synonyms in this talk

Church's Simple Type Theory

Classical Higher Order Logic (HOL)

HOL (Alonzo Church, 1940)

- simple types $\alpha, \beta ::= \iota | o | \alpha \to \beta$ (opt. further base types)
- HOL defined by

$$\begin{aligned} s, t \quad ::= \quad & p_\alpha \mid X_\alpha \\ & \mid (\lambda X_\alpha . s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \\ & \mid (\neg_{o\to o}\, s_o)_o \mid (s_o \vee_{o\to o\to o} t_o)_o \mid (\forall X_\alpha . t_o)_o \end{aligned}$$

- HOL is well understood
  - Origin (Church, J.Symb.Log., 1940)
  - Henkin semantics (Henkin, J.Symb.Log., 1950)
    (Andrews, J.Symb.Log., 1971, 1972)
  - Extens./Intens. (BenzmüllerEtAl., J.Symb.Log., 2004)
    (Muskens, J.Symb.Log., 2007)

HOL (Alonzo Church, 1940)

- simple types $\alpha, \beta ::= \iota | o | \alpha \to \beta$ (opt. further base types)
- HOL defined by

$$
\begin{aligned}
s, t \quad ::= \quad & p_\alpha \mid X_\alpha \\
& \mid (\lambda X_\alpha . s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\ t_\alpha)_\beta \\
& \mid (\neg_{o\to o}\ s_o)_o \mid (s_o \vee_{o\to o\to o}\ t_o)_o \mid (\Pi_{(\alpha\to o)\to o}\ (\lambda X_\alpha . t_o))_o
\end{aligned}
$$

- HOL is well understood
  - Origin (Church, J.Symb.Log., 1940)
  - Henkin semantics (Henkin, J.Symb.Log., 1950)
    (Andrews, J.Symb.Log., 1971, 1972)
  - Extens./Intens. (BenzmüllerEtAl., J.Symb.Log., 2004)
    (Muskens, J.Symb.Log., 2007)

HOL (Alonzo Church, 1940)

- simple types $\alpha, \beta ::= \iota | o | \alpha \to \beta$ (opt. further base types)
- HOL defined by

$$
\begin{aligned}
s, t \quad ::= \quad & p_\alpha \mid X_\alpha \\
& \mid (\lambda X_\alpha . s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \\
& \mid (\neg_{o\to o}\, s_o)_o \mid (s_o \vee_{o\to o\to o}\, t_o)_o \mid (\Pi_{(\alpha\to o)\to o}\, (\lambda X_\alpha . t_o))_o
\end{aligned}
$$

- HOL is well understood
  - Origin (Church, J.Symb.Log., 1940)
  - Henkin semantics (Henkin, J.Symb.Log., 1950)
    (Andrews, J.Symb.Log., 1971, 1972)
  - Extens./Intens. (BenzmüllerEtAl., J.Symb.Log., 2004)
    (Muskens, J.Symb.Log., 2007)

Opinions about HOL:

- HOL is expressive

but ...

- HOL can not be effectively automated
- HOL is a classical logic and not easily compatible with
  - modal logics
  - intuitionistic logic
  - ...
- HOL can not fruitfully serve as a basis for combining logics

Opinions about HOL: Position of this talk

- HOL is expressive and we exploit this here

but ...

- HOL can ~~not~~ be effectively automated (at least partly)
- HOL is a classical logic and ~~not~~ easily compatible with
  - (normal) modal logics
  - intuitionistic logic
  - ...
- HOL can ~~not~~ fruitfully serve as a basis for combining logics
  (interesting application area: multi-agent systems)

...I will give theoretical and practical evidence

Quantified Multimodal Logics (QML) as HOL Fragments

(jww Larry Paulson)

Quantified Multimodal Logics (QML)

- QML defined by

$$\begin{aligned} s, t \quad ::= \quad & P \mid (k\, X^1 \ldots X^n) \\ & \mid \neg s \mid s \vee t \\ & \mid \Box_r s \\ & \mid \forall^i X.s \mid \forall^p P.s \end{aligned}$$

- Kripke style semantics
  - notion of (QS5) models: (Fitting, J.Symb.Log., 2005)

    QS5π

    (BenzmüllerPaulson, Techn.Report, 2009)

Quantified Multimodal Logics (QML)

- QML defined by

$$\begin{aligned} s, t \quad ::= \quad & P \mid (k\, X^1 \ldots X^n) \\ & \mid \neg s \mid s \vee t \\ & \mid \Box_r s \\ & \mid \forall^i X.s \mid \forall^p P.s \end{aligned}$$

- Kripke style semantics
  - notion of (QS5) models: (Fitting, J.Symb.Log., 2005)

    QS5$\pi \longrightarrow$ **QK**π (correspondence to Henkin models)

    (BenzmüllerPaulson, Techn.Report, 2009)

2.2. Quantified Multimodal Logic

First-order quantification can be constant domain or varying domain. Below we only consider the constant domain case: every possible world has the same domain. We adapt the presentation of syntax and semantics of quantified modal logic from Fitting [18]. In contrast to Fitting we are not interested in S5 structures but in the more general case of K.

Let IV be a set of first-order (individual) variables, PV a set of propositional variables, and SYM a set of predicate symbols of any arity. Like Fitting, we keep our definitions simple by not having function or constant symbols; our language has no terms other than variables. While Fitting [18] studies quantified monomodal logic, we are interested in quantified multimodal logic. Hence, we introduce multiple $\Box_r$ operators for symbols r from an index set S. The grammar for our quantified multimodal logic QML is thus

$$s, t \quad ::= \quad P \mid k(X^1, \ldots, X^n) \mid \neg\, s \mid s \vee t \mid \forall X.\, s \mid \forall P.\, s \mid \Box_r\, s$$

where $P \in \text{PV}$, $k \in \text{SYM}$, and $X, X^i \in \text{IV}$.

Further connectives, quantifiers, and modal operators can be defined as usual. We also obey the usual definitions of free variable occurrences and substitutions.

Fitting introduces three different notions of semantics: QS5π^-, QS5π, and QS5π^+. We study related notions QKπ^-, QKπ, and QKπ^+ for a modal context K, and we support multiple modalities.

A *QKπ^- model* is a structure $M = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ such that $(W, (R_r)_{r\in S})$ is a multimodal frame (that is, W is the set of possible worlds and the R_r are accessibility relations between worlds in W), D is a non-empty set (the first-order domain), P is a non-empty collection of subsets of W (the propositional domain), and the I_w are interpretation functions mapping each n-place relation symbol $k \in \text{SYM}$ to some n-place relation on D in world w.

A *variable assignment* $g = (g^{iv}, g^{pv})$ is a pair of maps $g^{iv} : \text{IV} \longrightarrow D$ and $g^{pv} : \text{PV} \longrightarrow P$, where g^{iv} maps each individual variable in IV to a an object in D and g^{pv} maps each propositional variable in PV to a set of worlds in P.

Validity of a formula s for a model $M = (W, (R_r)_{r\in S}, D, P, I_w)$, a world $w \in W$, and a variable assignment $g = (g^{iv}, g^{pv})$ is denoted as $M, g, w \models s$ and defined as follows, where $[a/Z]g$ denotes the assignment identical to g except that $([a/Z]g)(Z) = a$:

$M, g, w \models k(X^1, \ldots, X^n)$	if and only if	$\langle g^{iv}(X^1), \ldots, g^{iv}(X^n)\rangle \in I_w(k)$
$M, g, w \models P$	if and only if	$w \in g^{pv}(P)$
$M, g, w \models \neg\, s$	if and only if	$M, g, w \not\models s$
$M, g, w \models s \vee t$	if and only if	$M, g, w \models s$ or $M, g, w \models t$
$M, g, w \models \forall X.\, s$	if and only if	$M, ([d/X]g^{iv}, g^{pv}), w \models s$ for all $d \in D$

$$M, g, w \models \forall Q_{\cdot}\, s \quad \text{if and only if} \quad M, (g^{iv}, [p/Q]g^{pv}), w \models s \text{ for all } p \in P$$

$$M, g, w \models \Box_r\, s \quad \text{if and only if} \quad M, g, v \models s \text{ for all } v \in W \text{ with } \langle w, v\rangle \in R_r$$

A QKπ^- model $M = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ is a *QKπ model* if for every variable assignment g and every formula $s \in$ QML, the set of worlds $\{w \in W \mid M, g, w \models s\}$ is a member of P.

A QKπ model $M = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ is a *QKπ^+ model* if every world $w \in W$ is member of an atom in P. The *atoms* of P are minimal non-empty elements of P: no proper subsets of an atom are also elements of P.

A QML formula s is *valid in model M for world w* if $M, g, w \models s$ for all variable assignments g. A formula s is *valid in model M* if $M, g, w \models s$ for all g and w. Formula s is *QKπ-valid* if s is valid in all QKπ models, when we write $\models^{\mathrm{QK}\pi} s$; we define QK$\pi^-$-valid and QK$\pi^+$-valid analogously.

In the remainder we mainly focus on QKπ models. These models naturally correspond to Henkin models, as we shall see in Section 4.

3. Embedding Quantified Multimodal Logic in STT

The idea of the encoding is simple. We choose type ι to denote the (non-empty) set of individuals and we reserve a second base type μ to denote the (non-empty) set of possible worlds. The type o denotes the set of truth values. Certain formulas of type $\mu \to o$ then correspond to multimodal logic expressions. The multimodal connectives $\neg$, $\vee$, and $\Box$, become λ-terms of types $(\mu \to o) \to (\mu \to o)$, $(\mu \to o) \to (\mu \to o) \to (\mu \to o)$, and $(\mu \to \mu \to o) \to (\mu \to o) \to (\mu \to o)$ respectively.

Quantification is handled as usual in higher-order logic by modeling $\forall X_{\cdot} s$ as $\Pi(\lambda X_{\cdot} s)$ for a suitably chosen connective Π, as we remarked in Section 2. Here we are interested in defining two particular modal Π-connectives: $\mathbf{\Pi}^{\iota}$, for quantification over individual variables, and $\mathbf{\Pi}^{\mu\to o}$, for quantification over modal propositional variables that depend on worlds, of types $(\iota \to (\mu \to o)) \to (\mu \to o)$ and $((\mu \to o) \to (\mu \to o)) \to (\mu \to o)$, respectively.

