Hybrid Logic,

Patrick Blackburn

INRIA Lorraine, Nancy, France

Compter Science
Roskilde University
13 June 2005

Today's talk

- Orthodox modal logic: from an Amsterdam perspective
- Problems with orthodox modal logic
- Basic hybrid logic
- Deduction in hybrid logic
- Concluding remarks

What is modal logic?

Slogan 1: Modal languages are simple yet expressive languages for talking about relational structures.

Slogan 2: Modal languages provide an internal, local perspective on relational structures.

Slogan 3: Modal languages are not isolated formal systems.

From *Modal Logic*, by Blackburn, de Rijke and Venema, Cambridge University Press, 2001.

Propositional Modal Logic

Given propositional symbols PROP $= \{\mathrm{p}, \mathrm{q}, \mathrm{r}, \ldots\}$, and modality symbols MOD $= \{m, m', m'', \ldots\}$ the basic modal language (over PROP and MOD) is defined as follows:

$$\begin{array}{rcl} \text{WFF} & := & \mathrm{p} \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi \\ & & \mid \varphi \rightarrow \psi \mid \langle m \rangle \varphi \mid [m]\varphi \end{array}$$

If there's just one modality symbol in the language, we usually write $\Diamond$ and $\Box$ for its diamond and box forms.

Kripke Models

- A Kripke model $\mathcal{M}$ is a triple $(W, \mathcal{R}, V)$, where:
  - W is a non-empty set, whose elements can be thought of possible worlds, or epistemic states, or times, or states in a transition system, or geometrical points, or people standing in various relationships, or nodes in a parse tree — indeed, pretty much anything you like.
  - $\mathcal{R}$ is a collection of binary relation on W (one for each modality)
  - V is a valuation assigning subsets of W to propositional symbols.

- The component $(W, \mathcal{R})$ traditionally call a frame.

Satisfaction Definition

$$
\begin{array}{lll}
\mathcal{M}, w \Vdash \mathrm{p} & \text{iff} & w \in V(\mathrm{p}),\ \text{where } \mathrm{p} \in \mathrm{PROP} \\
\mathcal{M}, w \Vdash \neg\varphi & \text{iff} & \mathcal{M}, w \nVdash \varphi \\
\mathcal{M}, w \Vdash \varphi \wedge \psi & \text{iff} & \mathcal{M}, w \Vdash \varphi \text{ and } \mathcal{M}, w \Vdash \psi \\
\mathcal{M}, w \Vdash \varphi \vee \psi & \text{iff} & \mathcal{M}, w \Vdash \varphi \text{ or } \mathcal{M}, w \Vdash \psi \\
\mathcal{M}, w \Vdash \varphi \rightarrow \psi & \text{iff} & \mathcal{M}, w \nVdash \varphi \text{ or } \mathcal{M}, w \Vdash \psi \\
\mathcal{M}, w \Vdash \langle m \rangle\varphi & \text{iff} & \exists w'(wR^m w' \ \&\ \mathcal{M}, w' \Vdash \varphi) \\
\mathcal{M}, w \Vdash [m]\varphi & \text{iff} & \forall w'(wR^m w' \ \Rightarrow\ \mathcal{M}, w' \Vdash \varphi).
\end{array}
$$

Note the internal perspective. This is a key modal intuition, gives rise to the notion of bisimulation we shall shortly discuss, and is the driving force for many traditional applications.

Tense logic

- $\langle \mathrm{F} \rangle$ means "at some Future state", and $\langle \mathrm{P} \rangle$ means "at some Past state".

- $\langle \mathrm{P} \rangle\, \mathrm{mia-unconscious}$ is true iff we can look back in time from the current state and see a state where Mia is unconscious. Works a bit like the sentence *Mia has been unconscious*.

- $\langle \mathrm{F} \rangle\, \mathrm{mia-unconscious}$ requires us to scan the states that lie in the future looking for one where Mia is unconscious. Works a bit like the sentence *Mia will be unconscious*.

