qué consiste este cálculo? Uno de los conceptos que CTL intenta capturar, es el de *direccionalidad*. Para esto posee un operador de composición $\bullet$ no conmutativo, lo que quiere decir que el orden de los términos afecta la composición: $(A \bullet B)$ no es lo mismo que $(B \bullet A)$. Por ejemplo, si decimos "salí a la calle y llovió", no es lo mismo que si decimos "llovió y salí a la calle". En este caso, y en general en el lenguaje natural, la composición no es conmutativa. El lenguaje posee además una implicación $\rightarrow$, pero dada la no conmutatividad de la composición, se quiere que $A \bullet (A \rightarrow B) \vdash B$, mientras que (por no ser $\bullet$ conmutativa) $(B \rightarrow A) \bullet A \nvdash B$. Entonces surge naturalmente la necesidad de una segunda implicación $\leftarrow$ que haga válido $(B \leftarrow A) \bullet A \vdash B$.

Hay distintas variaciones que se utilizan habitualmente de Lambek calculus. La versión que utilizaremos en esta tesis es la no asociativa, llamada NL. A la asociativa se la conoce como L, que es la más común.

La tripla de operadores $(\rightarrow, \bullet, \leftarrow)$ están gobernados por la propiedad llamada *residuación* [Birkhoff, 1967; Moortgat, 1996]. La propiedad de residuación se origina en el estudio de mapeos con preservación de orden [Blyth and Janowitz, 1872]. En cualquier orden parcial con una operación $\bullet$ que funcione como "producto", un *residual* (a derecha) para un elemento a con respecto a b, es el máximo c tal que $b \bullet c$ sea menor o igual a a. Cuando existe residual para todo par de elementos en la estructura, podemos definir una función $\rightarrow$ que devuelva dicho residual. Si esto ocurre, se dice que $\bullet$ y $\rightarrow$ son residuados. Si el operador $\bullet$ no es conmutativo, también se puede definir la noción de residuación a izquierda, notándola $\leftarrow$. Como ejemplo podemos tomar el conjunto de los números racionales sin el 0, y dado cualquier racional a, el residual con respecto a algún b será simplemente $\frac{a}{b}$, y las operaciones producto y división serán las funciones residuadas. Veamos formalmente la definición de residuación para pares y ternas residuadas.

Definición 3.1 (Residuación)**.** Sea $S_i = (A_i, \sqsubseteq_{A_i})$ un conjunto parcialmente ordenado. Un par de funciones (f, g) tal que $f : A_1 \to A_2$ y $g : A_2 \to A_1$ forman un *par residuado* cuando:

$$\forall x \in A_1, y \in A_2 \begin{pmatrix} fx & \sqsubseteq_{A_2} & y & sii \\ x & \sqsubseteq_{A_1} & gy & \end{pmatrix}$$

Una terna de funciones (f, g, h) tal que $f : A_1 \times A_2 \to A_3$, $g : A_1 \times A_3 \to A_2$, y $h : A_3 \times A_2 \to A_1$ forman una *terna residuada* cuando

$$\forall x \in A_1, y \in A_2, z \in A_3 \begin{pmatrix} f(x,y) & \sqsubseteq_{A_3} & z & sii \\ y & \sqsubseteq_{A_2} & g(x,z) & sii \\ x & \sqsubseteq_{A_1} & h(z,y) & \end{pmatrix}$$

Si tomamos el conjunto parcialmente ordenado en el que los elementos son las *FBF*, y la relación de orden es la de derivación, podemos analizar la residuación en el contexto de las lógicas.

En el Capítulo 1 ya vimos un par de operadores que cumplen esta propiedad: $(\langle P\rangle, [G])$ es un par residuado, ya que

$$\forall A, B \in FBF \begin{pmatrix} \langle P\rangle A & \models & B & sii \\ A & \models & [G]B & \end{pmatrix}$$

Podemos verlo como una clase de cancelación: una forma de volver al mismo lugar, ya que se cumple siempre que $p \Rightarrow [G]\langle P\rangle p$. Otro par residuado es también $(\langle F\rangle, [H])$.

Finalmente, en NL los conectivos $(\to, \bullet, \leftarrow)$ forman una terna residuada de operadores, ya que

$$\forall A, B, C \in FBF \begin{pmatrix} A \bullet B & \models & C & sii \\ A & \models & C \to B & sii \\ B & \models & A \leftarrow C & \end{pmatrix}$$

3.2. CTL como lógica modal

Se observó que CTL puede verse como una clase de lógica modal sin estructura booleana [Dosen, 1992]. No tener estructura booleana quiere decir que no se pueden expresar (al menos directamente) los conectivos clásicos de la lógica proposicional.

Los operadores modales de esta lógica ($\bullet$, $\leftarrow$, y $\to$) son binarios. Cuando notamos una fórmula $\Diamond\alpha$ con un operador modal $\Diamond$, la relación de accesibilidad asociada a $\Diamond$ es una relación binaria. En el caso de que los operadores modales sean binarios como α OP β, con OP operador modal, las relaciones en los modelos de Kripke asociadas a este operador serán ternarias. En general, cuando los operadores modales son *n-arios*, en los modelos de Kripke se los asocia con relaciones *(n+1)-arias.*

Definición 3.2 (Sintaxis para CTL)**.** Sean los siguientes conjuntos contables y disjuntos: *PROP* de símbolos de proposición, *REL* de símbolos de relación. Definimos las *fórmulas bien formadas* (*FBF*) de la lógica *CTL* como:

$$FBF ::= p \mid \varphi \bullet_r \psi \mid \varphi \to_r \psi \mid \varphi \leftarrow_r \psi$$

con $p \in PROP$, $r \in REL$, y $\varphi, \psi \in FBF$.

Definición 3.3 (Modelo para CTL)**.** Un modelo CTL es una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R \in REL}, V\rangle$ donde

$$\begin{array}{rl} W & \text{es un conjunto no vacío} \\ R^{\mathcal{M}} \subseteq W \times W \times W & \text{es una relación ternaria } \forall R \in REL \\ V(p) \subseteq W & \forall p \in PROP. \end{array}$$

La semántica asociada a modelos de Kripke para esta sintaxis es la siguiente.

Definición 3.4 (Interpretación de una fórmula de CTL)**.** Dado un lenguaje modal definido sobre $PROP$ y REL, y una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R \in REL}, V \rangle$, $a \in W$. La relación de satisfacibilidad $\models$ se define recursivamente como:

$$
\begin{array}{lll}
\mathcal{M}, a \models p & \text{sii} & a \in V(p),\ p \in PROP \\
\mathcal{M}, a \models \varphi \bullet_r \psi & \text{sii} & \exists b, c \in W \text{ tal que } (a,b,c) \in r^{\mathcal{M}},\ \mathcal{M}, b \models \varphi \text{ y } \mathcal{M}, c \models \psi,\ r \in REL \\
\mathcal{M}, c \models \varphi \rightarrow_r \psi & \text{sii} & \forall a, b \in W \text{ tal que } (a,b,c) \in r^{\mathcal{M}}, \text{ si } \mathcal{M}, b \models \varphi \text{ entonces } \mathcal{M}, a \models \psi,\ r \in REL \\
\mathcal{M}, b \models \psi \leftarrow_r \varphi & \text{sii} & \forall a, c \in W \text{ tal que } (a,b,c) \in r^{\mathcal{M}}, \text{ si } \mathcal{M}, c \models \varphi \text{ entonces } \mathcal{M}, a \models \psi,\ r \in REL\ .
\end{array}
$$

Cuando para todo $w \in W$ se cumpla $\mathcal{M}, w \models \varphi$ escribiremos $\mathcal{M} \models \varphi$.

Veamos un ejemplo para entender un poco mejor el funcionamiento de ésta lógica. Tomemos el siguiente modelo

$$\mathcal{M} = \langle \{a, a', b, c\}, \{R^{\mathcal{M}}(a,b,c), R^{\mathcal{M}}(a',b,c)\}, \{V(p) = \{a\}, V(q) = \{b\}, V(r) = \{c\}\} \rangle.$$

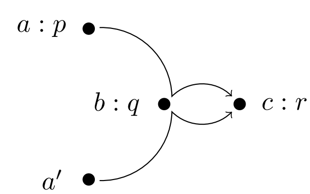

Figura 3.1: Modelo $\mathcal{M}$

Se cumple $\mathcal{M}, a \models q \bullet r$ porque existen dos mundos relacionados con a en $R^{\mathcal{M}}$, tal que el mundo donde evaluamos es el primero de la relación $R^{\mathcal{M}}(a,b,c)$, que en el segundo vale q, y en el tercero r. Como vemos, el operador $\bullet$ es similar al operador $\diamond$ en la LMB ya que se satisface con la *existencia* de mundos relacionadas con el actual que cumplan otras fórmulas argumento.

Consideremos ahora si vale $\mathcal{M}, c \models q \rightarrow p$. Debería ocurrir que para todo par de mundos en los que c sea el tercero de la relación, si en el segundo mundo de la relación vale q, entonces en el primero debe valer p. En este caso hay dos pares de mundos que cumplen la primer condición, ya que tenemos en $R^{\mathcal{M}}(a,b,c)$ y $R^{\mathcal{M}}(a',b,c)$. Para la primera de las dos relaciones esto se cumple ya que b satisface q, y a satisface p. Pero para la segunda relación esta implicación no vale ya que, como dijimos, b satisface q, pero a' no satisface p, por lo tanto $\mathcal{M}, c \not\models q \rightarrow p$. Como vemos, el operador $\rightarrow$ se comporta de alguna manera, análogamente al H en K_t, ya que no habla de existencia como $\diamond$ ó $\bullet$, sino de universalidad. El operador $\leftarrow$ lo podemos pensar de forma similar al operador $\rightarrow$, y veremos que $\mathcal{M}, b \not\models r \rightarrow p$.

En la versión asociativa L de la lógica, la diferencia puede generarse incluyendo restricciones sobre las clases de frames [Kurtonina, 1995].

Como podemos observar, los operadores $\leftarrow$ y $\rightarrow$ actúan de alguna manera "para atrás", ya que cuando analizamos el operador $\bullet$ nos ubicamos en el primer elemento de una relación, cuando vemos el $\leftarrow$ nos paramos en el segundo, y con $\rightarrow$ en el tercero, y en estos dos últimos casos predicando sobre el primer elemento de la relación. Ahora entonces podemos entender un poco mejor la analogía con K_t que hicimos en la Sección 3.1, y ver que $(\rightarrow, \bullet, \leftarrow)$ forman una terna residuada.

Una de las cosas que debemos notar es que toda fórmula de CTL es satisfacible, y lo que es más fuerte aún, que existe un modelo $\mathcal{M}_v$ que satisface toda fórmula:

$$\mathcal{M}_v = \langle \{a\}, \{R^{\mathcal{M}}(a,a,a)\}, \{V(p) = a, \forall p \in PROP\} \rangle$$

Podemos convencernos fácilmente de este hecho.

Proposición 3.1. *El modelo $\mathcal{M}_v$ en el mundo a satisface toda fórmula φ.*

Demostración. Veámoslo por inducción en la estructura de φ:

Como caso base, a satisface toda proposición por la definición de V, entonces $\mathcal{M}_v, a \models \varphi$.

Si $\varphi = \psi \bullet \psi'$, por HI a satisface ψ y ψ', y ya que también vale $R^{\mathcal{M}}(a, a, a)$, se sigue que $\mathcal{M}_v, a \models \varphi$.

Si $\varphi = \psi \rightarrow \psi'$, por HI a satisface otra vez ψ y ψ', y como $R^{\mathcal{M}}(a, a, a)$ es la única relación, $\mathcal{M}_v, a \models \varphi$. Y de igual manera para $\varphi = \psi \leftarrow \psi'$. □

Ya que existe un modelo que satisface toda fórmula, el problema de SAT en CTL es trivial, y esto es debido a que no existe negación en esta lógica. El problema realmente interesante en este lenguaje, y el que se va a estudiar en esta tesis, es el problema de entailment.

Veamos un ejemplo de entailment. Dados p y q, $p \bullet (p \rightarrow q) \models q$. Vayamos analizando su significado. Empezamos evaluando en un punto del modelo, llamémoslo w, tal que

$$\mathcal{M}, w \models p \bullet (p \rightarrow q)$$

esto implica que existen w' y w'' en W, tales que $R^{\mathcal{M}}(w, w', w'')$, y que

$$\begin{array}{lcl} \mathcal{M}, w' & \models & p \\ \mathcal{M}, w'' & \models & p \rightarrow q. \end{array}$$

Por estos dos y usando que $R^{\mathcal{M}}(w, w', w'')$, se infiere aplicando la definición del $\rightarrow$ que

$$\mathcal{M}, w \models q.$$

Pero esto es justamente lo que queríamos ver: partiendo de un modelo y mundo arbitrarios en el que vale $p \bullet (p \rightarrow q)$, llegamos a que vale q, por lo que $p \bullet (p \rightarrow q) \models q$.

En tanto complejidad computacional, se sabe que el problema de entailment de la lógica NL no sólo es decidible, sino también que su complejidad es PTIME como se mostró en [de Groote, 1999]. En cambio, en la versión asociativa L, es NP-complete [Pentus, 2006]. En el Capítulo 5 estudiaremos la relación entre NL extendido con operadores híbridos y lógica proposicional.

Capítulo 4

Hibridizando CTL

En este capítulo comenzaremos por definir la lógica hCTL($@, \downarrow$), la hibridización de CTL. Para ello le agregaremos nominales, y los operadores $@$ y $\downarrow$. Luego presentaremos un sistema de tableaux para esta lógica, que mostraremos completo. Veremos además que podemos utilizar este tableaux para resolver el problema de entailment de las lógicas hCTL($@, \downarrow$), hCTL($\downarrow$), hCTL($@$), y CTL.

4.1. La lógica hCTL($@, \downarrow$)

Dadas las introducciones de $\mathcal{H}(@, \downarrow)$ y CTL que dimos en los Capítulos 2 y 3 respectivamente, la presentación de hCTL($@, \downarrow$) es simple: nos basta con poner todos los ingredientes juntos. Definamos entonces la lógica hCTL($@, \downarrow$), comenzando por su sintaxis.

Definición 4.1. Sean los siguientes conjuntos contables y disjuntos: $PROP$ de símbolos de proposición, NOM de nominales, VAR de variables, REL de símbolos de relación, y llamemos *átomos* al conjunto $AT = PROP \cup NOM \cup VAR$, y *símbolos de estado* al conjunto $SE = NOM \cup VAR$. Definimos las *fórmulas bien formadas* (FBF) de la lógica hCTL($@, \downarrow$) como:

$$FBF ::= a \mid \varphi \bullet_r \psi \mid \varphi \rightarrow_r \psi \mid \varphi \leftarrow_r \psi \mid @_s\varphi \mid \downarrow x.\varphi$$

con $a \in AT$, $r \in REL$, $s \in SE$, $i \in NOM$, $x \in VAR$ y $\varphi, \psi \in FBF$.

Definición 4.2 (Modelo para hCTL)**.** Un modelo para la lógica hCTL es una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R \in REL}, V \rangle$, en donde[1]

$$\begin{array}{rl} W & \text{es un conjunto no vacío} \\ R^{\mathcal{M}} \subseteq W \times W \times W & \text{es una relación ternaria } \forall R \in REL \\ V(p) \subseteq W & \forall p \in PROP \\ V(i) = \{w\} \subseteq W & \forall i \in NOM. \end{array}$$

Notemos que estos modelos extienden a los de CTL con una interpretación para nominales. Para la definición semántica basta con tomar las cláusulas correspondientes a los diferentes operadores que dimos anteriormente.

Definición 4.3 (Interpretación de una fórmula de hCTL($@, \downarrow$))**.** Dados un lenguaje definido sobre NOM, $PROP$, VAR y REL, una estructura $\mathcal{M} = \langle W, (R^{\mathcal{M}})_{R \in REL}, V \rangle$, $a \in W$, y una asignación g, la relación de satisfacibilidad $\models$ está dada por:

[1] De aquí en adelante, confundiremos intencionalmente R y $R^{\mathcal{M}}$ por comodidad de notación, ya que no puede traer ambigüedades

$$
\begin{array}{lll}
\mathcal{M}, a, g \models p & \text{sii} & a \in V(p),\ p \in PROP \\
\mathcal{M}, a, g \models i & \text{sii} & a \in V(i),\ i \in NOM \\
\mathcal{M}, a, g \models x & \text{sii} & g(x) = a,\ x \in VAR \\
\mathcal{M}, a, g \models \varphi \bullet_r \psi & \text{sii} & \exists b, c \in W \text{ tal que } (a, b, c) \in r, \\
 & & \mathcal{M}, b, g \models \varphi \text{ y } \mathcal{M}, c, g \models \psi,\ r \in REL \\
\mathcal{M}, c, g \models \varphi \rightarrow_r \psi & \text{sii} & \forall a, b \in W \text{ tal que } (a, b, c) \in r, \\
 & & \text{si } \mathcal{M}, b, g \models \varphi \text{ entonces } \mathcal{M}, a, g \models \psi,\ r \in REL \\
\mathcal{M}, b, g \models \psi \leftarrow_r \varphi & \text{sii} & \forall a, c \in W \text{ tal que } (a, b, c) \in r, \\
 & & \text{si } \mathcal{M}, c, g \models \varphi \text{ entonces } \mathcal{M}, a, g \models \psi,\ r \in REL \\
\mathcal{M}, a, g \models \downarrow x.\varphi & \text{sii} & \mathcal{M}, a, g[x/w] \models \varphi \\
\mathcal{M}, a, g \models @_i\varphi & \text{sii} & \mathcal{M}, b, g \models \varphi \text{ donde } V(i) = \{b\},\ i \in NOM,\ \varphi \in FBF \\
\mathcal{M}, a, g \models @_x\varphi & \text{sii} & \mathcal{M}, b, g \models \varphi \text{ donde } g(x) = b,\ x \in VAR,\ \varphi \in FBF.
\end{array}
$$

Cuando para todo $w \in W$ se cumple $\mathcal{M}, w, g \models \varphi$ decimos que $\mathcal{M}, g \models \varphi$. Si además es válida para toda valuación g, lo notamos como $\mathcal{M} \models \varphi$.

Notemos que si tomamos el fragmento de hCTL en que no aparecen nominales, ni variables, ni operadores híbridos, la lógica que queda es CTL.

Veamos algunos ejemplos de fórmulas en esta lógica. Dado que tenemos nominales y el operador @, podemos expresar la fórmula tautológica $@_i i$, y con el operador ↓ la tautología $\downarrow x.x$. La fórmula $\downarrow x.(x \bullet x)$ vale cuando el mundo actual está relacionado con sí mismo vía R (i.e., $\mathcal{M}, a, g \models \downarrow x.(x \bullet x)$ cuando $(a, a, a) \in R$).

En los próximos capítulos trabajaremos con los fragmentos hCTL(@) y hCTL(↓) de ésta lógica. En estos, utilizaremos el símbolo $\top$, que representará a las tautologías. Observemos que el hacer uso de este símbolo no agrega ningún tipo de complejidad a la lógica, ya que podemos definirlo como un macro. En el caso de la lógica hCTL(@), definiremos $\top = @_i i$, con un i que no aparezca en el resto de la fórmula. Para hCTL(↓), $\top = \downarrow x.x$, también con un x fresco.

4.2. Un tableaux para hCTL(@, ↓)

Como dijimos en la Sección 1.4, uno de los posibles métodos de inferencia es el de tableaux[2]. Un tableaux es una forma sistemática de organizar la búsqueda de un modelo. Las reglas de tableaux indican cómo descomponer la fórmula (o conjunto de ellas) dada como entrada, siguiendo su estructura sintáctica e intentando imitar la semántica de la lógica. Durante esta descomposición se exploran diferentes alternativas, dando origen a una estructura de árbol. Cada rama de este árbol corresponde a un posible modelo. Para ilustrar concretamente al método, consideremos el método de tableaux para la lógica proposicional definida sobre el conjunto adecuado de operadores $\{\Rightarrow, \perp\}$.

$$
\Rightarrow^{+} \quad \frac{\varphi \Rightarrow \psi : T}{\varphi : F | \psi : T} \qquad\qquad \Rightarrow^{-} \quad \frac{\varphi \Rightarrow \psi : F}{\varphi : T, \psi : F}
$$

Con $\varphi, \psi \in FORM$

Figura 4.1: Reglas de Tableaux para LP

[2]Usaremos ambiguamente el término *tableaux* tanto para el método, como para el conjunto de sus reglas y para los árboles de derivación en dicho método.

Se suele llamar a la parte superior de una regla *numerador*, y *denominador* a la parte inferior. Cuando generamos el árbol y nos encontramos con una fórmula que coincide con el numerador de una regla, agregamos en esa rama las fórmulas correspondientes del denominador en una o más ramas, dependiendo de la regla. Los símbolos T y F indican si a la fórmula debemos interpretarla como verdadera o falsa (un tableaux que contiene estos símbolos se dice que es *signado*).

Veamos un ejemplo para el tableaux de la Figura 4.1. Supongamos que queremos encontrar un modelo para la fórmula $\varphi = (\bot \Rightarrow p) \Rightarrow p$. El árbol correspondiente es el siguiente:

$$
\begin{array}{rccl}
 & \multicolumn{2}{c}{1.(\bot \Rightarrow p) \Rightarrow p : T} & \\
(\Rightarrow^{+}) \text{ en } 1 \prec & 2a.\ \bot \Rightarrow p : F & 2b.\ p : T & \succ (\Rightarrow^{+}) \text{ en } 1 \\
(\Rightarrow^{-}) \text{ en } 2a \prec & 3a.\ p : F & & \\
(\Rightarrow^{-}) \text{ en } 2a \prec & 3b.\ \bot : T & & \\
 & \otimes & &
\end{array}
$$

En la raíz ubicamos la fórmula para la cual queremos encontrar un modelo. A excepción de la raíz, en cada nodo del árbol figura una fórmula que surja de aplicar una regla de tableaux sobre una fórmula anterior. Al costado de cada una se encuentra el detalle de que regla fue la aplicada: por ejemplo desde la raíz surgen dos ramas, porque fue aplicada la regla $\Rightarrow^{+}$ sobre 1 y dicha regla indica que deben generarse dos ramas hijas con las fórmulas.

Definición 4.4 (Clash)**.** Una rama B generada por el Tableaux 4.1 tiene *clash*, cuando las fórmulas $\varphi : T$ y $\varphi : F$ ($\varphi \in FBF$) pertenecen a B, o cuando $\bot : T$ pertenecen a B.

Definición 4.5 (Cerrada, Abierta y Saturada)**.** Una rama B generada por un tableaux se dice que está *cerrada* cuando contiene un clash, *abierta* cuando no está cerrada, y *saturada* cuando está abierta y no pueden aplicársele reglas de tableaux, generando así fórmulas que no estén ya en dicha rama.

En el ejemplo, al final de la rama izquierda podemos observar una cruz, esto significa que en dicha rama apareció un clash y por lo tanto está cerrada. Notemos que si una rama está cerrada, no podría haber modelo que la satisfaga. Por otro lado, la rama de la derecha está saturada, ya que no pueden aplicarse más reglas que aporten fórmulas nuevas. Cuando el sistema es correcto y completo, una rama saturada induce un modelo para la misma, y el conjunto de fórmulas Γ de la rama, debe satisfacerse en el modelo. La rama de la derecha esta implicando que cualquier modelo en que la proposición p sea verdadera, es modelo de la fórmula φ. Cuando la fórmula no sea satisfacible, todas las ramas del árbol deberán cerrarse.

Una aplicación bastante directa para la que podemos utilizar este sistema es la siguiente: si quisiéramos ver que determinada fórmula φ es una tautología, bastaría con ver que al iniciar un tableaux con $\varphi : F$, todas las ramas se cierren.

Podemos pensar que un modelo de la lógica proposicional es un modelo de Kripke con sólo un mundo y una relación de accesibilidad vacía. Por este motivo, no necesitamos identificar en qué mundo estamos en dicho tableaux. Cuando nos movemos a las lógicas modales con modelos de Kripke arbitrarios, necesitamos por un lado describir qué pasa en cada mundo, y cómo se relacionan esos mundos. Para describir lo que sucede en cada mundo, se usa un tableaux proposicional por cada mundo, y para relacionar los mundos, se utilizan auxiliares metalógicos llamados *etiquetas*. Con las etiquetas podemos marcar cada una de las fórmulas de la derivación de la siguiente manera: si l es una etiqueta y φ una FBF, una fórmula etiquetada será $l : \varphi$. Interpretaremos que en un mundo representado por la etiqueta l, se satisface la fórmula φ. Un tableaux en que las fórmulas aparecen etiquetadas se llama *tableaux etiquetado*. Entonces para un tableaux de LMB, bastará con identificar cada uno de estos tableaux proposicionales con una etiqueta, y luego relacionarlos. El operador @, los nominales, y los operadores modales existenciales, nos dan exactamente los medios para expresar esto sin recurrir a etiquetas metalógicas. Cualquier fórmula etiquetada la podemos expresar como una fórmula-@ [Blackburn, 2000a].

