Modal Subordination and Continuation Semantics

(CAuLD project)

Nicholas Asher[1] Sylvain Pogodalla[2]

[1]nicholas.asher@irit.fr
IRIT CNRS, Toulouse, France
[2]sylvain.pogodalla@loria.fr
LORIA/INRIA Nancy–Grand Est, France

CAuLD Workshop – Nancy, France
December 13–14, 2010

Outline

Epistemic Modal Facts Asymmetrically Depend on Non Modal Facts

Example

Epistemic modal facts depend on non modal facts, but not vice versa [Veltman(1996)]:

- *It might be sunny. But it's not sunny.*
- *#It's not sunny. But it might (for all I know) be sunny.*

Epistemic Modal Facts Asymmetrically Depend on Non Modal Facts

Example

Epistemic modal facts depend on non modal facts, but not vice versa [Veltman(1996)]:

- *It might be sunny. But it's not sunny.*
- *#It's not sunny. But it might (for all I know) be sunny.*

Update of Information State (Set of Worlds)

- Non modal sentences ϕ: $(s[\phi]) = \{w \in s | w \in ||\phi||\}$
- Modal sentences $\Diamond\phi$:$s[\Diamond\phi] = s$ iff $\exists w \in s \; w \in ||\phi||$

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.
- A wolf is outside. It growls.
- A wolf is outside. ?It would growl.
- A wolf is outside. It might growl.

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.
- A wolf is outside. It growls.
- A wolf is outside. ?It would growl.
- A wolf is outside. It might growl.

Example

- If John bought a book, he'll be home reading it by now. #It's a murder mystery.

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.
- A wolf is outside. It growls.
- A wolf is outside. ?It would growl.
- A wolf is outside. It might growl.

Example

- If John bought a book, he'll be home reading it by now. #It's a murder mystery.
- If John bought a book, he'll be home reading it by now. It'll be a murder mystery.

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.
- A wolf is outside. It growls.
- A wolf is outside. ?It would growl.
- A wolf is outside. It might growl.

Example

- If John bought a book, he'll be home reading it by now. #It's a murder mystery.
- If John bought a book, he'll be home reading it by now. It'll be a murder mystery.
- If John's at home he'll be reading a book. Actually, he's still at the office. #It'll be War and Peace.

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.
- A wolf is outside. It growls.
- A wolf is outside. ?It would growl.
- A wolf is outside. It might growl.

Example

- If John bought a book, he'll be home reading it by now. #It's a murder mystery.
- If John bought a book, he'll be home reading it by now. It'll be a murder mystery.
- If John's at home he'll be reading a book. Actually, he's still at the office. #It'll be War and Peace.
- If John's at home he'll be reading a book. He is. It's War and Peace.

Previous Accounts

Using DRT and Dynamic Frameworks

- Accommodation of DRSs [Roberts(1989)]
- Modals presuppose their domain [Geurts(1999)]
- Anaphoric context references and update of these with DRSs in the DRS syntax [Frank and Kamp(1997)]
- Compositional DRT extension [Stone and Hardt(1997)]
- Two-dimensionsal approach, accessibility relation and world ordering [van Rooij(2005)]
- DPL and sets of epistemic possibilities [Asher and McCready(2007)]

DRT Based Account

Example

A wolf might walk in.

DRT Based Account

Example

A wolf might walk in.

$$\begin{array}{|l|}\hline \\ \Diamond \begin{array}{|l|}\hline x \\ \hline \mathbf{wolf}(x) \\ \mathbf{enter}(x) \\ \hline \end{array} \\ \\ \\ \hline \end{array}$$

DRT Based Account

Example

A wolf might walk in.It would growl.

$$\Diamond \begin{array}{|l|}\hline x \\ \hline \textbf{wolf}(x) \\ \textbf{enter}(x) \\ \hline \end{array}$$

DRT Based Account

Example

A wolf might walk in.It would growl.

$$
\begin{array}{|l|}
\hline
\Diamond \begin{array}{|l|}\hline x \\ \hline \textbf{wolf}(x) \\ \textbf{enter}(x) \\ \hline\end{array} \\
\Box \begin{array}{|l|}\hline y \\ \hline \textbf{growl}(y) \\ \hline\end{array} \\
\hline
\end{array}
$$

DRT Based Account

Example

A wolf might walk in.It would growl.

$$\begin{array}{|l|}\hline \Diamond \begin{array}{|l|}\hline x \\ \hline \textbf{wolf}(x) \\ \textbf{enter}(x) \\ \hline \end{array} \\ \Box \begin{array}{|l|}\hline y \\ \hline \textbf{growl}(y) \\ \hline \end{array} \\ \hline \end{array}$$

Note:

- Accessibility conditions

DRT Based Account

Example

A wolf might walk in.It would growl.

$\Diamond$ [x | **wolf**(x), **enter**(x)]
$\Box$ [y | **growl**(y)]

$\Diamond$ [x | **wolf**(x), **enter**(x)]

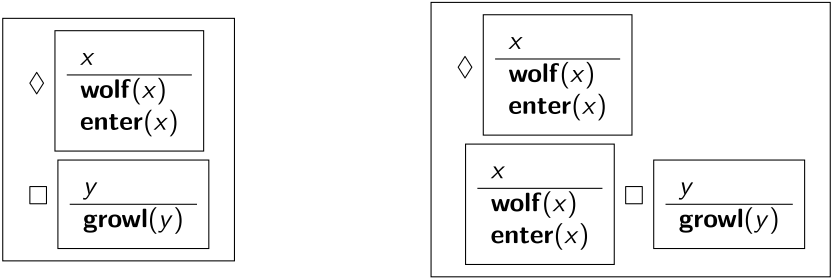
Note:

- Accessibility conditions

DRT Based Account

Example

A wolf might walk in.It would growl.



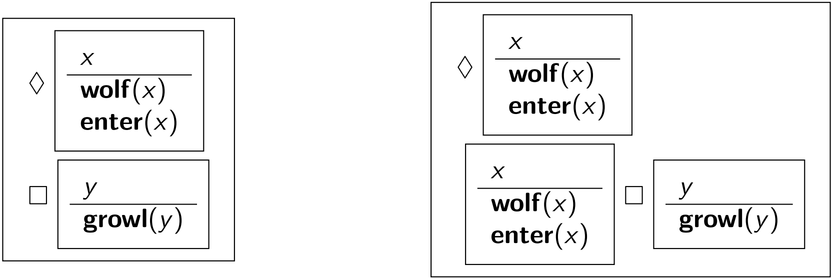
Note:

- Accessibility conditions

DRT Based Account

Example

A wolf might walk in.It would growl.