Description logic

Description logic

Consider the following $\mathcal{ALC}$ term:

$$\text{killer} \sqcap \exists \text{EMPLOYER}.\text{gangster}$$

Description logic

Consider the following $\mathcal{ALC}$ term:

$$\text{killer} \sqcap \exists\textsc{employer}.\text{gangster}$$

This means exactly the same thing as the modal formula:

$$\text{killer} \wedge \langle\textsc{employer}\rangle\text{gangster}$$

Description logic

Consider the following $\mathcal{ALC}$ term:

$$\text{killer} \sqcap \exists\textsc{employer}.\text{gangster}$$

This means exactly the same thing as the modal formula:

$$\text{killer} \wedge \langle\textsc{employer}\rangle\text{gangster}$$

Incidentally, this example shows that modal logic can be used as a tool in what would generally be regarded as the heartland of ordinary *extensional* logic: representing and reasoning about ordinary individuals.

But modal logic is merely one possibility among many

- There's nothing magic about frames or Kripke models. Frames $(W, \mathcal{R})$, are just a directed multigraphs (or labelled transition systems). Valuations simply decorate states with properties.

- So a Kripke model for the basic modal language are just (very simple) relational structures in the usual sense of first-order model theory.

- So we don't have to talk about Kripke models using modal logic — we could use first-order logic, or second-order logic, or infinitary logic, or fix-point logic, or indeed any logic interpreted over relational structures. Let's see how. . .

Doing it first-order style (I)

Consider the modal representation

$$\langle \mathrm{F} \rangle\, \mathrm{mia} - \mathrm{unconscious}$$

Doing it first-order style (I)

Consider the modal representation

$$\langle \mathrm{F} \rangle \, \mathrm{mia-unconscious}$$

we could use instead the first-order representation

$$\exists t (t_o < t \wedge \mathrm{MIA-UNCONSCIOUS}(t)).$$

Doing it first-order style (II)

And consider the modal representation

$$\text{killer} \wedge \langle \textsc{employer} \rangle \text{gangster}$$

Doing it first-order style (II)

And consider the modal representation

$$\mathsf{killer} \wedge \langle \text{EMPLOYER} \rangle \mathsf{gangster}$$

We could use instead the first-order representation

$$\mathsf{KILLER}(x) \wedge \exists y(\text{EMPLOYER}(x, y) \wedge \mathsf{GANGSTER}(y))$$

Standard Translation

And in fact, any modal representation can by converted into an equisatisfiable first-order representation:

$$
\begin{array}{lcl}
\mathrm{ST}_x(\mathrm{p}) & = & \mathrm{P}x \\
\mathrm{ST}_x(\neg\varphi) & = & \neg\,\mathrm{ST}_x(\varphi) \\
\mathrm{ST}_x(\varphi \wedge \psi) & = & \mathrm{ST}_x(\varphi) \wedge \mathrm{ST}_x(\psi) \\
\mathrm{ST}_x(\langle R \rangle \varphi) & = & \exists y(Rxy \wedge \mathrm{ST}_y(\varphi))
\end{array}
$$

Note that $\mathrm{ST}_x(\varphi)$ always contains exactly one free variable (namely x).

Proposition: For any modal formula φ, any Kripke model $\mathcal{M}$, and any state w in $\mathcal{M}$ we have that: $\mathcal{M}, w \Vdash \varphi$ iff $\mathcal{M} \models \mathrm{ST}_x(\varphi)[x \leftarrow w]$.

Aren't we better off with first-order logic . . . ?

- We've just seen that any modal formula can be systematically converted into an equisatifiable first-order formula.

- And as we'll later see, the reverse is not possible: first-order logic can describe models in far more detail that modal logic can. Some first-order formulas have no modal equivalent. That is, modal language are weaker than their corresponding first-order languages.

- So why bother with modal logic?

Reasons for going modal

- **Simplicity**. The standard translation shows us that modalities are essentially ‘macros’ encoding a quantification over related sates. Modal notation hides the bound variables, resulting in a compact, easy to read, representations.