In previous work [10] we have discussed first-order and higher-order modal logic, including a means of explicitly excluding terms of certain types. The idea was that no proper subterm of $t_{\mu\to o}$ should introduce a dependency on worlds. Here we skip this restriction. This leads to a simpler definition of a quantified multimodal language QMLSTT below, and it does not affect our soundness and completeness results.

Definition 3.1 (Modal operators). The modal operators $\neg$, $\vee$, $\Box$, $\mathbf{\Pi}^{\iota}$, and $\mathbf{\Pi}^{\mu\to o}$ are defined as follows:

$$\neg_{(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o\cdot}\lambda W_{\mu\cdot}\neg(\phi\, W)$$

$$\vee_{(\mu\to o)\to(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o\cdot}\lambda\psi_{\mu\to o\cdot}\lambda W_{\mu\cdot}\phi\, W \vee \psi\, W$$

$$\Box_{(\mu\to\mu\to o)\to(\mu\to o)\to(\mu\to o)} = \lambda R_{\mu\to\mu\to o}.\lambda\phi_{\mu\to o}.\lambda W_{\mu}.\forall V_{\mu}.\neg(R\,W\,V)\vee\phi\,V$$
$$\mathbf{\Pi}^{\iota}_{(\iota\to(\mu\to o))\to(\mu\to o)} = \lambda\phi_{\iota\to(\mu\to o)}.\lambda W_{\mu}.\forall X_{\iota}.\phi\,X\,W$$
$$\mathbf{\Pi}^{\mu\to o}_{((\mu\to o)\to(\mu\to o))\to(\mu\to o)} = \lambda\phi_{(\mu\to o)\to(\mu\to o)}.\lambda W_{\mu}.\forall P_{\mu\to o}.\phi\,P\,W$$

Note that our encoding actually only employs the second-order fragment of simple type theory enhanced with lambda-notation.

Further operators can be introduced, for example,

$$\top_{(\mu\to o)\to(\mu\to o)} = \forall P_{\mu\to o}.P\vee\neg P$$
$$\bot_{(\mu\to o)\to(\mu\to o)} = \neg\top$$
$$\wedge_{(\mu\to o)\to(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o}.\lambda\psi_{\mu\to o}.\neg(\neg\phi\vee\neg\psi)$$
$$\supset_{(\mu\to o)\to(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o}.\lambda\psi_{\mu\to o}.\neg\phi\vee\psi$$
$$\Diamond_{(\mu\to\mu\to o)\to(\mu\to o)\to(\mu\to o)} = \lambda R_{\mu\to\mu\to o}.\lambda\phi_{\mu\to o}.\neg(\Box\,R\,(\neg\phi))$$
$$\mathbf{\Sigma}^{\iota}_{(\iota\to(\mu\to o))\to(\mu\to o)} = \lambda\phi_{\iota\to(\mu\to o)}.\neg(\mathbf{\Pi}^{\iota}(\lambda X_{\iota}.\neg(\phi\,X)))$$
$$\mathbf{\Sigma}^{\mu\to o}_{((\mu\to o)\to(\mu\to o))\to(\mu\to o)} = \lambda\phi_{(\mu\to o)\to(\mu\to o)}.\neg(\mathbf{\Pi}^{\mu\to o}(\lambda P_{\mu\to o}.\neg(\phi\,P)))$$

We could also introduce further modal operators, such as the difference modality D, the global modality E, nominals with !, or the @ operator (cf. the recent work of Kaminski and Smolka [23] in the propositional hybrid logic context):

$$D_{(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o}.\lambda W_{\mu}.\exists V_{\mu}.W\neq V\wedge\phi V$$
$$E_{(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o}.\phi\vee D\,\phi$$
$$!_{(\mu\to o)\to(\mu\to o)} = \lambda\phi_{\mu\to o}.E\,(\phi\wedge\neg(D\,\phi))$$
$$@_{\mu\to(\mu\to o)\to(\mu\to o)} = \lambda W_{\mu}.\lambda\phi_{\mu\to o}.\phi\,W$$

For defining QMLSTT-propositions we fix a set IVSTT of individual variables of type ι, a set PVSTT of propositional variables of type $\mu\to o$, and a set SYMSTT of n-ary (curried) predicate constants of types $\underbrace{\iota\to\ldots\to\iota}_{n}\to(\mu\to o)$. The latter types will be abbreviated as $\iota^n\to(\mu\to o)$ in the remainder. Moreover, we fix a set SSTT of accessibility relation constants of type $\mu\to\mu\to o$.

Definition 3.2 (QMLSTT-propositions). QMLSTT-propositions are defined as the smallest set of simply typed λ-terms for which the following hold:

- Each variable $P_{\mu\to o}\in$ PVSTT is an atomic QMLSTT-proposition, and if $X^j_{\iota}\in$ IVSTT (for $j=1,\ldots,n$) and $k_{\iota^n\to(\mu\to o)}\in$ SYMSTT, then the term $(k\,X^1\,\ldots\,X^n)_{\mu\to o}$ is an atomic QMLSTT-proposition.
- If ϕ and ψ are QMLSTT-propositions, then so are $\neg\,\phi$ and $\phi\vee\psi$.
- If $r_{\mu\to\mu\to o}\in$ SSTT is an accessibility relation constant and if ϕ is an QMLSTT-proposition, then $\Box\,r\,\phi$ is a QMLSTT-proposition.
- If $X_{\iota}\in$ IVSTT is an individual variable and ϕ is a QMLSTT-proposition then $\mathbf{\Pi}^{\iota}(\lambda X_{\iota}.\phi)$ is a QMLSTT-proposition.
- If $P_{\mu\to o}\in$ PVSTT is a propositional variable and ϕ is a QMLSTT-proposition then $\mathbf{\Pi}^{\mu\to o}(\lambda P_{\mu\to o}.\phi)$ is a QMLSTT-proposition.

We write $\Box_r\,\phi$, $\forall X_\iota.\,\phi$, and $\forall P_{\mu\to o}.\,\phi$ for $\Box\,r\,\phi$, $\mathbf{\Pi}^\iota(\lambda X_\iota.\,\phi)$, and $\mathbf{\Pi}^{\mu\to o}(\lambda P_{\mu\to o}.\,\phi)$, respectively.

Because the defining equations in Definition 3.1 are themselves formulas in simple type theory, we can express proof problems in a higher-order theorem prover elegantly in the syntax of quantified multimodal logic. Using rewriting or definition expanding, we can reduce these representations to corresponding statements containing only the basic connectives $\neg$, $\vee$, $=$, Π^ι, and $\Pi^{\mu\to o}$ of simple type theory.

Example. The following QMLSTT proof problem expresses that in all accessible worlds there exists truth:

$$\Box_r\,\exists P_{\mu\to o}.\,P$$

The term rewrites into the following $\beta\eta$-normal term of type $\mu\to o$

$$\lambda W_\mu.\forall Y_\mu.\neg(r\,W\,Y)\vee(\neg\forall P_{\mu\to o}.\neg(P\,Y))$$

Next, we define validity of QMLSTT propositions $\phi_{\mu\to o}$ in the obvious way: a QML-proposition $\phi_{\mu\to o}$ is valid if and only if for all possible worlds w_μ we have $w_\mu\in\phi_{\mu\to o}$, that is, if and only if $\phi_{\mu\to o}\,w_\mu$ holds.

Definition 3.3 (Validity). Validity is modeled as an abbreviation for the following simply typed λ-term:

$$\text{valid}=\lambda\phi_{\mu\to o}.\forall W_\mu.\,\phi\,W$$

Alternatively, we could define validity simply as $\Pi_{(\mu\to o)\to o}$.

Example. We analyze whether the proposition $\Box_r\,\exists P_{\mu\to o}.\,P$ is valid or not. For this, we formalize the following proof problem

$$\text{valid}\;(\Box_r\,\exists P_{\mu\to o}.\,P)$$

Expanding this term leads to

$$\forall W_\mu.\forall Y_\mu.\neg(r\,W\,Y)\vee(\neg\forall X_{\mu\to o}.\neg(X\,Y))$$

It is easy to check that this term is valid in Henkin semantics: put $X=\lambda Y_\mu.\top$.

An obvious question is whether the notion of quantified multimodal logics we obtain via this embedding indeed exhibits the desired properties. In the next section, we prove soundness and completeness for a mapping of QML-propositions to QMLSTT-propositions.

4. Soundness and Completeness of the Embedding

In our soundness proof, we exploit the following mapping of QKπ models into Henkin models. We assume that the QML logic L under consideration is constructed as outlined in Section 2 from a set of individual variables IV, a set of propositional variables PV, and a set of predicate symbols SYM. Let $\Box_{r^1},\ldots,\Box_{r^n}$ for $r^i\in S$ be the box operators of L.

Definition 4.1 (QMLSTT logic $L^{\mathbf{STT}}$ for QML logic L). Given an QML logic L, define a mapping $\dot{}$ as follows:

$$\dot{X} = X_\iota \text{ for every } X \in \text{IV}$$
$$\dot{P} = P_{\mu\to o} \text{ for every } P \in \text{PV}$$
$$\dot{k} = k_{\iota^n\to(\mu\to o)} \text{ for every n-ary } k \in \text{SYM}$$
$$\dot{r} = r_{\mu\to\mu\to o} \text{ for every } r \in S$$

The QMLSTT logic L^{STT} is obtained from L by applying Def. 3.2 with $\text{IVSTT} = \{\dot{X} \mid X \in \text{IV}\}$, $\text{PVSTT} = \{\dot{P} \mid P \in \text{PV}\}$, $\text{SYMSTT} = \{\dot{k} \mid k \in \text{SYM}\}$, and $\text{SSTT} = \{\dot{r} \mid r \in S\}$. Our construction obviously induces a one-to-one correspondence $\dot{}$ between languages L and L^{STT}.

Moreover, let $g = (g^{iv} : \text{IV} \longrightarrow D, g^{pv} : \text{PV} \longrightarrow P)$ be a variable assignment for L. We define the corresponding variable assignment

$$\dot{g} = (\dot{g}^{iv} : \text{IVSTT} \longrightarrow D = D_\iota,\ \dot{g}^{pv} : \text{PVSTT} \longrightarrow P = D_{\mu\to o})$$

for L^{STT} so that $\dot{g}(X_\iota) = \dot{g}(\dot{X}) = g(X)$ and $\dot{g}(P_{\mu\to o}) = \dot{g}(\dot{P}) = g(P)$ for all $X_\iota \in \text{IVSTT}$ and $P_{\mu\to o} \in \text{PVSTT}$.