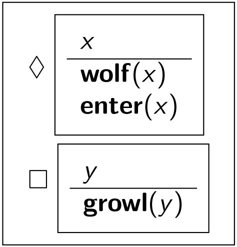
Note:

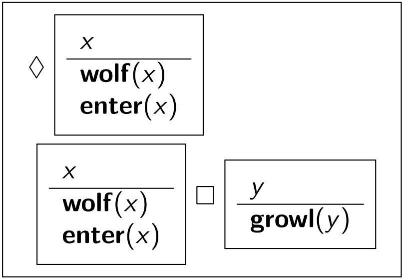
- Accessibility conditions
- Modal base and accommodation

DRT Based Account

Example

A wolf might walk in.It would growl.





Note:

- Accessibility conditions
- Modal base and accommodation
- No (explicit) modal logic in the interpretation

Compositional Interpretations

Methodological and Technical Issues

- Non-standard interpretation of formulas:
  - Interpretation as relations between pairs of worlds and assignment functions
  - $(\exists x.\phi) \wedge \psi \Leftrightarrow \exists x.(\phi \wedge \psi)$ (scope theorem)
- Destructive assignment and variable clash
- $[\![\phi \Rightarrow_{\diamond} \psi]\!] = ?$

Compositional Interpretations

Methodological and Technical Issues

- Non-standard interpretation of formulas:
  - Interpretation as relations between pairs of worlds and assignment functions
  - $(\exists x.\phi) \wedge \psi \Leftrightarrow \exists x.(\phi \wedge \psi)$ (scope theorem)
- Destructive assignment and variable clash
- $[\![\phi \Rightarrow_\diamond \psi]\!] = ?$

Formal Semanticist and Logician?

- What are the useful data to feed the context with? (entities, context referents. . .)
- How do discourse and sentences combine? (DRS merge, DPL relational composition, accessibility. . .)
- What are the semantic recipes of the lexical items
- Should I design a new logic?

Compositional Interpretations

Methodological and Technical Issues

- Non-standard interpretation of formulas:
  - Interpretation as relations between pairs of worlds and assignment functions
  - $(\exists x.\phi) \wedge \psi \Leftrightarrow \exists x.(\phi \wedge \psi)$ (scope theorem)
- Destructive assignment and variable clash
- $[\![\phi \Rightarrow_{\diamond} \psi]\!] = ?$

Formal Semanticist and Logician?

- What are the useful data to feed the context with? (entities, context referents...)
- How do discourse and sentences combine? (DRS merge, DPL relational composition, accessibility...)
- What are the semantic recipes of the lexical items
- Should I design a new logic? Continuation semantics

Continuation Semantics

Principles [de Groote(2006)]

$[\![s]\!]$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
&[\![s]\!] \\
&[\![np]\!] = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!]
\end{aligned}
$$

Continuation Semantics

Principles [de Groote(2006)]

$$\begin{array}{ll} [\![s]\!] & \\ [\![np]\!] & = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\ [\![n]\!] & = e \rightarrow [\![s]\!] \end{array}$$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
[\![s]\!] &= \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\
[\![np]\!] &= (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
[\![n]\!] &= e \rightarrow [\![s]\!]
\end{aligned}
$$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
[\![s]\!] &= \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\
[\![np]\!] &= (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
[\![n]\!] &= e \rightarrow [\![s]\!]
\end{aligned}
$$

Example

A man is sleeping.

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
[\![s]\!] &= \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\
[\![np]\!] &= (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
[\![n]\!] &= e \rightarrow [\![s]\!]
\end{aligned}
$$

Example

A man is sleeping.

$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

Continuation Semantics

Principles [de Groote(2006)]

$$\llbracket s \rrbracket = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \triangleq \Omega$$
$$\llbracket np \rrbracket = (e \rightarrow \llbracket s \rrbracket) \rightarrow \llbracket s \rrbracket$$
$$\llbracket n \rrbracket = e \rightarrow \llbracket s \rrbracket$$

Example

A man is sleeping. He is snoring.

$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

Continuation Semantics

Principles [de Groote(2006)]

$$\begin{array}{ll} [\![s]\!] & = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\ [\![np]\!] & = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\ [\![n]\!] & = e \rightarrow [\![s]\!] \end{array}$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
[\![s]\!] &= \gamma \to (\gamma \to t) \to t \quad \triangleq \Omega \\
[\![np]\!] &= (e \to [\![s]\!]) \to [\![s]\!] \\
[\![n]\!] &= e \to [\![s]\!]
\end{aligned}
$$

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i\ (\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$

$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))]\ (\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$
$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))]$$
$$(\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$
$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))]$$
$$(\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))$$
$$\rightarrow_\beta \quad \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))\ (x :: i))]$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i\ (\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$
$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))]\ (\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))$$
$$\rightarrow_\beta \quad \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))\ (x :: i))]$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$
$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))]$$
$$(\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))$$
$$\rightarrow_\beta \quad \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))\ (x :: i))]$$
$$\rightarrow_\beta \quad \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\mathbf{snoring}\ (\mathtt{sel}\ (x :: i)) \wedge (k\ (x :: i))))]$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$
$$\rightarrow_\beta\quad \lambda i\ k.[\lambda k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))]$$
$$(\lambda i'.(\textbf{snoring}\ (\texttt{sel}\ i')) \wedge (k\ i'))$$
$$\rightarrow_\beta\quad \lambda i\ k.[\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge ((\lambda i'.(\textbf{snoring}\ (\texttt{sel}\ i')) \wedge (k\ i'))\ (x :: i))]$$
$$\rightarrow_\beta\quad \lambda i\ k.[\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge ((\textbf{snoring}\ (\texttt{sel}\ (x :: i)) \wedge (k\ (x :: i))))]$$

The Basics

Context

$$
\begin{aligned}
i &: \gamma \triangleq t \\
k &: t \rightarrow t \\
:: &\; t \rightarrow t \rightarrow t
\end{aligned}
$$

The Basics

Context

$$\begin{array}{ll} i & : \gamma \stackrel{\Delta}{=} t \\ k & : t \rightarrow t \\ :: & t \rightarrow t \rightarrow t \end{array}$$

Logic

- for an atomic static formulas $p : t$: $\overline{p} = \lambda i\ k.p \wedge (k\ (p :: i))$