- **Computability**. First-order logic is undecidable over arbitrary models. Modal logic is decidable over arbitrary models (indeed, decidable in PSPACE). Modal logic trades expressivity for computability.

- **Internal perspective**. Intuitively attractive. Moreover, leads to an elegant characterization of what modal logic can say about models. Let’s take a closer look. . .

Bisimulation

Let $\mathcal{M} = (W, \mathcal{R}, V)$ and $\mathcal{M}' = (W', \mathcal{R}', V')$ be models for the same basic modal language. A relation $Z \subseteq W \times W'$ is a bisimulation between $\mathcal{M}$ and $\mathcal{M}'$ if the following conditions are met:

1. Atomic equivalence: if wZw' then $w \in V(p)$ iff $w' \in V'(p)$, for all propositional symbols p.
2. Forth: if wZw' and wRv then there is a v' such that $w'R'v'$ and vZv'.
3. Back: if wZw' and $w'R'v'$ then there is a v such that wRv and vZv'.

If w is a point in $\mathcal{M}$ and w' a point in $\mathcal{M}'$ such that wZw' then we say that w is bisimilar to w'.

The van Benthem Characterization Theorem

For all first-order formulas φ containing exactly one free variable, φ is equivalent to the standard translation of a modal formula iff φ is bisimulation-invariant.

Proof:

($\Rightarrow$) An easy induction.

($\Leftarrow$) Non-trivial (usually proved using elementary chains or by appealing to the existence of saturated models).

In short, modal logic is a simple notation for capturing exactly the bisimulation-invariant fragment of first-order logic.

Back to slogan 3

Slogan 3: Modal languages are not isolated formal systems.

Modal languages over models are essentially simple fragments of first-order logic. These fragments have a number of attractive properties such as robust decidability and bisimulation invariance. Traditional modal notation is essentially a nice (quantifier free) 'macro' notation for working with this fragment.

Back to slogan 2

Slogan 2: Modal languages provide an internal, local perspective on relational structures.

This is not just an intuition: the notion of bisimulation, and the results associated with it, shows that this is the key model theoretic fact at work in modal logic.

Back to slogan 1

Slogan 1: Modal languages are simple yet expressive languages for talking about relational structures.

You can use modal logic for just about anything. Anywhere you see a graph, you can use a modal language to talk about it.

That was the good news — now comes the bad

Orthodox modal languages have an obvious drawback for many applications: they don't let us refer to individual states (worlds, times, situations, nodes. . . .). That is, they don't allow us to say things like

- this happened $there$; or
- this happened $then$; or
- $this$ state has property φ; or
- node i is marked with the information p.

and so on.

Temporal logic

- Temporal representations in Artificial Intelligence based around temporal reference — and for good reasons.

- Worse, standard modal logics of time are completely inadequate for the temporal semantics of natural language. *Vincent accidentally squeezed the trigger* doesn't mean that at some completely unspecified past time Vincent did in fact accidentally squeeze the trigger, it means that at some *particular*, contextually determined, past time he did so. The representation, $\langle \mathrm{P} \rangle\, \mathrm{vincent-accidentally-squeeze-trigger}$ fails to capture this.

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

The states described by the first two sentences hold at the same time. The event described by the second takes place a little later. In orthodox modal logics there is no way assert the identity of the times needed for the first two sentences, nor to capture the move forward in time needed by the third.

In fact, modal languages for temporal representation have not been the tool of choice in natural language semantics for over 15 years.

Description logic I

As we have already said, there is a transparent correspondence between simple DL terms and modal formulas:

$$\text{killer} \sqcap \exists\textsc{employer}.\text{gangster}$$

$$\text{killer} \wedge \langle\textsc{employer}\rangle\text{gangster}$$

Nonetheless, this correspondence only involves what description logicians call the TBox (Terminological Box).

Description logic II

Orthodox modal logic does not have anything to say about the ABox (Assertional Box):

$$\text{mia} : \text{Beautiful}$$

$$(\text{jules}, \text{vincent}) : \text{Friends}$$

That is, it can't make assertions about individuals, for it has no tools for naming individuals.