Finally, a variable assignment $\dot{g}$ is lifted to an assignment for variables Z_α of arbitrary type by choosing $\dot{g}(Z_\alpha) = d \in D_\alpha$ arbitrarily, if $\alpha \neq \iota, \mu \to o$.

We assume below that L, L^{STT}, g and $\dot{g}$ are defined as above.

Definition 4.2 (Henkin model H^Q for QKπ model Q). Given a QKπ model $Q = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ for L, a Henkin model $H^Q = \langle \{D_\alpha\}_{\alpha\in \text{T}}, I\rangle$ for L^{STT} is constructed as follows. We choose

- the set D_μ as the set of possible worlds W,
- the set D_ι as the set of individuals D (cf. definition of $\dot{g}^{iv}$),
- the set $D_{\mu\to o}$ as the set of sets of possible worlds P (cf. definition of $\dot{g}^{pv}$),[2]
- the set $D_{\mu\to\mu\to o}$ as the set of relations $(R_r)_{r\in S}$,
- and all other sets $D_{\alpha\to\beta}$ as (not necessarily full) sets of functions from D_α to D_β; for all sets $D_{\alpha\to\beta}$ the rule that everything denotes must be obeyed, in particular, we require that the sets $D_{\iota^n\to(\mu\to o)}$ contain the elements $Ik_{\iota^n\to(\mu\to o)}$ as characterized below.

The interpretation I is as follows:

- Let $k_{\iota^n\to(\mu\to o)} = \dot{k}$ for $k \in \text{SYM}$ and let $X^i_\iota = \dot{X}^i$ for $X^i \in \text{IV}$. We choose $Ik_{\iota^n\to(\mu\to o)} \in D_{\iota^n\to(\mu\to o)}$ such that

  $$(I\,k)(\dot{g}(X^1_\iota), \ldots, \dot{g}(X^n_\iota), w) = T$$

  for all worlds $w \in D_\mu$ such that $Q, g, w \models k(X^1, \ldots, X^n)$, that is, if $\langle g(X^1), \ldots, g(X^n)\rangle \in I_w(k)$. Otherwise $(I\,k)(\dot{g}(X^1_\iota), \ldots, \dot{g}(X^n_\iota), w) = F$.

[2] To keep things simple, we identify sets with their characteristic functions.

- Let $r_{\mu\to\mu\to o} = \dot{r}$ for $r \in S$. We choose $Ir_{\mu\to\mu\to o} \in D_{\mu\to\mu\to o}$ such that $(Ir_{\mu\to\mu\to o})(w, w') = T$ if $\langle w, w'\rangle \in R_r$ in Q and $(Ir_{\mu\to\mu\to o})(w, w') = F$ otherwise.

It is not hard to verify that $H^Q = \langle \{D_\alpha\}_{\alpha\in \mathrm{T}}, I\rangle$ is a Henkin model.

Lemma 4.3. *Let $Q = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ be a QKπ model and let $H^Q = \langle \{D_\alpha\}_{\alpha\in T}, I\rangle$ be a Henkin model for Q. Furthermore, let $s_{\mu\to o} = \dot{s}$ for $s \in L$. Then for all worlds $w \in W$ and variable assignments g we have $Q, g, w \models s$ in Q if and only if $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(s_{\mu\to o} W_\mu) = T$ in H^Q.*

Proof. The proof is by induction on the structure of $s \in L$.

Let $s = P$ for $P \in \mathrm{PV}$. By construction of Henkin model H^Q and by definition of $\dot{g}$, we have for $P_{\mu\to o} = \dot{P}$ that $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(P_{\mu\to o}\, W_\mu) = \dot{g}(P_{\mu\to o})(w) = T$ if and only if $Q, g, w \models P$, that is, $w \in g(P)$.

Let $s = k(X^1, \ldots, X^n)$ for $k \in \mathrm{SYM}$ and $X^i \in \mathrm{IV}$. By construction of Henkin model H^Q and by definition of $\dot{g}$, we have for $\dot{k}(\dot{X}^1, \ldots, \dot{X}^n) = (k_{\iota^n\to(\mu\to o)}\, X^1_\iota \ \ldots\ X^n_\iota)$ that

$$\mathrm{V}_{[w/W_\mu]\dot{g}}\,((k_{\iota^n\to(\mu\to o)} X^1_\iota \ldots X^n_\iota)\, W_\mu) = (I\, k)(\dot{g}(X^1_\iota), \ldots, \dot{g}(X^n_\iota), w) = T$$

if and only if $Q, g, w \models k(X^1, \ldots, X^n)$, that is, $\langle g(X^1), \ldots, g(X^n)\rangle \in I_w(k)$.

Let $s = \neg\, t$ for $t \in L$. We have $Q, g, w \models \neg s$ if and only $Q, g, w \not\models s$, which is equivalent by induction to $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(t_{\mu\to o}\, W_\mu) = F$ and hence to $\mathrm{V}_{[w/W_\mu]\dot{g}}\,\neg(t_{\mu\to o}\, W_\mu) =_{\beta\eta} \mathrm{V}_{[w/W_\mu]\dot{g}}\,((\neg\ t_{\mu\to o})\, W_\mu) = T$.

Let $s = (t \vee l)$ for $t, l \in L$. We have $Q, g, w \models (t \vee l)$ if and only if $Q, g, w \models t$ or $Q, g, w \models l$. The latter condition is equivalent by induction to $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(t_{\mu\to o}\ W_\mu) = T$ or $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(l_{\mu\to o}\ W_\mu) = T$ and therefore to $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(t_{\mu\to o}\ W_\mu) \vee (l_{\mu\to o}\, W_\mu) =_{\beta\eta} \mathrm{V}_{[w/W_\mu]\dot{g}}\,(t_{\mu\to o}\ \vee\ l_{\mu\to o}\ W_\mu) = T$.

Let $s = \Box_r\, t$ for $t \in L$. We have $Q, g, w \models \Box_r\, t$ if and only if for all u with $\langle w, u\rangle \in R_r$ we have $Q, g, u \models t$. The latter condition is equivalent by induction to this one: for all u with $\langle w, u\rangle \in R_r$ we have $\mathrm{V}_{[u/V_\mu]\dot{g}}\,(t_{\mu\to o}\, V_\mu) = T$. That is equivalent to

$$\mathrm{V}_{[u/V_\mu][w/W_\mu]\dot{g}}\,(\neg(r_{\mu\to\mu\to o}\, W_\mu\, V_\mu) \vee (t_{\mu\to o}\, V_\mu)) = T$$

and thus to

$$\mathrm{V}_{[w/W_\mu]\dot{g}}\,(\forall Y_\mu\mathbf{.}\,(\neg(r_{\mu\to\mu\to o}\, W_\mu\, Y_\mu) \vee (t_{\mu\to o}\, Y_\mu))) =_{\beta\eta} \mathrm{V}_{[w/W_\mu]\dot{g}}\,(\Box_r\, t\, W_\mu) = T.$$

Let $s = \forall X\mathbf{.}\, t$ for $t \in L$ and $X \in \mathrm{IV}$. We have $Q, g, w \models \forall X\mathbf{.}\, t$ if and only if $Q, [d/X]g, w \models t$ for all $d \in D$. The latter condition is equivalent by induction to $\mathrm{V}_{[d/X_\iota][w/W_\mu]\dot{g}}\,(t_{\mu\to o}\, W_\mu) = T$ for all $d \in D_\iota$. That condition is equivalent to

$$\mathrm{V}_{[w/W_\mu]\dot{g}}\,(\Pi^\iota_{(\iota\to o)\to o}(\lambda X_\iota\mathbf{.}\, t_{\mu\to o}\, W_\mu)) =_{\beta\eta} \mathrm{V}_{[w/W_\mu]\dot{g}}\,((\lambda V_\mu\mathbf{.}\,(\Pi^\iota_{(\iota\to o)\to o}\,(\lambda X_\iota\mathbf{.}\, t_{\mu\to o}\, V_\mu)))\, W_\mu) = T$$

and so by definition of $\mathbf{\Pi}^\iota$ to $\mathrm{V}_{[w/W_\mu]\dot{g}}\,((\mathbf{\Pi}^\iota_{(\iota\to(\mu\to o))\to(\mu\to o)}\,(\lambda X_\iota\mathbf{.}\, t_{\mu\to o}))\, W_\mu) = \mathrm{V}_{[w/W_\mu]\dot{g}}\,((\forall X_\iota\mathbf{.}\ t_{\mu\to o})\, W_\mu) = T$.

The case for $s = \forall P\mathbf{.}\, t$ where $t \in L$ and $P \in \mathrm{PV}$ is analogous to $s = \forall X\mathbf{.}\, t$. □

We exploit this result to prove the soundness of our embedding.

Theorem 4.4 (Soundness for QKπ semantics). *Let $s \in L$ be a QML proposition and let $s_{\mu\to o} = \dot{s}$ be the corresponding QMLSTT proposition. If $\models^{STT} (valid\, s_{\mu\to o})$ then $\models^{QK\pi} s$.*

Proof. By contraposition, assume $\not\models^{\mathrm{QK}\pi} s$: that is, there is a QK$\pi$ model $Q = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$, a variable assignment g and a world $w \in W$, such that $Q, g, w \not\models s$. By Lemma 4.3, we have $\mathrm{V}_{[w/W_\mu]\dot{g}}\,(s_{\mu\to o}\, W_\mu) = F$ in a Henkin model H^Q for Q. Thus, $\mathrm{V}_{\dot{g}}\,(\forall W_{\mu\blacksquare}(s_{\mu\to o}\, W)) =_{\beta\eta} \mathrm{V}_{\dot{g}}\,(\mathrm{valid}\, s_{\mu\to o}) = F$. Hence, $\not\models^{\mathrm{STT}} (\mathrm{valid}\, s_{\mu\to o})$. □

In order to prove completeness, we reverse our mapping from Henkin models to QKπ models.

Definition 4.5 (QML logic $L^{\mathbf{QML}}$ for QMLSTT logic L). The mapping $\bar{\_}$ is defined as the reverse map of $\dot{\_}$ from Def. 4.1.