The Basics

Context

$$i \quad : \gamma \triangleq t$$
$$k \quad : t \rightarrow t$$
$$:: \quad t \rightarrow t \rightarrow t$$

Logic

- for an atomic static formulas $p : t$: $\overline{p} = \lambda i\ k.p \wedge (k\ (p :: i))$
- $\mathbb{T} \triangleq \lambda i.\top$ the trivial continuation

The Basics

Context

$$i \quad : \gamma \triangleq t$$
$$k \quad : t \rightarrow t$$
$$:: \quad t \rightarrow t \rightarrow t$$

Logic

- for an atomic static formulas $p : t$: $\overline{p} = \lambda i\ k.p \wedge (k\ (p :: i))$
- $\mathbb{T} \triangleq \lambda i.\top$ the trivial continuation
- for $P : \Omega$ a dynamic proposition:
  - $\neg_d P \triangleq \lambda i\ k.(\neg(P\ i\mathbb{T})) \wedge (k\ ((\neg P\ i\ \mathbb{T}) :: i))$

The Basics

Context

$i \quad : \gamma \triangleq t$
$k \quad : t \rightarrow t$
$:: \quad t \rightarrow t \rightarrow t$

Logic

- for an atomic static formulas $p : t$: $\overline{p} = \lambda i\ k.p \wedge (k\ (p :: i))$
- $\mathbb{T} \triangleq \lambda i.\top$ the trivial continuation
- for $P : \Omega$ a dynamic proposition:
  - $\neg_d P \triangleq \lambda i\ k.(\neg(P\ i \mathbb{T})) \wedge (k\ ((\neg P\ i\ \mathbb{T}) :: i))$
  - $P \wedge_d Q \triangleq \lambda i\ k.P\ i\ (\lambda i'.Q\ i'\ k)$

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\texttt{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception `Halt`

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\textsc{test}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception Halt
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\textsc{test}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception Halt
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is sunny

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\texttt{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception Halt
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is sunny $\qquad S = \lambda i\ k.\mathbf{sunny} \wedge (k\ (\mathbf{sunny} :: i))$

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\text{TEST}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\text{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception `Halt`
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is sunny $\quad S = \lambda i\ k.\mathbf{sunny} \wedge (k\ (\mathbf{sunny} :: i))$

(start) It might be sunny

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\textsc{test}\ P = \lambda i\ k.\mathtt{if}\ (\mathtt{EVAL}\ P\ i\ \mathbb{T})\ \mathtt{then}\ (k\ i)\ \mathtt{else}\ (\mathtt{raise\ Halt})$
- With the exception Halt
- $\mathtt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is sunny	$S = \lambda i\ k.\textbf{sunny} \wedge (k\ (\textbf{sunny} :: i))$
(start) It might be sunny	$(\Diamond_d S)\top = \lambda k.(\textsc{test}\ P)\ \top\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\textsc{test}\ P = \lambda i\ k.\mathtt{if}\ (\mathtt{EVAL}\ P\ i\ \mathbb{T})\ \mathtt{then}\ (k\ i)\ \mathtt{else}\ (\mathtt{raise\ Halt})$
- With the exception `Halt`
- $\mathtt{EVAL} : \Omega \to t \to k \to t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is sunny	$S = \lambda i\ k.\mathbf{sunny} \wedge (k\ (\mathbf{sunny} :: i))$
(start) It might be sunny	$(\Diamond_d S)\top = \lambda k.(\textsc{test}\ P)\ \top\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
$\mathtt{EVAL}\ S\ \top\ \mathbb{T}$	$= \top$

Veltman's Test

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\text{TEST}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
- $\text{TEST}\ P = \lambda i\ k.\mathtt{if}\ (\mathtt{EVAL}\ P\ i\ \mathbb{T})\ \mathtt{then}\ (k\ i)\ \mathtt{else}\ (\mathtt{raise\ Halt})$
- With the exception `Halt`
- $\mathtt{EVAL} : \Omega \to t \to k \to t$: returns $\top$ iff the theory made of $P\ \top\ \mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is sunny	$S = \lambda i\ k.\textbf{sunny} \wedge (k\ (\textbf{sunny} :: i))$
(start) It might be sunny	$(\Diamond_d S)\top = \lambda k.(\text{TEST}\ P)\ \top\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
$\mathtt{EVAL}\ S\ \top\ \mathbb{T}$	$= \top$
(start) It might be sunny	$(\Diamond_d S)\top = \lambda k.k\ \top \wedge (\Diamond(S\ \top\ \mathbb{T}))$

Veltman's Test (cont'd)

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\texttt{TEST}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.
- $\texttt{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception Halt
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is not sunny

Veltman's Test (cont'd)

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\texttt{TEST}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.
- $\texttt{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception Halt
- $\texttt{EVAL} : \Omega \to t \to k \to t$: returns $\top$ iff the theory made of $P\mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is not sunny $\quad S = \lambda i\ k.\neg\textbf{sunny} \wedge (k\ ((\neg\textbf{sunny}) :: i))$

Veltman's Test (cont'd)

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\text{TEST}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.
- $\texttt{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception `Halt`
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is not sunny $S = \lambda i\ k.\neg\textbf{sunny} \wedge (k\ ((\neg\textbf{sunny}) :: i))$

$(\neg\textbf{sunny}))$ It might be sunny $(\Diamond_d S)(\neg\textbf{sunny})$

Veltman's Test (cont'd)

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.
- $\textsc{test}\ P = \lambda i\ k.\mathtt{if}\ (\mathtt{EVAL}\ P\ i\ \mathbb{T})\ \mathtt{then}\ (k\ i)\ \mathtt{else}\ (\mathtt{raise\ Halt})$
- With the exception Halt
- $\mathtt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is not sunny (¬**sunny**)) It might be sunny

$S = \lambda i\ k.\neg\mathbf{sunny} \wedge (k\ ((\neg\mathbf{sunny}) :: i))$
$(\Diamond_d S)(\neg\mathbf{sunny})$
$= \lambda k.(\textsc{test}\ P)\ (\neg\mathbf{sunny})\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$

Veltman's Test (cont'd)