Ambition

- Want to be able to refer to states, but want to do so without destroying the simplicity of propositional modal logic.

- But how can we do this — propositional modal logic has very few moving parts?

- Answer: sort the atomic symbols. Use formulas as terms.

This will fix the obvious shortcoming — and as we shall learn, it will fix a lot more besides.

Extension #1

- Take a language of basic modal logic (with propositional variables p, q, r, and so on) and add a second sort of atomic formula.
- The new atoms are called nominals, and are typically written i, j, k, and l.
- Both types of atom can be freely combined to form more complex formulas in the usual way; for example,

$$\Diamond(i \wedge \mathrm{p}) \wedge \Diamond(i \wedge \mathrm{q}) \rightarrow \Diamond(\mathrm{p} \wedge \mathrm{q})$$

  is a well formed formula.
- Insist that each nominal be true at exactly one world in any model. A nominal names a state by being true there and nowhere else.

We already have a richer logic

Consider the orthodox formula

$$\Diamond(\mathrm{r} \wedge \mathrm{p}) \wedge \Diamond(\mathrm{r} \wedge \mathrm{q}) \rightarrow \Diamond(\mathrm{p} \wedge \mathrm{q})$$

This is easy to falsify.

We already have a richer logic

Consider the orthodox formula

$$\Diamond(\mathrm{r} \wedge \mathrm{p}) \wedge \Diamond(\mathrm{r} \wedge \mathrm{q}) \rightarrow \Diamond(\mathrm{p} \wedge \mathrm{q})$$

This is easy to falsify.

On the other hand, the hybrid formula

$$\Diamond(i \wedge \mathrm{p}) \wedge \Diamond(i \wedge \mathrm{q}) \rightarrow \Diamond(\mathrm{p} \wedge \mathrm{q})$$

is valid (unfalsifiable). Nominals name, and this adds to the expressive power at our disposal.

Extension #2

- Add formulas of form $@_i\varphi$.

- Such formulas assert that φ is satisfied at the point named by the nominal i.

- Expressions of the form $@_i$ are called satisfaction operators.

Tense logic

- On the road to capturing Allen's logic of temporal reference; @ plays the role of **Holds**.

- $\langle \mathsf{P} \rangle (i \wedge$Vincent-accidentally-squeeze-the-trigger$)$
locates the trigger-squeezing not merely in the past, but at a specific temporal state there, namely the one named by i — capturing the meaning of *Vincent accidentally squeezed the trigger*.

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

$$P(i \wedge \textsf{vincent-wake-up})$$

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

$$P(i \wedge \textsf{vincent-wake-up})$$

$$\wedge \quad P(j \wedge \textsf{something-feel-very-wrong})$$

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

$$P(i \wedge \textsf{vincent-wake-up})$$

$$\wedge \quad P(j \wedge \textsf{something-feel-very-wrong}) \qquad \wedge \quad @_j i$$

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

$$
\begin{array}{llll}
 & P(i \wedge \textsf{vincent-wake-up}) & & \\
\wedge & P(j \wedge \textsf{something-feel-very-wrong}) & \wedge & @_j i \\
\wedge & P(k \wedge \textsf{vincent-reach-under-pillow-for-uzi}) & &
\end{array}
$$

Tense in text

Vincent woke up. Something felt very wrong. Vincent reached under his pillow for his Uzi.

$$
\begin{array}{llll}
 & P(i \wedge \textsf{vincent-wake-up}) & & \\
\wedge & P(j \wedge \textsf{something-feel-very-wrong}) & \wedge & @_j i \\
\wedge & P(k \wedge \textsf{vincent-reach-under-pillow-for-uzi}) & \wedge & @_k P i
\end{array}
$$

Description logic (I)

We can now make ABox statements. For example, to capture the effect of the (conceptual) ABox assertion

$$\text{mia} : \text{Beautiful}$$

we can write

$$@_{\text{mia}}\text{Beautiful}$$

Description logic (II)

Similarly, to capture the effect of the (relational) ABox assertion

$$(\mathsf{jules}, \mathsf{vincent}) : \mathsf{Friends}$$

we can write

$$@_{\mathsf{jules}}\langle \mathsf{Friends} \rangle \mathsf{vincent}$$

Basic hybrid language clearly modal

- Neither syntactical nor computational simplicity, nor general ‘style’ of modal logic, has been compromised.