Veamos un ejemplo concreto de una de las reglas que vamos a utilizar en el tableaux signado para hCTL(@, ↓):

$$\bullet^+ \qquad \frac{@_a(\varphi \bullet \psi) : T}{@_a(b \bullet c) : T,\ @_b\varphi : T,\ @_c\psi : T}$$

con b y c nominales nuevos para la rama.

La regla $\bullet^+$ actúa de la siguiente manera: si en una rama hay una fórmula de la forma $@_a\varphi \bullet \psi : T$, se agregarán a la misma rama el conjunto de fórmulas $\{@_a(b \bullet c) : T,\ @_b\varphi : T,\ @_c\psi : T\}$. La idea consiste en que si en un modelo (que la rama representa), en el mundo que satisface el nominal a, satisface la fórmula $\varphi \bullet \psi$, entonces en el modelo deben valer el resto de las condiciones. Debe valer $@_a(b \bullet c) : T$: que haya dos mundos que satisfagan b y c relacionados con a $R(a, b, c)$, que el mundo en que vale b satisfaga φ, y en el que vale c, ψ. De haber aparecido una fórmula de la forma $@_a\varphi : F$, interpretaríamos que la fórmula φ no debe ser satisfecha en a. Como vemos, la idea es que coincida con la interpretación semántica, pero hasta que no demostremos que el sistema es correcto y completo, no son más que un conjunto de reglas puramente sintácticas.

$$\bullet^+ \qquad \frac{@_a(\varphi \bullet \psi) : T}{@_a(b \bullet c) : T,\ @_b\varphi : T,\ @_c\psi : T}\ {}^1 \qquad\qquad \bullet^- \qquad \frac{@_a(\varphi \bullet \psi) : F,\ @_a(b \bullet c) : T}{@_b\varphi : F \mid @_c\psi : F}$$

$$\to^+ \qquad \frac{@_a(\varphi \to \psi) : T,\ @_c(b \bullet a) : T}{@_b\varphi : F \mid @_c\psi : T} \qquad\qquad \to^- \qquad \frac{@_a(\varphi \to \psi) : F}{@_b\varphi : T,\ @_c\psi : F,\ @_c(b \bullet a) : T}\ {}^1$$

$$\leftarrow^+ \qquad \frac{@_a(\varphi \leftarrow \psi) : T,\ @_c(a \bullet b) : T}{@_b\psi : F \mid @_c\varphi : T} \qquad\qquad \leftarrow^- \qquad \frac{@_a(\varphi \leftarrow \psi) : F}{@_b\psi : T,\ @_c\varphi : F,\ @_c a \bullet b : T}\ {}^1$$

$$@^+ \qquad \frac{@_a @_b\varphi : T}{@_b\varphi : T} \qquad\qquad @^- \qquad \frac{@_a @_b\varphi : F}{@_b\varphi : F}$$

$$\mathrm{ID} \qquad \frac{@_a\varphi : T,\ @_a b : T}{@_b\varphi : T} \qquad\qquad \mathrm{NOM} \qquad \frac{@_a b : T}{@_b a : T}$$

$$R_2 \qquad \frac{@_a(b \bullet c) : T,\ @_b d : T}{@_a(d \bullet c) : T} \qquad\qquad R_3 \qquad \frac{@_a(b \bullet c) : T,\ @_c d : T}{@_a(b \bullet d) : T}$$

$$\downarrow^+ \qquad \frac{@_a{\downarrow}x.\varphi : T}{@_a\varphi[x/a] : T} \qquad\qquad \downarrow^- \qquad \frac{@_a{\downarrow}x.\varphi : F}{@_a\varphi[x/a] : F}$$

Con $a, b, c, d \in NOM$; $\varphi, \psi \in FORM$; y $p \in PROP$.

[1] con b y c nuevos.

Figura 4.2: Reglas de Tableaux para hCTL(@, ↓)

Definición 4.6 (Clash)**.** Una rama B generada por el tableaux de la Figura 4.2 tiene *clash*, cuando las fórmulas $\varphi : T$ y $\varphi : F$ ($\varphi \in \mathit{FBF}$) pertenecen a B, o cuando $@_a a : F$ ($a \in \mathit{NOM}$) pertenece a B.

Vale la pena observar que el lenguaje se presenta como multi-modal, aunque las reglas se dan como si hubiese tan sólo una relación. Eso es por simplicidad, ya que basta agregar reglas idénticas para cada una de las relaciones de accesibilidad que queramos utilizar.

Veamos un ejemplo de como se utiliza el tableaux, buscando un modelo para la fórmula

$$\downarrow x.((@_i i \leftarrow @_i i) \bullet \downarrow y.(@_x((@_i i) \bullet y))).$$

$$\begin{array}{ll}
1. @_a \downarrow x.((@_i i \leftarrow @_i i) \bullet \downarrow y.(@_x((@_i i) \bullet y))) : T & \\
2. @_a((@_i i \leftarrow @_i i) \bullet \downarrow y.(@_a((@_i i) \bullet y))) : T & \succ(\downarrow^+) \text{ en } 1 \\
3. @_a(b \bullet c) : T & \succ(\bullet^+) \text{ en } 2 \\
4. @_b(@_i i \leftarrow @_i i) : T & \succ(\bullet^+) \text{ en } 2 \\
5. @_c \downarrow y.(@_a((@_i i) \bullet y)) : T & \succ(\bullet^+) \text{ en } 2 \\
6. @_c @_a((@_i i) \bullet c) : T & \succ(\downarrow^+) \text{ en } 5 \\
7. @_a((@_i i) \bullet c) : T & \succ(@^+) \text{ en } 6 \\
8. @_a(d \bullet e) : T & \succ(\bullet^+) \text{ en } 7 \\
9. @_d(@_i i) : T & \succ(\bullet^+) \text{ en } 7 \\
10. @_e c : T & \succ(\bullet^+) \text{ en } 7 \\
11. @_c e : T & \succ(\text{NOM}) \text{ en } 10 \\
12. @_i i : T & \succ(@^+) \text{ en } 9
\end{array}$$

$$\begin{array}{llll}
(\leftarrow^+) \text{ en 3 y 4} \prec & 13\text{a}. @_c @_i i : F & 13\text{b}. @_a @_i i : T & \succ(\leftarrow^+) \text{ en 3 y 4} \\
(@^-) \text{ en 13a} \prec & 14\text{a}. @_i i : F & 14\text{b}. @_i i : T & \succ(@^+) \text{ en 13b} \\
 & \otimes & &
\end{array}$$

Mirando con un poco de cuidado el sistema de tableaux presentado en la Figura 4.2, es sencillo convencernos de la correctitud del mismo: si tomamos el numerador de una de las reglas y suponemos que se satisface en un modelo, el denominador debe satisfacerse también. Lo que debemos analizar con mayor cuidado es la completitud de dicho sistema.

El teorema de completitud lo demostraremos al final del presente capítulo.

Teorema 4.1 (Completitud)**.** *El sistema de tableaux para hCTL($@, \downarrow$) presentado en la Figura 4.2 es completo.*

El problema que principalmente nos va a ocupar en esta tesis, es el de entailment. En este problema, como dijimos, queremos saber si siempre que en un mundo se cumple una fórmula φ, en ese mismo mundo se cumple ψ, notándolo $\varphi \models \psi$. Este problema es el mismo al de ver si puede ocurrir que en un mismo mundo, valga φ y no valga ψ. La existencia de dicho mundo en algún modelo es equivalente a decir que el entailment $\varphi \models \psi$ no es correcto, por lo que si aseguramos que no hay un modelo que satisfaga φ y no ψ sabemos que $\varphi \models \psi$.

Usando el tableaux que definimos para hCTL($@, \downarrow$), podemos resolver el problema de entailment: si generamos un tableaux a partir de $\{@_a\varphi : T; @_a\psi : F\}$, y a éste se le cierran todas las ramas, significa por completitud del sistema que no hay un modelo que satisfaga ambas condiciones, y por lo tanto $\varphi \models \psi$.

Vamos a querer utilizar en el próximo capítulo las lógicas hCTL($@$) y hCTL($\downarrow$). Estas lógicas son los fragmentos sintácticos de hCTL($@, \downarrow$), los cuales no contienen respectivamente los operadores $\downarrow$ y $@$. Observemos que si queremos estudiar el problema de entailment para alguna de estas lógicas, o incluso para CTL, podemos valernos de este mismo tableaux por ser fragmentos de la lógica hCTL($@, \downarrow$). Para el caso de la lógica hCTL($\downarrow$), aunque no contenga el operador $@$, sirve igualmente como sistema de prueba. Aunque no pertenezca a la lógica objeto dicho operador, tal y como el valor de verdad con el cual interpretamos la fórmula (T ó F) es metalógico, podemos utilizarlo. Esto se debe a que la prueba de completitud del tableaux, consiste en dar un modelo para una rama saturada y abierta. Si tomamos dos fórmulas φ y ψ de alguna de estas lógicas, por ser fragmentos de la otra, podemos de igual manera aplicar el método de tableaux al conjunto $\{@_a\varphi : T; @_a\psi : F\}$ y ver si tiene modelo o no.

Ahora lo único que nos falta es demostrar el Teorema 4.1: vamos a ver que el sistema de tableaux presentado en la Figura 4.2 es un sistema de pruebas completo para la lógica hCTL(@, ↓). Para ver esto, alcanza con mostrar que toda rama abierta y saturada tiene un modelo. Entonces vamos a querer a partir de una rama saturada y abierta B, generar un modelo $\mathcal{M} = \langle W, R, V\rangle$ que satisfaga todas las fórmulas incluidas en la rama.

En B aparecerán distintos nominales, y sabemos que esos nominales valdrán en determinados mundos, así que podemos empezar pensando que cada nominal que ocurre en B representa a un mundo distinto. Llamemos W' a un conjunto de mundos tal que en cada uno vale exactamente un nominal de los que ocurren en B. Pero si nuestra intención es representar la rama en un modelo, también querremos que cuando aparezca una fórmula de estilo $@_ab : T$ en B, los nominales a y b valgan en el mismo mundo. Para lograr esto, vamos a colapsar el mundo en el que vale a y el mundo en el que vale b en un mismo mundo. Para lo cual, vamos a definir una relación con la que cocientar el conjunto W', y así obtendremos el conjunto de mundos W que buscamos

$$\sim = \{(a, a) \mid a \in NOM\} \cup \{(a, b) \mid @_ab : T \in B, a, b \in NOM\}$$

Para poder efectivamente tomar un cociente, es necesario que ésta sea una relación de equivalencia

Lema 4.2. *$\sim$es una relación de equivalencia*

Demostración. Veamos entonces que $\sim$ es efectivamente una relación de equivalencia:

- Reflexividad: es reflexiva porque $a \sim a$, para todo a en NOM.
- Simetría: es simétrica porque si tomamos a y b distintos que estén relacionados $a \sim b$, por definición $@_ab : T$ debe estar en B. Pero como B está saturada, en particular está clausurada por la regla NOM, por lo que también debe pertenecer $@_ba : T$ a B. Pero entonces por definición de $\sim$, $b \sim a$.
- Transitividad: es transitiva porque si existen a, b, y c tales que $a \sim b$ y $b \sim c$, $@_ab : T$ y $@_bc : T$ van a pertenecer a B. Entonces, por NOM aplicado sobre $@_ab : T$, también pertenecerá $@_ba : T$ a B. Aplicando ahora ID entre $@_bc : T$ y $@_ba : T$, obtenemos que $@_ac : T \in B$, lo que por definición nos da $a \sim c$.

□

Ya que $\sim$ es una relación de equivalencia, definimos W como la cocientación de W' por dicha relación

$$W = W'/\sim.$$

La clase de equivalencia a la que pertenece cada a en W', la notaremos $[a]$.

Definamos la valuación V: por un lado, los nominales deben valer en los mundos colapsados que los representan, sus clases de equivalencia:

$$\{[a]\} = V(a), \forall a \in W'.$$

Y para las proposiciones, definiremos que cada una valga cuando hay algún nominal a' en la clase en el cual deba valer:

$$[a] \in V(p) \text{ sii existe } a' \in [a] \text{ tal que } @_{a'}p : T \in B.$$

Para que sea coherente, debería valer cada una de estas proposiciones no sólo en algún nominal de la clase, sino en todos. Veremos con el siguiente lema que si incluimos a algún $[a]$ en $V(p)$, entonces p se satisface no sólo en uno, sino en todos los elementos de la clase.

Lema 4.3. $[a] \in V(p) \Rightarrow \forall a' \in [a], @_{a'}p : T \in B$.

Demostración. Si $[a] \in V(p)$, es porque existe un $a' \in [a]$ tal que $@_{a'}p : T$ está en B. Tomemos algún otro b de la misma clase $[a]$: por b pertenecer a la clase de a', en B debe estar $@_{a'}b : T$. Y por la regla ID sabemos que $@_b p : T \in B$. □

Como en el caso de la valuación, definiremos que tres clases de equivalencias están relacionadas en R cuando hay elementos en ellas que estén relacionados

$$R([a],[b],[c]) \iff \exists a,b,c / a \in [a], b \in [b], c \in [c] \text{ y } @_a b \bullet c : T \in B$$

Y veremos a continuación que si hay tres elementos de las clases que deban estar relacionados, todos los elementos de esa clase deberán estar relacionados también

Lema 4.4. $R([a],[b],[c]) \Rightarrow \forall a',b',c' / a' \in [a], b' \in [b], c' \in [c],\ @_{a'}b' \bullet c' : T \in B$

Demostración. Como $R([a],[b],[c])$, por definición $a \in [a], b \in [b]$ y $c \in [c]$. Tomamos a', b' y c' en las clases de equivalencia de a, b y c respectivamente. Queremos ver entonces que $@_{a'}b' \bullet c' : T \in B$. Sabemos por definición de R que $@_a b \bullet c : T \in B$. Por estar a', b' y c' en las mismas clases de equivalencia que a, b y c, tenemos que

$$\{@_a a' : T, @_b b' : T, @_c c' : T\} \subseteq B$$

entonces aplicándole ID a $@_a b \bullet c : T \in B$ tenemos que $@_{a'} b \bullet c : T \in B$, aplicando R_2 $@_{a'} b' \bullet c : T \in B$, y por R_3 $@_{a'} b' \bullet c' : T \in B$, que es lo que queríamos probar. □

Definición 4.7 (Modelo Inducido)**.** Dada B una rama abierta y saturada, definimos un modelo $\mathcal{M}$ de hCTL inducido para B, sobre los conjuntos $PROP$ de símbolos de proposición que aparecen en B, REL de símbolos de relación mencionados en B, NOM de símbolos de nominales que ocurren en B, como

$$\mathcal{M} = \langle W, R, V \rangle$$

Con

$$\begin{array}{lcl} \sim & = & \{(a,a) \mid a \in NOM\} \cup \{(a,b) \mid @_a b : T \in B, a,b \in NOM\} \\ W & = & NOM/\sim \\ & & \\ V(a) & = & \{[a]\}, \forall a \in NOM \\ V(p) & = & \{[a] \mid \exists a' \in [a] / @_{a'}p : T \in B\}, \forall p \in PROP \\ & & \\ R & = & \{R([a],[b],[c]) \mid a' \in [a], b' \in [b], c' \in [c] / @_{a'} b' \bullet c' : T \in B\}. \end{array}$$

Veremos ahora que toda fórmula de una rama abierta y saturada B, es verdadera en el modelo $\mathcal{M}$ inducido por dicha rama. Notar que como B está saturada, va a estar clausurada por cada una de las reglas del tableaux: si a una fórmula de B puede aplicársele una regla, esta regla va a haber sido aplicada y uno de los grupos de fórmulas obtenidas deberán pertenecer a la rama también.

Teorema 4.5 (Completitud)**.** *El sistema de tableaux para hCTL($@,\downarrow$) presentado en la Figura 4.2 es completo.*

Empecemos por definir un orden entre las fórmulas de una rama B.

Definición 4.8 (menor ($<$))**.** Sean $\alpha, \beta \in FBF$, definiremos que $\alpha < \beta$ cuando α es subfórmula propia de β, o cuando $\alpha = \beta[x/a]$, con $x \in VAR, a \in AT$.

Demostración. Sea B una rama abierta y saturada, y $\mathcal{M}$ el modelo inducido por B.
Queremos ver que todas las fórmulas en B signadas con T son satisfacibles en $\mathcal{M}$, y las marcadas con F no lo son.

Veamos entonces por inducción estructural sobre α, que se cumple la siguiente proposición:

$$@_a\alpha : T \in B \Rightarrow \mathcal{M},\ [a], g \models \alpha, \forall g$$
$$@_a\alpha : F \in B \Rightarrow \mathcal{M},\ [a], g \not\models \alpha, \forall g$$

Casos base

- $@_a b : T \in B$
  Si $@_a b : T$ está en la rama, el mundo de a es el mismo que el de b por definición de $\sim$ ($[a] = [b]$). Y como además sabemos que $[b] \in V(b)$, entonces también $[a] \in V(b)$, lo que es lo mismo que decir que $\mathcal{M}, [a], g \models b, \forall g$.

- $@_a b : F \in B$
  Supongamos que $\mathcal{M}, [a], g \models b$ para algún g e intentemos llegar a un absurdo.

  Sabemos que $a \neq b$ ya que sino habría clash en la rama, pero como $[a] \in V(b)$ entonces pertenecen a la misma clase de equivalencia y $[a] = [b]$. Como están en la misma clase de equivalencia, deben pertenecer a B $@_a b : T$ ó $@_b a : T$. Por NOM, $@_a b : T \in B$. Pero como sabíamos que $@_a b : F \in B$, esto generaría un clash en B.

  Este absurdo viene de suponer que $\mathcal{M}, [a], g \models b$ para algún g, por lo tanto $\mathcal{M}, [a] \not\models b, \forall g$.

- $@_a p : T \in B$
  Si ocurre $@_a p : T$ en B, por definición de V, $[a] \in V(p)$. De lo que se deduce que $\mathcal{M}, [a], g \models p, \forall g$.

- $@_a p : F \in B$
  Supongamos por el absurdo que $[a] \in V(p)$. Sabemos que $a \in [a]$. Usando el Lema 4.3, tenemos entonces que $@_a p : T \in B$, con lo que habría clash en B, cosa que no puede ocurrir.

  Absurdo que viene de suponer que $[a] \in V(p)$, por lo que podemos concluir que $[a] \notin V(p)$, y así $\mathcal{M}, [a], g \not\models p, \forall g$.

Casos inductivos

Sea α una *FBF* de la lógica hCTL(@, ↓). Supongamos como hipótesis inductiva que nuestra proposición es cierta para toda fórmula menor a α, y supongamos que α pertenece a B:

- $@_a\varphi \bullet \psi : T \in B$
  Como en B está $@_a\varphi \bullet \psi : T$, por la regla $\bullet^+$ también lo están las fórmulas $@_a b \bullet c : T$, $@_b\varphi : T$, y $@_c\psi : T$.

  Dado que $@_b\varphi : T$ y $@_c\psi : T \in B$, aplicamos la hipótesis inductiva en ambos y obtenemos que $\mathcal{M}, [b], g \models \varphi$, y que $\mathcal{M}, [c], g \models \psi$. Y como además $@_a b \bullet c : T$ está en B por construcción de $R([a], [b], [c])$, y por lo tanto $\mathcal{M}, [a], g \models \varphi \bullet \psi$.

- $@_a\varphi \bullet \psi : F \in B$
  Supongamos que $\mathcal{M}, [a], g \models \varphi \bullet \psi$. De ser así, existen $[b]$ y $[c]$ en W tales que $R([a], [b], [c])$, y se cumplen $\mathcal{M}, [b], g \models \varphi$ y $\mathcal{M}, [c], g \models \psi$. Como $R([a], [b], [c])$, por construcción de R existe $a' \in [a]$ tal que en B está $@_{a'} b \bullet c : T$. Usando el Lema 4.4 tenemos que $@_a b \bullet c : T \in B$.

  Aplicando $\bullet^-$ entre $@_a b \bullet c : T$ y $@_a\varphi \bullet \psi : F$ tenemos que $@_b\varphi : F$ ó $@_c\psi : F$ estarán en B.

  Si es el caso que $@_b\varphi : F \in B$, por hipótesis inductiva $\mathcal{M}, [b], g \not\models \varphi$. Pero esto es absurdo, ya que habíamos dicho que $\mathcal{M}, [b], g \models \varphi$.

  De no ser así, $@_c\psi : F \in B$, lo que otra vez por hipótesis inductiva $\mathcal{M}, [c], g \not\models \psi$, lo cual también es absurdo ya que $\mathcal{M}, [c], g \models \psi$.

  Habíamos partido del supuesto que $\mathcal{M}, [a], g \models \varphi \bullet \psi$ y llegamos a que es absurdo, por lo tanto $\mathcal{M}, [a], g \not\models \varphi \bullet \psi$.

- $@_a\varphi \to \psi : T \in B$
  Supongamos que $\mathcal{M}, [a], g \not\models \varphi \to \psi$, entonces existen $[b]$ y $[c]$ en W tales que $R([c], [b], [a])$, $\mathcal{M}, [b], g \models \varphi$, y que $\mathcal{M}, [c], g \not\models \psi$.

  Por la definición de R sabemos que existen a', b' y c' en W, con $a' \in [a]$, $b' \in [b]$ y $c' \in [c]$ tales que $@_{c'}b' \bullet a' : T \in B$. Por el Lema 4.4 tenemos que $@_{c'}b' \bullet a : T \in B$.

  Aplicando $\to^+$ entre $@_a\varphi \to \psi : T$ y $@_{c'}b' \bullet a : T \in B$ obtenemos que $@_{b'}\varphi : F$ ó $@_{c'}\psi : T$ pertenecen a B.

  Si $@_{b'}\varphi : F \in B$, por hipótesis inductiva obtenemos $\mathcal{M}, [b], g \not\models \varphi$, lo cual es absurdo.

  Si $@_{c'}\psi : T \in B$, por hipótesis inductiva otra vez tenemos que $\mathcal{M}, [c], g \models \psi$, lo cual también es absurdo.

  Con lo que llegamos a la conclusión de que no puede ocurrir que $\mathcal{M}, [a], g \not\models \varphi \to \psi$, y por lo tanto $\mathcal{M}, [a], g \models \varphi \to \psi$.

- $@_a\varphi \to \psi : F \in B$
  Aplicando la regla $\to^-$ sobre $@_a\varphi \to \psi : F$ obtenemos que existen b y c tales que $@_b\varphi : T$, $@_c\psi : F$, $@_c b \bullet a : T$ aparecen en B.

  Por hipótesis inductiva en los dos primeros tenemos que $\mathcal{M}, [b], g \models \varphi$ y $\mathcal{M}, [c], g \not\models \psi$.

  Como $@_c b \bullet a : T \in B$ entonces por la definición de R, vale $R([c], [b], [a])$, y por lo tanto $\mathcal{M}, [a], g \not\models \varphi \to \psi$.

- Los casos para $\leftarrow$ son análogos a los de $\to$.

- $@_a@_b\varphi : T \in B$
  Si $@_a@_b\varphi : T$ está en B, entonces por $@^+$ también lo está $@_b\varphi : T$. Si le aplicamos hipótesis inductiva, obtenemos que $\mathcal{M}, [b], g \models \varphi$. Por la definición del operador @ y la construcción de V, $\mathcal{M}, [b], g \models @_b\varphi$, lo que implica que $\mathcal{M}, [a], g \models @_b\varphi$.

- $@_a@_b\varphi : F \in B$
  Si $@_a@_b\varphi : F$ se encuentra en B, por la regla $@^-$ sabemos que también tendremos $@_b\varphi : F$ en B. Si aplicamos hipótesis inductiva, sabemos que $\mathcal{M}, [b], g \not\models \varphi$, lo que como en el caso anterior implica que $\mathcal{M}, [b], g \not\models @_b\varphi$, y éso a su vez que $\mathcal{M}, [a], g \not\models @_b\varphi$.