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\texttt{TEST}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.
- $\texttt{TEST}\ P = \lambda i\ k.\texttt{if}\ (\texttt{EVAL}\ P\ i\ \mathbb{T})\ \texttt{then}\ (k\ i)\ \texttt{else}\ (\texttt{raise Halt})$
- With the exception Halt
- $\texttt{EVAL} : \Omega \rightarrow t \rightarrow k \rightarrow t$: returns $\top$ iff the theory made of $P\mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is not sunny	$S = \lambda i\ k.\neg\mathbf{sunny} \wedge (k\ ((\neg\mathbf{sunny}) :: i))$
$(\neg\mathbf{sunny}))$ It might be sunny	$(\Diamond_d S)(\neg\mathbf{sunny})$
	$= \lambda k.(\texttt{TEST}\ P)\ (\neg\mathbf{sunny})\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
$\texttt{EVAL}\ S\ (\neg\mathbf{sunny})\ \mathbb{T}$	$= \bot$

Veltman's Test (cont'd)

Modal Logic

- $\Diamond_d P \triangleq \lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.
- $\textsc{test}\ P = \lambda i\ k.\mathtt{if}\ (\mathtt{EVAL}\ P\ i\ \mathbb{T})\ \mathtt{then}\ (k\ i)\ \mathtt{else}\ (\mathtt{raise\ Halt})$
- With the exception `Halt`
- $\mathtt{EVAL} : \Omega \to t \to k \to t$: returns $\top$ iff the theory made of $P\mathbb{T}$ and i is consistent (the conjunction can be satisfied), relative to any model

Example

It is not sunny	$S = \lambda i\ k.\neg\mathbf{sunny} \wedge (k\ ((\neg\mathbf{sunny}) :: i))$
$(\neg\mathbf{sunny})$) It might be sunny	$(\Diamond_d S)(\neg\mathbf{sunny})$
	$= \lambda k.(\textsc{test}\ P)\ (\neg\mathbf{sunny})\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$
$\mathtt{EVAL}\ S\ (\neg\mathbf{sunny})\ \mathbb{T}$	$= \bot$
$(\neg\mathbf{sunny})$) It might be sunny	$(\Diamond_d S)(\neg\mathbf{sunny}) \to_\beta \mathtt{raise\ Halt}$

Some Modal Facts Depend on other Modal Facts

Example

- A wolf might enter. It would growl.
- A wolf might enter. #It will growl.
- A wolf is outside. It growls.
- A wolf is outside. ?It would growl.
- A wolf is outside. It might growl.

Modals Depend on Modals: First Attempt

Lexicon

$$\begin{aligned}
[\![c_{enter}]\!] &= \lambda s.s\,(\lambda x\,i\,k.(\mathbf{enter}\,x) \wedge (k\,i)) \\
[\![c_{growl}]\!] &= \lambda s.s\,(\lambda x\,i\,k.(\mathbf{growl}\,x) \wedge (k\,i)) \\
[\![c_{wolf}]\!] &= \lambda x\,i\,k.(\mathbf{wolf}\,x) \wedge (k\,i) \\
[\![c_{a}]\!] &= \lambda P\,Q.\lambda i\,k.\exists x.P\,x\,(x :: i)\,(\lambda i'.Q\,x\,i'\,k) \\
[\![c_{it}]\!] &= \lambda P\,i\,k.P\,(\mathtt{sel}\,i)\,i\,k
\end{aligned}$$

Modals Depend on Modals: First Attempt

Lexicon

$$
\begin{aligned}
[\![c_{enter}]\!] &= \lambda s.s\,(\lambda x\,i\,k.(\mathbf{enter}\,x) \wedge (k\,i))\\
[\![c_{growl}]\!] &= \lambda s.s\,(\lambda x\,i\,k.(\mathbf{growl}\,x) \wedge (k\,i))\\
[\![c_{wolf}]\!] &= \lambda x\,i\,k.(\mathbf{wolf}\,x) \wedge (k\,i)\\
[\![c_{a}]\!] &= \lambda P\,Q.\lambda i\,k.\exists x.P\,x\,(x :: i)\,(\lambda i'.Q\,x\,i'\,k)\\
[\![c_{it}]\!] &= \lambda P\,i\,k.P\,(\mathtt{sel}\,i)\,i\,k
\end{aligned}
$$

Modals

$$
\begin{aligned}
[\![c_{might}]\!] &= \lambda v\,s.\lambda i\,k.\Diamond(v\,s\,i\,k) \qquad (\text{or } \lambda v\,s.\lambda i\,k.v\,s\,i\,(\lambda i'.\Diamond(k\,i')))\\
[\![c_{would}]\!] &= \lambda v\,s.\lambda i\,k.\Box(v\,s\,i\,k)
\end{aligned}
$$

Modals Depend on Modals: First Attempt

Lexicon

$$[\![c_{enter}]\!] = \lambda s.s\,(\lambda x\,i\,k.(\mathbf{enter}\,x) \wedge (k\,i))$$
$$[\![c_{growl}]\!] = \lambda s.s\,(\lambda x\,i\,k.(\mathbf{growl}\,x) \wedge (k\,i))$$
$$[\![c_{wolf}]\!] = \lambda x\,i\,k.(\mathbf{wolf}\,x) \wedge (k\,i)$$
$$[\![c_{a}]\!] = \lambda P\,Q.\lambda i\,k.\exists x.P\,x\,(x :: i)\,(\lambda i'.Q\,x\,i'\,k)$$
$$[\![c_{it}]\!] = \lambda P\,i\,k.P\,(\mathtt{sel}\,i)\,i\,k$$

Modals

$$[\![c_{might}]\!] = \lambda v\,s.\lambda i\,k.\Diamond(v\,s\,i\,k) \qquad (\text{or } \lambda v\,s.\lambda i\,k.v\,s\,i\,(\lambda i'.\Diamond(k\,i')))$$
$$[\![c_{would}]\!] = \lambda v\,s.\lambda i\,k.\Box(v\,s\,i\,k)$$

Example ($t_0 = c_{might}\ c_{enter}\ (c_a\ c_{wolf})$ and $t_1 = c_{would}\ c_{growl}\ c_{it}$)

$$[\![t_0]\!] = \lambda i\,k.\Diamond(\exists x.(\mathbf{wolf}\,x) \wedge ((\mathbf{enter}\,x) \wedge (k\,(x :: i))))$$
$$[\![t_1]\!] = \lambda i\,k.\Box((\mathbf{growl}\,(\mathtt{sel}\,i)) \wedge (k\,i))$$