- Satisfaction operators are normal modal operators, nominals just atomic formulas.

Still computable

Enriching ordinary propositional modal logic with both nominals and satisfaction operators does effect computability. The basic hybrid logic is decidable. Indeed we have:

Theorem: The satisfiability problem for basic hybrid languages over arbitrary models is PSPACE-complete (Areces, Blackburn, and Marx).

That is (up to a polynomial) the hybridized language has the same complexity as the orthodox modal language we started with.

Remark: things don't always work out as straightforwardly as this.

Standard Translation

Any hybrid representation can by converted into an equisatisfiable first-order representation. All we have to do is add a first-order constant i for each nominal$i$ and translate as follows (note the use of equality):

$$\begin{array}{lcl} \mathrm{ST}_x(p) & = & Px \\ \mathrm{ST}_x(i) & = & (x = i) \\ \mathrm{ST}_x(\neg\varphi) & = & \neg\,\mathrm{ST}_x(\varphi) \\ \mathrm{ST}_x(\varphi \wedge \psi) & = & \mathrm{ST}_x(\varphi) \wedge \mathrm{ST}_x(\psi) \\ \mathrm{ST}_x(\langle R \rangle \varphi) & = & \exists y(Rxy \wedge \mathrm{ST}_y(\varphi)) \\ \mathrm{ST}_x(@_i\varphi) & = & \mathrm{ST}_i(\varphi) \end{array}$$

Note that $\mathrm{ST}_x(\varphi)$ always contains at most free variable (namely x).

Proposition: For any hybrid formula φ, any Kripke model $\mathcal{M}$, and any state w in $\mathcal{M}$ we have that: $\mathcal{M}, w \Vdash \varphi$ iff $\mathcal{M} \models \mathrm{ST}_x(\varphi)[x \leftarrow w]$.

But what *is* basic hybrid logic?

We have seen many examples of what basic hybrid logic can do in various applications.

We've also seen that a number of the properties we liked about modal logic are inherited by the basic hybrid language.

This is all very nice — but none of it gives us a clear mathematical characterization of what basic hybrid logic actually is.

And it is possible to give such a characterization, and a genuinely modal one at that. Let's take a look . . .

Bisimulation-with-constants

Let $\mathcal{M} = (W, \mathcal{R}, V)$ and $\mathcal{M}' = (W', \mathcal{R}', V')$ be models for the same basic hybrid language. A relation $Z \subseteq W \times W'$ is a bisimulation-with-constants between $\mathcal{M}$ and $\mathcal{M}'$ if the following conditions are met:

1. Atomic equivalence: if wZw' then $w \in V(p)$ iff $w' \in V'(p)$, for all propositional symbols p, and all nominals i.

2. Forth: if wZw' and wRv then there is a v' such that $w'R'v'$ and vZv'.

3. Back: if wZw' and $w'R'v'$ then there is a v such that wRv and vZv'.

4. All points named by nominals are related by Z.

Lifting the van Benthem Characterization theorem

For all first-order formulas φ (in the correspondence language with constants and equality) containing at most one free variable, φ is bisimulation-with-constants invariant iff φ is equivalent to the standard translation of a basic hybrid formula iff (Areces, Blackburn, ten Cate, and Marx)

In short, basic hybrid logic is a simple notation for capturing exactly the bisimulation-invariant fragment of first-order logic when we make use of constants and equality.

Proof:

($\Rightarrow$) An easy induction.

($\Leftarrow$) Can be proved using elementary chains or by appealing to the existence of saturated models.

Hybrid deduction

- Hybridization has clearly fixed the referential failure of modal logic in a natural way.

- But it has also fixed a deeper problem: developing decent proof systems for modal logic. Let's take a brief look at this.