- $@_a{\downarrow}x.\varphi : T \in B$
  Aplicando $\downarrow^+$ en $@_a{\downarrow}x.\varphi : T$ obtenemos que $@_a\varphi[x/a] : T$ está en B. Con la hipótesis inductiva llegamos a que $\mathcal{M}, [a], g \models \varphi[x/a]$, y como también por construcción sabemos que $\mathcal{M}, [a], g \models a$, obtenemos $\mathcal{M}, [a], g \models @_a\varphi[x/a]$, de lo que podemos deducir que $\mathcal{M}, [a], g \models @_a{\downarrow}x.\varphi$.

- $@_a{\downarrow}x.\varphi : F \in B$
  Similarmente al caso anterior comenzamos aplicando $\downarrow^-$ en $@_a{\downarrow}x.\varphi : F$, lo que implica que $@_a\varphi[x/a] : F$ está en B. La hipótesis inductiva nos da que $\mathcal{M}, [a], g \not\models \varphi[x/a]$, y como por construcción $\mathcal{M}, [a], g \models a$, entonces $\mathcal{M}, [a], g \not\models @_a\varphi[x/a]$, con lo que tenemos que $\mathcal{M}, [a], g \not\models @_a{\downarrow}x.\varphi$.

$\square$

Capítulo 5

Cota inferior para hCTL(@)

En los próximos capítulos, vamos a analizar la complejidad computacional del problema de entailment para diversos fragmentos de hCTL(@, ↓). Para esto vamos a utilizar la técnica de reducir un problema en otro: si sabemos que un problema tiene determinada complejidad y logramos encontrar una forma de resolverlo con otro problema, sabremos entonces que la complejidad de éste será al menos tan alta como la del problema original.

Vamos a comenzar viendo una traducción que nos servirá para comparar el problema SAT de la lógica proposicional con el de entailment de hCTL(@). Luego pasaremos por la traducción del problema SAT de $\mathcal{H}(\downarrow)$ con el de entailment de hCTL(@, ↓), para resolver por último el problema SAT de $\mathcal{H}(\downarrow)$ con el problema de entailment de hCTL(↓), y así veremos que este problema también es indecidible.

Comenzaremos comparando la lógica hCTL(@) con la lógica proposicional. Esta primera comparación, que nos dará una cota inferior para la complejidad computacional de hCTL(@), servirá particularmente para observar cómo podemos comparar el problema de entailment de una hibridización de CTL, con el problema de satisfacibilidad de una lógica con base booleana.

Comencemos observando las similitudes que tienen las reglas del tableaux de la Figura 4.1 y las reglas $\rightarrow^+$ y $\rightarrow^-$ del tableaux de la Figura 4.2. Si no tomamos en cuenta en qué mundos estamos evaluando en hCTL(@) y las relaciones entre los mismos, las reglas actúan de la misma forma. Ahora bien, si quisiéramos emular el problema SAT de la lógica proposicional con estas dos reglas, logrando resolver el problema de los mundos, aún nos faltaría alguna forma de expresar $\perp$ (o la negación), ya que hCTL(@) es una lógica en la que no puede expresarse directamente. Como el problema con que vamos a comparar la satisfacibilidad de LP, es el problema de entailment de hCTL(@), tenemos una ventaja: podemos utilizar dos fórmulas en lugar de sólo una. Es más, podemos decir que queremos comparar la satisfacibilidad de una fórmula α de LP, con el *no* cumplimiento de determinado entailment de hCTL(@). De esta forma tendremos un tipo de negación: el entailment $\varphi \models \psi$ no se cumple cuando existe un modelo que tenga un mundo en que valga φ y no valga ψ. Si como ψ ponemos una proposición, digamos f, cuando f ocurra en φ actuará como $\perp$: si se puede inferir f a partir de φ, el entailment será falso y es lo que queríamos que ocurra, lo que es análogo a $\perp$ en la satisfacibilidad en LP. Por ejemplo, tomamos $p \bullet (p \rightarrow f) \not\models f$, como del antecedente puede inferirse f, la negación del entailment no se cumple, y es una forma de generar el concepto de $\perp$, que junto con la implicación, puede actuar como implicación. El problema del que hablamos antes, de que las implicaciones ocurren en distintos mundos, podemos solucionarlas arreglando las relaciones de la forma que más nos convenga.

Como dijimos, dada una fórmula α de proposicional, queremos traducirla en un entailment $A \models B$ de hCTL(@), tal que α sea satisfacible si y sólo si $A \not\models B$. Para esto propondremos a continuación una traducción *TProp* de la fórmula α, para que pueda ser expresada en esta lógica híbrida.

Definición 5.1 (*TProp*)**.** La traducción *TProp* de una fórmula α de la lógica proposicional a una de hCTL(@) se define recursivamente como

$$\begin{array}{lcl} TProp(p) & = & @_i p \\ TProp(\bot) & = & @_i f \\ TProp(\varphi \Rightarrow \psi) & = & @_i(TProp(\varphi) \to TProp(\psi)), \end{array}$$

con $\varphi, \psi \in FBF$, $p \in PROP$, y donde $i \in NOM$, y $f \in PROP$ no ocurren en α.

Como dijimos, esta traducción es bastante transparente: estamos utilizando la similitud del $\Rightarrow$ con el $\to$, la proposición f como $\bot$, y siempre haciendo que las proposiciones se evalúen en un mismo mundo, ya que un modelo de Kripke lo podemos ver como varios modelos de lógica proposicional interrelacionados, y en este caso sólo necesitamos uno. Ahora nos falta ocuparnos de las relaciones entre los mundos para que se comporte adecuadamente el operador $\to$, y definir el entailment propiamente dicho. Usaremos el siguiente entailment:

$$@_i(i \bullet i) \bullet TProp(\alpha) \models @_i\ f.$$

Con $@_i(i \bullet i)$ estamos obligando a que se relacione el mundo i reflexivamente. Con $TProp(\alpha)$ a que ocurra la traducción de α. Con la consecuencia, estamos logrando que una proposición (que debe ser nueva) actúe como $\bot$. Demostraremos ahora nuestra primera cota inferior.

Teorema 5.1. *Dada α una fórmula bien formada de LP, se cumple lo siguiente:*

$$\alpha \textit{ es satisfacible sii } @_i(i \bullet i) \bullet TProp(\alpha) \not\models @_i\ f,$$

con $f \in PROP$ tal que no ocurre en α, e $i \in NOM$.

La demostración de este teorema la separaremos en dos partes. Para la primera parte, supondremos que la fórmula α es satisfacible. Entonces existe una valuación que la satisface. A partir de esa valuación, construiremos un modelo para hCTL que no satisfaga el entailment del Teorema. Para la segunda parte, supondremos que el entailment no se cumple, entonces sabemos que el conjunto $\{@_i(i \bullet i) \bullet TProp(\alpha) : T, @_i\ f\ : F\}$ es satisfacible y por lo tanto tiene al menos una rama de tableaux abierta y saturada. A partir de dicha rama, tomaremos su modelo inducido y generaremos una valuación que satisfaga α.

Pero antes de esto, veremos una propiedad y una definición que necesitaremos.

Propiedad 5.2. *Sea $\mathcal{M} = \langle W, R, V \rangle$ tal que el mundo $w \in W$ está relacionado en R sólo de la forma (w, w, w), entonces:*

$$\begin{array}{lcl} \mathcal{M}, w, g \models \varphi \bullet \psi & \textit{sii} & \mathcal{M}, w, g \models \varphi \textit{ y } \mathcal{M}, w, g \models \psi \\ \mathcal{M}, w, g \models \varphi \to \psi & \textit{sii} & \textit{si } \mathcal{M}, w, g \models \varphi \textit{ entonces } \mathcal{M}, w, g \models \psi \\ \mathcal{M}, w, g \models \varphi \leftarrow \psi & \textit{sii} & \textit{si } \mathcal{M}, w, g \models \psi \textit{ entonces } \mathcal{M}, w, g \models \varphi \\ \mathcal{M}, w, g \models \top \to \varphi & \textit{sii} & \mathcal{M}, w, g \models \varphi \\ \mathcal{M}, w, g \models \varphi \leftarrow \top & \textit{sii} & \mathcal{M}, w, g \models \varphi. \end{array}$$

La demostración de esta Propiedad es inmediata a partir de la definición de la semántica de hCTL(@), y de las propiedades del modelo.

Definición 5.2 (Valuación σ')**.** Veremos la función σ' como la extensión de la valuación σ para fórmulas de la lógica proposicional, definida de la siguiente manera:

$$\begin{array}{lcl} \sigma'(p) & = & \sigma(p), \forall p \in PROP \\ \sigma'(\bot) & = & 0 \\ \sigma'(\varphi \Rightarrow \psi) & = & \begin{cases} 1 & \text{cuando } \sigma'(\varphi) = 0 \text{ ó } \sigma'(\psi) = 1 \\ 0 & \text{en caso contrario} \end{cases} \end{array}$$

De ahora en adelante utilizaremos indistintamente σ tanto para σ como para σ'. Estamos ahora listos para demostrar el Teorema 5.1.

Demostración del Teorema 5.1, parte 1: α satisfacible implica $@_i(i \bullet i) \bullet TProp(\alpha) \not\models @_i\ f$.

Supongamos que α, fórmula de la lógica proposicional y definida sobre $PROP$, es satisfacible. Por ser satisfacible, existe una valuación σ que la satisface. Queremos ver que no se cumple $@_i(i \bullet i) \bullet TProp(\alpha) \models @_i\ f$. Para esto vamos a generar, a partir de σ, un modelo de hCTL(@) que satisfaga $@_i(i \bullet i) \bullet TProp(\alpha)$ pero que no satisfaga $@_i\ f$.

Definiremos el modelo $\mathcal{M} = \langle W, R, V' \rangle$ sobre $PROP' = PROP(\alpha) \cup \{f\}, NOM = \{i\}$, con

$$\begin{array}{lcl} W & = & \{w\} \\ R & = & \{(w,w,w)\} \\ V'(p) & = & \{w\} \text{ cuando } \sigma(p) = 1, p \in PROP(\alpha) \\ V'(i) & = & \{w\} \\ V'(f) & = & \{\}. \end{array}$$

Veamos ahora que $\mathcal{M}, w \models @_i(i \bullet i) \bullet TProp(\alpha)$ y que $\mathcal{M}, w \not\models @_i f$.

Dado que $V'(f) = \{\}$, $\mathcal{M}, w \not\models @_i f$. $\mathcal{M}, w \models @_i(i \bullet i) \bullet TProp(\alpha)$, por la Propiedad 5.2, siempre que $\mathcal{M}, w \models @_i(i \bullet i)$ y $\mathcal{M}, w \models TProp(\alpha)$. Como $V'(i) = \{w\}$, $\mathcal{M}, w \models @_i(i \bullet i)$ es equivalente a $\mathcal{M}, w \models i \bullet i$, que se cumple. Sólo nos quedaría ahora probar que $\mathcal{M}, w \models TProp(\alpha)$. Para esto veremos que $\mathcal{M}, w \models TProp(\alpha)$ si y sólo si σ satisface α (i.e., $\sigma(\alpha) = 1$) por inducción en el largo de α. Comencemos por los casos base:

- p
  Si $\alpha = p$, como p debe pertenecer a $PROP$, $V'(p) = w$ sii $\sigma(p) = 1$, por lo tanto $\mathcal{M}, w \models p$ cuando σ satisface p, y por $V'(i) = w$, $\mathcal{M}, w \models @_i p$.

- $\perp$
  Si $\alpha = \perp$, sabemos que no es satisfacible, entonces debemos ver que el modelo no satisface $@_i f$. Pero sabemos que esto no ocurre porque $V'(f) = \{\}$.

Veamos ahora el caso inductivo, suponiendo que la proposición vale para las sub-fórmulas φ y ψ.

- $\varphi \Rightarrow \psi$
  Sabemos que $\sigma(\varphi \Rightarrow \psi) = 1$ sii $\sigma(\varphi) = 0$ ó $\sigma(\psi) = 1$. Por hipótesis inductiva, podemos decir que sucede si y sólo si

  $$\mathcal{M}, w \not\models TProp(\varphi) \text{ o } \mathcal{M}, w \models TProp(\psi).$$

  Por la Propiedad 5.2, es equivalente a

  $$\mathcal{M}, w \models TProp(\varphi) \to TProp(\psi).$$

  Pero como $V'(i) = \{w\}$, sii

  $$\mathcal{M}, w \models @_i(TProp(\varphi) \to TProp(\psi)).$$

  Y como además $TProp(\alpha) = @_i(TProp(\varphi) \to TProp(\psi))$, esto es a lo que queríamos llegar.

□

Demostración del Teorema 5.1, parte 2: $@_i(i \bullet i) \bullet TProp(\alpha) \not\models @_i\ f$ implica α satisfacible.

Supongamos ahora que $@_i(i \bullet i) \bullet TProp(\alpha) \not\models @_i\ f$, con lo cual el conjunto $\Gamma = \{@_a(@_i(i \bullet i) \bullet TProp(\alpha)) : T, @_a @_i\ f\ : F\}$, con a un nominal nuevo, es consistente. Por ser consistente, existe una rama B de tableaux, abierta y saturada a partir de Γ. Sea $\mathcal{M} = \langle W, R, V \rangle$ el modelo inducido a partir de B, y llamemos w a $[i]$, el mundo en que vale i. Generaremos una valuación σ a partir de $\mathcal{M}$, que satisfaga α. La valuación σ la definiremos de la siguiente manera:

$$\sigma(p) = 1 \textit{ sii } w \in V(p).$$

Comencemos analizando la rama B que se inicia con Γ

$$\begin{array}{ll}
1.@_a(@_i(i \bullet i) \bullet \mathit{TProp}(\alpha)) : T & \\
2.@_a @_i f : F & \\
3.@_a(b \bullet c) : T & \succ(\bullet^+) \text{ en } 1 \\
4.@_b(@_i(i \bullet i)) : T & \succ(\bullet^+) \text{ en } 1 \\
5.@_c \mathit{TProp}(\alpha) : T & \succ(\bullet^+) \text{ en } 1 \\
6.@_i(i \bullet i) : T & \succ(@^+) \text{ en } 4 \\
7.\mathit{TProp}(\alpha) : T & \succ(@^+) \text{ en } 5^1 \\
8.@_i f : F & \succ(@^+) \text{ en } 2
\end{array}$$

Sabemos entonces que $@_i f : F$ y que $\mathit{TProp}(\alpha) : T$ están en la rama, a partir de lo cual veremos que $\sigma(\alpha) = 1$.

Demostraremos por inducción en la estructura de α que

$$\mathit{TProp}(\alpha) : T \in B \text{ entonces } \sigma(\alpha) = 1$$

$$\mathit{TProp}(\alpha) : F \in B \text{ entonces } \sigma(\alpha) = 0$$

- $\mathit{TProp}(p) : T$
  Si $@_i p : T$ está en B, por construcción de modelo inducido sabemos que $w \in V(p)$, y por lo tanto $\sigma(p) = 1$.

- $\mathit{TProp}(p) : F$
  Si $@_i p : F$ está en B, por construcción de modelo inducido sabemos que $w \notin V(p)$, y por lo tanto $\sigma(p) = 0$.

- $\mathit{TProp}(\bot) : T$
  $@_i f : T$ nunca podría estar en B ya que ésta era un rama abierta, y sabemos que $@_i f : F$ está en la misma.

- $\mathit{TProp}(\bot) : F$
  Es trivial, ya que por definición $\sigma(\bot) = 0$.

- $\mathit{TProp}(\varphi \Rightarrow \psi) : T$
  Veamos qué debe suceder con todo árbol que contenga $\mathit{TProp}(\varphi \Rightarrow \psi) : T$.

$$\begin{array}{c}
1.@_i(i \bullet i) : T \quad \succ(4) \text{ en árbol inicial} \\
2.@_i(\mathit{TProp}(\varphi) \rightarrow \mathit{TProp}(\psi)) : T
\end{array}$$

$$\begin{array}{llll}
(\rightarrow^+) \text{ en 1 y 2} \prec & 3a.@_i \mathit{TProp}(\varphi) : F & 3b.@_i \mathit{TProp}(\psi) : T & \succ(\rightarrow^+) \text{ en 1 y 2} \\
(@^-) \text{ en } 3a^1 \prec & 4a.\mathit{TProp}(\varphi) : F & 4b.\mathit{TProp}(\psi) : T & \succ(@^+) \text{ en } 3b^1
\end{array}$$

Deben suceder entonces $4a$ ó $4b$, pero por hipótesis inductiva en ambas fórmulas, sabemos que suceden $\sigma(\varphi) = 0$ ó $\sigma(\psi) = 1$, y por lo tanto $\sigma(\varphi \Rightarrow \psi) = 1$.

[1] Por ser $\mathit{TProp}(\alpha)$ una fórmula-@.

- $TProp(\varphi \Rightarrow \psi) : F$
  Analicemos el árbol:

$$
\begin{array}{cl}
1.@_i(TProp(\varphi) \rightarrow TProp(\psi)) : F & \\
2.@_a(b \bullet i) : T & \succ(\rightarrow^-) \text{ en } 1 \\
3.@_b TProp(\varphi) : T & \succ(\rightarrow^-) \text{ en } 1 \\
4.@_a TProp(\psi) : F & \succ(\rightarrow^-) \text{ en } 1 \\
5. TProp(\varphi) : T & \succ(@^+) \text{ en } 3 \\
6. TProp(\psi) : F & \succ(@^-) \text{ en } 4
\end{array}
$$

  Por hipótesis inductiva en 5 y 6, sabemos que $\sigma(\varphi) = 1$ y $\sigma(\psi) = 0$, y por lo tanto $\sigma(\varphi \Rightarrow \psi) = 0$.

□

Por lo tanto demostramos que podemos resolver el problema de satisfacibilidad de la lógica proposicional con el problema de entailment de hCTL(@) con una sola relación, lo que nos da una cota inferior para la complejidad de este último. De hecho, el resultado es más fuerte, ya que lo resolvimos con el fragmento de hCTL(@) que no incluye $\leftarrow$. Incluso, el resultado es el mismo si en lugar de $\rightarrow$ utilizamos $\leftarrow$, ya que sólo usamos este operador en contextos en que se comportan igual, resolviéndolo entonces con el fragmento de hCTL(@) que no incluye $\rightarrow$. Ya que estamos resolviendo SAT de LP con la negación del problema de entailment, y el problema SAT de LP es NP-complete, la complejidad computacional del problema de entailment es co-NP-hard.

Corolario 5.3. *El problema de entailment en la lógica hCTL(@) (sin $\leftarrow$ ó $\rightarrow$) con una relación es co-NP-hard.*

Capítulo 6

La lógica hCTL(@, ↓) es indecidible

En este capítulo vamos a dar una traducción similar a la que dimos en el Capítulo 5, pero que permitirá resolver el problema de SAT para $\mathcal{H}(\downarrow)$ utilizando el de entailment de hCTL$(@, \downarrow)$. Con esto mostraremos que el problema de entailment para hCTL$(@, \downarrow)$ es indecidible.

Podemos codificar los símbolos de proposición, la implicación, y $\bot$, de forma análoga a lo que hicimos en el capítulo anterior, usando una modalidad R_1^3. El problema es que ahora tenemos varios mundos sobre los que movernos. Para dar cuenta de la modalidad de $\mathcal{H}(\downarrow)$, usaremos una segunda modalidad[1] de hCTL$(@, \downarrow)$, R_2^3. Evidentemente, no son iguales las relaciones de una y otra lógica, en hCTL las relaciones son ternarias en lugar de binarias. Representaremos una relación binaria $R(w_1, w_2)$ con $R_2(w_1, w_0, w_2)$, para algún w_0 fijo. En esta nueva traducción nos moveremos por los distintos mundos del modelo con la relación R_2, por lo cual tenemos que tener un poco más de cuidado al ver dónde evaluamos las subfórmulas. Veamos como sería la traducción que proponemos.

Definición 6.1 (*THib*)**.** La traducción *THib* de una fórmula α de la lógica $\mathcal{H}(\downarrow)$ a una de hCTL$(@, \downarrow)$ la definimos recursivamente como[2]:

$$
\begin{array}{lcl}
\mathit{THib}(a) & = & a \\
\mathit{THib}(\bot) & = & @_i f \\
\mathit{THib}(\Diamond\varphi) & = & i \bullet_2 \mathit{THib}(\varphi) \\
\mathit{THib}(\downarrow x.\varphi) & = & \downarrow x.\mathit{THib}(\varphi) \\
\mathit{THib}(\varphi \Rightarrow \psi) & = & \downarrow x.(@_i(@_x \mathit{THib}(\varphi) \rightarrow @_x \mathit{THib}(\psi)))
\end{array}
$$

con $a \in \mathit{NOM} \cup \mathit{PROP} \cup \mathit{VAR}$; $x \in \mathit{VAR}$, $\varphi, \psi \in \mathit{FORM}$, y con $f \in \mathit{PROP}$ e $i \in \mathit{NOM}$ que no aparecen en α.

Conviene comparar esta traducción con la de la Definición 5.1. Podemos ver por un lado, que los átomos ahora no tienen un @ adelante, por lo que valdrán en el nodo en que estén. Vemos también que para traducir el símbolo $\bot$, estamos saltando siempre al mundo en que vale i, tal y como lo hacíamos en la otra traducción. Para la traducción de las modalidades, como dijimos, hacemos que pase por un mundo intermedio mediante otra relación R_2^3, y siga evaluando en el tercer mundo de la relación. El caso $\downarrow$ es claro, pero miremos un poco más de cerca la traducción de la implicación. Estamos recordando el mundo actual en una variable nueva, luego pasamos por el mundo en que vale i, y evaluamos la implicación en ese mundo, pero refiriéndonos al mundo del que partimos, en el que valía x. Observar que si además podemos garantizar que se satisface $@_i(i \bullet i)$, entonces con la traducción estaremos capturando apropiadamente la semántica de la implicación.

[1] Para simplificar la notación, cuando no esté explicitamente indicado de otra forma $\leftarrow$, $\bullet$ y $\rightarrow$ denotarán $\leftarrow_1$, $\bullet_1$ y $\rightarrow_1$ respectivamente.

[2] En la regla de $\mathit{THib}(\varphi \Rightarrow \psi)$, x no debe ocurrir en $\mathit{THib}(\varphi)$ ni en $\mathit{THib}(\psi)$.

Para que funcione la traducción de $\perp$ necesitamos que $@_i f$ sea falso, y por otro lado, necesitamos que el mundo en que vale i esté relacionado con si mismo. Como en el capítulo anterior, este último requerimiento lo agregamos al antecedente usando $\bullet$ para imitar una conjunción. El primero saldrá de la negación del consecuente.

Ahora estamos listos para plantear el siguiente Teorema.

Teorema 6.1. *Sea α una fórmula bien formada y sin variables libres de $\mathcal{H}(\downarrow)$,*

$$\alpha \text{ es satisfacible sii } @_i\, THib(\alpha) \bullet @_i(i \bullet i) \not\models @_i\ f$$

con $f \in PROP$ e $i \in NOM$, tales que no ocurren en α.

Ya que el problema SAT de $\mathcal{H}(\downarrow)$ es indecidible, esto significa que la negación del problema de entailment de hCTL($@, \downarrow$) con dos relaciones también lo es. Obviamente, esto implica que el problema de entailment de hCTL($@, \downarrow$) es también indecidible. Nuevamente sólo utilizamos el fragmento de hCTL($@, \downarrow$) que no contiene $\leftarrow$, y si reemplazamos $\rightarrow$ por $\leftarrow$, la codificación funciona de la misma forma, por lo que también vale para el fragmento de hCTL($@, \downarrow$) que no contiene $\rightarrow$.

Corolario 6.2. *El problema de entailment en la lógica hCTL($@, \downarrow$) (sin $\leftarrow$ ó $\rightarrow$) con dos relaciones es indecidible.*

Dedicaremos el resto del capítulo a demostrar el Teorema 6.1. La demostración seguirá la misma estructura que la del Teorema 5.1, y la separaremos en dos partes. Comenzaremos suponiendo que α es satisfacible, por lo que existe un mundo en un modelo que la satisface. Basándonos en este modelo, construiremos uno para hCTL tal que no satisfaga el entailment propuesto. Luego supondremos que no se cumple el entailment, por lo que es consistente el conjunto $\{@_i\, THib(\alpha) \bullet @_i(i \bullet i) : T, @_i\ f\ : F\}$, y por lo tanto tiene al menos una rama de tableaux abierta y saturada. A partir de esa rama, tomaremos su modelo inducido y generaremos otro modelo de $\mathcal{H}(\downarrow)$ que satisfaga α en un mundo.

Pero antes de la demostración, veremos un lema que nos será necesario en la misma.

Lema 6.3. *Sea α una fórmula sin variables libres de $\mathcal{H}(\downarrow)$, entonces $THib(\alpha)[y/b] = THib(\alpha[y/b])$, $y \in VAR$, $b \in NOM$.*

Demostración.