Modals Depend on Modals: First Attempt

Lexicon

$$[\![c_{enter}]\!] = \lambda s.s\,(\lambda x\,i\,k.(\mathbf{enter}\,x) \wedge (k\,i))$$
$$[\![c_{growl}]\!] = \lambda s.s\,(\lambda x\,i\,k.(\mathbf{growl}\,x) \wedge (k\,i))$$
$$[\![c_{wolf}]\!] = \lambda x\,i\,k.(\mathbf{wolf}\,x) \wedge (k\,i)$$
$$[\![c_{a}]\!] = \lambda P\,Q.\lambda i\,k.\exists x.P\,x\,(x :: i)\,(\lambda i'.Q\,x\,i'\,k)$$
$$[\![c_{it}]\!] = \lambda P\,i\,k.P\,(\mathtt{sel}\,i)\,i\,k$$

Modals

$$[\![c_{might}]\!] = \lambda v\,s.\lambda i\,k.\Diamond(v\,s\,i\,k) \qquad (\text{or } \lambda v\,s.\lambda i\,k.v\,s\,i\,(\lambda i'.\Diamond(k\,i')))$$
$$[\![c_{would}]\!] = \lambda v\,s.\lambda i\,k.\Box(v\,s\,i\,k)$$

Example ($t_0 = c_{might}\ c_{enter}\ (c_a\ c_{wolf})$ and $t_1 = c_{would}\ c_{growl}\ c_{it}$)

$$[\![t_0]\!] = \lambda i\,k.\Diamond(\exists x.(\mathbf{wolf}\,x) \wedge ((\mathbf{enter}\,x) \wedge (k\,(x :: i))))$$
$$[\![t_1]\!] = \lambda i\,k.\Box((\mathbf{growl}\,(\mathtt{sel}\,i)) \wedge (k\,i))$$
$$[\![t_0\ .\ t_1]\!] = \lambda i\,k.\Diamond(\exists x.(\mathbf{wolf}\,x) \wedge ((\mathbf{enter}\,x) \wedge (\Box((\mathbf{growl}\,(\mathtt{sel}\,(x :: i))) \wedge (k\,(x :: i))$$

Problem

$$[\![t_0\ .\ t_1]\!] = \lambda i\, k.\Diamond(\exists x.(\mathbf{wolf}\; x) \wedge ((\mathbf{enter}\; x) \wedge (\Box((\mathbf{growl}\; (\mathtt{sel}\; (x :: i))) \wedge (k\; (x :: i)))))$$

If x is a wolf in the first accessible world but a tiger in all the next ones...

Introducing the Modal Base

The Basics

- The context is a record: $\gamma = \{\mathsf{m\_ref} : \gamma'; \mathsf{base} : t\}$

Introducing the Modal Base

The Basics

- The context is a record: $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t\}$
- $x ::_b i = \{\text{m\_ref} = x :: (i.\text{m\_ref}); \text{base} = i.\text{base}\}$
- $\mathtt{sel}_b\, i = \mathtt{sel}\, (i.\text{m\_ref})$
- $p \wedge_b i = \{\text{m\_ref} = (i.\text{m\_ref}); \text{base} = p \wedge (i.\text{base})\}$

Introducing the Modal Base

The Basics

- The context is a record: $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t\}$
- $x ::_b i = \{\text{m\_ref} = x :: (i.\text{m\_ref}); \text{base} = i.\text{base}\}$
- $\texttt{sel}_b\, i = \texttt{sel}\,(i.\text{m\_ref})$
- $p \wedge_b i = \{\text{m\_ref} = (i.\text{m\_ref}); \text{base} = p \wedge (i.\text{base})\}$

Lexicon

$$[\![c_{enter}]\!] = \lambda s.s\,(\lambda x\, i\, k.(\mathbf{enter}\, x) \wedge (k\,((\mathbf{enter}\, x) \wedge_b i)))$$

Introducing the Modal Base

The Basics

- The context is a record: $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t\}$
- $x ::_b i = \{\text{m\_ref} = x :: (i.\text{m\_ref}); \text{base} = i.\text{base}\}$
- $\texttt{sel}_b\, i = \texttt{sel}\,(i.\text{m\_ref})$
- $p \wedge_b i = \{\text{m\_ref} = (i.\text{m\_ref}); \text{base} = p \wedge (i.\text{base})\}$

Lexicon

$$
\begin{aligned}
[\![c_{enter}]\!] &= \lambda s.s\,(\lambda x\, i\, k.(\mathbf{enter}\, x) \wedge (k\,((\mathbf{enter}\, x) \wedge_b i))) \\
[\![c_{growl}]\!] &= \lambda s.s\,(\lambda x\, i\, k.(\mathbf{growl}\, x) \wedge (k\,((\mathbf{growl}\, x) \wedge_b i))) \\
[\![c_{wolf}]\!] &= \lambda x\, i\, k.(\mathbf{wolf}\, x) \wedge (k\,((\mathbf{wolf}\, x) \wedge_b i)) \\
[\![c_{a}]\!] &= \lambda P\, Q.\lambda i\, k.\exists x.P\, x\,(x :: i)\,(\lambda i'.Q\, x\, i'\, k) \\
[\![c_{it}]\!] &= \lambda P\, i\, k.P\,(\texttt{sel}_b\, i)\, i\, k
\end{aligned}
$$

Introducing the Modal Base

The Basics

- The context is a record: $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t\}$
- $x ::_b i = \{\text{m\_ref} = x :: (i.\text{m\_ref}); \text{base} = i.\text{base}\}$
- $\texttt{sel}_b\, i = \texttt{sel}\,(i.\text{m\_ref})$
- $p \wedge_b i = \{\text{m\_ref} = (i.\text{m\_ref}); \text{base} = p \wedge (i.\text{base})\}$

Lexicon

$$
\begin{aligned}
[\![c_{enter}]\!] &= \lambda s.s\,(\lambda x\, i\, k.(\mathbf{enter}\, x) \wedge (k\,((\mathbf{enter}\, x) \wedge_b i))) \\
[\![c_{growl}]\!] &= \lambda s.s\,(\lambda x\, i\, k.(\mathbf{growl}\, x) \wedge (k\,((\mathbf{growl}\, x) \wedge_b i))) \\
[\![c_{wolf}]\!] &= \lambda x\, i\, k.(\mathbf{wolf}\, x) \wedge (k\,((\mathbf{wolf}\, x) \wedge_b i)) \\
[\![c_{a}]\!] &= \lambda P\, Q.\lambda i\, k.\exists x.P\, x\,(x :: i)\,(\lambda i'.Q\, x\, i'\, k) \\
[\![c_{it}]\!] &= \lambda P\, i\, k.P\,(\texttt{sel}_b\, i)\, i\, k \\
[\![c_{might}]\!] &= \lambda v\, s.\lambda i\, k.\Diamond(i.\text{base} \Rightarrow v\, s\, i\, k)
\end{aligned}
$$