Different models, different logics

Key fact about modal logic: when you work with different kinds of models (graphs) the logic typically changes. For example:

- $\Box p \wedge \Box q \rightarrow \Box(p \wedge q)$ is valid on all models: it's part of the basic, universally applicable, logic.
- But $\Diamond\Diamond p \rightarrow \Diamond p$ is only valid on transitive graphs. It's not part of the basic logic, rather it's part of the special (stronger) logic that we need to use when working with transitive models.

Modal deduction should be general

- Modal logicians have insisted on developing proof methods which are general — that is, which can be easily adapted to cope with the logics of many kinds of models (transitive, reflexive, symmetric, dense, and so on).
- They achieve this goal by making use of Hilbert-style systems (that is, axiomatic systems).
- There is a basic axiomatic systems (called **K**) for dealing with arbitrary models.
- To deal with special classes of models, further axioms are added to **K**. For example, adding $\Diamond\Diamond \mathrm{p} \rightarrow \Diamond p$ as an axiom gives us the logic of transitive frames.

Generality clashes with easy of use

- Unfortunately, Hilbert systems are hard to use and completely unsuitable for computational implementation.

- For ease of use we want (say) natural deduction systems or tableau systems. For computational implementation we want (say) resolution systems or tableau systems.

- But it is hard to develop tableau, or natural deduction, or resolution in a general way in orthodox modal logic.

- Why is this?

Getting behind the diamonds

- The difficulty is extracting information from under the scope of diamonds.

- That is, given $\Diamond\phi$, how do we lay hands on ϕ? And given $\neg\Box\phi$ (that is, $\Diamond\neg\phi$), how do we lay hands on $\neg\phi$?

- In first order logic, the analogous problem is trivial. There is a simple rule for stripping away existential quantifiers: from $\exists x\phi$ we conclude $\phi[x \leftarrow a]$ for some brand new constant a (this rule is usually called Existential Elimination).

- But in orthodox modal logic there is no simple way of stripping off the diamonds.

Hybrid deduction

- Hybrid deduction is based on a simple observation: it's easy to get at the information under the scope of diamonds — for there is an natural way of stripping the diamonds away.

- I'll illustrate this idea in the setting of tableaux — but it can (and has been) used in a variety of proof styles, including resolution and natural deduction.

- Moreover, once the tableau system for reasoning about arbitrary models has been defined, it is straightforward to extend it to to cover the logics of special classes of models. That is, hybridization enables us to achieve the traditional modal goal of generality without resorting to Hilbert-systems.

$$[\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\, (hip \wedge cute)$$

$$[\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$$

1 $\neg @_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute))$

$$[\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\, (hip \wedge cute)$$

1	$\neg @_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\, (hip \wedge cute))$
2	$@_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute)$
$2'$	$\neg @_i\langle\text{HATE}\rangle\, (hip \wedge cute)$

$$[\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \to \langle\text{HATE}\rangle\, (hip \wedge cute)$$

1	$\neg @_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \to \langle\text{HATE}\rangle\, (hip \wedge cute))$
2	$@_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute)$
$2'$	$\neg @_i\langle\text{HATE}\rangle\, (hip \wedge cute)$
3	$@_i[\text{HATE}]\, hip$
$3'$	$@_i\langle\text{HATE}\rangle\, cute$

$$[\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\, (hip \wedge cute)$$

1	$\neg @_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\, (hip \wedge cute))$
2	$@_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute)$
$2'$	$\neg @_i\langle\text{HATE}\rangle\, (hip \wedge cute)$
3	$@_i[\text{HATE}]\, hip$
$3'$	$@_i\langle\text{HATE}\rangle\, cute$
4	$@_i\langle\text{HATE}\rangle\, j$

$$[\mathrm{HATE}]\, hip \wedge \langle\mathrm{HATE}\rangle\, cute \rightarrow \langle\mathrm{HATE}\rangle\, (hip \wedge cute)$$