The QML logic L^{QML} is obtained from QMLSTT logic L by choosing $\mathrm{IV} = \{\bar{X}_\iota \mid X_\iota \in \mathrm{IVSTT}\}$, $\mathrm{PV} = \{\bar{P}_{\mu\to o} \mid P_{\mu\to o} \in \mathrm{PVSTT}\}$, $\mathrm{SYM} = \{\bar{k}_{\iota^n\to(\mu\to o)} \mid k_{\iota^n\to(\mu\to o)} \in \mathrm{SYMSTT}\}$, and $S = \{\bar{r}_{\mu\to\mu\to o} \mid r_{\mu\to\mu\to o} \in \mathrm{SSTT}\}$.

Moreover, let $g : \mathrm{IVSTT}\cup\mathrm{PVSTT} \longrightarrow D\cup P$ be a variable assignment for L. The corresponding variable assignment $\bar{g} : \mathrm{IV}\cup\mathrm{PV} \longrightarrow D\cup P$ for L^{QML} is defined as follows: $\bar{g}(X) = \bar{g}(\bar{X}_\iota) = g(X_\iota)$ and $\bar{g}(P) = \bar{g}(\bar{P}_{\mu\to o}) = g(P_{\mu\to o})$ for all $X \in \mathrm{IV}$ and $P \in \mathrm{PV}$.

We assume below that L, L^{QML}, g and $\bar{g}$ are defined as above.

Definition 4.6 (QKπ^- model Q^H for Henkin model H). Given a Henkin model $H = \langle\{D_\alpha\}_{\alpha\in\mathrm{T}}, I\rangle$ for QMLSTT logic L, we construct a QML model $Q^H = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ for L^{QML} by choosing $W = D_\mu$, $D = D_\iota$, and $P = D_{\mu\to o}$. Moreover, let $k = \bar{k}_{\iota^n\to(\mu\to o)}$ and let $X^i = \bar{X}^i_\iota$. We choose $I_w(k)$ such that $\langle\bar{g}(X^1), \ldots, \bar{g}(X^n)\rangle \in I_w(k)$ if and only if

$$(I\, k)(g(X^1_\iota), \ldots, g(X^n_\iota), w) = T.$$

Finally, let $r = \bar{r}_{\mu\to\mu\to o}$. We choose R_r such that $\langle w, w'\rangle \in R_r$ if and only if $(Ir_{\mu\to\mu\to o})(w, w') = T$.

It is not hard to verify that $Q^H = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ meets the definition of QKπ^- models. Below we will see that it also meets the definition of QKπ models.

Lemma 4.7. *Let $Q^H = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ be a QKπ^- model for a given Henkin model $H = \langle\{D_\alpha\}_{\alpha\in T}, I\rangle$. Furthermore, let $s = \bar{s}_{\mu\to o}$. For all worlds $w \in W$ and variable assignments g we have $V_{[w/W_\mu]g}\,(s_{\mu\to o} W_\mu) = T$ in H if and only if $Q^H, \bar{g}, w \models s$ in Q^H.*

Proof. The proof is by induction on the structure of $s_{\mu\to o} \in L$ and it is similar to the proof of Lemma 4.3. □

With the help of Lemma 4.7, we now show that the QKπ^- models we construct in Def. 4.6 are in fact always QKπ models. Thus, Henkin models never relate to QKπ^- models that do not already fulfill the QKπ criterion.

Lemma 4.8. *Let $Q^H = (W, (R_r)_{r\in S}, D, P, (I_w)_{w\in W})$ be a QKπ^- model for a given Henkin model $H = \langle \{D_\alpha\}_{\alpha\in T}, I\rangle$. Then Q^H is also a QKπ model.*

Proof. We need to show that for every variable assignment $\bar{g}$ and formula $s = \bar{s}_{\mu\to o}$ the set $\{w \in W \mid Q^h, \bar{g}, w \models s\}$ is a member of P in Q^H. This is a consequence of the rule that everything denotes in the Henkin model H. To see this, consider $\mathrm{V}_g s_{\mu\to o} = \mathrm{V}_g(\lambda V_\mu . s_{\mu\to o} V)$ for variable V_μ not occurring free in $s_{\mu\to o}$. By definition of Henkin models this denotes that function from $D_\mu = W$ to truth values $D_o = \{T, F\}$ whose value for each argument $w \in D_\mu$ is $\mathrm{V}_{[w/V_\mu]g}(s\, V)$, that is, $s_{\mu\to o}$ denotes the characteristic function $\lambda w \in W . \mathrm{V}_{[w/V_\mu]g}(s_{\mu\to o} V_\mu) = T$ which we identify with the set $\{w \in W \mid \mathrm{V}_{[w/V_\mu]g}(s_{\mu\to o} V_\mu) = T\}$. Hence, we have $\{w \in W \mid \mathrm{V}_{[w/V_\mu]g}(s_{\mu\to o} V_\mu) = T\} \in D_{\mu\to o}$. By the choice of $P = D_{\mu\to o}$ in the construction of Q^H we know $\{w \in W \mid \mathrm{V}_{[w/V_\mu]g}(s_{\mu\to o} V_\mu) = T\} \in P$. By Lemma 4.7 we get $\{w \in W \mid Q^h, \bar{g}, w \models s\} \in P$. □

Theorem 4.9 (Completeness for QKπ models). *Let $s_{\mu\to o}$ be a QMLSTT proposition and let $s = \bar{s}_{\mu\to o}$ be the corresponding QML proposition. If $\models^{QK\pi} s$ then $\models^{STT} (valid\, s_{\mu\to o})$.*

Proof. By contraposition, assume $\not\models^{\mathrm{STT}} (\mathrm{valid}\, s_{\mu\to o})$: there is a Henkin model $H = \langle \{D_\alpha\}_{\alpha\in \mathrm{T}}, I\rangle$ and a variables assignment g such that $\mathrm{V}_g(\mathrm{valid}\, s_{\mu\to o}) = F$. Hence, for some world $w \in D_\mu$ we have $\mathrm{V}_{[w/W_\mu]g}(s_{\mu\to o} W_\mu) = F$. By Lemma 4.7 we then get $Q^H, \bar{g}, w \not\models^{\mathrm{QK}\pi^-} s$ for $s = \bar{s}_{\mu\to o}$ in QKπ^- model Q^H for H. By Lemma 4.8 we know that Q^H is actually a QKπ model. Hence, $\not\models^{\mathrm{QK}\pi} s$. □

Our soundness and completeness results obviously also apply to fragments of QML logics.

Corollary 4.10. *The reduction of our embedding to propositional quantified multimodal logics (which only allow quantification over propositional variables) is sound and complete.*

Corollary 4.11. *The reduction of our embedding to first-order multimodal logics (which only allow quantification over individual variables) is sound and complete.*

Corollary 4.12. *The reduction of our embedding to propositional multimodal logics (no quantification) is sound and complete.*

5. Conclusion

We have presented a straightforward embedding of quantified multimodal logics in simple type theory and we have shown that this embedding is sound

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\lambda W_\iota.\neg(\phi\, W)$$
$$\vee = \lambda\phi_{\iota\to o}.\lambda\psi_{\iota\to o}.\lambda W_\iota.\phi\, W \vee \psi\, W$$
$$\Box = \lambda R_{\iota\to\iota\to o}.\lambda\phi_{\iota\to o}.\lambda W_\iota.\forall V_\iota.\neg(R\, W\, V) \vee \phi\, V$$
$$(\forall^i) \quad \Pi^\mu = \lambda\tau.\lambda W.\forall X.(\tau\, X)\, W$$
$$(\forall^p) \quad \Pi^{\iota\to o} = \lambda\tau.\lambda W.\forall P.(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\neg(\phi\, W)$$

$$\vee = \lambda\phi_{\iota\to o}.\,\lambda\psi_{\iota\to o}.\,\lambda W_{\iota}.\,\phi\, W \vee \psi\, W$$

$$\Box = \lambda R_{\iota\to\iota\to o}.\,\lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\forall V_{\iota}.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^{i}) \quad \Pi^{\mu} = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall X_{\mu}.\,(\tau\, X)\, W$$

$$(\forall^{p}) \quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\neg(\phi\, W)$$

$$\vee\ \phi = \lambda\psi_{\iota\to o}.\,\lambda W_{\iota}.\,\phi\, W \vee \psi\, W$$

$$\Box = \lambda R_{\iota\to\iota\to o}.\,\lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\forall V_{\iota}.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^{i}) \quad \Pi^{\mu} = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall X_{\mu}.\,(\tau\, X)\, W$$

$$(\forall^{p}) \quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_\iota.\,\neg(\phi\, W)$$

$$\vee\ \phi\ \psi = \lambda W_\iota.\,\phi\, W \vee \psi\, W$$

$$\Box = \lambda R_{\iota\to\iota\to o}.\,\lambda\phi_{\iota\to o}.\,\lambda W_\iota.\,\forall V_\iota.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^i)\quad \Pi^\mu = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_\iota.\,\forall X_\mu.\,(\tau\, X)\, W$$

$$(\forall^p)\quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_\iota.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\neg(\phi\, W)$$

$$(\vee\,\phi\,\psi)\,W = \phi\, W \vee \psi\, W$$

$$\Box = \lambda R_{\iota\to\iota\to o}.\,\lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\forall V_{\iota}.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^{i}) \quad \Pi^{\mu} = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall X_{\mu}.\,(\tau\, X)\, W$$

$$(\forall^{p}) \quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\neg(\phi\, W)$$

$$\vee = \lambda\phi_{\iota\to o}.\,\lambda\psi_{\iota\to o}.\,\lambda W_{\iota}.\,\phi\, W \vee \psi\, W$$

$$\Box = \lambda R_{\iota\to\iota\to o}.\,\lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\forall V_{\iota}.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^{i}) \quad \Pi^{\mu} = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall X_{\mu}.\,(\tau\, X)\, W$$

$$(\forall^{p}) \quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\neg(\phi\, W)$$

$$\vee = \lambda\phi_{\iota\to o}.\,\lambda\psi_{\iota\to o}.\,\lambda W_{\iota}.\,\phi\, W \vee \psi\, W$$

$$\Box_R = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\forall V_{\iota}.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^i) \quad \Pi^{\mu} = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall X_{\mu}.\,(\tau\, X)\, W$$

$$(\forall^p) \quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

— related, but significantly extending (Ohlbach, 1988/93) —

Straightforward encoding

- base type ι: non-empty set of possible worlds
- base type μ: non-empty set of individuals