A wolf might walk in. It would growl

Example

$$[\![t_0]\!] = \lambda i\, k.\Diamond(i.\text{base} \wedge \exists x.(\textbf{wolf}\ x) \wedge ((\textbf{enter}\ x) \wedge (k\ ((\textbf{enter}\ x) \wedge_b (\textbf{wolf}\ x) \wedge_b (x ::_b i)))))$$

A wolf might walk in. It would growl

Example

$$[\![t_0]\!] = \lambda i\, k.\Diamond(i.\text{base} \wedge \exists x.(\mathbf{wolf}\; x) \wedge ((\mathbf{enter}\; x) \wedge (k\; ((\mathbf{enter}\; x) \wedge_b (\mathbf{wolf}\; x) \wedge_b (x ::_b i)))))$$

$$[\![t_1]\!] = \lambda i\, k.\Box(i.\text{base} \Rightarrow (\mathbf{growl}\,(\mathtt{sel}_b\, i)) \wedge (k\; ((\mathbf{growl}\,(\mathtt{sel}_b\, i)) \wedge_b i)))$$

A wolf might walk in. It would growl

Example

$$[\![t_0]\!] = \lambda i\,k.\Diamond(i.\text{base} \wedge \exists x.(\mathbf{wolf}\,x) \wedge ((\mathbf{enter}\,x) \wedge(k\,((\mathbf{enter}\,x) \wedge_b (\mathbf{wolf}\,x) \wedge_b (x ::_b i)))))$$

$$[\![t_1]\!] = \lambda i\,k.\Box(i.\text{base} \Rightarrow (\mathbf{growl}\,(\mathtt{sel}_b\,i)) \wedge (k\,((\mathbf{growl}\,(\mathtt{sel}_b\,i)) \wedge_b i)))$$

$$[\![t_0\ .\ t_1]\!] = \lambda ik.(\Diamond(i.\text{base} \wedge \exists x.(\mathbf{wolf}\,x) \wedge ((\mathbf{enter}\,x) \wedge(\Box((((\mathbf{enter}\,x) \wedge_b (\mathbf{wolf}\,x) \wedge_b (x ::_b i))).\text{base} \Rightarrow (\mathbf{growl}\,(\mathtt{sel}_b\,(x ::_b i) \wedge(k\,((\mathbf{growl}\,(\mathtt{sel}\,(x :: i.\text{m\_ref}))) \wedge_b ((\mathbf{enter}\,x) \wedge_b (\mathbf{wolf}\,x) \wedge_b (x ::_b i)))$$

With empty $= \{\text{m\_ref} = \mathtt{nil}; \text{base} = \top\}$ and $\mathbb{T} = (\lambda i.\top)$:

$$[\![t_0\ .\ t_1]\!]\,\text{empty}\,\mathbb{T} = \Diamond(\top \wedge \exists x.(\mathbf{wolf}\,x) \wedge ((\mathbf{enter}\,x) \wedge(\Box(((\mathbf{enter}\,x) \wedge (\mathbf{wolf}\,x)) \Rightarrow (\mathbf{growl}\,(\mathtt{sel}\,((x :: \mathtt{nil}))))))))$$

Models

Possible axioms on the models:

- Secondary reflexivity (so that the $\Box$ claim also holds of the world selected by *might*)
- Euclidean alternativeness relation (so that **growl** x also holds in all epistemic possibilities)
- Or replace $\Rightarrow$ by $\wedge$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow (\gamma \rightarrow t) \rightarrow (t \rightarrow t \rightarrow t) \rightarrow t \triangleq \Omega$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \to (\gamma \to t) \to (\gamma \to t) \to (t \to t \to t) \to t \stackrel{\Delta}{=} \Omega$
- $[\![S_1 \; . \; S_2]\!] = \lambda i\ k_1\ k_2\ f.[\![S_1]\!]\ i\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2 \Pi_1)\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2 \Pi_2)\ f$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow (\gamma \rightarrow t) \rightarrow (t \rightarrow t \rightarrow t) \rightarrow t \triangleq \Omega$
- $[\![S_1 \,.\, S_2]\!] = \lambda i\ k_1\ k_2\ f.[\![S_1]\!]\ i\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2 \Pi_1)\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2 \Pi_2)\ f$

lexicon

$$[\![c_a^m]\!] = \lambda P\,Q.\lambda i\,k_1\ k_2\ f.f(\exists x.P\,x\,\{i\ \texttt{with}\ \text{m\_ref} = x :: i.\text{m\_ref}\}\ (\lambda i'.Q\,x\,i'\,k_1\ k_2\ \Pi_1)\ k_2 \Pi_1)\ (k_2\ i)$$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \to (\gamma \to t) \to (\gamma \to t) \to (t \to t \to t) \to t \triangleq \Omega$
- $[\![S_1 \ . \ S_2]\!] = \lambda i \ k_1 \ k_2 \ f.[\![S_1]\!] \ i \ (\lambda i'.[\![S_2]\!] \ i' \ k_1 \ k_2 \Pi_1) \ (\lambda i'.[\![S_2]\!] \ i' \ k_1 \ k_2 \Pi_2) \ f$

lexicon

$$[\![c_a^m]\!] = \lambda P\, Q.\lambda i\, k_1 \ k_2 \ f.f(\exists x.P\, x\, \{i \ \texttt{with} \ \text{m\_ref} = x :: i.\text{m\_ref}\} \ (\lambda i'.Q\, x\, i' \, k_1 \ k_2 \ \Pi_1) \ k_2 \Pi_1) \ (k_2 \ i)$$

$$[\![c_a^f]\!] = \lambda P\, Q.\lambda i\, k_1 \ k_2 \ f.\exists x.f[k_1 \ \{i \ \texttt{with} \ \text{f\_ref} = x :: i\}] \ [P\, x\, \{i \ \texttt{with} \ \text{f\_ref} = x :: i\}\, k_1\, (\lambda i'.Q\, x\, i'\, k_1\, k_2 \Pi_2) \ \Pi_2]$$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \to (\gamma \to t) \to (\gamma \to t) \to (t \to t \to t) \to t \triangleq \Omega$
- $[\![S_1 \ . \ S_2]\!] = \lambda i\ k_1\ k_2\ f.[\![S_1]\!]\ i\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2 \Pi_1)\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2 \Pi_2)\ f$