Probémoslo por inducción en la estructura de la fórmula α.

- $\perp$

$$\begin{array}{rcl} THib(\perp)[y/b] & = & (@_i f)[y/b] \quad = \\ & & @_i f \quad = \\ & & THib(\perp) \quad = \quad THib(\perp[y/b]). \end{array}$$

- a, $a \in NOM \cup PROP \cup VAR$

  Para esta traducción, separaremos en dos casos: cuando a e y coincidan, y cuando no.

  - $a = y$

  $$\begin{array}{rcl} THib(y)[y/b] & = & y[y/b] \quad = \\ & & b \quad = \\ & & THib(b) \quad = \quad THib(y[y/b]). \end{array}$$

  - $a \neq y$

  $$\begin{array}{rcl} THib(a)[y/b] & = & a[y/b] \quad = \\ & & a \quad = \\ & & THib(a) \quad = \quad THib(a[y/b]). \end{array}$$

- $\Diamond\varphi$

$$\begin{array}{llll} \mathit{THib}(\Diamond\varphi)[y/b] & = & (i \bullet_2 \mathit{THib}(\varphi))[y/b] & = \\ & & i \bullet_2 (\mathit{THib}(\varphi)[y/b]) & = \text{ por hipótesis inductiva} \\ & & i \bullet_2 \mathit{THib}(\varphi[y/b]) & = \\ & & \mathit{THib}(\Diamond(\varphi[y/b])) & = \mathit{THib}((\Diamond\varphi)[y/b]). \end{array}$$

- $\downarrow x.\varphi$

  Para esta traducción, separaremos nuevamente en dos casos: cuando x e y coincidan, y cuando no.

  - $x = y$

$$\begin{array}{llll} \mathit{THib}(\downarrow x.\varphi)[x/b] & = & (\downarrow x.\mathit{THib}(\varphi))[x/b] & = \\ & & \downarrow x.\mathit{THib}(\varphi) & = \\ & & \mathit{THib}(\downarrow x.\varphi) & = \mathit{THib}((\downarrow x.\varphi)[x/b]). \end{array}$$

  - $x \neq y$

$$\begin{array}{llll} \mathit{THib}(\downarrow x.\varphi)[y/b] & = & (\downarrow x.\mathit{THib}(\varphi))[y/b] & = \\ & & \downarrow x.(\mathit{THib}(\varphi)[y/b]) & = \text{ por hipótesis inductiva} \\ & & \downarrow x.(\mathit{THib}(\varphi[y/b])) & = \\ & & \mathit{THib}(\downarrow x.(\varphi[y/b])) & = \mathit{THib}((\downarrow x.\varphi)[y/b]). \end{array}$$

- $\varphi \Rightarrow \psi$

  Para este caso, como por la definición de la traducción sabemos que x es una variable fresca, x será distinta de y.

$$\begin{array}{lll} \mathit{THib}(\varphi \Rightarrow \psi)[y/b] & = & \\ (\downarrow x.(@_i(@_x \mathit{THib}(\varphi) \rightarrow @_x \mathit{THib}(\psi))))[y/b] & = & \\ \downarrow x.(@_i(@_x(\mathit{THib}(\varphi)[y/b]) \rightarrow @_x(\mathit{THib}(\psi)[y/b]))) & = & \text{por hipótesis inductiva} \\ \downarrow x.(@_i(@_x \mathit{THib}(\varphi[y/b]) \rightarrow @_x \mathit{THib}(\psi[y/b]))) & = & \\ \mathit{THib}((\varphi[y/b] \Rightarrow \psi[y/b])) & = & \mathit{THib}((\varphi \Rightarrow \psi)[y/b]). \end{array}$$

□

Demostración del Teorema 6.1, parte 1: α satisfacible implica $@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i) \not\models @_i\ f$.

Supongamos que α, fórmula sin variables libres de $\mathcal{H}(\downarrow)$, es satisfacible. Entonces existe un modelo $\mathcal{M}$ y un mundo w_0, tales que $\mathcal{M}, w_0, g \models \alpha$, $\forall g$. A partir de este modelo, construiremos un modelo $\mathcal{M}'$ de hCTL(@, ↓) con un mundo w_0 tal que $\mathcal{M}', w_0, g \models @_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i)$, pero $\mathcal{M}', w_0, g \not\models f$.

Sea $\mathcal{M} = \langle W, R^2, V\rangle$ un modelo de $H(\downarrow)$ definido sobre los conjuntos de proposiciones $PROP$, y nominales NOM tal que $\mathcal{M}, w_0, g \models \alpha$. Sea $\mathcal{M}' = \langle W, \{R_1^3, R_2^3\}, V'\rangle$, modelo de hCTL(@, ↓) sobre las proposiciones $PROP' = PROP \cup \{f\}$, y nominales $NOM' = NOM \cup \{i\}$ tal que:

$$\begin{array}{lll} R_1^3 & = & \{(w_0, w_0, w_0)\} \\ R_2^3 & = & \{(w_j, w_0, w_k) \mid (w_j, w_k) \in R^2\} \\ V'(a) & = & V(a), \forall a \in PROP \cup NOM \\ V'(i) & = & \{w_0\} \\ V'(f) & = & \{\}. \end{array}$$

Queremos demostrar que $\mathcal{M}', w_0, g \models \mathit{THib}(\alpha) \bullet @_i(i \bullet i)$, y que $\mathcal{M}', w_0, g \not\models f$. Por un lado sabemos que $\mathcal{M}', w_0, g \not\models f$ porque definimos que $V'(f) = \{\}$. Por la Propiedad 5.2 sabemos que $\mathcal{M}', w_0, g \models @_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i)$ sii $\mathcal{M}', w_0, g \models @_i \mathit{THib}(\alpha)$ y $\mathcal{M}', w_0, g \models @_i(i \bullet i)$. Lo segundo sabemos que se cumple porque $V'(i) = \{w_0\}$ y $R_1^3(w_0, w_0, w_0)$. Ahora, sabiendo que $\mathcal{M}, w_0, g \models \alpha$, nos faltaría probar que $\mathcal{M}', w_0, g \models \mathit{THib}(\alpha)$. Para esto demostraremos la siguiente propiedad:

$$\mathcal{M}, w, g \models \varphi \text{ sii } \mathcal{M}', w, g \models \mathit{THib}(\varphi).$$

$\forall w \in W$, $\forall g$, y pudiendo φ contener variables libres.

Si lo demostramos, tendremos lo que necesitamos. Notar que sólo la implicación a derecha es la que requerimos, pero para que la demostración se pueda llevar a cabo, necesitaremos también la otra implicación.

Lo demostraremos por inducción en la estructura de φ; comencemos por los casos base.

- $\perp$
  - $\mathcal{M}, w, g \models \perp$
    Sabemos que esto no podría ocurrir.
  - $\mathcal{M}, w, g \not\models \perp$
    El modelo $\mathcal{M}'$ no satisface $@_i f$ ya que definimos $V'(f) = \{\}$.
- $a, a \in \mathit{PROP} \cup \mathit{NOM}$
  Se satisface $\mathcal{M}, w, g \models a$ sii $w \in V(a)$, y por construcción $w \in V'(a)$, lo que es equivalente a $\mathcal{M}', w, g \models a$, o sea, $\mathcal{M}', w, g \models \mathit{THib}(a)$.
- $x, x \in \mathit{VAR}$
  Ocurre que $\mathcal{M}, w, g \models x$ sii $g(x) = w$, sii $\mathcal{M}', w, g \models x$ que ocurre sii $\mathcal{M}', w, g \models \mathit{THib}(x)$.

Veamos ahora los casos inductivos.

- $\downarrow x.\varphi$, $x \in \mathit{VAR}$, $\varphi \in \mathit{FORM}$
  Ocurre que $\mathcal{M}, w, g \models \downarrow x.\varphi$, si y sólo si $\mathcal{M}, w, g[x/w] \models \varphi$. Aplicando la hipótesis inductiva, tenemos que es equivalente a $\mathcal{M}', w, g[x/w] \models \mathit{THib}(\varphi)$. Por la definición de $\downarrow$, esto es equivalente a $\mathcal{M}', w, g \models \downarrow x.\mathit{THib}(\varphi)$, que a su vez es igual a $\mathcal{M}', w, g \models \mathit{THib}(\downarrow x.\varphi)$.
- $\Diamond\varphi$
  - $\mathcal{M}, w, g \models \Diamond\varphi$
    Como $\mathcal{M}, w, g \models \Diamond\varphi$, existe un w' en W tal que $R^2(w, w')$ y $\mathcal{M}, w', g \models \varphi$. Por hipótesis inductiva, sabemos que $\mathcal{M}', w', g \models \mathit{THib}(\varphi)$. Y como $R^2(w, w')$, entonces por construcción sabemos que $R^3_2(w, w_0, w')$, sabiendo que $V'(i) = \{w_0\}$, tenemos que $\mathcal{M}', w, g \models i \bullet_2 .\mathit{THib}(\varphi)$.
  - $\mathcal{M}, w, g \not\models \Diamond\varphi$
    Queremos ver que $\mathcal{M}', w, g \not\models \mathit{THib}(\Diamond\varphi)$, o sea, $\mathcal{M}', w, g \not\models i \bullet_2 \mathit{THib}(\varphi)$. Definimos el conjunto A con todos los mundos que están relacionados con el actual en R^2: $A = \{w' \mid wR^2w'\}$. Como $\mathcal{M}, w, g \not\models \Diamond\varphi$, entonces $\forall w' \in A$, $\mathcal{M}, w', g \not\models \varphi$. Entonces, por hipótesis inductiva, $\mathcal{M}', w', g \not\models \mathit{THib}(\varphi)$.

    Supongamos primero que no hay elementos en A, i.e., $\#A = 0$. Por construcción no existe un w' tal que $(w, w_0, w') \in R^3_2$. Entonces no hay α y β tales que $\mathcal{M}', w, g \models \alpha \bullet_2 \beta$, y en particular $\mathcal{M}', w, g \not\models i \bullet_2 \mathit{THib}(\varphi)$.

    Supongamos entonces ahora que $\#A > 0$, y tomemos algún $w' \in A$. Por construcción, $(w, w_0, w') \in R^3_2$. Como $w' \in A$, $\mathcal{M}', w', g \not\models \mathit{THib}(\varphi)$. Esto ocurre para todo w' tal que $R^3_2(w, w_0, w')$, por lo que $\mathcal{M}', w, g \not\models i \bullet_2 \mathit{THib}(\varphi)$.
- $\varphi \Rightarrow \psi$

  Queremos llegar a que $\mathcal{M}, w, g \models \varphi \Rightarrow \psi$ sii $\mathcal{M}', w, g \models \downarrow x.(@_i(@_x \mathit{THib}(\varphi) \rightarrow @_x \mathit{THib}(\psi)))$. Esto sucede si $\mathcal{M}', w, g[x/w] \models @_i(@_x \mathit{THib}(\varphi) \rightarrow @_x \mathit{THib}(\psi))$. Dado que sabemos que $V'(i) = \{w_0\}$, esto es equivalente a $\mathcal{M}', w_0, g[x/w] \models @_x \mathit{THib}(\varphi) \rightarrow @_x \mathit{THib}(\psi)$.

  Como por definición w_0 está relacionado en R^3_1 sólo de la forma $R^3_1(w_0, w_0, w_0)$, podemos concluir, usando la Propiedad 5.2, que $\mathcal{M}', w_0, g[x/w] \not\models @_x \mathit{THib}(\varphi)$ o bien que $\mathcal{M}', w_0, g[x/w] \models @_x \mathit{THib}(\psi)$.

Dado que $g[x/w](x) = w$, eso a su vez es equivalente a que suceda $\mathcal{M}', w, g[x/w] \not\models \mathit{THib}(\varphi)$ o suceda $\mathcal{M}', w, g[x/w] \models \mathit{THib}(\psi)$.

Ya que no puede ocurrir x en las fórmulas φ y ψ por definición de *THib*, es igual a que ver que ocurra $\mathcal{M}', w, g \not\models \mathit{THib}(\varphi)$ o que $\mathcal{M}', w, g \models \mathit{THib}(\psi)$.

Volviendo al principio, dijimos que sabemos que ocurre $\mathcal{M}, w, g \models \varphi \Rightarrow \psi$, cuando ocurre que $\mathcal{M}, w, g \not\models \varphi$ o que $\mathcal{M}, w, g \models \psi$.

Por hipótesis inductiva, sabemos o bien que $\mathcal{M}', w, g \not\models \mathit{THib}(\varphi)$ o bien $\mathcal{M}', w, g \models \mathit{THib}(\psi)$. Pero como dijimos, esto es exactamente equivalente a $\mathcal{M}', w, g \models {\downarrow}x.(@_i(@_x \mathit{THib}(\varphi) \rightarrow @_x \mathit{THib}(\psi)))$, que es a lo que queríamos llegar.

$\square$

Demostración del Teorema 6.1, parte 2: $@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i) \not\models @_i\ f$ *implica* α *satisfacible.*

Supongamos ahora que no se cumple el entailment $@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i) \models @_i\ f$. Por lo tanto el conjunto $\{@_a(@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i)) : T, @_a @_i\ f\ : F\}$ es consistente. Por ser consistente tiene al menos una rama de tableaux abierta y saturada. A partir del modelo inducido por tal rama, generaremos otro modelo, de $\mathcal{H}(\downarrow)$ que satisfaga α en algún mundo.

Sea B una rama abierta y saturada para el tableaux de la Figura 4.2, a partir de las fórmulas $\{@_a(@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i)) : T, @_a @_i\ f\ : F\}$, y sea $\mathcal{M} = \langle W, \{R_1^3, R_2^3\}, V\rangle$ el modelo de hCTL inducido a partir de B, definido sobre los conjuntos de proposiciones *PROP*, y nominales *NOM*, con $V(i) = \{w_0\}, w_0 \notin V(f)$, para $f \in \mathit{PROP}$ e $i \in \mathit{NOM}$ distinguidos.

Definimos $\mathcal{M}' = \langle W, R^2, V'\rangle$, modelo de $\mathcal{H}(\downarrow)$ sobre la misma signatura, de la siguiente manera:

$$\begin{array}{lcl} R^2 & = & \{(w, w') \mid (w, w_0, w') \in R_2^3\} \\ V'(a) & = & V(a), \forall a \in \mathit{PROP} \cup \mathit{NOM}. \end{array}$$

Nuestro objetivo es ver que $\mathcal{M}', w_0, g \models \alpha$, donde $V'(i) = \{w_0\}$.

Comencemos analizando la rama B que inicia con $\{@_a(@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i)) : T, @_a @_i\ f\ : F\}$.

$$\begin{array}{ll} 1.@_a(@_i \mathit{THib}(\alpha) \bullet @_i(i \bullet i)) : T & \\ 2.@_a @_i f : F & \\ 3.@_a(b \bullet c) : T & \succ(\bullet^+) \text{ en } 1 \\ 4.@_c(@_i(i \bullet i)) : T & \succ(\bullet^+) \text{ en } 1 \\ 5.@_b @_i \mathit{THib}(\alpha) : T & \succ(\bullet^+) \text{ en } 1 \\ 6.@_i(i \bullet i) : T & \succ(@^+) \text{ en } 4 \\ 7.@_i \mathit{THib}(\alpha) : T & \succ(@^+) \text{ en } 5 \\ 8.@_i f : F & \succ(@^+) \text{ en } 2 \end{array}$$

Sabemos entonces que $@_i \mathit{THib}(\alpha) : T$ está en B, a partir de lo cual demostraremos que se cumple $\mathcal{M}', w_0, g \models \alpha$. Para esto, deberemos probar antes dos implicaciones:

$$@_a \mathit{THib}(\varphi) : T \in B \text{ entonces } \mathcal{M}', [a], g \models \varphi$$

$$@_a \mathit{THib}(\varphi) : F \in B \text{ entonces } \mathcal{M}', [a], g \not\models \varphi$$

para todo φ fórmula sin variables libres de $\mathcal{H}(\downarrow)$, tal que i y f no son subfórmulas de φ, $\forall a \in \mathit{NOM}$, para toda $g : \mathit{VAR} \rightarrow W$.

Si probamos que es cierto, por $@_i \mathit{THib}(\alpha) : T$ pertenecer a B, sabremos que $\mathcal{M}', w_0, g \models \alpha$ que es lo que queríamos ver. Otra vez, la implicación interesante es la primera, pero la segunda es necesaria para la prueba. Lo probaremos por inducción en la estructura de φ, comenzando por los casos base:

- $@_a \mathit{THib}(\bot) : T \in B$
  Como $@_a \mathit{THib}(\bot) = @_a @_i f$, aplicando la regla $@^+$ tenemos que $@_i f : T \in B$. Habíamos dicho que $@_i f : F \in B$, pero como B estaba abierta, esto no puede ocurrir.

- $@_a\,THib(\bot) : F \in B$
  Es trivial que $\mathcal{M}', [a], g \not\models \bot$.

- $@_a\,THib(b) : T \in B$
  Sabemos por un lado que b no puede ser una variable, porque supusimos que es una fórmula sin variables libres, y la traducción *THib* no las agrega. Tenemos que $@_a b : T \in B$, y por la construcción del modelo inducido sabemos que $\mathcal{M}, [a], g \models b$, con $b \in NOM \cup PROP$, por la forma en que construimos los mundos del modelo inducido. Pero entonces, por la definición de $\mathcal{M}'$, ocurre que $\mathcal{M}', [a], g \models b$.

- $@_a\,THib(b) : F \in B$
  Como en el caso anterior, b no puede ser variable, y sabemos que $\mathcal{M}, [a], g \not\models b$, lo que implica que $\mathcal{M}', [a], g \not\models b$.

Los casos inductivos ahora, suponiendo que la propiedad se cumple para las subfórmulas.

- $@_a\,THib(\Diamond\varphi) : T \in B$

  Analicemos qué ocurre en toda rama que contenga dicha fórmula.

$$\begin{array}{ll} 1.\ @_a i \bullet_2 THib(\varphi) : T & \\ 2.\ @_a b \bullet_2 c : T & \succ(\bullet^+) \text{ en } 1 \\ 3.\ @_b i : T & \succ(\bullet^+) \text{ en } 1 \\ 4.\ @_c\,THib(\varphi) : T & \succ(\bullet^+) \text{ en } 1 \end{array}$$

  Como $@_a b \bullet_2 c : T$ está en B, sabemos que por construcción del modelo inducido, $R_2^3([a],[b],[c])$. Como en $[b]$ vale i por 3, $w_0 = [b]$, por lo tanto $R^2([a],[c])$. Como además sabemos que $@_c\,THib(\varphi) : T$ está en B, por hipótesis inductiva sabemos que $\mathcal{M}', [c], g \models \varphi$. Como $R^2([a],[c])$ y $\mathcal{M}', [c], g \models \varphi$, entonces $\mathcal{M}', [a], g \models \Diamond\varphi$, que es lo que queríamos ver.

- $@_a\,THib(\Diamond\varphi) : F \in B$
  Razonando por el absurdo, supongamos que $\mathcal{M}', [a], g \models \Diamond\varphi$. Entonces sabemos que existe un $[b]$ tal que $R^2([a],[b])$, y que $\mathcal{M}', [b], g \models \varphi$. Como $R^2([a],[b])$, por definición de $\mathcal{M}'$, $R_2^3([a], w_0, [b])$ (donde $w_0 = [i]$). Esto, por Lema 4.4, nos dice que existe b tal que $@_a i \bullet_2 b : T \in B$. Veamos entonces qué sucede en una rama abierta y saturada que contenga ambas fórmulas:

$$\begin{array}{c} 1.\ @_a i \bullet_2 THib(\varphi) : F \\ 2.\ @_a i \bullet_2 b : T \end{array}$$

$$\begin{array}{cccc} (\bullet^-) \text{ en 1 y 2} \prec & 3a.\ @_i i : F & 3b.\ @_b\,THib(\varphi) : F & \succ(\bullet^-) \text{ en 1 y 2} \\ & \otimes & & \end{array}$$

  Como B está abierta, no puede suceder que $@_i i : F$ esté en la misma, por lo que concluimos que $@_b\,THib(\varphi) : F$ lo está. Aplicando hipótesis inductiva obtenemos que $\mathcal{M}', [b], g \not\models \varphi$, pero habíamos dicho que $\mathcal{M}', [b], g \models \varphi$. Absurdo que viene de suponer que $\mathcal{M}', [a], g \models \Diamond\varphi$, por lo que $\mathcal{M}', [a], g \not\models \Diamond\varphi$.

- $@_a\,THib(\downarrow x.\varphi) : T \in B$
  Como $@_a{\downarrow}x.THib(\varphi) : T$ está en B, y B está saturada, $@_a\,THib(\varphi)[x/a] : T$ también debe estar por la regla $\downarrow^+$. Como $@_a\,THib(\varphi)[x/a] : T$ es igual a $@_a\,THib(\varphi[x/a]) : T$ por Lema 6.3, aplicando hipótesis inductiva obtenemos que $\mathcal{M}', [a], g \models \varphi[x/a]$. Pero dado que sabemos por la construcción del modelo inducido y de $\mathcal{M}'$ que $\mathcal{M}', [a], g \models a$, $\mathcal{M}', [a], g \models \varphi[x/a]$, sii $\mathcal{M}', [a], g \models \downarrow x.\varphi$, que es lo que queríamos ver.

- $@_a\,THib(\downarrow x.\varphi) : F \in B$
  Análogamente al caso anterior, si $@_a{\downarrow}x.THib(\varphi) : F$ está en B, $@_a\,THib(\varphi)[x/a] : F$ también

está por $\downarrow^-$. El Lema 6.3 dice que $@_a\,THib(\varphi)[x/a] : F$ es $@_a\,THib(\varphi[x/a]) : F$. Aplicando hipótesis inductiva obtenemos entonces que $\mathcal{M}', [a], g \not\models \varphi[x/a]$. Y dado que por construcción $\mathcal{M}', [a], g \models a$, $\mathcal{M}', [a], g \not\models \varphi[x/a]$ es verdadero sii $\mathcal{M}', [a], g \not\models \downarrow x.\varphi$, que es lo que queríamos probar.

- $@_a\,THib(\varphi \Rightarrow \psi) : T \in B$

  Analicemos qué ocurre en una rama que contenga esa fórmula.

  $$\begin{array}{c}
  1.\ @_a{\downarrow}x.(@_i(@_x\,THib(\varphi) \to @_x\,THib(\psi))) : T \\
  2.\ @_a(@_i(@_a\,THib(\varphi) \to @_a\,THib(\psi))) : T \qquad \succ({\downarrow}^+) \text{ en } 1^{[3]} \\
  3.\ @_i(@_a\,THib(\varphi) \to @_a\,THib(\psi)) : T \qquad \succ(@^+) \text{ en } 2 \\
  4.\ @_i(i \bullet i) : T \qquad \succ(6) \text{ en reglas iniciales}
  \end{array}$$

  $$\begin{array}{lll}
  (\to^+) \text{ en 3 y 4} \prec \quad 5a.\ @_i@_a\,THib(\varphi) : F & & 5b.\ @_i@_a\,THib(\psi) : T \quad \succ(\to^+) \text{ en 3 y 4} \\
  (@^-) \text{ en 5a} \prec \quad 6a.\ @_a\,THib(\varphi) : F & & 6b.\ @_a\,THib(\psi) : T \quad \succ(\to^+) \text{ en 5b}
  \end{array}$$

  O bien 6a está en la rama, o bien 6b lo está. Por hipótesis inductiva en 6a y 6b, sabemos que $\mathcal{M}', [a], g \not\models \varphi$ ó $\mathcal{M}', [a], g \models \psi$. Pero esto es equivalente a $\mathcal{M}', [a], g \models \varphi \Rightarrow \psi$, que es lo que queríamos probar.

- $@_a\,THib(\varphi \Rightarrow \psi) : F$

  Veamos una vez más qué es lo que ocurre en toda rama que contenga esta fórmula.

  $$\begin{array}{c}
  1.\ @_a{\downarrow}x.(@_i(@_x\,THib(\varphi) \to @_x\,THib(\psi))) : F \\
  2.\ @_a(@_i(@_a\,THib(\varphi) \to @_a\,THib(\psi))) : F \qquad \succ({\downarrow}^-) \text{ en } 1^{[4]} \\
  3.\ @_i(@_a\,THib(\varphi) \to @_a\,THib(\psi)) : F \qquad \succ(@^-) \text{ en } 2 \\
  4.\ @_c(b \bullet i) : T \qquad \succ(\to^-) \text{ en } 3 \\
  5.\ @_b@_a\,THib(\varphi) : T \qquad \succ(\to^-) \text{ en } 3 \\
  6.\ @_c@_a\,THib(\psi) : F \qquad \succ(\to^-) \text{ en } 3 \\
  7.\ @_a\,THib(\varphi) : T \qquad \succ(@^+) \text{ en } 5 \\
  8.\ @_a\,THib(\psi) : F \qquad \succ(@^-) \text{ en } 6
  \end{array}$$

  Por hipótesis inductiva en 7 y 8, sabemos que $\mathcal{M}', [a], g \models \varphi$ y $\mathcal{M}', [a], g \not\models \psi$. Por lo tanto, podemos decir que $\mathcal{M}', [a], g \not\models \varphi \Rightarrow \psi$.