1	$\neg @_i([\mathrm{HATE}]\, hip \wedge \langle\mathrm{HATE}\rangle\, cute \rightarrow \langle\mathrm{HATE}\rangle\, (hip \wedge cute))$
2	$@_i([\mathrm{HATE}]\, hip \wedge \langle\mathrm{HATE}\rangle\, cute)$
2′	$\neg @_i\langle\mathrm{HATE}\rangle\, (hip \wedge cute)$
3	$@_i[\mathrm{HATE}]\, hip$
3′	$@_i\langle\mathrm{HATE}\rangle\, cute$
4	$@_i\langle\mathrm{HATE}\rangle\, j$
4′	$@_j cute$

$$[\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$$

1	$\neg @_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute))$
2	$@_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute)$
2′	$\neg @_i \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$
3	$@_i [\mathrm{HATE}]\, hip$
3′	$@_i \langle \mathrm{HATE} \rangle\, cute$
4	$@_i \langle \mathrm{HATE} \rangle\, j$
4′	$@_j cute$
5	$@_j hip$

$$[\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\,(hip \wedge cute)$$

1	$\neg @_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute \rightarrow \langle\text{HATE}\rangle\,(hip \wedge cute))$
2	$@_i([\text{HATE}]\, hip \wedge \langle\text{HATE}\rangle\, cute)$
2′	$\neg @_i\langle\text{HATE}\rangle\,(hip \wedge cute)$
3	$@_i[\text{HATE}]\, hip$
3′	$@_i\langle\text{HATE}\rangle\, cute$
4	$@_i\langle\text{HATE}\rangle\, j$
4′	$@_j cute$
5	$@_j hip$
6	$\neg @_j(hip \wedge cute)$

$$[\text{HATE}]\, hip \wedge \langle \text{HATE} \rangle\, cute \rightarrow \langle \text{HATE} \rangle\, (hip \wedge cute)$$

1	$\neg @_i([\text{HATE}]\, hip \wedge \langle \text{HATE} \rangle\, cute \rightarrow \langle \text{HATE} \rangle\, (hip \wedge cute))$	
2	$@_i([\text{HATE}]\, hip \wedge \langle \text{HATE} \rangle\, cute)$	
$2'$	$\neg @_i \langle \text{HATE} \rangle\, (hip \wedge cute)$	
3	$@_i [\text{HATE}]\, hip$	
$3'$	$@_i \langle \text{HATE} \rangle\, cute$	
4	$@_i \langle \text{HATE} \rangle\, j$	
$4'$	$@_j cute$	
5	$@_j hip$	
6	$\neg @_j (hip \wedge cute)$	
7	$\neg @_j hip$	$\neg @_j cute$

$$[\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$$

1		$\neg @_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute))$	
2		$@_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute)$	
2′		$\neg @_i \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$	
3		$@_i [\mathrm{HATE}]\, hip$	
3′		$@_i \langle \mathrm{HATE} \rangle\, cute$	
4		$@_i \langle \mathrm{HATE} \rangle\, j$	
4′		$@_j cute$	
5		$@_j hip$	
6		$\neg @_j (hip \wedge cute)$	
7	$\neg @_j hip$		$\neg @_j cute$
	$\perp_{5,7}$		

$$[\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$$

1		$\neg @_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute \rightarrow \langle \mathrm{HATE} \rangle\, (hip \wedge cute))$	
2		$@_i([\mathrm{HATE}]\, hip \wedge \langle \mathrm{HATE} \rangle\, cute)$	
$2'$		$\neg @_i \langle \mathrm{HATE} \rangle\, (hip \wedge cute)$	
3		$@_i [\mathrm{HATE}]\, hip$	
$3'$		$@_i \langle \mathrm{HATE} \rangle\, cute$	
4		$@_i \langle \mathrm{HATE} \rangle\, j$	
$4'$		$@_j cute$	
5		$@_j hip$	
6		$\neg @_j (hip \wedge cute)$	
7	$\neg @_j hip$		$\neg @_j cute$
	$\bot_{5,7}$		$\bot_{4',7}$