lexicon

$$[\![c_a^m]\!] = \lambda P\,Q.\lambda i\,k_1\ k_2\ f.f(\exists x.P\,x\,\{i\ \texttt{with}\ \text{m\_ref} = x :: i.\text{m\_ref}\}\ (\lambda i'.Q\,x\,i'\,k_1\ k_2\ \Pi_1)\ k_2 \Pi_1)\ (k_2\ i)$$

$$[\![c_a^f]\!] = \lambda P\,Q.\lambda i\,k_1\ k_2\ f.\exists x.f[k_1\ \{i\ \texttt{with}\ \text{f\_ref} = x :: i\}]\ [P\,x\,\{i\ \texttt{with}\ \text{f\_ref} = x :: i\}\,k_1\,(\lambda i'.Q\,x\,i'\,k_1\,k_2\Pi_2)\ \Pi_2]$$

$$[\![c_{it}^m]\!] = \lambda P\,i\,k_1\ k_2\ f.P\,(\texttt{sel}_b\,i.\text{m\_ref} \cup i.\text{f\_ref})\,i\,k_1\ k_2\ f$$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow (\gamma \rightarrow t) \rightarrow (t \rightarrow t \rightarrow t) \rightarrow t \stackrel{\Delta}{=} \Omega$
- $[\![S_1 \ . \ S_2]\!] = \lambda i \ k_1 \ k_2 \ f.[\![S_1]\!] \ i \ (\lambda i'.[\![S_2]\!] \ i' \ k_1 \ k_2 \Pi_1) \ (\lambda i'.[\![S_2]\!] \ i' \ k_1 \ k_2 \Pi_2) \ f$

lexicon

$$
\begin{aligned}
[\![c_a^m]\!] &= \lambda P\, Q.\lambda i\, k_1 \ k_2 \ f.f(\exists x.P\, x\, \{i \ \texttt{with} \ \text{m\_ref} = x :: i.\text{m\_ref}\} \\
&\qquad (\lambda i'.Q\, x\, i'\, k_1 \ k_2 \ \Pi_1) \ k_2 \Pi_1) \ (k_2 \ i) \\
[\![c_a^f]\!] &= \lambda P\, Q.\lambda i\, k_1 \ k_2 \ f.\exists x.f[k_1 \ \{i \ \texttt{with} \ \text{f\_ref} = x :: i\}] \\
&\qquad [P\, x\, \{i \ \texttt{with} \ \text{f\_ref} = x :: i\}\, k_1\, (\lambda i'.Q\, x\, i'\, k_1\, k_2 \Pi_2) \ \Pi_2] \\
[\![c_{it}^m]\!] &= \lambda P\, i\, k_1 \ k_2 \ f.P\, (\texttt{sel}_b\, i.\text{m\_ref} \cup i.\text{f\_ref})\, i\, k_1 \ k_2 \ f \\
[\![c_{it}^f]\!] &= \lambda P\, i\, k_1 \ k_2 \ f.P\, (\texttt{sel}_b\, i.\text{f\_ref})\, i\, k_1 \ k_2 f
\end{aligned}
$$

Interactions Between Actual and Modal Contexts

The Basics

- $\gamma = \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'\}$
- Two continuations: $[\![s]\!] = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow (\gamma \rightarrow t) \rightarrow (t \rightarrow t \rightarrow t) \rightarrow t \stackrel{\Delta}{=} \Omega$
- $[\![S_1\ .\ S_2]\!] = \lambda i\ k_1\ k_2\ f.[\![S_1]\!]\ i\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2\Pi_1)\ (\lambda i'.[\![S_2]\!]\ i'\ k_1\ k_2\Pi_2)\ f$

lexicon

$$
\begin{aligned}
[\![c_a^m]\!] &= \lambda P\, Q.\lambda i\, k_1\ k_2\ f.f(\exists x.P\, x\, \{i\ \texttt{with}\ \text{m\_ref} = x :: i.\text{m\_ref}\} \\
&\qquad (\lambda i'.Q\, x\, i'\, k_1\ k_2\ \Pi_1)\ k_2\Pi_1)\ (k_2\ i) \\
[\![c_a^f]\!] &= \lambda P\, Q.\lambda i\, k_1\ k_2\ f.\exists x.f[k_1\ \{i\ \texttt{with}\ \text{f\_ref} = x :: i\}] \\
&\qquad [P\, x\, \{i\ \texttt{with}\ \text{f\_ref} = x :: i\}\, k_1\, (\lambda i'.Q\, x\, i'\, k_1\, k_2\Pi_2)\ \Pi_2] \\
[\![c_{it}^m]\!] &= \lambda P\, i\, k_1\ k_2\ f.P\, (\texttt{sel}_b\, i.\text{m\_ref} \cup i.\text{f\_ref})\, i\, k_1\ k_2\ f \\
[\![c_{it}^f]\!] &= \lambda P\, i\, k_1\ k_2\ f.P\, (\texttt{sel}_b\, i.\text{f\_ref})\, i\, k_1\ k_2 f \\
[\![c_{might}]\!] &= \lambda v\, s.\lambda i\, k_1\ k_2\ f.f(\Diamond(i.\text{base} \wedge (v\, s\, i\, k_1\ k_2\Pi_1)))\ (k_2\ i)
\end{aligned}
$$

Examples

Example ($t_2 = c_{will}\ c_{growl}\ c_{it}$)

$$[\![t_0]\!] = \lambda i\, k_1\ k_2 f.f[\Diamond(i.\mathsf{base} \wedge \exists x.(\mathbf{wolf}\ x) \wedge ((\mathbf{enter}\ x) \wedge(k_1\ \{i\ \mathtt{with}\ \mathsf{m\_ref} = x :: i.\mathsf{m\_ref}\ \mathtt{and}\ \mathsf{base} = (\mathbf{wolf}\ x) \wedge (\mathbf{enter}\ x) \wedge i.\mathsf{base}\})))$$