QML formulas $\longrightarrow$ HOL terms of type $\iota \to o$

QML operators as abbreviations for specific HOL terms

$$\neg = \lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\neg(\phi\, W)$$

$$\vee = \lambda\phi_{\iota\to o}.\,\lambda\psi_{\iota\to o}.\,\lambda W_{\iota}.\,\phi\, W \vee \psi\, W$$

$$\Box = \lambda R_{\iota\to\iota\to o}.\,\lambda\phi_{\iota\to o}.\,\lambda W_{\iota}.\,\forall V_{\iota}.\,\neg(R\, W\, V) \vee \phi\, V$$

$$(\forall^{i}) \quad \Pi^{\mu} = \lambda\tau_{\mu\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall X_{\mu}.\,(\tau\, X)\, W$$

$$(\forall^{p}) \quad \Pi^{\iota\to o} = \lambda\tau_{(\iota\to o)\to(\iota\to o)}.\,\lambda W_{\iota}.\,\forall P_{\iota\to o}.\,(\tau\, P)\, W$$

(Normal) QML as Fragment of HOL

Encoding of validity

$$\mathsf{valid} \quad = \quad \lambda\phi_{\iota\to o}.\forall W_{\iota}.\phi\, W$$

Example: In all r-accessible worlds exists truth

Formulate problem in HOL using original QML syntax

$$\text{valid}\ \Box_r\ \exists^p P_{\iota\to o}.\ P$$

then automatically rewrite abbreviations

$$\begin{aligned}
\Box_r &\xrightarrow{\text{rewrite}} \ldots \\
\exists^p &\xrightarrow{\text{rewrite}} \ldots \\
\text{valid} &\xrightarrow{\text{rewrite}} \ldots \\
&\xrightarrow{\beta\eta\downarrow} \forall W_\iota.\forall Y_\iota.\neg r\ W\ Y \vee (\neg\forall P_{\iota\to o}.\neg(P\ Y))
\end{aligned}$$

and prove automatically (LEO-II, IsabelleP, TPS, Satallax, ...
here the provers need to generate witness term $P = \lambda Y_\iota.\top$)

Example: In all r-accessible worlds exists truth

Formulate problem in HOL using original QML syntax

$$\text{valid}\ \Box_r\ \exists^p P_{\iota\to o}.\ P$$

then automatically rewrite abbreviations

$$\begin{aligned}
\Box_r &\xrightarrow{\text{rewrite}} \ldots \\
\exists^p &\xrightarrow{\text{rewrite}} \ldots \\
\text{valid} &\xrightarrow{\text{rewrite}} \ldots \\
&\xrightarrow{\beta\eta\downarrow} \forall W_\iota.\forall Y_\iota.\neg r\ W\ Y \vee (\neg\forall P_{\iota\to o}.\neg(P\ Y))
\end{aligned}$$

and prove automatically (LEO-II, IsabelleP, TPS, Satallax, ...
here the provers need to generate witness term $P = \lambda Y_\iota.\top$)

Example: In all r-accessible worlds exists truth

Formulate problem in HOL using original QML syntax

$$\text{valid}\ \Box_r\ \exists^p\ P_{\iota\to o}.\ P$$

then automatically rewrite abbreviations

$$\begin{aligned}
\Box_r &\xrightarrow{\text{rewrite}} \ldots \\
\exists^p &\xrightarrow{\text{rewrite}} \ldots \\
\text{valid} &\xrightarrow{\text{rewrite}} \ldots \\
&\xrightarrow{\beta\eta\downarrow} \forall W_\iota.\forall Y_\iota.\neg r\ W\ Y \vee (\neg\forall P_{\iota\to o}.\neg(P\ Y))
\end{aligned}$$

and prove automatically (LEO-II, IsabelleP, TPS, Satallax, ...
here the provers need to generate witness term $P = \lambda Y_\iota.\top$)

Soundness and Completeness

Soundness and Completeness Theorem:

$$\models^{QML}_{\mathbf{QK}\pi} s \quad \text{if and only if} \quad \models^{HOL}_{Henkin} \text{valid}\, s_{\iota\to o}$$

(BenzmüllerPaulson, Techn.Report, 2009)

Soundness and Completeness Theorem for Propositional Multimodal Logic

(BenzmüllerPaulson, Log.J.IGPL, 2010)

Further interesting Fragments of HOL

- Intuitionistic Logic
  (exploiting Gödel's translation to S4)
  (BenzmüllerPaulson, Log.J.IGPL, 2010)
- Access Control Logics
  (exploiting a translation by Garg and Abadi)
  (Benzmüller, IFIP SEC, 2009)
- Region Connection Calculus — later in this talk
- ...

Reasoning *about* Combinations of Logics

Reasoning *about* Combinations of Logics: Correspondence

Correspondences between properties of accessibility relations like

$$\begin{aligned} \text{symmetric} &= \lambda R.\forall S,T.\, R\,S\,T \Rightarrow R\,T\,S \\ \text{serial} &= \lambda R.\forall S.\exists T.\, R\,S\,T \end{aligned}$$

and corresponding axioms

$$\forall R.\,\text{symmetric}\,R \overset{0.0s}{\Leftarrow} \overset{0.0s}{\Rightarrow} \text{valid}\ \forall^{p}\phi.\,\phi \supset \Box_R \Diamond_R \phi \qquad (B)$$

$$\forall R.\,\text{serial}\,R \overset{0.0s}{\Leftarrow} \overset{0.0s}{\Rightarrow} \text{valid}\ \forall^{p}\phi.\,\Box_R \phi \supset \Diamond_R \phi \qquad (D)$$

Such proofs — including axioms D, M, 4, B, 5 — can be automated with LEO-II in no-time!

Reasoning *about* Combinations of Logics: Correspondence

Correspondences between properties of accessibility relations like

$$\begin{aligned} \text{symmetric} &= \lambda R.\forall S,T.\, R\,S\,T \Rightarrow R\,T\,S \\ \text{serial} &= \lambda R.\forall S.\exists T.\, R\,S\,T \end{aligned}$$

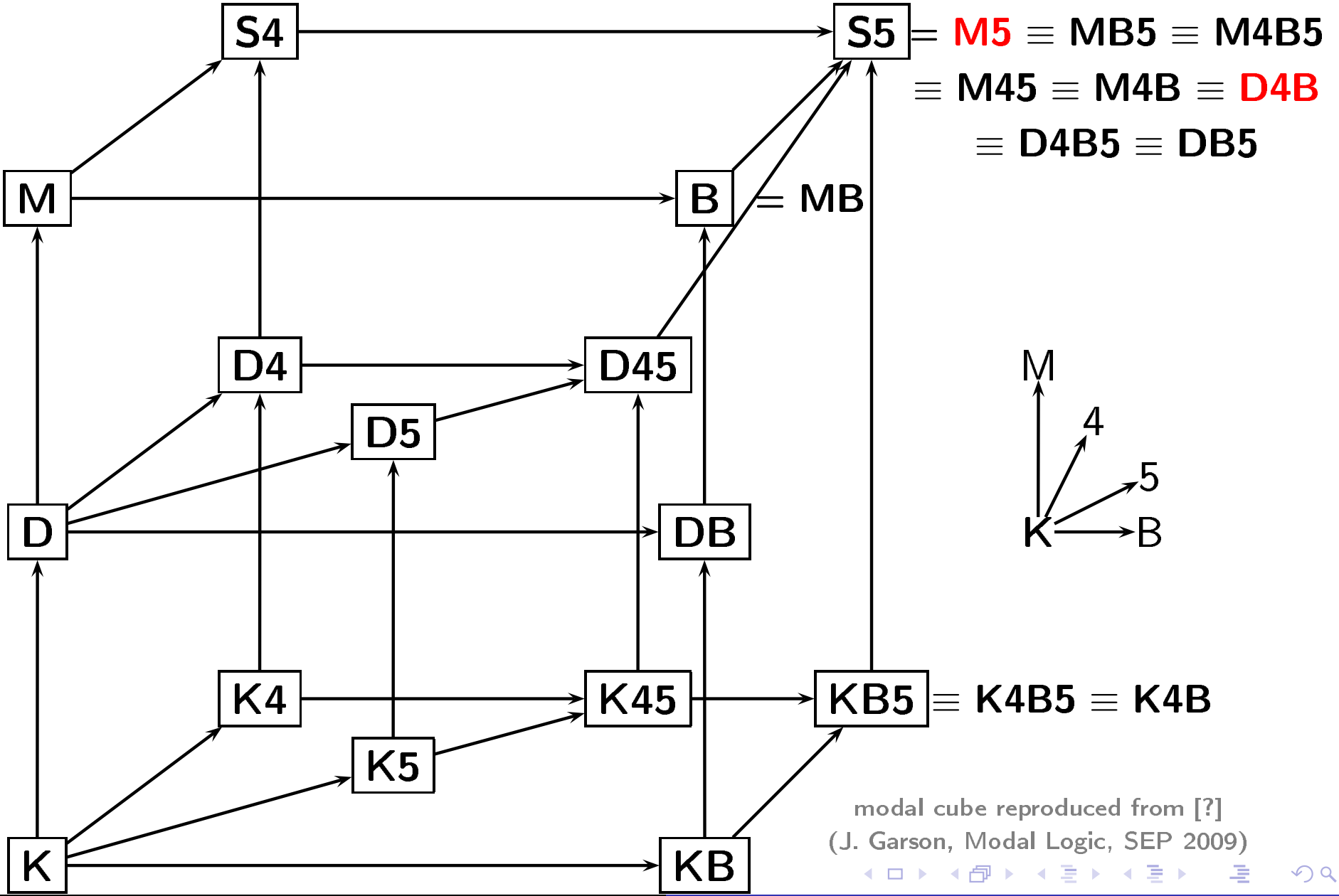
and corresponding axioms

$$\forall R.\,\text{symmetric}\,R \overset{0.0s}{\Leftarrow} \overset{0.0s}{\Rightarrow} \text{valid}\ \forall^p \phi.\, \phi \supset \Box_R \Diamond_R \phi \qquad (B)$$

$$\forall R.\,\text{serial}\,R \overset{0.0s}{\Leftarrow} \overset{0.0s}{\Rightarrow} \text{valid}\ \forall^p \phi.\, \Box_R \phi \supset \Diamond_R \phi \qquad (D)$$

Such proofs — including axioms D, M, 4, B, 5 — can be automated with LEO-II in no-time!