$\square$

Queda entonces con esto probado el Teorema 6.1, y por consiguiente la indecidibilidad del problema de entailment en la lógica hCTL(@, ↓).

[3] Ya que x no ocurre en $THib(\varphi)$ ni en $THib(\psi)$, éstas no se ven afectadas por la substitución.
[4] Ya que x no ocurre en $THib(\varphi)$ ni en $THib(\psi)$, éstas no se ven afectadas por la substitución.

Capítulo 7

Indecidibilidad sin @

Como dijimos en la Sección 1.5, nuestra motivación era encontrar una hibridización decidible que incluya al operador $\downarrow$. En ese sentido, el resultado del capítulo anterior es negativo: hCTL(@, $\downarrow$) es indecidible. Sin embargo, la traducción allí utilizada se basa fuertemente en la interacción del @ con el resto de los operadores, la cual permite simular fórmulas con estructura booleana. Esto nos lleva a una pregunta natural: ¿podremos lograr este mismo efecto si no contamos con el operador @?, Vamos a ver en este capítulo que es así, que la expresividad del operador $\downarrow$ logra que esta lógica también sea indecidible. Para probar esto, vamos a dar una forma de reducir satisfacibilidad de $\mathcal{H}(\downarrow)$ a entailment de hCTL($\downarrow$).

En líneas generales, la idea de la traducción seguirá la que vimos en el Capítulo 6 para la lógica hCTL(@, $\downarrow$). Claramente, el problema ahora radica en que no contamos con el operador @. Para salvar esto, deberemos utilizar una combinación de recursos. El resultado final será una traducción bastante más compleja; es por ello que comenzaremos por motivar cada una de sus partes.

Veamos para qué usábamos el operador @. Empecemos por la traducción de $\bot$, $@_i f$. Sabemos que en i no vale f, y necesitamos una expresión que sea "globalmente falsa". Sabemos que la fórmula $i \Rightarrow \neg f$ es "globalmente verdadera", y por tanto su negación, $\neg(i \Rightarrow \neg f)$ es falsa. Ahora notemos que $\neg(i \Rightarrow \neg f)$ es equivalente a $\neg(\neg i \vee \neg f)$, que a su vez es igual a $i \wedge f$. Supongamos por un momento que tuviéramos una macro AND que permita expresar la conjunción clásica; entonces podríamos expresar esta fórmula globalmente falsa con $AND(i, f)$.

Otro caso que utilizaba @ de manera crucial, era el de la traducción del $\Rightarrow$. En ésta, valiéndonos de dicho operador, volvíamos cada vez al mundo en que valía i, porque sabíamos que estaba relacionado consigo mismo, lo que nos permitía simular la implicación con el operador $\rightarrow$. Si logramos, en cambio, que todo mundo relevante, esté relacionado consigo mismo de esta manera, entonces podremos traducir la implicación directamente con un $\rightarrow$. Pero el problema es ahora ¿cómo podemos garantizar que todos estos mundos sean reflexivos?. Simplemente pidiendo que nuestra traducción TH^* cumpla que $TH^*(\Diamond\varphi) = i \bullet_2 \downarrow x.(AND((x \bullet x), TH^*(\varphi)))$. Observar que también en este caso suponemos que contamos con una macro AND.

Nos resta ver cómo expresar una conjunción entre φ y ψ. Simplemente nos valdremos de esta fórmula:

$$(\varphi \leftarrow \top) \bullet (\top \rightarrow \psi).$$

Si $\mathcal{M}, a, g \models (\varphi \leftarrow \top) \bullet (\top \rightarrow \psi)$, entonces existen b y c tales que $R(a, b, c)$, $\mathcal{M}, b \models (\varphi \leftarrow \top)$ y $\mathcal{M}, c \models (\top \rightarrow \psi)$. Por ser $\top$ tautología, esto garantiza $\mathcal{M}, a, g \models \varphi$ y $\mathcal{M}, a, g \models \psi$, que es lo que buscamos. Utilizaremos la siguiente definición para simplificar la notación

Definición 7.1 ($\wedge$).
$$\varphi \wedge \psi := (\varphi \leftarrow \top) \bullet (\top \rightarrow \psi).$$

Propiedad 7.1. *Si el mundo a está relacionado mediante la relación R sólo de la forma $R(a, a, a)$, entonces $\mathcal{M}, a, g \models \varphi \wedge \psi$ si y sólo si $\mathcal{M}, a, g \models \varphi$ y $\mathcal{M}, a, g \models \psi$.*

Demostración. Se demuestra directamente aplicando la Propiedad 5.2 tres veces sobre a. □

La idea general de la traducción es esencialmente la que acabamos de esbozar. Sin embargo, todavía necesitamos algunos ingredientes adicionales. Para entender su necesidad, conviene observar porqué no funcionaría una traducción naive basada en las ideas recién expuestas. Para ello, consideremos entonces la siguiente traducción TH^* similar a las traducciones anteriores[1].

$$\begin{array}{lcl} TH^*(a) & = & a \\ TH^*(\Diamond\varphi) & = & i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^*(\varphi)) \\ TH^*(\varphi \Rightarrow \psi) & = & TH^*(\varphi) \rightarrow TH^*(\psi) \\ TH^*(\downarrow x.\varphi) & = & \downarrow x.TH^*(\varphi) \\ TH^*(\bot) & = & i \wedge f \end{array}$$

con $a \in NOM \cup PROP \cup VAR$; $x \in VAR$, $\varphi, \psi \in FORM$, y con $f \in PROP$ e $i \in NOM$ que no aparecen en la fórmula.

Esperaríamos que el entailment que se utilice junto con esta traducción, sea el siguiente:

$$(i \wedge TH^*(\alpha)) \wedge (i \bullet i) \not\models f.$$

En este entailment estamos poniendo, como habíamos dicho, una f como consecuente, para poder conseguir que funcione la traducción de $\bot$. Pedimos además, usando $\wedge$, que en el mundo inicial valga el nominal i, y que esté relacionado consigo mismo. Obviamente, pedimos además que este mundo inicial satisfaga la traducción de α.

Esta traducción, tiene un problema esencial, que veremos más claramente con el siguiente ejemplo. Consideremos la fórmula:

$$\varphi := (p \Rightarrow p) \Rightarrow \bot$$

Evidentemente, φ es una contradicción. Si la codificación del problema de satisfacibilidad que propusimos fuera correcta, debería ocurrir $(i \wedge TH^*(\varphi)) \wedge (i \bullet i) \models f$. O lo que es equivalente, que en todo mundo de todo modelo que satisfaga $(i \wedge TH^*(\varphi)) \wedge (i \bullet i)$, también valga f. Veamos el modelo $\mathcal{M} = \langle W, R_1^3, V \rangle$, de la Figura 7.1, definido de la siguiente manera:

$$\begin{array}{lcl} W & = & \{w_1, w_2, w_3, w_4, w_5, w_6, w_7\} \\ R_1^3 & = & \{(w_1, w_1, w_1), (w_1, w_2, w_3), (w_4, w_5, w_1), (w_7, w_6, w_5)\} \\ V(i) & = & \{w_1\} \\ V(p) & = & \{w_5, w_6\}. \end{array}$$

Analicemos desde el mundo w_1. Este punto del modelo no satisface f, por lo que debería suceder $\mathcal{M}, w_1, g \not\models (i \wedge TH^*(\varphi)) \wedge (i \bullet i)$. Sin embargo, como veremos a continuación, éste no es el caso.

Veremos que $\mathcal{M}, w_1, g \models (i \wedge TH^*(\varphi)) \wedge (i \bullet i)$, i.e., $\mathcal{M}, w_1, g \models ((i \wedge TH^*(\varphi)) \leftarrow \top) \bullet (\top \rightarrow (i \bullet i))$. Usando la relación $R(w_1, w_2, w_3)$, basta ver que $\mathcal{M}, w_2, g \models (i \wedge TH^*(\varphi)) \leftarrow \top$ y $\mathcal{M}, w_3, g \models \top \rightarrow (i \bullet i)$. Pero esto vale cuando $\mathcal{M}, w_1, g \models i \wedge TH^*(\varphi)$ y $\mathcal{M}, w_1, g \models i \bullet i$. La segunda vale ya que $V(i) = \{w_1\}$ y $R(w_1, w_1, w_1)$. La primera, siguiendo el mismo análisis de recién, y usando también que $R(w_1, w_2, w_3)$, es equivalente a $\mathcal{M}, w_1, g \models i$ y $\mathcal{M}, w_1, g \models TH^*(\varphi)$. La primera se satisface ya que $V(i) = \{w_1\}$. Por lo que tendríamos que ver $\mathcal{M}, w_1, g \models TH^*(\varphi)$, o sea, que $\mathcal{M}, w_1, g \models TH^*((p \Rightarrow p) \Rightarrow \bot)$, sii, $\mathcal{M}, w_1, g \models TH^*((p \Rightarrow p)) \rightarrow TH^*(\bot))$. Como w_1 es el tercero de la relación en dos casos, debemos analizar ambos. Por un lado debe ocurrir que $\mathcal{M}, w_1, g \not\models TH^*(p \Rightarrow p)$ ó $\mathcal{M}, w_1, g \models TH^*(\bot)$, y por otro que $\mathcal{M}, w_5, g \not\models TH^*(p \Rightarrow p)$ ó $\mathcal{M}, w_4, g \models TH^*(\bot)$. Entonces, nos va a alcanzar con ver que $\mathcal{M}, w_1, g \not\models TH^*(p \Rightarrow p)$ y $\mathcal{M}, w_5, g \not\models TH^*(p \Rightarrow p)$, es decir, $\mathcal{M}, w_1, g \not\models p \rightarrow p$ y $\mathcal{M}, w_5, g \not\models p \rightarrow p$. Pero w_1 no satisface

[1] Para el caso de $TH^*(\Diamond\varphi)$, x no debe ocurrir en $TH^*(\varphi)$.

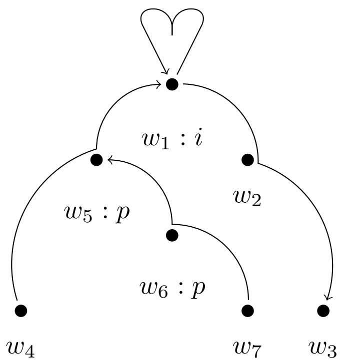

Figura 7.1: Modelo φ

$p \to p$ porque $\mathcal{M}, w_5, g \models p$, y $\mathcal{M}, w_4, g \not\models p$; mientras que w_5 no satisface $p \to p$ porque $\mathcal{M}, w_6, g \models p$, y $\mathcal{M}, w_7, g \not\models p$.

Vemos entonces que hay un problema con esta traducción, pero ¿dónde?. Podemos ver mejor lo que está pasando usando el tableaux, pero antes veamos un lema.

Lema 7.2. *Sea R una rama abierta y saturada, $a \in NOM$, y $\varphi, \psi \in FORM$. Si $@_a(\varphi \wedge \psi) : T$ pertenece a R, entonces $@_a\varphi : T$ y $@_a\psi : T$ también pertenecen a R.*

Demostración.

1. $@_a(\varphi \leftarrow \top) \bullet (\top \to \psi) : T$

2. $@_a(b \bullet c) : T$ $\quad \succ(\bullet^+)$ en 1

3. $@_b(\varphi \leftarrow \top) : T$ $\quad \succ(\bullet^+)$ en 1

4. $@_c(\top \to \psi) : T$ $\quad \succ(\bullet^+)$ en 1

$(\leftarrow^+)$ en 3 y 2 $\prec$ 5a. $@_c\top : F$ $\otimes$ | 5b. $@_a\varphi : T$ $\succ(\leftarrow^+)$ en 3 y 2

$(\to^+)$ en 4 y 2 $\prec$ 6a. $@_b\top : F$ $\otimes$ | 6b. $@_a\psi : T$ $\succ(\to^+)$ en 4 y 2

Observemos entonces que por 5b y 6b queda probado el Lema. $\square$

Ahora sí pasaremos al tableaux que nos referíamos.

$$
\begin{array}{ll}
1.\ @_a((i \wedge TH^*(\varphi)) \wedge (i \bullet i)) : T & \\
2.\ @_a f : F & \\
3.\ @_a(i \wedge TH^*(\varphi)) : T & \succ\text{(Lema 7.2) en 1} \\
4.\ @_a(i \bullet i) : T & \succ\text{(Lema 7.2) en 1} \\
5.\ @_a i : T & \succ\text{(Lema 7.2) en 3} \\
6.\ @_a TH^*(\varphi) : T & \succ\text{(Lema 7.2) en 3} \\
7.\ @_i a : T & \succ(NOM)\text{ en 5} \\
8.\ @_a(a \bullet i) : T & \succ(R_2)\text{ en 4 y 7} \\
9.\ @_a(a \bullet a) : T & \succ(R_3)\text{ en 4 y 7} \\
10.\ @_i f : F & \succ\text{(ID) en 5 y 2} \\
11.\ @_a TH^*((p \Rightarrow p) \Rightarrow \bot) : T & \succ(expandir)\text{ en 3} \\
12.\ @_a(TH^*((p \Rightarrow p)) \rightarrow TH^*(\bot)) : T & \succ(expandir)\text{ en 11}
\end{array}
$$

$$
\begin{array}{rl|ll}
(\rightarrow^+)\text{ en 12 y 9} \prec & 13a.\ @_a TH^*((p \Rightarrow p)) : F & 13b.\ @_a TH^*(\bot) : T & \succ(\rightarrow^+)\text{ en 12 y 9} \\
(expandir)\text{ en 13a} \prec & 14a.\ @_a(p \rightarrow p) : F & 14b.\ @_a(i \wedge f) : T & \succ(expandir)\text{ en 13b} \\
(\rightarrow^-)\text{ en 13a} \prec & 15a.\ @_c(b \bullet a) : F & 15b.\ @_a f : T & \succ\text{(Lema 7.2) en 14b} \\
(\rightarrow^-)\text{ en 13a} \prec & 16a.\ @_b p : T & 16b.\ @_a i : T & \succ\text{(Lema 7.2) en 14b} \\
(\rightarrow^-)\text{ en 13a} \prec & 17a.\ @_c p : F & 17b.\ @_i f : T & \succ\text{(ID) en 15b y 16b} \\
 & & \otimes &
\end{array}
$$

Como vemos, al expandir el $\rightarrow$ de 14a, esperaríamos que se forme un clash al intentar evaluar la implicación en el mundo en que vale i, pero como vemos, las proposiciones p positiva y negativa, ocurren en mundos que no son el mismo. En particular, podemos observar que el problema radica en que $\rightarrow$, al estar en un contexto negativo (i.e., tiene polaridad F en el tableaux), funciona como un diamante (recordemos que $\neg\Diamond\varphi \equiv \Box\neg\varphi$). Por este motivo, nos gustaría decir entonces algo del estilo $(p \wedge i) \rightarrow (p \wedge i)$, Pero la subfórmula generada en ella, es de polaridad negativa, y no se lleva bien con el $\rightarrow$ de la definición de $\wedge$. En el Capítulo 6, esto no nos traía inconvenientes, ya que con @ volvíamos al mundo en que valía i, y nos quedaban en conjunción ambas partes de la implicación con la polaridad correcta. Para resolver este conflicto, deberemos por un lado, saber de antemano si vamos a evaluar una fórmula determinada (la traducción de la implicación) en un contexto positivo o negativo. Las implicaciones en contexto negativo necesitarán formulaciones que aseguren que volvamos al mundo correcto y con la polaridad intencionada. Para ésto vamos a proponer dos traducciones, una para cuando el contexto es positivo, TH^+, en la que funcionaría una traducción como la que propusimos más arriba, y otra para cuando el contexto es negativo, TH^-.

Veamos como podemos interpretar la implicación para un contexto negativo, pero primero notemos que los operadores $\leftarrow$ y $\rightarrow$ generan cambios de polaridades, por lo que los utilizaremos de distintas maneras para obtener el resultado deseado. La traducción propuesta es:

$$TH^-(\varphi \Rightarrow \psi) = \downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))).$$

Asumimos que el mundo inicial (digamos a) está relacionado consigo mismo. En ese caso $\leftarrow$ actúa como una implicación, por la Propiedad 5.2, pero como el contexto es negativo, valdrá cuando en particular en a se cumpla $(x \wedge TH^+(\varphi))$ y no valga $(TH^-(\psi) \bullet \top)$, con $g(x) = a$. Como se cumple $(x \wedge TH^+(\varphi))$ en a, se cumple $TH^+(\varphi)$ ahí mismo, que era una de las condiciones que pretendíamos. Ya que no se cumple $(TH^-(\psi)\bullet\top)$, y sabemos que a está relacionado consigo mismo, y que obviamente vale $\top$ ahí, debe no valer $TH^-(\psi)$, que es la otra condición que queríamos.

Si ahora pensamos en la traducción tentativa que dimos para el $\Diamond$, $i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^*(\varphi))$, en la misma proponíamos que al pasar a un mundo nuevo, se creara la relación de este mundo nuevo consigo mismo. Pero ahora que estamos analizando las polaridades, es interesante notar que cuando el $\Diamond$ está en un contexto negativo, se comporta como un $\Box$, digamos como ejemplo en $\Diamond\varphi \Rightarrow \bot$. En este caso, no tendría sentido traducirlo como si se tratara de un $\Diamond$, con esta forma de asegurar las relaciones reflexivas. Podríamos simplemente traducirlo como $i \bullet_2 TH^-(\varphi)$ (aunque pondremos una variación de esta fórmula).

Definición 7.2 (TH^+ y TH^-)**.** Las traducciones TH^+ y TH^- de una fórmula α de la lógica $\mathcal{H}(\downarrow)$ a una de hCTL($\downarrow$) la definimos recursivamente como:

$$
\begin{array}{lcl}
TH^+(a) & = & a \\
TH^-(a) & = & a \\
TH^+(\Diamond\varphi) & = & i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^+(\varphi))^2 \\
TH^-(\Diamond\varphi) & = & i \bullet_2 \downarrow x.TH^-(\varphi)^3,{}^4 \\
TH^+(\varphi \Rightarrow \psi) & = & TH^-(\varphi) \rightarrow TH^+(\psi) \\
TH^-(\varphi \Rightarrow \psi) & = & \downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi)))^5 \\
TH^+(\downarrow x.\varphi) & = & \downarrow x.TH^+(\varphi) \\
TH^-(\downarrow x.\varphi) & = & \downarrow x.TH^-(\varphi) \\
TH^+(\bot) & = & i \wedge f \\
TH^-(\bot) & = & i \wedge f
\end{array}
$$

con $a \in NOM \cup PROP \cup VAR$; $x \in VAR$, $\varphi, \psi \in FORM$, y con $f \in PROP$ e $i \in NOM$ que no aparecen en α.

Veamos que con ésta traducción que toma en cuenta las polaridades, el ejemplo que habíamos dado antes, se comporta como esperábamos. Pero antes observemos lo siguiente:

Observación 7.1. Si R es una rama abierta y saturada, y a un nominal, entonces $@_a\top : F$ no puede pertenecer a R. Si, perteneciese, entonces como $\top$ es una abreviación para $\downarrow x.x$, al aplicar a $@_a\downarrow x.x : F$ la regla $\downarrow^-$, también $@_a a : F$ debería pertenecer a R, pero eso generaría un clash y R era una rama abierta y saturada.

Veamos ahora al ejemplo anterior, y veamos como se comporta con la nueva traducción:

---

[2] x no debe ocurrir en $TH^+(\varphi)$.

[3] x no debe ocurrir en $TH^-(\varphi)$.

[4] Vale la pena notar que ésta traducción tiene un $\downarrow x$. "innecesario", ya que dicho x no ocurre en el resto de la fórmula, y por lo tanto es equivalente a no ponerlo. El hecho de que lo hayamos incluido en la traducción, es debido a que simplifica la demostración de un lema, el Lema A.1.

[5] x no debe ocurrir en $TH^-(\psi)$ ni en $TH^+(\varphi)$.

1. $@_a((i \wedge TH^+(\varphi)) \wedge (i \bullet i)) : T$
2. $@_a f : F$
3. $@_a(i \wedge TH^+(\varphi)) : T$ $\succ$(Lema 7.2) en 1
4. $@_a(i \bullet i) : T$ $\succ$(Lema 7.2) en 1
5. $@_a i : T$ $\succ$(Lema 7.2) en 3
6. $@_a TH^+(\varphi) : T$ $\succ$(Lema 7.2) en 3
7. $@_i a : T$ $\succ(NOM)$ en 5
8. $@_a(a \bullet i) : T$ $\succ(R_2)$ en 4 y 7
9. $@_a(a \bullet a) : T$ $\succ(R_3)$ en 4 y 7
10. $@_i f : F$ $\succ$(ID) en 5 y 2
11. $@_a TH^+((p \Rightarrow p) \Rightarrow \bot) : T$ $\succ(expandir)$ en 3
12. $@_a(TH^-((p \Rightarrow p)) \rightarrow TH^+(\bot)) : T$ $\succ(expandir)$ en 11

$(\rightarrow^+)$ en 12 y 9 $\prec$ 13a. $@_a TH^-((p \Rightarrow p)) : F$	13b. $@_a TH^+(\bot) : T$ $\succ(\rightarrow^+)$ en 12 y 9
$(expandir)$ en 13a $\prec$ 14a. $@_a{\downarrow}x.((p \bullet \top) \leftarrow (x \wedge p)) : F$	14b. $@_a(i \wedge f) : T$ $\succ(expandir)$ en 13b
$(\downarrow^-)$ en 13a $\prec$ 15a. $@_a(p \bullet \top) \leftarrow (a \wedge p) : F$	15b. $@_a f : T$ $\succ$(Lema 7.2) en 14b
$(\leftarrow^-)$ en 15a $\prec$ 16a. $@_c(a \bullet b) : T$	16b. $@_a i : T$ $\succ$(Lema 7.2) en 14b
$(\leftarrow^-)$ en 15a $\prec$ 17a. $@_b(a \wedge p) : T$	17b. $@_i f : T$ $\succ$(ID) en 15b y 16b
$(\leftarrow^-)$ en 15a $\prec$ 18. $@_c(p \bullet \top) : F$	$\otimes$

$(\bullet^-)$ en 18 y 16a $\prec$ 19a. $@_b \top : F$	19b. $@_a p : F$ $\succ(\bullet^-)$ en 18 y 16a
$\otimes$	20. $@_b a : T$ $\succ$(Lema 7.2) en 17a
	21. $@_b p : T$ $\succ$(Lema 7.2) en 9
	22. $@_a p : T$ $\succ$(ID) en 20 y 21
	$\otimes$

Podemos apreciar que esta traducción, resuelve efectivamente el problema visto anteriormente. En este caso se genera un clash por ocurrir la proposición p, tanto con polaridad positiva como negativa en a, entre 19b y 22.

Pasemos ahora a definir el entailment:

$$(i \wedge TH^+(\alpha)) \wedge (i \bullet i) \not\models f.$$

Como vemos, es similar al que mencionamos anteriormente, sólo que comenzando con la polaridad que corresponde, positiva.

Teorema 7.3. *Sea α una fórmula bien formada de $\mathcal{H}(\downarrow)$, sin variables libres,*

$$\alpha \text{ es satisfacible sii } (i \wedge TH^+(\alpha)) \wedge (i \bullet i) \not\models f,$$

siendo $f \in PROP$, $i \in NOM$, tales que no aparecen en α.

Corolario 7.4. *El problema de entailment en la lógica hCTL($\downarrow$) con dos relaciones es indecidible.*

Para demostrar este teorema, seguiremos la misma estructura que para el Teorema 6.1: una primera parte suponiendo que la fórmula α es satisfacible, y a partir de un modelo para ella, generaremos otro que no satisfaga el entailment anterior. Para la segunda parte supondremos que el entailment no se cumple, y tomaremos una rama de tableaux que lo represente. Luego, construiremos un modelo de $\mathcal{H}(\downarrow)$ que satisfaga α a partir del modelo inducido por la rama.

Demostración del Teorema 7.3, parte 1: α satisfacible implica $(i \wedge TH^+(\alpha)) \wedge (i \bullet i) \not\models f$.