Examples

Example ($t_2 = c_{will}\ c_{growl}\ c_{it}$)

$$\llbracket t_0 \rrbracket = \lambda i\, k_1\ k_2 f.f[\Diamond(i.\mathsf{base} \wedge \exists x.(\mathbf{wolf}\ x) \wedge ((\mathbf{enter}\ x)$$
$$\wedge(k_1\ \{i\ \mathtt{with}\ \mathsf{m\_ref} = x :: i.\mathsf{m\_ref}\ \mathtt{and}\ \mathsf{base} = (\mathbf{wolf}\ x) \wedge (\mathbf{enter}\ x) \wedge i.\mathsf{base}\}))$$
$$\llbracket t_1 \rrbracket = \lambda i\, k_1\ k_2 f.f[\Box(i.\mathsf{base} \Rightarrow ((\mathbf{growl}\ (\mathtt{sel}\ i.\mathsf{m\_ref} \cup i.\mathsf{f\_ref}))$$
$$\wedge(k_1\ \{i\ \mathtt{with}\ \mathsf{base} = (\mathbf{growl}\ (\mathtt{sel}\ i.\mathsf{m\_ref} \cup i.\mathsf{f\_ref})) \wedge i.\mathsf{base}\})))]\ [k_2\ i]$$

Examples

Example ($t_2 = c_{will}\ c_{growl}\ c_{it}$)

$$\begin{aligned} [\![t_0]\!] &= \lambda i\, k_1\ k_2 f.f[\Diamond(i.\mathrm{base} \wedge \exists x.(\mathbf{wolf}\ x) \wedge ((\mathbf{enter}\ x) \\ &\quad \wedge(k_1\ \{i\ \mathtt{with}\ \mathrm{m\_ref} = x :: i.\mathrm{m\_ref}\ \ \mathtt{and}\ \ \mathrm{base} = (\mathbf{wolf}\ x) \wedge (\mathbf{enter}\ x) \wedge i.\mathrm{base}\}))) \\ [\![t_1]\!] &= \lambda i\, k_1\ k_2 f.f[\Box(i.\mathrm{base} \Rightarrow ((\mathbf{growl}\ (\mathtt{sel}\ i.\mathrm{m\_ref} \cup i.\mathrm{f\_ref})) \\ &\quad \wedge(k_1\ \{i\ \mathtt{with}\ \mathrm{base} = (\mathbf{growl}\ (\mathtt{sel}\ i.\mathrm{m\_ref} \cup i.\mathrm{f\_ref})) \wedge i.\mathrm{base}\})))]\ [k_2\ i] \\ [\![t_2]\!] &= \lambda i\, k_1\ k_2 f.f[k_1\ i]\ [(\mathbf{growl}\ (\mathtt{sel}\ i.\mathrm{f\_ref})) \wedge (k_2\ i)] \end{aligned}$$

Examples

Example (*A wolf might enter. It would growl*)

$$\begin{aligned}\llbracket t_0 \ . \ t_1 \rrbracket \ \text{empty} \ \mathbb{T} \ \mathbb{T} \ \text{Conj} \quad = [\Diamond(\top \wedge (\exists x.(\textbf{wolf}\ x) \wedge ((\textbf{enter}\ x) \\ \wedge(\Box(((\textbf{wolf}\ x) \wedge (\textbf{enter}\ x)) \\ \Rightarrow (\textbf{growl}\ (\texttt{sel}\ ((x :: \texttt{nil}) \cup \texttt{nil})))))))))] \wedge \top\end{aligned}$$

Examples

Example (*A wolf might enter. It would growl*)

$$\llbracket t_0 \ . \ t_1 \rrbracket \ \text{empty} \ \mathbb{T} \ \mathbb{T} \ \text{Conj} = [\Diamond(\top \wedge (\exists x.(\textbf{wolf}\ x) \wedge ((\textbf{enter}\ x) \wedge(\Box(((\textbf{wolf}\ x) \wedge (\textbf{enter}\ x)) \Rightarrow (\textbf{growl}\ (\texttt{sel}\ ((x :: \texttt{nil}) \cup \texttt{nil}))))))))] \wedge \top$$

Example (*A wolf might enter. #It will growl*)

$$\llbracket t_0 \ . \ t_2 \rrbracket \ \text{empty} \ \mathbb{T} \ \mathbb{T} \ \text{Conj} = [\Diamond(\top \wedge (\exists x.(\textbf{wolf}\ x) \wedge ((\textbf{enter}\ x) \wedge \top)))] \wedge [\textbf{growl}\ (\texttt{sel}\ \texttt{nil})]$$

Examples

Example (*A wolf is outside. He might eat you.*)

$\exists x.[\Diamond(\top \wedge (\mathbf{eat\ you}\ (\mathtt{sel\ nil} \cup (x :: \mathtt{nil}))))] \wedge [(\mathbf{wolf}\ x) \wedge ((\mathbf{Outside}\ x))]$

Putting Everything Together

The Basics

$$\gamma \triangleq \{\text{m\_ref} : \gamma'; \text{base} : t; \text{f\_ref} : \gamma'; \text{theory} : t\}$$

$$[\![c_{might}]\!] = \lambda v\, s.\lambda i\, k_1\ k_2\ f.(\lambda P.(\text{TEST}\ P)\ i.\text{theory}\ (\lambda i'\ o'_1\ o'_2\ f'.f'(\Diamond(i'.\text{base} \wedge (P\ i'\ o'_1\ o'_2\ Pi_1))))))\ (v\ s)\ i\ k_1\ k_2\ f$$

Conclusion and Perspectives

Summary

- Generality of the left context: extension to new areas without changing the logic
- Modularity
- Reset operator to empty the modal context
- Duplication of the content (see [Martin and Pollard(2010)])

Perspectives

- Move to TY2 and attitude reports, Hob and Nobs
- Discussion on models
- Negation contexts, counterfactuals...

N. Asher and E. McCready.
Were, would, might and a compositional account of counterfactuals.
Journal of Semantics, 24(2), 2007.

P. de Groote.
Towards a montagovian account of dynamics.
In *Proceedings of Semantics and Linguistic Theory XVI*, 2006.

A. Frank and H. Kamp.
On Context Dependence in Modal Constructions.
In *Proceedings of SALT VII*, 1997.

B. Geurts.
Presuppositions and Pronouns.
Current Research in the Semantics/Pragmatics Interface. Elsevier, 1999.

S. Martin and C. Pollard.
Hyperintensional dynamic semantics: Analyzing definiteness with enriched contexts.
In *Proceedings of Formal Grammar (FG) 15*, 2010.
To appear in Lecture Notes in Computer Science.

C. Roberts.
Modal subordination and pronominal anaphora in discourse.
Linguistics and Philosophy, 12, 1989.

M. Stone and D. Hardt.
Dynamic discourse referents for tense and modals.
In *Proceedings of IWCS 2*, 1997.

R. van Rooij.
A modal analysis of presupposition and modal subordination.
Journal of Semantics, 2005.

F. Veltman.
Defaults in update semantics.
Journal of Philosophical Logic, 25(3):221–261, 1996.