Link with first-order deduction

Hybrid Logic	First Order Logic
$@_i\Diamond\phi$	

Link with first-order deduction

Hybrid Logic	First Order Logic
$@_i \Diamond \phi$	$\exists y (Riy \wedge ST_y(\phi))$

Link with first-order deduction

Hybrid Logic	First Order Logic
$@_i\Diamond\phi$	$\exists y(Riy \wedge ST_y(\phi))$
$@_i\Diamond j$ $@_j\phi$	

Link with first-order deduction

Hybrid Logic	First Order Logic
$@_i \Diamond \phi$	$\exists y(Riy \wedge ST_y(\phi))$
	$Rij \wedge ST_j(\phi)$
$@_i \Diamond j$	
$@_j \phi$	

Link with first-order deduction

Hybrid Logic	First Order Logic
$@_i\Diamond\phi$	$\exists y(Riy \wedge ST_y(\phi))$
	$Rij \wedge ST_j(\phi)$
$@_i\Diamond j$	Rij
$@_j\phi$	$ST_j(\phi)$

Completeness

- The tableaux system is complete with respect to the class of all models (easy Henkin/Hintikka proof).
- Moreover, any pure axiomatic extension of the tableau system (that is, an extension using axioms whose only atoms are nominals) is complete with respect to the class of models defined by the axioms (easy Henkin/Hintikka proof).
- In short, hybridization allows one to achieve the modal goal of generality without resorting to Hilbert systems — and this can be proved by adapting classical tools.

Concluding remarks

Where is hybrid logic going?

Concluding remarks

Where is hybrid logic going?

- Stronger hybrid languages in which nominals can be bound to states: $\downarrow i.\phi$ binds i to the point of evaluation. General interpolation results.

Concluding remarks

Where is hybrid logic going?

- Stronger hybrid languages in which nominals can be bound to states: $\downarrow i.\phi$ binds i to the point of evaluation. General interpolation results.
- First-order hybrid logic, First-order hybrid logic with predicate abstraction, type theory. Basic meta-theory (completeness, interpolation, Beth definability) has been lifted to First-order hybrid logic case.

Concluding remarks

Where is hybrid logic going?

- Stronger hybrid languages in which nominals can be bound to states: $\downarrow i.\phi$ binds i to the point of evaluation. General interpolation results.

- First-order hybrid logic, First-order hybrid logic with predicate abstraction, type theory. Basic meta-theory (completeness, interpolation, Beth definability) has been lifted to First-order hybrid logic case.

- Applications — traditional and novel.

Computational tools and applications

- Developing hybrid theorem proving methods. Most important so far is HyLoRes (Areces) a resolution based theorem prover that covers some very rich hybrid logics.
- Exploring links with description logic, to take advantage of DL theorem proving technology. For example, Areces, Blackburn and Marx on compiling parts of Hybrid Logic into RACER-format description logic.

Advertisements

- Hybrid logic homepage: www.hylo.net. Information, papers, HyLoRes, and lots of other stuff.

- "Reasoning, Representation and Relational Structures: the Hybrid Logic Manifesto", by Patrick Blackburn. Overview of Hybrid Logic; probably a good follow-up to this talk. Can be downloaded from hybrid logic homepage.

A Brief History

A Brief History

Early days (1970): Robert Bull

A Brief History

Early days (1970): Robert Bull

Sofia School (1980s): Soloman Passy, George Gargov, Tinko Tinchev, Valentin Goranko

A Brief History

Early days (1970): Robert Bull

Sofia School (1980s): Soloman Passy, George Gargov, Tinko Tinchev, Valentin Goranko

Recent Work (1990s till present): Carlos Areces, Torben Brauner, Balder ten Cate, Maarten Marx, Jerry Seligman, Miroslava Tzakova.

A Brief History

Early days (1970): Robert Bull

Sofia School (1980s): Soloman Passy, George Gargov, Tinko Tinchev, Valentin Goranko

Recent Work (1990s till present): Carlos Areces, Torben Brauner, Balder ten Cate, Maarten Marx, Jerry Seligman, Miroslava Tzakova.

But one crucial name is missing . . .

Arthur Prior, 1914-1969