Supongamos que α es satisfacible; entonces existe un modelo $\mathcal{M}$ y un mundo w_0, tales que $\forall g$, $\mathcal{M}, w_0, g \models \alpha$. A partir de este modelo, generaremos $\mathcal{M}'$, un modelo de hCTL($\downarrow$), tal que $\mathcal{M}', w_0, g \models (i \wedge TH^+(\alpha)) \wedge (i \bullet i)$, y que $\mathcal{M}', w_0, g \not\models f$.

Sea $\mathcal{M} = \langle W, R^2, V \rangle$,definimos $\mathcal{M}' = \langle W, \{R_1^3, R_2^3\}, V' \rangle$, modelo de hCTL($\downarrow$) sobre las proposiciones $PROP' = PROP \cup \{f\}$, y nominales $NOM' = NOM \cup \{i\}$ de la siguiente manera:

$$
\begin{array}{lcl}
R_1^3 & = & \{(w, w, w) \mid w \in W'\} \\
R_2^3 & = & \{(w, w_0, w') \mid (w, w') \in R^2\} \\
V'(a) & = & V(a), \forall a \in PROP \cup NOM \\
V'(i) & = & \{w_0\} \\
V'(f) & = & \{\}.
\end{array}
$$

Notemos que la Propiedad 5.2 aplica a todo el modelo $\mathcal{M}'$ con la relación R_1^3.

Por una parte, podemos ver que $\mathcal{M}', w_0, g \not\models f$. Queremos ver ahora que $\mathcal{M}', w_0, g \models (i \wedge TH^+(\alpha)) \wedge (i \bullet i)$. Como para la relación R_1^3 los mundos están relacionados sólo consigo mismos, es equivalente por la Propiedad 7.1 a que $\mathcal{M}', w_0, g \models TH^+(\alpha)$, $\mathcal{M}', w_0, g \models i$, y $\mathcal{M}', w_0, g \models i \bullet i$. Sabemos que los dos últimos se cumplen, nos faltaría ver que $\mathcal{M}', w_0, g \models TH^+(\alpha)$. Para ver esto probaremos por inducción en la estructura de α que

$$\text{si } \mathcal{M}, w, g \models \varphi \text{ entonces } \mathcal{M}', w, g \models TH^+(\varphi),$$

$$\text{si } \mathcal{M}, w, g \not\models \varphi \text{ entonces } \mathcal{M}', w, g \not\models TH^-(\varphi).$$

La segunda implicación, como en la primera parte de la demostración del Teorema 6.1, está planteada para que pueda funcionar la inducción, pero en realidad el resultado que necesitamos es el dado por la primera implicación.

- $\mathcal{M}, w, g \models \bot$
  Está claro que esto no puede ocurrir.

- $\mathcal{M}, w, g \not\models \bot$
  Supongamos por el absurdo que $\mathcal{M}', w, g \models TH^-(\bot)$. Es decir, $\mathcal{M}', w, g \models i \wedge f$ que por la Propiedad 7.1 es equivalente a ver que $\mathcal{M}', w, g \models i$ y que $\mathcal{M}', w, g \models f$. Esto último no puede ocurrir porque definimos $V'(f) = \{\}$. El absurdo viene de suponer que $\mathcal{M}', w, g \models TH^-(\bot)$, por lo que $\mathcal{M}', w, g \not\models TH^-(\bot)$.

- $\mathcal{M}, w, g \models a, a \in PROP \cup NOM$
  Si $\mathcal{M}, w, g \models a$ entonces $w \in V(a)$, y por definición sabemos $w \in V'(a)$, lo que nos dice que $\mathcal{M}', w, g \models a$ que es igual que $\mathcal{M}', w, g \models TH^+(a)$.

- $\mathcal{M}, w, g \not\models a, a \in PROP \cup NOM$
  Análogamente, si $\mathcal{M}, w, g \not\models a$ entonces $w \notin V(a)$, y por definición sabemos $w \notin V'(a)$, lo que nos dice que $\mathcal{M}', w, g \not\models a$, equivalentemente, $\mathcal{M}', w, g \not\models TH^-(a)$.

- $\mathcal{M}, w, g \models x, x \in VAR$
  $\mathcal{M}, w, g \models x$ implica que $g(x) = w$, pero entonces $\mathcal{M}', w, g \models x$ que es lo igual que $\mathcal{M}', w, g \models TH^+(x)$.

- $\mathcal{M}, w, g \not\models x, x \in VAR$
  Análogamente, $\mathcal{M}, w, g \not\models x$ implica que $g(x) \neq w$, pero entonces $\mathcal{M}', w, g \not\models x$ que es equivalente a $\mathcal{M}', w, g \not\models TH^-(x)$.

- $\mathcal{M}, w, g \models \downarrow x.\varphi$, $x \in VAR$, $\varphi \in FORM$
  Si $\mathcal{M}, w, g \models \downarrow x.\varphi$ entonces $\mathcal{M}, w, g[x/w] \models \varphi$. Aplicando la hipótesis inductiva, tenemos que $\mathcal{M}', w, g[x/w] \models TH^+(\varphi)$. Por la definición de $\downarrow$, esto sucede sii $\mathcal{M}', w, g \models \downarrow x.TH^+(\varphi)$, que a su vez es lo mismo que $\mathcal{M}', w, g \models TH^+(\downarrow x.\varphi)$.

- $\mathcal{M}, w, g \not\models \downarrow x.\varphi$, $x \in VAR$, $\varphi \in FORM$
  Nuevamente en forma análoga al caso anterior, si $\mathcal{M}, w, g \not\models \downarrow x.\varphi$ entonces $\mathcal{M}, w, g[x/w] \not\models \varphi$. Aplicando la hipótesis inductiva, tenemos que $\mathcal{M}', w, g[x/w] \not\models TH^-(\varphi)$. Por la definición de $\downarrow$, esto es equivalente a $\mathcal{M}', w, g \not\models \downarrow x.TH^-(\varphi)$, que a su vez es igual a $\mathcal{M}', w, g \not\models TH^-(\downarrow x.\varphi)$.

- $\mathcal{M}, w, g \models \Diamond\varphi$
  Queremos llegar a que $\mathcal{M}', w, g \models TH^+(\Diamond\varphi)$, o sea, que $\mathcal{M}', w, g \models i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^+(\varphi))$.

  Como $\mathcal{M}, w, g \models \Diamond\varphi$, entonces existe un mundo w' tal que $R^2(w, w')$, y $\mathcal{M}, w', g \models \varphi$. De $R^2(w, w')$ concluimos que $R^3_2(w, w_0, w')$. Aplicando hipótesis inductiva sobre $\mathcal{M}, w', g \models \varphi$, tenemos que $\mathcal{M}', w', g \models TH^+(\varphi)$. Por construcción de $\mathcal{M}'$, sabemos que $\mathcal{M}, w_0, g \models i$.

  A partir de esto, intentaremos ver que $\mathcal{M}', w', g \models \downarrow x.((x \bullet x) \wedge TH^+(\varphi))$, si y sólo si $\mathcal{M}', w', g[x/w'] \models (x \bullet x) \wedge TH^+(\varphi)$. Por la Propiedad 7.1, es equivalente a que $\mathcal{M}', w', g[x/w'] \models x \bullet x$ y $\mathcal{M}', w', g[x/w'] \models TH^+(\varphi)$. Lo primero vale por la definición de R^3_1. Como sabemos que x no ocurre en φ, $\mathcal{M}', w', g[x/w'] \models TH^+(\varphi)$ es equivalente a $\mathcal{M}', w', g \models TH^+(\varphi)$, pero esto ya vimos que vale.

  Usando ahora que $\mathcal{M}', w', g \models TH^+(\varphi)$ y que $\mathcal{M}, w_0, g \models i$, obtenemos finalmente que $\mathcal{M}', w, g \models i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^+(\varphi))$.

- $\mathcal{M}, w, g \not\models \Diamond\varphi$
  Queremos ver que $\mathcal{M}', w, g \not\models TH^-(\Diamond\varphi)$, o sea, $\mathcal{M}', w, g \not\models i \bullet_2 \downarrow x.TH^-(\varphi)$. Definimos el conjunto A con todos los mundos que están relacionados con el actual en R^2: $A = \{w' \mid wR^2w'\}$. Como $\mathcal{M}, w, g \not\models \Diamond\varphi$, entonces $\forall w' \in A$, $\mathcal{M}, w', g \not\models \varphi$. Entonces, por hipótesis inductiva, $\mathcal{M}', w', g \not\models TH^-(\varphi)$.

  Supongamos primero que no hay elementos en A, i.e., que $\#A = 0$. Por construcción no existe un w' tal que $R^3_2(w, w_0, w')$. Entonces no existen α y β tales que $\mathcal{M}', w, g \models \alpha \bullet_2 \beta$, y en particular $\mathcal{M}', w, g \not\models i \bullet_2 \downarrow x.TH^-(\varphi)$.

  Supongamos ahora que $\#A > 0$, y tomemos algún $w' \in A$. Por construcción, $(w, w_0, w') \in R^3_2$. Como $w' \in A$, $\mathcal{M}', w', g \not\models TH^-(\varphi)$. Pero esto es equivalente a $\mathcal{M}', w', g \not\models \downarrow x.TH^-(\varphi)$ para un x que no ocurra en $TH^-(\varphi)$, que es justamente lo que la definición de TH^- enuncia. Y esto ocurre para todo w' tal que $R^3_2(w, w_0, w')$, por lo que $\mathcal{M}', w, g \not\models i \bullet_2 \downarrow x.TH^-(\varphi)$.

- $\mathcal{M}, w, g \models \varphi \Rightarrow \psi$
  Queremos ver que $\mathcal{M}', w, g \models TH^+(\varphi \Rightarrow \psi)$ sii $\mathcal{M}', w, g \models TH^-(\varphi) \rightarrow TH^+(\psi)$. Por Propiedad 5.2, esto sucede cuando $\mathcal{M}', w, g \not\models TH^-(\varphi)$ ó $\mathcal{M}', w, g \models TH^+(\psi)$.

  Como $\mathcal{M}, w, g \models \varphi \Rightarrow \psi$, sabemos que $\mathcal{M}, w, g \not\models \varphi$ ó $\mathcal{M}, w, g \models \psi$.

  Supongamos que sucede $\mathcal{M}, w, g \not\models \varphi$. Por hipótesis inductiva, $\mathcal{M}', w, g \not\models TH^-(\varphi)$. Y por lo tanto $\mathcal{M}', w, g \models TH^-(\varphi) \rightarrow TH^+(\psi)$.

  Supongamos ahora que $\mathcal{M}, w, g \models \psi$. Por hipótesis inductiva nuevamente, $\mathcal{M}', w, g \models TH^+(\psi)$, con lo que $\mathcal{M}', w, g \models TH^-(\varphi) \rightarrow TH^+(\psi)$.

- $\mathcal{M}, w, g \not\models \varphi \Rightarrow \psi$
  Queremos ver que $\mathcal{M}', w, g \not\models TH^-(\varphi \Rightarrow \psi)$, que es equivalente a $\mathcal{M}', w, g \not\models \downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi)))$.

  Como $\mathcal{M}, w, g \not\models \varphi \Rightarrow \psi$, entonces $\mathcal{M}, w, g \models \varphi$ y $\mathcal{M}, w, g \not\models \psi$. Dado que $\mathcal{M}, w, g \models \varphi$, por hipótesis inductiva $\mathcal{M}', w, g \models TH^+(\varphi)$. Y dado que $\mathcal{M}, w, g \not\models \psi$, por hipótesis inductiva otra vez, $\mathcal{M}', w, g \not\models TH^-(\psi)$.

  Supongamos entonces por el absurdo que $\mathcal{M}', w, g \models \downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi)))$, entonces $\mathcal{M}', w, g[x/w] \models (TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))$. Aplicando la Propiedad 5.2, sabemos que o bien $\mathcal{M}', w, g[x/w] \not\models (x \wedge TH^+(\varphi))$ o bien $\mathcal{M}', w, g[x/w] \models (TH^-(\psi)) \bullet \top$.

  Supongamos que $\mathcal{M}', w, g[x/w] \not\models (x \wedge TH^+(\varphi))$. Aplicando la Propiedad 7.1 tenemos que $\mathcal{M}', w, g[x/w] \not\models x$ o que $\mathcal{M}', w, g[x/w] \not\models TH^+(\varphi)$. $\mathcal{M}', w, g[x/w] \not\models x$ no puede ocurrir porque $g[x/w](x) = w$. Si $\mathcal{M}', w, g[x/w] \not\models TH^+(\varphi)$, como x no puede aparecer en $TH^+(\psi)$, entonces tendríamos $\mathcal{M}', w, g \not\models TH^+(\varphi)$, lo cual es absurdo porque habíamos visto que $\mathcal{M}', w, g \models TH^+(\varphi)$.

Supongamos ahora que $\mathcal{M}', w, g[x/w] \models (TH^-(\psi)) \bullet \top$. Con la Propiedad 5.2 tenemos que $\mathcal{M}', w, g[x/w] \models \top$ y que $\mathcal{M}', w, g[x/w] \models TH^-(\psi)$. Como x no puede ocurrir en $TH^-(\psi)$, esto es equivalente a $\mathcal{M}', w, g \models TH^-(\psi)$, lo cual es absurdo porque vimos que $\mathcal{M}', w, g \not\models TH^-(\psi)$.

Este absurdo vino de suponer que $\mathcal{M}', w, g \models TH^-(\varphi \Rightarrow \psi)$, por lo tanto, tenemos que $\mathcal{M}', w, g \not\models TH^-(\varphi \Rightarrow \psi)$.

$\square$

Antes de adentrarnos en la demostración, veremos algunos resultados que nos serán útiles más adelante.

Lema 7.5. *Sea α una fórmula bien formada sin variables libres de $\mathcal{H}(\downarrow)$, R una rama abierta y saturada para el tableaux de la Figura 4.2, a partir del conjunto $\{@_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T, @_a f : F\}$, siendo $f \in PROP$, $i \in NOM$, tales que no aparecen en α.Si $@_a(b \bullet_2 c) : T$ está en la rama R, entonces $@_c(c \bullet_1 c) : T$ también lo está.*

La demostración de este lema, debido a su extensión se encuentra en el Apéndice A.

Lema 7.6. *Sea α una fórmula sin variables libres de $\mathcal{H}(\downarrow)$, entonces $TH^+(\alpha)[y/b] = TH^+(\alpha[y/b])$, y $TH^-(\alpha)[y/b] = TH^-(\alpha[y/b])$, $y \in VAR$, $b \in NOM$.*

Demostración.

Probémoslo por inducción para cada una de las posibles estructuras de α.

- $TH^+(a)$
$$\begin{array}{lllll} TH^+(a)[y/b] & = & a[y/b] & = & \\ & & a & = & \\ & & TH^+(a) & = & TH^+(a[y/b]) \end{array}$$
- $TH^-(a)$
Este caso se ve en forma análoga al anterior.
- $TH^+(\bot)$
$$\begin{array}{lllll} TH^+(\bot)[y/b] & = & (i \wedge f)[y/b] & = & \\ & & i \wedge f & = & \\ & & TH^+(\bot) & = & TH^+(\bot[y/b]). \end{array}$$
- $TH^-(\bot)$
También se analiza de la misma forma, ya que tienen la misma traducción.
- $TH^+(\downarrow x.\varphi)$
Para esta traducción, separaremos en dos casos: cuando coincidan x e y, y cuando no.
  - $x = y$
$$\begin{array}{lllll} TH^+(\downarrow x.\varphi)[x/b] & = & (\downarrow x.TH^+(\varphi))[x/b] & = & \\ & & \downarrow x.TH^+(\varphi) & = & \\ & & TH^+(\downarrow x.\varphi) & = & TH^+((\downarrow x.\varphi)[x/b]). \end{array}$$
  - $x \neq y$
$$\begin{array}{lllll} TH^+(\downarrow x.\varphi)[y/b] & = & (\downarrow x.TH^+(\varphi))[y/b] & = & \\ & & \downarrow x.(TH^+(\varphi)[y/b]) & = & \text{[por hipótesis inductiva]} \\ & & \downarrow x.TH^+(\varphi[y/b]) & = & \\ & & TH^+(\downarrow x.(\varphi[y/b])) & = & TH^+((\downarrow x.\varphi)[y/b]). \end{array}$$

- $TH^-(\downarrow x.\varphi)$
  Este es otro caso que se prueba en forma análoga a su predecesor.

- $TH^+(\Diamond\varphi)$

$$\begin{array}{lcll}
TH^+(\Diamond\varphi)[y/b] & = & (i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^+(\varphi)))[y/b] & = \text{[como } x \text{ no ocurre en } TH^+(\varphi)] \\
 & & i \bullet_2 \downarrow x.((x \bullet x) \wedge (TH^+(\varphi)[y/b])) & = \text{[por hipótesis inductiva]} \\
 & & i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^+(\varphi[y/b])) & = \\
 & & TH^+(\Diamond(\varphi[y/b])) & = \ TH^+((\Diamond\varphi)[y/b]).
\end{array}$$

- $TH^-(\Diamond\varphi)$

$$\begin{array}{lcll}
TH^-(\Diamond\varphi)[y/b] & = & (i \bullet_2 \downarrow x.TH^-(\varphi))[y/b] & = \text{[como } x \text{ no ocurre en } TH^-(\varphi)] \\
 & & i \bullet_2 \downarrow x.(TH^-(\varphi)[y/b]) & = \text{[por hipótesis inductiva]} \\
 & & i \bullet_2 \downarrow x.TH^-(\varphi[y/b]) & = \\
 & & TH^-(\Diamond(\varphi[y/b])) & = \ TH^-((\Diamond\varphi)[y/b]).
\end{array}$$

- $TH^+(\varphi \Rightarrow \psi)$

$$\begin{array}{lcll}
TH^+(\varphi \Rightarrow \psi)[y/b] & = & (TH^-(\varphi) \to TH^+(\psi))[y/b] & = \\
 & & (TH^-(\varphi)[y/b]) \to (TH^+(\psi)[y/b]) & = \text{[por hipótesis inductiva]} \\
 & & TH^-(\varphi[y/b]) \to TH^+(\psi[y/b]) & = \\
 & & TH^+((\varphi[y/b]) \Rightarrow (\psi[y/b])) & = \ TH^+((\varphi \Rightarrow \psi)[y/b]).
\end{array}$$

- $TH^-(\varphi \Rightarrow \psi)$

$$\begin{array}{ll}
TH^-(\varphi \Rightarrow \psi)[y/b] & = \\
(\downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))))[y/b] & = \text{[}x \text{ no ocurre en } TH^-(\psi) \text{ ni } TH^+(\varphi)] \\
\downarrow x.(((TH^-(\psi)[y/b]) \bullet \top) \leftarrow (x \wedge (TH^+(\varphi)[y/b]))) & = \text{[por hipótesis inductiva]} \\
\downarrow x.((TH^-(\psi[y/b]) \bullet \top) \leftarrow (x \wedge TH^+(\varphi[y/b]))) & = \\
TH^-((\varphi[y/b]) \Rightarrow (\psi[y/b])) & = \ TH^-((\varphi \Rightarrow \psi)[y/b]).
\end{array}$$

$\square$

Demostración del Teorema 7.3, parte 2: $(i \wedge TH^+(\alpha)) \wedge (i \bullet i) \not\models f$ *implica* α *satisfacible.*

Supondremos ahora que $(i \wedge TH^+(\alpha)) \wedge (i \bullet i) \not\models f$. Analizaremos entonces una rama saturada y abierta generada a partir de $\{@_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T, @_a f : F\}$. Como en los casos anteriores, a partir de su modelo inducido $\mathcal{M}$, generaremos un modelo de $\mathcal{H}(\downarrow)$ que satisfaga α.

Sea R una rama abierta y saturada para el tableaux de la Figura 4.2, a partir de las fórmulas $\{@_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T, @_a f : F\}$, y sea $\mathcal{M} = \langle W, \{R_1^3, R_2^3\}, V' \rangle$ el modelo de hCTL inducido a partir de R, definido sobre los conjuntos de proposiciones $PROP$, y nominales NOM, con $V'(i) = \{w_0\}$.Definimos $\mathcal{M}' = \langle W, R^2, V \rangle$, modelo de $\mathcal{H}(\downarrow)$ sobre la misma signatura (aunque no nos interesará i, f, ni R_1^3), de la siguiente manera:

$$\begin{array}{lcl}
R^2 & = & \{(w, w') \mid (w, w_0, w') \in R_2^3\} \\
V(a) & = & V'(a), \forall a \in PROP \cup NOM.
\end{array}$$

Nuestro objetivo es ver que $\mathcal{M}', w_0, g \models \alpha$, donde $V(i) = \{w_0\}$.

Observemos que en este caso, la traducción de modelos que proponemos, es la misma que usamos en el Capítulo 6.

Comencemos por analizar la rama R.

$$
\begin{array}{ll}
1.\ @_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T & \\
2.\ @_a f : F & \\
3.\ @_a(i \wedge TH^+(\alpha)) : T & \succ\text{(Lema 7.2) en 1} \\
4.\ @_a(i \bullet i) : T & \succ\text{(Lema 7.2) en 1} \\
5.\ @_a i : T & \succ\text{(Lema 7.2) en 3} \\
6.\ @_a TH^+(\alpha) : T & \succ\text{(Lema 7.2) en 3} \\
7.\ @_i a : T & \succ(NOM)\text{ en 5} \\
8.\ @_a(a \bullet i) : T & \succ(R_2)\text{ en 4 y 7} \\
9.\ @_a(a \bullet a) : T & \succ(R_3)\text{ en 4 y 7} \\
10.\ @_i f : F & \succ\text{(ID) en 5 y 2}
\end{array}
$$

Sabemos entonces que $@_i f : F$ y $@_a TH^+(\varphi)) : T$ están en la R, a partir de lo cual demostraremos que el modelo $\mathcal{M}'$ en el mundo a satisface φ.

Como en los casos anteriores, vamos a tener dos implicaciones, de las cuales la primera es la que necesitaremos, pero sin la segunda la inducción no funcionaría. Queremos ver que para toda *FBF* φ de $\mathcal{H}(\downarrow)$ sin variables libres, para todo $g : VAR \to W$, para todo $a \in NOM$, si $@_a(a \bullet a) : T \in R$, y si f e i no son subfórmulas de φ, entonces:

$$@_a TH^+(\varphi) : T \in R \text{ entonces } \mathcal{M}', a, g \models \varphi$$

$$@_a TH^-(\varphi) : F \in R \text{ entonces } \mathcal{M}', a, g \not\models \varphi$$

La hipótesis que pedimos de que el mundo en el que realicemos la inducción esté relacionado de esta forma, se debe a que todos los mundos más interesantes, los que imitan el comportamiento de los de la otra lógica, están relacionados consigo mismos por la relación R_1, y necesitaremos este hecho para la prueba.

Probemos esta proposición por inducción en la estructura de φ y suponiendo que pertenece a R.

- $@_a TH^+(\bot) : T \in R$
  Veamos que esto no puede ocurrir, y supongamos por el absurdo que sí:

  $$
  \begin{array}{ll}
  1.\ @_a(i \wedge f) : T & \\
  2.\ @_a f : T & \succ\text{(Lema 7.2) en 1} \\
  3.\ @_a i : T & \succ\text{(Lema 7.2) en 1} \\
  4.\ @_i f : T & \succ\text{(ID) en 2 y 3}
  \end{array}
  $$

  Pero sabíamos que $@_i f : F \in R$, y esto es absurdo porque R es una rama abierta y saturada, por lo que llegamos a que no puede pasar que $@_a TH^+(\bot) : T$.

- $@_a TH^-(\bot) : F \in R$
  Por definición, $\mathcal{M}', [a], g \not\models \bot$.

- $@_a TH^+(b) : T \in R$
  No hay que considerar el caso de que b sea una variable, ya que suponemos que la fórmula no contiene variables libres, y el método de tableaux no las agrega tampoco. Como $@_a b : T \in R$, por la construcción del modelo inducido sabemos entonces que $\mathcal{M}, [a], g \models b$, lo que por la construcción de $\mathcal{M}'$ nos da que $\mathcal{M}', [a], g \models b$.

- $@_a TH^-(b) : F \in R$
  Tampoco debemos considerar el caso de que b sea variable, por el mismo motivo. Como en el caso anterior, $a : b : F \in R$, por la construcción del modelo inducido sabemos que $\mathcal{M}, [a], g \not\models b$, lo que por construcción de $\mathcal{M}'$, tenemos que $\mathcal{M}', [a], g \not\models b$.