Reasoning *about* Combinations of Logics: Modal Cube



modal cube reproduced from [?]
(J. Garson, Modal Logic, SEP 2009)

Reasoning *about* Combinations of Logics: M5 $\Leftrightarrow$ D4B

$$\forall R.$$

$$\left.\begin{array}{ll} & \text{valid } \forall^p \phi. \Box_R \phi \supset \phi \\ \wedge & \text{valid } \forall^p \phi. \Diamond_R \phi \supset \Box_R \Diamond_R \phi \end{array}\right\} M5$$

$$\Leftrightarrow$$

$$\left.\begin{array}{ll} & \text{valid } \forall^p \phi. \Box_R \phi \supset \Diamond_R \phi \\ \wedge & \text{valid } \forall^p \phi. \Box_R \phi \supset \Box_R \Box_R \phi \\ \wedge & \text{valid } \forall^p \phi. \phi \supset \Box_R \Diamond_R \phi \end{array}\right\} D4B$$

Reasoning *about* Combinations of Logics: M5 $\Leftrightarrow$ D4B

$\forall R.$

$$\left.\begin{array}{ll} & \text{valid } \forall^p \phi. \Box_R \phi \supset \phi \\ \wedge & \text{valid } \forall^p \phi. \Diamond_R \phi \supset \Box_R \Diamond_R \phi \end{array}\right\} M5$$

$\Leftrightarrow$

$$\left.\begin{array}{ll} & \text{serial } R \\ \wedge & \text{valid } \forall^p \phi. \Box_R \phi \supset \Box_R \Box_R \phi \\ \wedge & \text{symmetric } R \end{array}\right\} D4B$$

Reasoning *about* Combinations of Logics: M5 $\Leftrightarrow$ D4B

$\forall R.$

$$\left.\begin{array}{ll} & \text{reflexive } R \\ \wedge & \text{euclidean } R \end{array}\right\} M5$$

$\Leftrightarrow$

$$\left.\begin{array}{ll} & \text{serial } R \\ \wedge & \text{transitive } R \\ \wedge & \text{symmetric } R \end{array}\right\} D4B$$

Reasoning *about* Combinations of Logics: M5 $\Leftrightarrow$ D4B

$\forall R.$

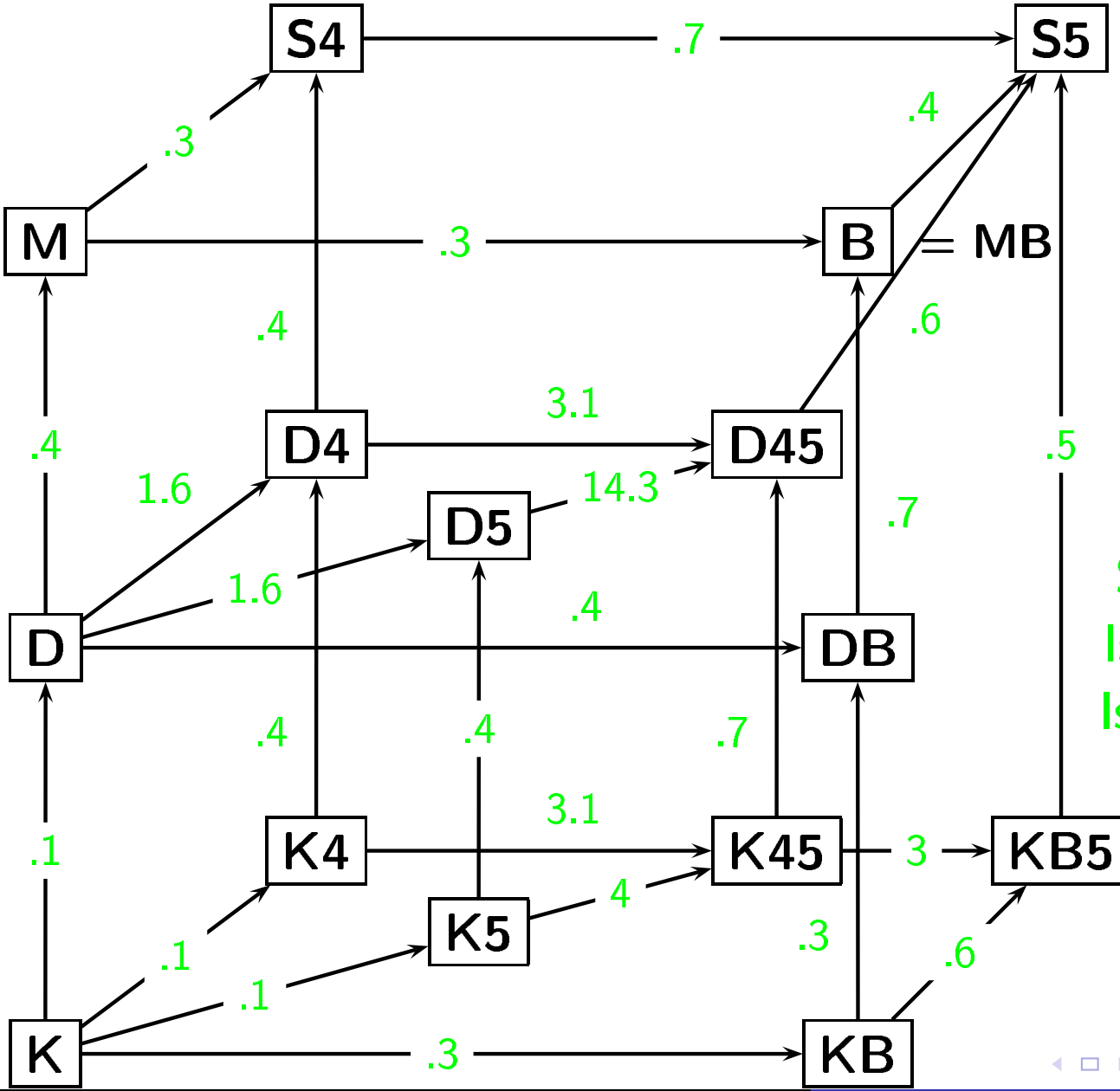
$$\left.\begin{array}{ll} & \text{reflexive } R \\ \wedge & \text{euclidean } R \end{array}\right\} M5$$

$\overset{0.1s}{\Leftrightarrow}$

$$\left.\begin{array}{ll} & \text{serial } R \\ \wedge & \text{transitive } R \\ \wedge & \text{symmetric } R \end{array}\right\} D4B$$

Proof with LEO-II in 0.1s

Reasoning *about* Combinations of Logics: Cube Verification



$= \mathbf{M5} \overset{.1}{\equiv} \mathbf{MB5} \overset{.2}{\equiv} \mathbf{M4B5} \overset{.2}{\equiv} \mathbf{M45} \overset{.1}{\equiv} \mathbf{M4B} \overset{.1}{\equiv} \mathbf{D4B} \overset{.2}{\equiv} \mathbf{D4B5} \overset{.1}{\equiv} \mathbf{DB5}$

fastest results by:

LEO-II (prover)
Satallax (prover)
TPS (prover)
Satallax (mod.find.)
IsabelleN (mod.find.)
IsabelleM (mod.find.)

$\overset{.2}{\equiv} \mathbf{K4B5} \overset{.1}{\equiv} \mathbf{K4B}$

$\sum < 40\text{sec.}$

Reasoning *about* Combinations of Logics: Segerberg

(Segerberg, 1973) discusses a 2-dimensional logic providing two S5 modalities $\Box_a$ and $\Box_b$. He adds further axioms stating that these modalities are commutative and orthogonal. It actually turns out that orthogonality is already implied in this context.

reflexive a, transitive a, euclid. a,

reflexive b, transitive b, euclid. b,

valid $\forall \phi.\ \Box_a \Box_b \phi \Leftrightarrow \Box_b \Box_a \phi$

$\vdash^{HOL}$

valid $\forall \phi, \psi.\ \Box_a (\Box_a \phi \vee \Box_b \psi) \supset (\Box_a \phi \vee \Box_a \psi)$

$\wedge$

valid $\forall \phi, \psi.\ \Box_b (\Box_a \phi \vee \Box_b \psi) \supset (\Box_b \phi \vee \Box_b \psi)$

Reasoning *about* Combinations of Logics: Segerberg

(Segerberg, 1973) discusses a 2-dimensional logic providing two S5 modalities $\Box_a$ and $\Box_b$. He adds further axioms stating that these modalities are commutative and orthogonal. It actually turns out that orthogonality is already implied in this context.

$$\begin{array}{ll} & \text{reflexive } a, \text{transitive } a, \text{euclid. } a, \\ & \text{reflexive } b, \text{transitive } b, \text{euclid. } b, \\ & \text{valid } \forall \phi. \ \Box_a \Box_b \phi \Leftrightarrow \Box_b \Box_a \phi \\ \vdash^{HOL} & \qquad\qquad \text{proof by LEO-II in 0.2s} \\ & \text{valid } \forall \phi, \psi. \ \Box_a (\Box_a \phi \vee \Box_b \psi) \supset (\Box_a \phi \vee \Box_a \psi) \\ & \wedge \\ & \text{valid } \forall \phi, \psi. \ \Box_b (\Box_a \phi \vee \Box_b \psi) \supset (\Box_b \phi \vee \Box_b \psi) \end{array}$$

Reasoning *within* Combined Logics

Wise Men Puzzle

Once upon a time, a king wanted to find the wisest out of his three wisest men. He arranged them in a circle and told them that he would put a white or a black spot on their foreheads and that one of the three spots would certainly be white. The three wise men could see and hear each other but, of course, they could not see their faces reflected anywhere. The king, then, asked to each of them to find out the color of his own spot. After a while, the wisest correctly answered that his spot was white.

Wise Men Puzzle

Once upon a time, a king wanted to find the wisest out of his three wisest men. He arranged them in a circle and told them that he would put a white or a black spot on their foreheads and that one of the three spots would certainly be white. The three wise men could see and hear each other but, of course, they could not see their faces reflected anywhere. The king, then, asked to each of them to find out the color of his own spot. After a while, the wisest correctly answered that his spot was white.

(adapted from (Baldoni, PhD, 1998))

- epistemic modalities:

  $\Box_a, \Box_b, \Box_c$: three wise men

  $\Box_{fool}$: common knowledge

- predicate constant:

  ws: 'has white spot'

Wise Men Puzzle

Once upon a time, a king wanted to find the wisest out of his three wisest men. He arranged them in a circle and told them that he would put a white or a black spot on their foreheads and that one of the three spots would certainly be white. The three wise men could see and hear each other but, of course, they could not see their faces reflected anywhere. The king, then, asked to each of them to find out the color of his own spot. After a while, the wisest correctly answered that his spot was white.