- $@_a TH^+(\downarrow x.\varphi) : T \in R$
Como $@_a \downarrow x.TH^+(\varphi) : T$ está en R, y R está saturada, $@_a TH^+(\varphi)[x/a] : T$ también debe estar por la regla $\downarrow^+$. Como $@_a TH^+(\varphi)[x/a] : T$ es igual a $@_a TH^+(\varphi[x/a]) : T$ por Lema 7.6, aplicando hipótesis inductiva obtenemos que $\mathcal{M}', [a], g \models \varphi[x/a]$. Pero dado que sabemos por la construcción del modelo inducido y de $\mathcal{M}'$ que $\mathcal{M}', [a], g \models a$, $\mathcal{M}', [a], g \models \varphi[x/a]$, sii $\mathcal{M}', [a], g \models \downarrow x.\varphi$, que es lo que queríamos ver.

- $@_a TH^-(\downarrow x.\varphi) : F \in R$
En forma similar al caso anterior, si $@_a \downarrow x.TH^-(\varphi) : F$ está en R, $@_a TH^-(\varphi)[x/a] : F$ también está por $\downarrow^-$. El Lema 7.6 dice que $@_a TH^-(\varphi)[x/a] : F$ es $@_a TH^-(\varphi[x/a]) : F$. Aplicando hipótesis inductiva obtenemos entonces que $\mathcal{M}', [a], g \not\models \varphi[x/a]$. Y dado que por construcción $\mathcal{M}', [a], g \models a$, $\mathcal{M}', [a], g \not\models \varphi[x/a]$ que es verdadero sii $\mathcal{M}', [a], g \not\models \downarrow x.\varphi$, que es lo que queríamos probar.

- $@_a TH^+(\Diamond\varphi) : T \in R$
Analicemos qué sucede en toda rama que contenga esta fórmula.

1. $@_a(i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^+(\varphi))) : T$
2. $@_a(b \bullet_2 c) : T \qquad \succ(\bullet^+)$ en 1
3. $@_b i : T \qquad \succ(\bullet^+)$ en 1
4. $@_c \downarrow x.((x \bullet x) \wedge TH^+(\varphi)) : T$
5. $@_c(c \bullet c) \wedge TH^+(\varphi) : T \qquad \succ(\downarrow^+)$ en 4
6. $@_c(c \bullet c) : T \qquad \succ$(Lema 7.2) en 5
7. $@_c TH^+(\varphi) : T \qquad \succ$(Lema 7.2) en 5
8. $@_a(i \bullet_2 c) : T \qquad \succ(R_2)$ en 2 y 3

Dado que $@_a i \bullet_2 c : T \in R$, sabemos por la construcción del modelo inducido $\mathcal{M}$, que $R_2^3([a], [i], [c])$, y por la construcción de $\mathcal{M}'$, como $V'(i) = \{w_0\}$, $w_0 = [i]$. Por lo que $R_2^3([a], w_0, [c])$, que otra vez por la construcción de $\mathcal{M}'$ sabemos que $R_2([a], [c])$.

Por otro lado, podemos aplicar la hipótesis inductiva usando 6 y 7, y obtenemos que $\mathcal{M}', [c], g \models \varphi$. Pero esto significa entonces que $\mathcal{M}', [a], g \models \Diamond\varphi$, que es lo que queríamos probar.

- $@_a TH^-(\Diamond\varphi) : F \in R$
Sabemos que $@_a(i \bullet_2 \downarrow x.TH^-(\varphi)) : F$ pertenece a la rama R, y por hipótesis también pertenece $@_a(a \bullet_2 a) : T$.

Supongamos que no se cumple la proposición, y analicemos por el absurdo qué ocurre cuando $\mathcal{M}', [a], g \models \Diamond\varphi$. Entonces existe $[b]$ tal que $R^2([a], [b])$ y que $\mathcal{M}', [b], g \models \varphi$. Como $R^2([a], [b])$, por definición de $\mathcal{M}'$, $R_2^3([a], w_0, [b])$ (donde $w_0 = [i]$), y por Lema 4.4, existe b tal que $@_a(i \bullet_2 b) : T \in R$. Aplicando el Lema 7.5 tenemos que $@_b(b \bullet_2 b) : T \in R$. Aplicando $\bullet^-$ entre $@_a(i \bullet_2 \downarrow x.TH^-(\varphi)) : F$ y $@_a(i \bullet_2 b) : T$ tenemos que ocurre $@_i i : F$ ó $@_b \downarrow x.TH^-(\varphi) : F$ en R. Lo primero no puede ser porque tendríamos clash en una rama abierta. Por lo tanto $@_b \downarrow x.TH^-(\varphi) : F \in R$. Aplicando $\downarrow^-$, tenemos que $@_b TH^-(\varphi)[x/b] : F$. Pero como sabemos, por definición de TH^-, que x no ocurre en $TH^-(\varphi)$, concluimos que $@_b TH^-(\varphi) : F \in R$. Pero aplicando la hipótesis inductiva (observar que podemos aplicarla porque sabemos que $@_b b \bullet_2 b : T \in R$), obtenemos que $\mathcal{M}', [b], g \not\models \varphi$. Aunque habíamos dicho más arriba que $\mathcal{M}', [b], g \models \varphi$. Absurdo que viene de suponer que $\mathcal{M}', [a], g \models \Diamond\varphi$, por lo que tenemos que $\mathcal{M}', [a], g \not\models \Diamond\varphi$.

- $@_a TH^+(\varphi \Rightarrow \psi) : T \in R$
Sabemos que $@_a TH^-(\varphi) \rightarrow TH^+(\psi) : T \in R$, y además por hipótesis sabemos que también $@_a a \bullet a : T \in R$. Analicemos la rama.

$$\begin{array}{c}
1.\ @_a TH^-(\varphi) \to TH^+(\psi) : T \\
2.\ @_a a \bullet a : T
\end{array}$$

$$\begin{array}{cccc}
(\to^+) \text{ en 1 y 2} \prec & 3\text{a}.@_a TH^-(\varphi) : F & 3\text{b}.@_a TH^+(\psi) : T & \succ(\to^+) \text{ en 1 y 2}
\end{array}$$

O bien $3a$ o bien $3b$ deben pertenecer a la rama. Si es el primer caso, entonces valiéndonos de 2 podemos aplicar la hipótesis inductiva y obtener que $\mathcal{M}', [a], g \not\models \varphi$, de lo que deducimos $\mathcal{M}', [a], g \models \varphi \Rightarrow \psi$. De ser el segundo caso, aplicando también la hipótesis inductiva con 2 tenemos que $\mathcal{M}', [a], g \models \psi$, con lo que también llegamos a que $\mathcal{M}', [a], g \models \varphi \Rightarrow \psi$. Por lo tanto, en ambos casos llegamos a lo que queríamos ver, que $\mathcal{M}', [a], g \models \varphi \Rightarrow \psi$.

- $@_a TH^-(\varphi \Rightarrow \psi) : F \in R$
  Sabemos que $@_a{\downarrow}x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))) : F \in R$, y por hipótesis que $@_a a \bullet a : T$.

$$\begin{array}{c}
1.\ @_a{\downarrow}x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))) : F \\
2.\ @_a(a \bullet a) : T \\
3.\ @_a(TH^-(\psi) \bullet \top) \leftarrow (a \wedge TH^+(\varphi)) : F \quad \succ({\downarrow}^-) \text{ en } 1 \\
4.\ @_c(a \bullet b) : T \quad \succ(\leftarrow^-) \text{ en } 3 \\
5.\ @_b(a \wedge TH^+(\varphi)) : T \quad \succ(\leftarrow^-) \text{ en } 3 \\
6.\ @_c(TH^-(\psi) \bullet \top) : F \quad \succ(\leftarrow^-) \text{ en } 3
\end{array}$$

$$\begin{array}{cccc}
(\bullet^-) \text{ en 6 y 4} \prec & 7\text{a}.\ @_b \top : F & 7\text{b}.\ @_a TH^-(\psi) : F & \succ(\bullet^-) \text{ en 6 y 4} \\
 & \otimes & 8.@_b a : T & \succ(\text{Lema 7.2}) \text{ en 5} \\
 & & 9.@_b TH^+(\varphi) : T & \succ(\text{Lema 7.2}) \text{ en 5} \\
 & & 10.\ @_a TH^+(\varphi) : T & \succ(\text{ID}) \text{ en 8 y 9}
\end{array}$$

Aplicando la hipótesis inductiva entre 2 y $7b$, tenemos que $\mathcal{M}', [a], g \not\models \psi$. Aplicando nuevamente la hipótesis inductiva entre 2 y 10, obtenemos que $\mathcal{M}', [a], g \models \varphi$. Con lo que llegamos a que $\mathcal{M}', [a], g \not\models \varphi \Rightarrow \psi$, que es lo que queríamos probar.

$\square$

Capítulo 8

Conclusiones

En esta tesis comenzamos presentando una vista general de las lógicas modales, con su sintaxis y semántica. Observamos que las lógicas más expresivas, suelen tener asociado un mayor costo computacional.

Vimos en el Capítulo 2 las lógicas híbridas, que son una familia de extensiones de las lógicas modales. Estas poseen características particularmente interesantes, como la capacidad de expresar una noción de identidad, que con la lógica modal básica no puede lograrse. Para expresar la identidad nos valemos de nominales. Presentamos el operador híbrido @, que nos permite movernos por los distintos mundos arbitrariamente. También vimos el operador $\downarrow$, con el que podemos nombrar mundos en tiempo de evaluación. Observamos que, generalmente, al incluir en una lógica el operador híbrido $\downarrow$, la lógica resultante se vuelve indecidible. En esta tesis investigamos justamente el efecto, en términos de complejidad computacional, que tiene el agregado del operador $\downarrow$ en una lógica modal. En particular, nos interesa estudiar $\downarrow$ sobre una lógica tan simple como sea posible, para poder delinear cuidadosamente el límite entre decidibilidad e indecidibilidad.

La complejidad computacional del operador $\downarrow$ ha sido investigada principalmente sobre lógicas con estructura booleana, es decir, lógicas que son extensiones de la lógica proposicional. Claramente, la presencia de operadores booleanos facilita la codificación de problemas difíciles en el lenguaje. La combinación de operadores booleanos más el $\downarrow$, lleva rápidamente a la indecidibilidad. En esta tesis, mostramos que el operador $\downarrow$ torna indecidible aun una lógica sub-booleana como CTL. En este caso, codificar problemas indecidibles es mucho más complejo, como explicamos en los Capítulos 6 y 7.

Presentamos en el Capítulo 3 el cálculo CTL (o NL), que puede ser visto como una lógica modal. Esta lógica que es ampliamente usada en, por ejemplo, el área de la lingüística computacional, no posee operadores booleanos. El problema de satisfacibilidad para CTL es trivial; el problema interesante en esta lógica es el de entailment. En el caso de la versión no asociativa, que es la que nos interesa, este problema está en PTIME. Por estos motivos, la tomamos de base para nuestro estudio.

Comenzamos en el Capítulo 4 juntando los ingredientes que teníamos hasta el momento, y presentamos una hibridización de la lógica CTL. En esta nueva lógica, llamada hCTL$(@, \downarrow)$, agregamos a CTL, además de nominales, los operadores híbridos @ y $\downarrow$. Para poder trabajar con ella, definimos un sistema de tableaux para esta lógica, y probamos su completitud (cf. Teorema 4.1).

Para demostrar este teorema, mostramos que toda rama abierta y saturada del tableaux, tiene un modelo que satisface todas sus fórmulas. Explicamos además, que este mismo método de inferencia podemos utilizarlo en la resolución de los problemas de entailment para aquellas lógicas que son fragmentos de hCTL$(@, \downarrow)$, en particular hCTL$(@)$ y hCTL$(\downarrow)$.

En el Capítulo 5 encontramos una cota inferior para el entailment de la lógica hCTL$(@)$ (incluso a su fragmento con una relación y sin $\leftarrow$ ó $\rightarrow$). Esto lo logramos resolviendo el problema de satisfacibilidad de la lógica proposicional con el de entailment de hCTL$(@)$ (cf. Teorema 5.1). Como sabemos que el problema SAT para LP es NP-complete, nos da una cota inferior para la complejidad del problema de entailment en hCTL$(@)$ (cf. Corolario 5.3). Esto muestra que sólo la presencia de nominales y @, incrementa considerablemente su complejidad computacional. Lo cual es sorprendente, ya que recordemos, en la LMB al agregarle estas características se vuelve más expresiva, pero manteniendo la complejidad computacional. Queda, para trabajos posteriores, la pregunta de saber cuál es la complejidad exacta de hCTL(@), de la cual probamos aquí una cota inferior.

En el Capítulo 5 no sólo conseguimos este resultado, sino que también observamos una forma de poder representar una base booleana en hCTL(@). En este capítulo mostramos una forma simplificada de la mecánica que utilizamos para resolver otros problemas en los capítulos subsiguientes. Fue importante el uso de $\Rightarrow$ y $\perp$ como conjunto de operadores adecuados, por estar más cercano a lo que hCTL(@) puede expresar.

Ya en el Capítulo 6, analizamos la complejidad de hCTL(@, $\downarrow$). Este es un ejemplo de lógica modal sub-booleana que posee el operador híbrido $\downarrow$. Vimos una forma de codificar el problema de satisfacibilidad de la lógica $\mathcal{H}(\downarrow)$, que es indecidible, utilizando el problema de entailment en la lógica hCTL(@, $\downarrow$) (en particular, de su fragmento con dos relaciones y sin $\leftarrow$ ó $\rightarrow$) (cf. Teorema 6.1), consiguiendo la complejidad del problema de entailment mencionado (cf. Corolario 6.2).

Como dijimos anteriormente, buscábamos analizar el comportamiento computacional de una lógica sin base booleana, pero que contenga el operador $\downarrow$. El resultado al que llegamos fue que esta lógica es también indecidible. Observamos que en esta prueba, el operador @ jugaba un rol fundamental, por lo que nos preguntamos si la indecidibilidad no podría ser evitada al no incluir dicho operador, y así acercarnos más al conocimiento de qué es lo que lleva a dicha indecidibilidad. Por lo que en el Capítulo 7, analizamos la complejidad de la lógica hCTL($\downarrow$) (con dos relaciones). Para dicho análisis nos valimos de métodos similares a los que utilizamos en el capítulo anterior, aunque debimos apelar a diversos recursos técnicos para suplir la ausencia del @. Aún así, conseguimos codificar el problema SAT de $\mathcal{H}(\downarrow)$ también en esta lógica (cf. Teorema 7.3), infiriendo la indecidibilidad del mismo (cf. Corolario 7.4).

Vale la pena resaltar que el sólo hecho de agregarle el operador $\downarrow$ a una lógica sub-booleana como lo es CTL, que incluso tiene complejidad PTIME para el problema de entailment, hace que este mismo problema se torne indecidible. Este salto expresivo es realmente sorprendente, y nos da una pauta del gran poder expresivo que posee el operador $\downarrow$.

Apéndice A

Demostración del Lema 7.5

Lema 7.5. *Sea α una fórmula bien formada sin variables libres de $\mathcal{H}(\downarrow)$, R una rama abierta y saturada para el tableaux de la Figura 4.2, a partir del conjunto $\{@_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T, @_a f : F\}$, siendo $f \in PROP$, $i \in NOM$, tales que no aparecen en α.Si $@_a(b \bullet_2 c) : T$ está en la rama R, entonces $@_c(c \bullet_1 c) : T$ también lo está.*

Antes de demostrar este lema, necesitaremos ver un conjunto de definiciones y resultados que serán cruciales para dicha demostración.

Definición A.1 (*FormTH*)**.** Definimos el conjunto de fórmulas *FormTH* de la siguiente manera:

$$FormTH = \{@_a TH^+(\alpha) : V\} \cup \{@_a TH^-(\alpha) : V\}$$

con $a \in NOM$, $\alpha \in FBF$ de $\mathcal{H}(\downarrow)$ sin variables libres, $\alpha \neq b$, $b \in NOM$, $V \in \{T, F\}$, $(\alpha = b \bullet c) \Rightarrow (V = F)$.

Sea $\varphi = @_a TH^+(\alpha) : V$, $\varphi \in FormTH$, llamamos a V el *símbolo* de φ, y a α el parámetro. Si es de la forma $@_a TH^+(\alpha) : V$ decimos que tiene *polaridad* positiva, caso contrario diremos que es negativa.

Definición A.2 (*FormD*)**.** Definimos el conjunto de fórmulas *FormD* como sigue:

$$FormD = \{@_a(\alpha \bullet_2 \beta) : V\}$$

con $a \in NOM$, $\alpha, \beta \in FBF$ de hCTL($\downarrow$), $V \in \{T, F\}$, y con α ó β distintos de nominales y variables.

Definición A.3 (correcta)**.** Sea $\varphi \in FormTH$, diremos que es correcta cuando si su símbolo es T tiene polaridad positiva, y si su símbolo es F tiene polaridad negativa.

Definición A.4 (adecuada)**.** Sea $\varphi \in FormD$, diremos que es adecuada cuando además $\varphi \in FormTH$.

Lema A.1. *Si en una rama abierta ocurre una fórmula $\varphi \in FormTH$ que es correcta:*

- *Si φ participa de la derivación de una fórmula $\psi \in FormTH$, entonces ψ es correcta.*
- *Si φ participa de la derivación de una fórmula $\psi \in FormD$, entonces ψ es adecuada.*

Demostración. Veámoslo por inducción en la estructura del parámetro α, pero primero observemos que el que determinada fórmula $@_a\varphi : V$ sea correcta, no depende de la etiqueta a. Por lo que con analizar sólo una de las posibles expansiones de la fórmula nos es suficiente. Dado que las identidades (i.e., $@_a b : V$) podrían influenciar sólo de esta forma, preservando la correctitud, las podemos ignorar al no ser claves en la generación de fórmulas incorrectas, y tampoco ser *FormTH* correctas. Lo propio haremos con las fórmulas $@_a(b \bullet c) : V$ y $@_a(b \bullet_2 c) : V$, ya que por un lado no están en *FormTH*, y cuando puedan servir para activar alguna regla, como en los casos $\bullet^-$, $\rightarrow^+$ y $\leftarrow^-$, lo consideraremos en el momento: supondremos que siempre hay alguna relación para activar cada una de las posibles reglas, aunque en la derivación no ocurra. Con eso nos alcanza, ya que lo que nos interesa es analizar la estructura de las posibles fórmulas que puedan llegar a generarse. Y por el mismo motivo, tampoco nos interesará por el momento la aplicación de las reglas R_2 y R_3, ya que tan sólo pueden generar relaciones de esa forma.

Estos mismos análisis aplican también en la segunda parte del lema: que sea adecuada una fórmula *FormD* no depende de la etiqueta, y ni las igualdades ni relaciones de nominales son fórmulas de *FormD*.

Comencemos ahora la inducción por los casos base, e iremos demostrando ambas partes del lema en simultaneo.

- $a, a \in \mathit{PROP} \cup \mathit{NOM}$

  Si fuera un nominal o una proposición, sería una fórmula correcta, y no podría participar para generar otro $\varphi \in \mathit{FormTH}$ correcto (salvo renombres, que ignoraremos).

  Ni es, ni puede participar de la generación de una *FormD* (salvo etiquetas).

- $a, a \in \mathit{VAR}$

  Esto no puede ocurrir, ya que supusimos que α no contiene variables libres.

- $\bot$

  Como $@_a TH^+(i \wedge f) : T = @_a((i \leftarrow \top) \bullet (\top \rightarrow f)) : T$, analicemos qué sucede con la derivación

  1. $@_a((i \leftarrow \top) \bullet (\top \rightarrow f)) : T$
  2. $@_a(b \bullet c) : T$ $\quad \succ(\bullet^+)$ en 1
  3. $@_b(i \leftarrow \top) : T$ $\quad \succ(\bullet^+)$ en 1
  4. $@_c(\top \rightarrow f) : T$ $\quad \succ(\bullet^+)$ en 1

  $(\leftarrow^+)$ en 3 y 2 $\prec$ 5a. $@_c\top : F$ $\otimes$ $\quad$ 5b. $@_a i : T$ $\succ(\leftarrow^+)$ en 3 y 2

  $(\rightarrow^+)$ en 4 y 2 $\prec$ 6a. $@_b\top : F$ $\otimes$ $\quad$ 6b. $@_a f : T$ $\succ(\rightarrow^+)$ en 4 y 2

  Dijimos que 2 y 5*b* podíamos descartarlas. 3 y 4 no son *FormTH* ya que ni i ni f pueden pertenecer al vocabulario de α. 6*b* no puede generar más *FormTH*s.

  Ninguna es, ni puede participar de la generación de una *FormD*.

  Por otro lado $@_a TH^-(i \wedge f) : F = @_a((i \leftarrow \top) \bullet (\top \rightarrow f)) : F$. Si hay algún $@_a(b \bullet c)$ para desencadenar la única regla con la que puede interactuar, entonces generaría:

  1. $@_a((i \leftarrow \top) \bullet (\top \rightarrow f)) : F$
  2. $@_a(b \bullet c) : T$

  $(\bullet^-)$ en 1 y 2 $\prec$ 3a. $@_b(i \leftarrow \top) : F$ $\quad$ 3b. $@_c(\top \rightarrow f) : F$ $\succ(\bullet^-)$ en 1 y 2

  $(\leftarrow^-)$ en 3a $\prec$ 4a. $@_d(b \bullet e) : T$ $\quad$ 4b. $@_d(e \bullet c) : T$ $\succ(\rightarrow^-)$ en 3b

  $(\leftarrow^-)$ en 3a $\prec$ 5a. $@_d i : F$ $\quad$ 5b. $@_d f : F$ $\succ(\rightarrow^-)$ en 3b

  $(\leftarrow^-)$ en 3a $\prec$ 6a. $@_e\top : T$ $\quad$ 6b. $@_e\top : T$ $\succ(\rightarrow^-)$ en 3b

  Otra vez, podemos descartar 4a, 5a, 4b, y 5b. 3a y 3b también por no poder ocurrir i ni f en α. 6a y 6b son $@_a\top : T$, que como dijimos es un macro para $@_e{\downarrow}x.x : T$, que visto como *FormTH* tiene símbolo positivo, y como es equivalente $\downarrow x.x$ a $\downarrow x.TH^+(x)$ y a $TH^+(\downarrow x.x)$, donde en ambos casos son correctos, e incluso aplicando la regla $\downarrow^+$ nos quedaría una identidad que descartaríamos, por lo que es correcta.

Ninguna es, ni puede participar de la generación de una *FormD*.

- $\downarrow x.\varphi$

  Como $@_a TH^+(\downarrow x.\varphi) : T = @_a \downarrow x.TH^+(\varphi) : T$, sólo puede aplicarse $\downarrow^+$, y así obtener $@_a TH^+(\varphi)[x/a] : T$, que sabemos por Lema 7.6 es igual a $@_a TH^+(\varphi[x/a]) : T$, que es una fórmula correcta en la que podemos aplicar la hipótesis inductiva. Además, de ser *FormD*, por definición es adecuada. Y de participar de la generación de alguna *FormD* por hipótesis inductiva es adecuada.

  El caso de $@_a TH^-(\downarrow x.\varphi) : F$ es análogo.

- $\varphi \Rightarrow \psi$

  La fórmula $@_a(TH^-(\varphi) \rightarrow TH^+(\psi)) : T$, suponiendo que haya un $@_c(b \bullet a)$ para activarla, genera $@_b TH^-(\varphi) : F$ y $@_c TH^+(\psi) : T$. Ambas son correctas, y en ambas podemos aplicar la hipótesis inductiva. De ser *FormD*, por definición son adecuadas. De participar de la generación de alguna *FormD* por hipótesis inductiva, es adecuada también. Ninguna otra fórmula es una *FormD*.

  Analicemos el árbol de $@_a \downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))) : F$.

  1. $@_a \downarrow x.((TH^-(\psi) \bullet \top) \leftarrow (x \wedge TH^+(\varphi))) : F$
  2. $@_a((TH^-(\psi) \bullet \top) \leftarrow (a \wedge TH^+(\varphi))) : F$ $\quad \succ(\downarrow^-)$ en 1
  3. $@_c(a \bullet b) : F$ $\quad \succ(\leftarrow^-)$ en 2
  4. $@_b(a \wedge TH^+(\varphi)) : T$ $\quad \succ(\leftarrow^-)$ en 2
  5. $@_c(TH^-(\psi) \bullet \top) : F$ $\quad \succ(\leftarrow^-)$ en 2
  6. $@_c(d \bullet e) : T$ $\quad \succ$(*supongo*) en hipótesis
  7. $@_d TH^-(\psi) : F$ $\quad \succ(\bullet^-)$ en 5 y 6

  $(\bullet^-)$ en 5 y 6 $\prec$ 8a.$@_e \top : F$ $\otimes$ | 8b. $@_b((a \leftarrow \top) \bullet (\top \rightarrow TH^+(\varphi))) : T$ $\quad \succ$(*expandir*) en 4

  9. $@_b(j \bullet k) : T$ $\quad \succ(\bullet^+)$ en 8b
  10. $@_j(a \leftarrow \top) : T$ $\quad \succ(\bullet^+)$ en 8b
  11. $@_k(\top \rightarrow TH^+(\varphi)) : T$ $\quad \succ(\bullet^+)$ en 8b

  $(\leftarrow^+)$ en 10 y 9 $\prec$ 12a. $@_k \top : F$ $\otimes$ | 12b.$@_b a : T$ $\quad \succ(\leftarrow^+)$ en 10 y 9

  $(\rightarrow^+)$ en 11 y 9 $\prec$ 13a.$@_j \top : F$ $\otimes$ | 13b.$@_b TH^+(\varphi) : T$ $\quad \succ(\rightarrow^+)$ en 11 y 9

  3, 6, 9 y 12b como dijimos, las podemos ignorar. 5, 8b y 4 no son *FormTH* ya que no hay ningún *FormTH* con $\bullet$ como operador principal. 2 y 10 tampoco, ya que no hay ningún *FormTH* con $\leftarrow$ como operador principal. 7 y 13b son *FormTHs* correctas, sobre las que podemos aplicar la hipótesis inductiva. 11 podría compararse con $TH^+(\varphi \Rightarrow \psi)$, que a su vez es correcta, y ya exploramos su árbol. 7, 11 y 12b, de ser *FormD*, por definición son adecuadas. De participar de la generación de alguna *FormD* por hipótesis inductiva, es adecuada también. Ninguna otra fórmula es una *FormD*. El resto que se generarían son identidades.

- $\Diamond\varphi$

Supongamos que hay algún $@_a(b \bullet_2 c) : T$.

$$1.\ @_a(i \bullet_2 {\downarrow} x.TH^-(\varphi)) : F$$
$$2.\ @_a(b \bullet_2 c) : T$$

$(\bullet^-)$ en 1 y 2 $\prec$ 3a. $@_b i : F$ 3b.$@_c{\downarrow}x.TH^-(\varphi) : F$ $\succ(\bullet^-)$ en 1 y 2

4.$@_c TH^-(\varphi) : F$ $\succ({\downarrow}^-)$ en 3b

Podemos ignorar 3a. En 3b, ${\downarrow}x.TH^-(\varphi)$ es equivalente a $TH^-({\downarrow}x.\varphi)$ por definición, y como 4, resulta ser una *FormTH* correcta a quien le podemos aplicar la hipótesis inductiva. Además, de ser *FormD*, por definición es adecuada. Y de participar de la generación de alguna *FormD* por hipótesis inductiva es adecuada. Ninguna otra fórmula es, ni puede participar de la generación de una *FormD*.

Pasemos a analizar el otro caso entonces.

$$1.\ @_a(i \bullet_2 {\downarrow}x.((x \bullet x) \wedge TH^+(\varphi))) : T$$
$$2.\ @_a(b \bullet_2 c) : T \qquad \succ(\bullet^+) \text{ en } 1$$
$$3.\ @_b i : T \qquad \succ(\bullet^+) \text{ en } 1$$
$$4.\ @_c{\downarrow}x.((x \bullet x) \wedge TH^+(\varphi)) : T \qquad \succ(\bullet^+) \text{ en } 1$$
$$5.\ @_c((c \bullet c) \wedge TH^+(\varphi)) : T \qquad \succ({\downarrow}^+) \text{ en } 4$$
$$6.\ @_c((c \bullet c) \leftarrow \top) \bullet (\top \to TH^+(\varphi)) : T \qquad \succ(\textit{expandir}) \text{ en } 5$$
$$7.\ @_c(d \bullet e) : T \qquad \succ(\bullet^+) \text{ en } 6$$
$$8.\ @_d(c \bullet c) \leftarrow \top : T \qquad \succ(\bullet^+) \text{ en } 6$$
$$9.\ @_e(\top \to TH^+(\varphi)) : T \qquad \succ(\bullet^+) \text{ en } 6$$

$(\leftarrow^+)$ en 8 y 7 $\prec$ 10a. $@_e\top : F$ 10b.$@_c(c \bullet c) : T$ $\succ(\leftarrow^+)$ en 8 y 7

$\otimes$

$(\to^+)$ en 9 y 7 $\prec$ 11a.$@_d\top : F$ 11b.$@_c TH^+(\varphi) : T$ $\succ(\to^+)$ en 9 y 7

$\otimes$

$$12.\ @_c(j \bullet k) : T \qquad \succ(\bullet^+) \text{ en } 10$$
$$13.\ @_c j : T \qquad \succ(\bullet^+) \text{ en } 10$$
$$14.\ @_c k : T \qquad \succ(\bullet^+) \text{ en } 10$$

Las fórmulas 2, 3, 7, 13 y 14 las podemos ignorar. 5, 6, 10b y 12 no son *FormTH* ya que no hay ningún *FormTH* con $\bullet$ como operador principal. 8 tampoco, ya que no hay ningún *FormTH* con $\leftarrow$ como operador principal. 4 no es *FormTH* porque no puede haber ninguno con ${\downarrow}x.$ seguido de $\bullet$ como operador principal. 11b es una *FormTH* correcta, en la que podemos aplicar la hipótesis inductiva. En tanto a 9, vamos a analizarla: $@_e\top \to TH^+(\varphi) : T$ podría compararse con $TH^+(\varphi \Rightarrow \psi)$, que a su vez es correcta, y ya exploramos su árbol. 11b y 9, de ser *FormD*, por definición son adecuadas. De participar de la generación de alguna *FormD* por hipótesis inductiva, es adecuada también. Ninguna otra fórmula es una *FormD*. El resto que se generarían son identidades.

$\square$

Corolario A.2. *Partiendo de las fórmulas $@_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T$, y $@_a f : F$, no pueden generarse fórmulas FormTH que no sean correctas, ni FormD que no sean adecuadas.*

Demostración. Comencemos analizando el árbol correspondiente.

$$1.\ @_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T$$
$$2.\ @_a f : F$$
$$3.\ @_a(((i \wedge TH^+(\alpha)) \leftarrow \top) \bullet (\top \to (i \bullet i))) : T \qquad \succ(\textit{expandir}) \text{ en } 1$$
$$4.\ @_a(b \bullet c) : T \qquad \succ(\bullet^+) \text{ en } 5$$
$$5.\ @_b((i \wedge TH^+(\alpha)) \leftarrow \top) : T \qquad \succ(\bullet^+) \text{ en } 5$$
$$6.\ @_c(\top \to (i \bullet i)) : T \qquad \succ(\bullet^+) \text{ en } 5$$

$$(\leftarrow^+) \text{ en } 5 \text{ y } 4 \prec \quad 7a.\ @_c\top : F \qquad\qquad 7b. @_a(i \wedge TH^+(\alpha)) : T \qquad \succ(\leftarrow^+) \text{ en } 5 \text{ y } 4$$
$$\otimes$$

$$(\to^+) \text{ en } 6 \text{ y } 4 \prec \quad 8a. @_b\top : F \qquad\qquad 8b. @_a(i \bullet i) : T \qquad \succ(\to^+) \text{ en } 6 \text{ y } 4$$
$$\otimes$$
$$9. @_a((i \leftarrow \top) \bullet (\top \to TH^+(\alpha))) : T \qquad \succ(\textit{expandir}) \text{ en 7b}$$
$$10.\ @_a(d \bullet e) : T \qquad \succ(\bullet^+) \text{ en } 9$$
$$11.\ @_d(i \leftarrow \top) : T \qquad \succ(\bullet^+) \text{ en } 9$$
$$12.\ @_e(\top \to TH^+(\alpha)) : T \qquad \succ(\bullet^+) \text{ en } 9$$

$$(\leftarrow^+) \text{ en } 11 \text{ y } 10 \prec 13a.\ @_e\top : F \qquad\qquad 13b. @_a i : T \qquad \succ(\leftarrow^+) \text{ en } 11 \text{ y } 10$$
$$\otimes$$

$$(\to^+) \text{ en } 12 \text{ y } 10 \prec 14a. @_d\top : F \qquad 14b. @_a TH^+(\alpha) : T \qquad \succ(\to^+) \text{ en } 12 \text{ y } 10$$
$$\otimes$$

4, 8b, 10 y 13b las podemos ignorar. 3, 7b y 9 no son *FormTH* ya que no hay ningún *FormTH* con $\bullet$ como operador principal. 5 y 11 tampoco, ya que no hay ningún *FormTH* con $\leftarrow$ como operador principal. 14b es una *FormTH* correcta, en la que podemos aplicar la hipótesis inductiva. 6 podría haberse comparado con $TH^+(\varphi \Rightarrow \psi)$, pero $(i \bullet i)$ no puede ser una *FormTH* por tener como operador principal un $\bullet$. 12 puede verse como $TH^+(\varphi \Rightarrow \psi)$, que a su vez es correcta, y ya exploramos su árbol.

14b de ser *FormD*, por definición es adecuada. De participar de la generación de alguna *FormD* por hipótesis inductiva, es adecuada también. 12 de ser *FormD*, es adecuada también. Ninguna otra fórmula es una *FormD*.

El resto que se generarían son identidades, con lo que queda probado el Corolario.

□

Lema A.3. *Sea R una rama de tableaux, y Γ las fórmulas iniciales del tableaux, si Γ no contiene $\to_2$ ni $\leftarrow_2$, R tampoco las contiene.*

Podemos convencernos de la veracidad del Lema A.3 observando las reglas del Tableaux correspondiente. En los divisores de las reglas, nunca hay un $\to_2$ ni $\leftarrow_2$, por lo que al aplicar cualquier regla, sólo ocurrirían si están en una anterior, cosa que supusimos que no pasa.

Ahora sí, ya estamos listos para demostrar el Lema 7.5.

Demostración del Lema 7.5. Veamos por inducción en la cantidad de pasos necesarios para derivar una fórmula φ en R, que si $\varphi = @_a(b \bullet_2 c) : T \in R$, entonces $@_c(c \bullet_1 c) : T \in R$, y que no existe asignación g tal que al aplicarle g a una fórmula de la rama distinta de $@_a(b \bullet_2 c)$, contenga como subfórmula propia a $b \bullet_2 c$.

Nuestro caso base serán las formulas iniciales del tableaux, $@_a((i \wedge TH^+(\alpha)) \wedge (i \bullet i)) : T$ y $@_a f : F$

Ninguna de estas dos fórmulas es de la forma de una relación ni puede tener (con algún reemplazo de variables por nominales) como subfórmula propia a $b \bullet_2 c$.

Supongamos entonces que la hipótesis es cierta para todas las anteriores fórmulas en la derivación, y veremos que se cumple para la fórmula actual, mirando la última regla aplicada.

Por el Lema A.3, las reglas $\rightarrow^{+}$, $\rightarrow^{-}$, $\leftarrow^{+}$ y $\leftarrow^{-}$ no pueden aplicarse ya que en las ya que las fórmulas iniciales de R no contienen $\leftarrow_2$ ni $\rightarrow_2$, por lo que no podemos encontrarnos en R ninguno de estos operadores.

Dado que como ninguna de la rama puede contener $b \bullet_2 c$ como subfórmula propia, salvo quizás con un reemplazo de variables por nominales, las reglas $@^{+}$, $@^{-}$, $\bullet^{-}$, $\downarrow^{+}$, $\downarrow^{-}$, y NOM no pueden generar algo que no cumpla la hipótesis.

Si fue generado por la regla R_2, entonces se generó $@_a(b \bullet_2 c) : T$ a partir de $@_a(b' \bullet_2 c) : T$, pero como esta última pertenecía a la rama, por hipótesis inductiva $@_c(c \bullet_1 c) : T$ ya pertenecía a la rama.

Si se generó por la regla ID, ocurre algo similar al generarse $@_a(b \bullet_2 c) : T$ a partir de un $@'_a(b \bullet_2 c) : T$, en cuyo caso debía pertenecer $@_c(c \bullet_1 c) : T$ a su vez a la rama.

Si fue la regla R_3 que produjo $@_a(b \bullet_2 c) : T$, fue a partir de fórmulas $@_a(b \bullet_2 c') : T$ y $@'_c c : T$. Como $@_a(b \bullet_2 c') : T$ estaba en la rama, $@'_c(c' \bullet_1 c') : T$ debe pertenecer también por hipótesis inductiva. Analicemos entonces algunas fórmulas que también deben pertenecer a la rama, ya que R está saturada.

$$\begin{array}{rll}
1. & @'_c c : T & \\
2. & @'_c(c' \bullet_1 c') : T & \\
3. & @_c(c' \bullet_1 c') : T & \succ(\text{ID}) \text{ en 1 y 2} \\
4. & @_c(c \bullet_1 c') : T & \succ(R_2) \text{ en 1 y 3} \\
5. & @_c(c \bullet_1 c) : T & \succ(R_3) \text{ en 1 y 4}
\end{array}$$

Por lo que $@_c(c \bullet_1 c) : T$ debe también pertenecer a la rama.

Finalmente, analizaremos el caso de si fue generado $@_a(b \bullet_2 c) : T$ a partir de la aplicación de la regla $\bullet^{+}$ en un $\beta = @_a(\varphi \bullet_2 \psi) : T$ perteneciente R. Separemos en dos casos: de ser tanto φ como ψ nominales, digamos que $\varphi = b'$, $\psi = c'$, y por hipótesis inductiva $@'_c(c' \bullet_1 c') : T$ ya estaba en R, y con el mismo análisis del caso anterior, $@_c(c \bullet_1 c) : T$ también lo está.

De ser alguno de los dos distinto de nominal, y sabiendo que ninguno puede tampoco ser una variable por no haber comenzado con ninguna fórmula que contenga variables libres, por definición β está en *FormD*. Por el Corolario A.2, sabemos que entonces es adecuada, con lo que coincide con una *FormTH*. Por ser una *FormTH* con operador principal $\bullet_2$, puede ser tanto $@_a\, TH^{+}(\Diamond\delta) : T$ como $@_a\, TH^{-}(\Diamond\delta) : T$, con δ *FBF* de $\mathcal{H}(\downarrow)$. Como es una *FormTH*, el Corolario A.2 también nos dice que es correcta, por lo que β tiene que ser $@_a\, TH^{+}(\Diamond\delta) : T$. Veamos que ocurre en este caso.

$$\begin{array}{rll}
1. & @_a(i \bullet_2 \downarrow x.((x \bullet x) \wedge TH^{+}(\delta))) : T & \\
2. & @_a(b \bullet_2 c) : T & \succ(\bullet^{+}) \text{ en 1} \\
3. & @_b i : T & \succ(\bullet^{+}) \text{ en 1} \\
4. & @_c \downarrow x.((x \bullet x) \wedge TH^{+}(\delta)) : T & \\
5. & @_c(c \bullet c) \wedge TH^{+}(\delta) : T & \succ(\downarrow^{+}) \text{ en 4} \\
6. & @_c(c \bullet c) : T & \succ(\text{Lema 7.2}) \text{ en 5} \\
7. & @_c\, TH^{+}(\delta) : T & \succ(\text{Lema 7.2}) \text{ en 5} \\
8. & @_a(i \bullet_2 c) : T & \succ(R_2) \text{ en 2 y 3}
\end{array}$$

Como vemos, cuando se aplica la regla $\bullet^{+}$ es el momento en que se genera un $@_a(b \bullet_2 c) : T$ "nuevo". Podemos ver que también en la rama debe estar $@_c(c \bullet c) : T$, haciendo valer la hipótesis. Además se cumplen las condiciones para aplicar la hipótesis inductiva en $@_c \downarrow x.((x \bullet x) \wedge TH^{+}(\delta)) : T$, ya que no existe asignación alguna tal que al aplicársele a una subfórmula propia contenga a $d \bullet_2 e$, por ser subfórmula de la anterior.

$\square$

Bibliografía

[Areces and Bernardi, 2004] C. Areces and R. Bernardi. Analyzing the core of categorial grammar. *Journal of Logic, Language and Information*, 13(2):121–137, 2004.

[Areces and de Rijke, 2001] C. Areces and M. de Rijke. From description to hybrid logics, and back. In F. Wolter, H. Wansing, M. de Rijke, and M. Zakharyaschev, editors, *Advances in Modal Logic. Volume 3*. CSLI Publications, 2001.

[Areces and Gorín, 2005] C. Areces and D. Gorín. Ordered resolution with selection for $\mathcal{H}(@)$. In F. Baader and A. Voronkov, editors, *Proceedings of LPAR 2004*, volume 3452 of *LNCS*, pages 125–141, Montevideo, Uruguay, 2005. Springer.

[Areces and ten Cate, 2006] C. Areces and B. ten Cate. Hybrid logics. In P. Blackburn, F. Wolter, and J. van Benthem, editors, *Handbook of Modal Logics*. Elsevier, 2006.

[Areces *et al.*, 1999] C. Areces, P. Blackburn, and M. Marx. A road-map on complexity for hybrid logics. In J. Flum and M. Rodríguez-Artalejo, editors, *Computer Science Logic*, number 1683 in LNCS, pages 307–321, Madrid, Spain, 1999. Springer. Proceedings of the 8th Annual Conference of the EACSL, Madrid, September 1999.

[Areces *et al.*, 2001] C. Areces, R. Bernardi, and M. Moortgat. Galois connections in categorial type logic. In *Proceedings of Mathematics of Language and Formal Grammar*, Helsinki, Finland, Augustus 2001.

[Areces *et al.*, 2008] C. Areces, A. Koller, and K. Striegnitz. Referring expressions as formulas of description logic. In *Proceedings of the 5th International Natural Language Generation*, Salt Fork, OH, USA, 2008.

[Baader *et al.*, 2003] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[Bacchus and Kabanza, 2000] F. Bacchus and F. Kabanza. Using temporal logics to express search control knowledge for planning. *Artificial Intelligence*, 116, 2000.

[Bell, 2005] J. Bell. The development of categorical logic, 2005.

[Birkhoff, 1967] G. Birkhoff. *Lattice theory*, volume 25. 1967.

[Blackburn and Seligman, 1995] P. Blackburn and J. Seligman. Hybrid languages. *Journal of Logic, Language and Information*, 4:251–272, 1995.

[Blackburn and Seligman, 1998] P. Blackburn and J. Seligman. What are hybrid languages? In M. Kracht, M. de Rijke, H. Wansing, and M. Zakharyaschev, editors, *Advances in Modal Logic*, volume 1, pages 41–62. CSLI Publications, Stanford University, 1998.

[Blackburn and Tzakova, 1999] P. Blackburn and M. Tzakova. Hybrid languages and temporal logic. *Logic Journal of the IGPL*, 7(1):27–54, 1999.

[Blackburn *et al.*, 2002] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2002.

[Blackburn, 2000a] P. Blackburn. Discipline as logic: treating labels as first class citizens. In *Labelled deduction*, pages 81–105, Norwell, MA, USA, 2000. Kluwer Academic Publishers.

[Blackburn, 2000b] P. Blackburn. Internalizing labelled deduction. *Journal of Logic and Computation*, 10(1):137–168, 2000.

[Blyth and Janowitz, 1872] T. Blyth and F. Janowitz. *Residuation Theory.* New York, 1872.

[Bochenski, 1961] I. Bochenski. *A history of formal logic / Translated from the German edition of 1956 and edited by T. Ivo.* University of Notre Dame Press, Notre Dame, 1961.

[Church, 1936] A. Church. A note on the entscheidungsproblem. *Journal of Symbolic Logic*, 1(1):40–41, 1936.

[Cook, 1971] S. Cook. The complexity of theorem-proving procedures. In *STOC '71: Proceedings of the third annual ACM symposium on theory of computing*, pages 151–158, New York, NY, USA, 1971. ACM Press.

[D'Agostino *et al.*, 1999] Marcello D'Agostino, Dov M. Gabbay, Reiner Hähnle, and Joachim Posegga, editors. *Handbook of Tableau Methods*. Springer, 1999.

[Davis and Putnam, 1960] M. Davis and H. Putnam. A computing procedure for quantification theory. *Journal of ACM*, 7(3):201–215, 1960.

[de Groote, 1999] P. de Groote. The non-associative lambek calculus with product in polynomial time. In *TABLEAUX '99: Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*, pages 128–139, London, UK, 1999. Springer-Verlag.

[Dosen, 1992] K. Dosen. A brief survey of frames for the lambek calculus. 38:179–187, 1992.

[Fischer and Ladner, 1979] M. J. Fischer and R. Ladner. Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences*, 18(2):194–211, 1979.

[Goldblatt, 2001] R. Goldblatt. Mathematical modal logic: A view of its evolution. Draft, written for A History of Mathematical Logic", 2001.

[Goldblatt, 2003] R. Goldblatt. Mathematical modal logic: a view of its evolution. *Journal of Applied Logic*, 1(5-6):309–392, 2003.

[Gorín, 2004] D. Gorín. Tesis de licenciatura: Resolución con orden y selección para la lógica $\mathcal{H}(@)$. *Departamento de computación, FCEyN, UBA*, 2004.

[Kripke, 1959] S. Kripke. A completeness theorem in modal logic. *Journal of Symbolic Logic*, 24:1–14, 1959.

[Kripke, 1963a] S. Kripke. Semantic analysis of modal logics I, normal propositional calculi. *Zeitschrift fürr mathemathische Logik und Grundlagen der Mathematik*, 9:67–96, 1963.

[Kripke, 1963b] S. Kripke. Semantical consideration on modal logics. *Acta Philosophica Fennica*, 16:83–94, 1963.

[Kurtonina, 1995] N. Kurtonina. *Frames and Labels. A Modal Analysis of Categorial Inference.* PhD thesis, Research Institute for Language and Speech, University of Utrecht., 1995.

[Ladner, 1977] R. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM Journal on Computing*, 6(3):467–480, 1977.

[Lambek and Scott, 1986] J. Lambek and P. J. Scott. *Introduction to higher order categorical logic*. Cambridge University Press, New York, NY, USA, 1986.

[Lambek, 1968] J. Lambek. Deductive systems and categories i. syntactic calculus and residuated categories. *Mathematical Systems Theory*, 2(4):287–318, 1968.

[Lewis and Langford, 1932] C. Lewis and C. Langford. *Symbolic Logic*. The Century Co, New York and London, 1932.

[Lewis, 1918] C. Lewis. *A Survey of Symbolic Logic*. Univ. of California Press, Berkeley, 1918. Reprint of Chapters I–IV by Dover Publications, 1960, New York.

[Manna and Pnueli, 1992] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems - Specification*. Springer-Verlag, 1992.

[Marx, 2002] M. Marx. Narcissists, stepmothers and spies. In *In Proceedings of DL'02, volume 53. CEUR*, 2002.

[Moortgat, 1996] M. Moortgat. Multimodal linguistic inference. *Journal of Logic, Language and Information*, 5(3/4):349–385, 1996.

[Mundhenk and Schneider, 2007] M. Mundhenk and T. Schneider. Undecidability of multi-modal hybrid logics. *Electronic Notes in Theoretical Computer Science*, 174(6):29–43, 2007.

[Ortiz, 2006] M. Ortiz. A fully internalized sequent calculus for hybrid categorial logics. In *Proceedings of the Student Session of the 18th European Summer School of Logic, Language and Information (ESSLLI'06)*, Malaga, Spain, 2006. Janneke Huitink and Sophia Katrenko. ortiz@kr.tuwien.ac.at.

[Pentus, 2006] M. Pentus. Lambek calculus is np-complete. *Theoretical Computer Science*, 357(1):186–201, 2006.

[Prior, 1957] A. Prior. *Time and Modality*. Oxford Unviersity Press, 1957.

[Prior, 1967] A. Prior. *Past, Present and Future*. Oxford University Press, 1967.

[Prior, 1968] A. Prior. *Papers on time and tense*. Oxford UP, 1968.

[Robinson, 1965] J. Robinson. A machine-oriented logic based on the resolution principle. *Journal of ACM*, 12(1):23–41, 1965.

[Schneider, 2007] T. Schneider. *The Complexity of Hybrid Logics over Restricted Frame Classes*. PhD thesis, Department of Computer Science, University of Jena, 2007.

[Turing, 1936] A. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceeding of the London Mathematical Society, series 2*, 42:230–265, 1936.

[Vermaat, 2005] W. Vermaat. *The Logic of Variation*. PhD thesis, Utrecht, 2005.

[Whitehead and Russell, 1925] A. Whitehead and B. Russell. *Principia Mathematica*. Cambridge University Press, 1925.