(adapted from (Baldoni, PhD, 1998))

- common knowledge:
  at least one of the wise men has a white spot

$$\text{valid}\ \Box_{\text{fool}}\,(ws\,a)\ \vee\ (ws\,b)\ \vee\ (ws\,c)$$

if X one has a white spot then Y can see this

$$(\text{valid}\ \Box_{\text{fool}}\,(ws\,X) \Rightarrow \Box_{Y}\,(ws\,X))$$

if X has not a white spot then Y can see this

$$\text{valid}\ \Box_{\text{fool}}\,\neg\,(ws\,X) \Rightarrow \Box_{Y}\,\neg\,(ws\,X))$$

$$X \neq Y \in \{a, b, c\}$$

Wise Men Puzzle

Once upon a time, a king wanted to find the wisest out of his three wisest men. He arranged them in a circle and told them that he would put a white or a black spot on their foreheads and that one of the three spots would certainly be white. The three wise men could see and hear each other but, of course, they could not see their faces reflected anywhere. The king, then, asked to each of them to find out the color of his own spot. After a while, the wisest correctly answered that his spot was white.

(adapted from (Baldoni, PhD, 1998))

- if X knows ϕ then Y knows this

$$\text{valid}\ \forall^{p}\phi_{\bullet}\ (\Box_X\ \phi \Rightarrow \Box_Y\ \Box_X\ \phi)$$

- if X does not know ϕ then Y knows this

$$\text{valid}\ \forall^{p}\phi_{\bullet}\ (\neg\ \Box_X\ \phi \Rightarrow \Box_Y\ \neg\ \Box_X\ \phi)$$

$$X \neq Y \in \{a, b, c\}$$

- axioms for common knowledge

$$\text{valid}\ \forall^{p}\phi_{\bullet}\ \Box_{\text{fool}}\ \phi \Rightarrow \phi \qquad \text{(M)}$$

$$\text{valid}\ \forall^{p}\phi_{\bullet}\ \Box_{\text{fool}}\ \phi \Rightarrow \Box_{\text{fool}}\ \Box_{\text{fool}}\ \phi \qquad \text{(4)}$$

$$\forall R_{\bullet}\ \text{valid}\ \forall^{p}\phi_{\bullet}\ \Box_{\text{fool}}\ \phi \Rightarrow \Box_R\ \phi$$

Wise Men Puzzle

Once upon a time, a king wanted to find the wisest out of his three wisest men. He arranged them in a circle and told them that he would put a white or a black spot on their foreheads and that one of the three spots would certainly be white. The three wise men could see and hear each other but, of course, they could not see their faces reflected anywhere. The king, then, asked to each of them to find out the color of his own spot. After a while, the wisest correctly answered that his spot was white.

(adapted from (Baldoni, PhD, 1998))

- a, b do not know that they have a white spot

$$\text{valid} \neg \Box_a (ws\ a) \qquad \text{valid} \neg \Box_b (ws\ b)$$

- prove that c does know he has a white spot:

$$\ldots \vdash^{HOL} \text{valid}\, \Box_c (ws\ c)$$

Wise Men Puzzle

Once upon a time, a king wanted to find the wisest out of his three wisest men. He arranged them in a circle and told them that he would put a white or a black spot on their foreheads and that one of the three spots would certainly be white. The three wise men could see and hear each other but, of course, they could not see their faces reflected anywhere. The king, then, asked to each of them to find out the color of his own spot. After a while, the wisest correctly answered that his spot was white.

(adapted from (Baldoni, PhD, 1998))

- a, b do not know that they have a white spot

  $\mathsf{valid} \neg \Box_a (ws\ a)$ $\mathsf{valid} \neg \Box_b (ws\ b)$

- prove that c does know he has a white spot:

  $\ldots \vdash^{HOL} \mathsf{valid}\, \Box_c (ws\ c)$

LEO-II can prove this result in 0.4s

Reasoning *within* Combined Logics: Epistemic & Spatial

Region Connection Calculus (RCC) (RandellCuiCohn, 1992)
as fragment of HOL:

$$
\begin{array}{rcl}
\text{disconnected}: & dc & = \lambda X_\tau.\lambda Y_\tau.\neg(c\ X\ Y) \\
\text{part of}: & p & = \lambda X_\tau.\lambda Y_\tau.\forall Z.((c\ Z\ X) \Rightarrow (c\ Z\ Y)) \\
\text{identical with}: & eq & = \lambda X_\tau.\lambda Y_\tau.((p\ X\ Y) \wedge (p\ Y\ X)) \\
\text{overlaps}: & o & = \lambda X_\tau.\lambda Y_\tau.\exists Z.((p\ Z\ X) \wedge (p\ Z\ Y)) \\
\text{partially o}: & po & = \lambda X_\tau.\lambda Y_\tau.((o\ X\ Y) \wedge \neg(p\ X\ Y) \wedge \neg(p\ Y\ X)) \\
\text{ext. connected}: & ec & = \lambda X_\tau.\lambda Y_\tau.((c\ X\ Y) \wedge \neg(o\ X\ Y)) \\
\text{proper part}: & pp & = \lambda X_\tau.\lambda Y_\tau.((p\ X\ Y) \wedge \neg(p\ Y\ X)) \\
\text{tangential pp}: & tpp & = \lambda X_\tau.\lambda Y_\tau.((pp\ X\ Y) \wedge \exists Z.((ec\ Z\ X) \wedge (ec\ Z\ Y))) \\
\text{nontang. pp}: & ntpp & = \lambda X_\tau.\lambda Y_\tau.((pp\ X\ Y) \wedge \neg\exists Z.((ec\ Z\ X) \wedge (ec\ Z\ Y)))
\end{array}
$$

Reasoning *within* Combined Logics: Epistemic & Spatial

A trivial problem for RCC:

Catalunya is a border region of Spain	(*tpp* catalunya spain),
Spain and France share a border	(*ec* spain france),
Paris is a region inside France	(*ntpp* paris france)
	$\vdash^{\text{HOL}}$
Catalunya and Paris are disconnected	(*dc* catalunya paris)
	$\wedge$
Spain and Paris are disconnected	(*dc* spain paris)

Reasoning *within* Combined Logics: Epistemic & Spatial

A trivial problem for RCC:

Catalunya is a border region of Spain	(*tpp* catalunya spain),
Spain and France share a border	(*ec* spain france),
Paris is a region inside France	(*ntpp* paris france)
	$\vdash^{\text{HOL}}_{2.3s}$
Catalunya and Paris are disconnected	(*dc* catalunya paris)
	$\wedge$
Spain and Paris are disconnected	(*dc* spain paris)

Reasoning *within* Combined Logics: Epistemic & Spatial

$$
\begin{aligned}
& \text{valid } \forall \phi_{\bullet}\ \Box_{\mathsf{fool}}\,\phi \supset \Box_{\mathsf{bob}}\,\phi, \\
& \text{valid } \Box_{\mathsf{fool}}\,(\lambda W_{\bullet}\,(\mathit{ec}\ \mathsf{spain}\ \mathsf{france})), \\
& \text{valid } \Box_{\mathsf{bob}}\,(\lambda W_{\bullet}\,(\mathit{tpp}\ \mathsf{catalunya}\ \mathsf{spain})), \\
& \text{valid } \Box_{\mathsf{bob}}\,(\lambda W_{\bullet}\,(\mathit{ntpp}\ \mathsf{paris}\ \mathsf{france})) \\
\vdash^{\mathit{HOL}}\ & \text{valid } \Box_{\mathsf{bob}}\,(\lambda W_{\bullet}\,((\mathit{dc}\ \mathsf{catalunya}\ \mathsf{paris}) \wedge (\mathit{dc}\ \mathsf{spain}\ \mathsf{paris})))
\end{aligned}
$$

Reasoning *within* Combined Logics: Epistemic & Spatial

valid $\forall \phi.\ \Box_{\mathsf{fool}}\, \phi \supset \Box_{\mathsf{bob}}\, \phi$,

valid $\Box_{\mathsf{fool}}\, (\lambda W.\, (ec\ \mathsf{spain}\ \mathsf{france}))$,

valid $\Box_{\mathsf{bob}}\, (\lambda W.\, (tpp\ \mathsf{catalunya}\ \mathsf{spain}))$,

valid $\Box_{\mathsf{bob}}\, (\lambda W.\, (ntpp\ \mathsf{paris}\ \mathsf{france}))$

$\vdash^{HOL}_{20.4s}$ valid $\Box_{\mathsf{bob}}\, (\lambda W.\, ((dc\ \mathsf{catalunya}\ \mathsf{paris}) \wedge (dc\ \mathsf{spain}\ \mathsf{paris})))$

Reasoning *within* Combined Logics: Epistemic & Spatial

valid $\forall\phi.\ \Box_{\mathsf{fool}}\,\phi \supset \Box_{\mathsf{bob}}\,\phi$,

valid $\Box_{\mathsf{fool}}\,(\lambda W.\,(ec\ \mathsf{spain}\ \mathsf{france}))$,

valid $\Box_{\mathsf{bob}}\,(\lambda W.\,(tpp\ \mathsf{catalunya}\ \mathsf{spain}))$,

valid $\Box_{\mathsf{bob}}\,(\lambda W.\,(ntpp\ \mathsf{paris}\ \mathsf{france}))$

$\vdash^{HOL}_{20.4s}$ valid $\Box_{\mathsf{bob}}\,(\lambda W.\,((dc\ \mathsf{catalunya}\ \mathsf{paris}) \wedge (dc\ \mathsf{spain}\ \mathsf{paris})))$

$\not\vdash^{HOL}$ valid $\Box_{\mathsf{fool}}\,(\lambda W.\,((dc\ \mathsf{catalunya}\ \mathsf{paris}) \wedge (dc\ \mathsf{spain}\ \mathsf{paris})))$

Reasoning *within* Combined Logics: Epistemic & Spatial

valid $\forall \phi.\ \Box_{\mathsf{fool}}\,\phi \supset \Box_{\mathsf{bob}}\,\phi$,

valid $\Box_{\mathsf{fool}}\,(\lambda W.\,(ec\ \mathsf{spain}\ \mathsf{france}))$,

valid $\Box_{\mathsf{bob}}\,(\lambda W.\,(tpp\ \mathsf{catalunya}\ \mathsf{spain}))$,

valid $\Box_{\mathsf{bob}}\,(\lambda W.\,(ntpp\ \mathsf{paris}\ \mathsf{france}))$

$\vdash^{HOL}_{20.4s}$ valid $\Box_{\mathsf{bob}}\,(\lambda W.\,((dc\ \mathsf{catalunya}\ \mathsf{paris}) \wedge (dc\ \mathsf{spain}\ \mathsf{paris})))$

$\not\vdash^{HOL}_{39.7s}$ valid $\Box_{\mathsf{fool}}\,(\lambda W.\,((dc\ \mathsf{catalunya}\ \mathsf{paris}) \wedge (dc\ \mathsf{spain}\ \mathsf{paris})))$

Reasoning *within* Combined Logics: Epistemic & Spatial

valid $\forall\phi.\ \Box_{\mathsf{fool}}\,\phi \supset \Box_{\mathsf{bob}}\,\phi$,

valid $\Box_{\mathsf{fool}}\,(\lambda W.\,(ec\ \text{spain}\ \text{france}))$,

valid $\Box_{\mathsf{bob}}\,(\lambda W.\,(tpp\ \text{catalunya}\ \text{spain}))$,

valid $\Box_{\mathsf{bob}}\,(\lambda W.\,(ntpp\ \text{paris}\ \text{france}))$

$\vdash^{HOL}_{20.4s}$ valid $\Box_{\mathsf{bob}}\,(\lambda W.\,((dc\ \text{catalunya}\ \text{paris}) \wedge (dc\ \text{spain}\ \text{paris})))$

$\nvdash^{HOL}_{39.7s}$ valid $\Box_{\mathsf{fool}}\,(\lambda W.\,((dc\ \text{catalunya}\ \text{paris}) \wedge (dc\ \text{spain}\ \text{paris})))$

Key idea is "Lifting" of RCC propositions to modal predicates:

$$\underbrace{(tpp\ \text{catalunya}\ \text{spain})}_{\text{type } o} \longrightarrow \underbrace{(\lambda W.\,(tpp\ \text{catalunya}\ \text{spain}))}_{\text{type } \iota\to o}$$

Conclusion

- HOL seems well suited as framework for combining logics
- automation of object-/meta-level reasoning — scalability?
- embeddings can possibly be fully verified in Isabelle/HOL? (consistency of QML embedding: 3.8s – IsabelleN)
- current work: application to ontology reasoning (SUMO)

You can use this framework right away! Try it!

- new TPTP infrastructure for automated HOL reasoning (SutcliffeBenzmüller, J.Formalized Reasoning, 2010)
  - standardized input / output language (THF)
  - problem library: 3000 problems
  - yearly CASC competitions
- provers and examples are online; demo: `http://tptp.org`
  Wise Men Puzzle:

`http://www.cs.miami.edu/~tptp/cgi-bin/SeeTPTP?Category=Problems&Domain=PUZ&File=PUZ087^1.p`

Current Work

Application to Ontology Reasoning

- possible worlds semantics for SUMO ontology
- mapping of modal operators in SUMO to appropriate modal logic operators
- logic combinations
- automation with LEO-II (and other THF0 reasoners)

$\longrightarrow$ see my presentation ARCOE-10 (tomorrow)

SUMO ontology and Sigma ontology engineering tool

$\longrightarrow$ two more presentations at IKBET-10 (tomorrow) and ARCOE-10 (today)

LEO-II

(EPRSC grant EP/D070511/1 at Cambridge University)

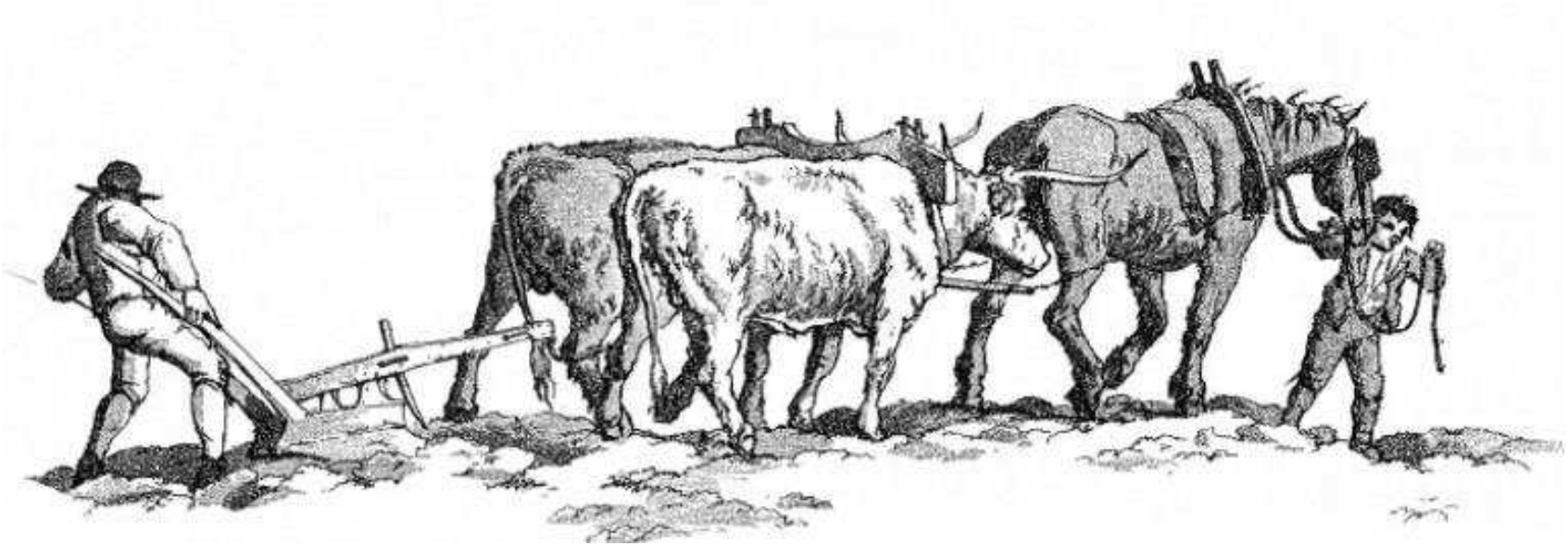
Thanks to Larry Paulson

LEO-II

UNIVERSITY OF CAMBRIDGE

UNIVERSITÄT DES SAARLANDES

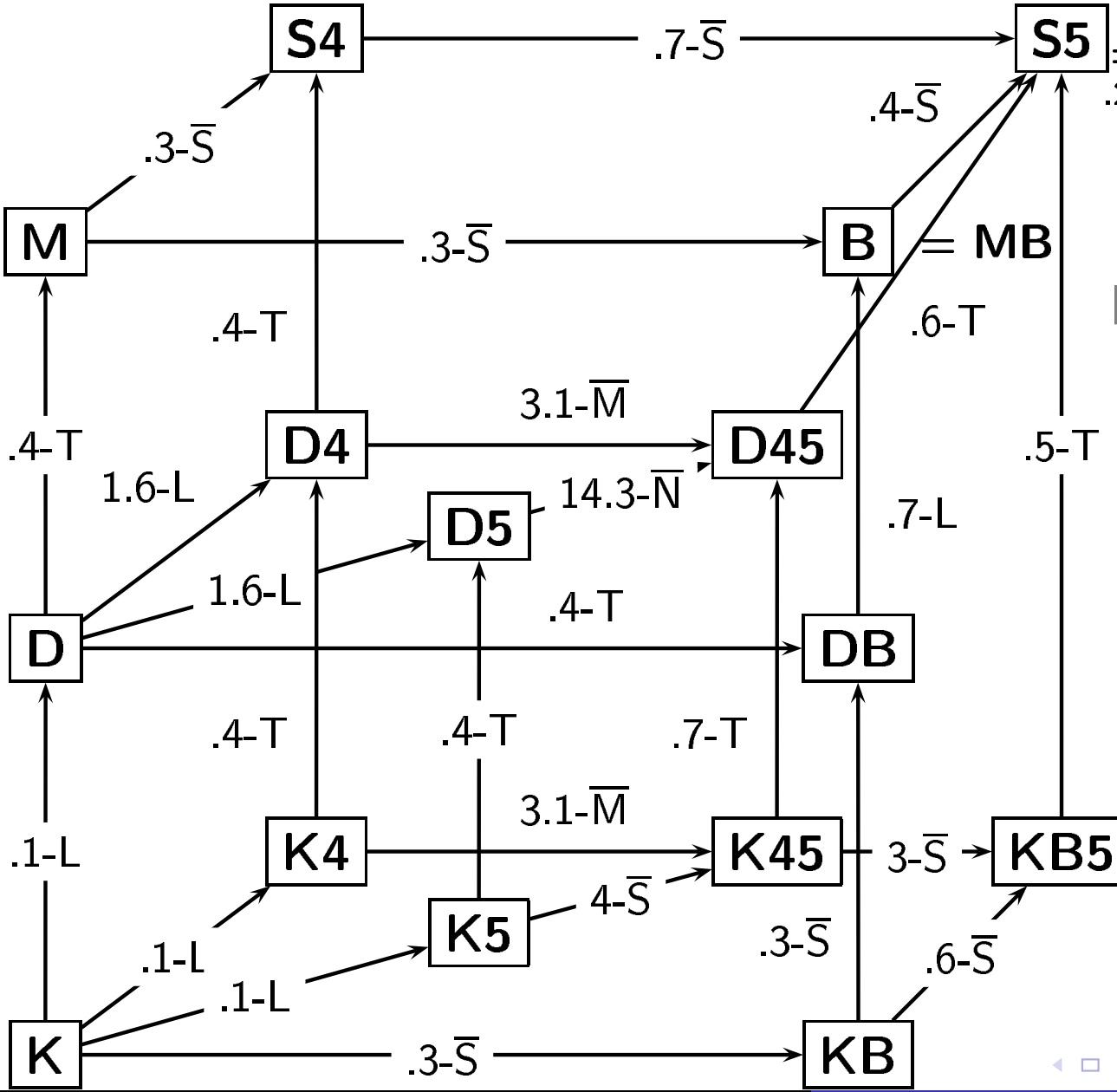
An Effective Higher-Order Theorem Prover



LEO-II employs FO-ATPs: **E, Spass, Vampire**

`http://www.ags.uni-sb.de/~leo`

Reasoning *about* Combinations of Logics: Cube Verification



$= \mathbf{M5} \overset{.1\text{-L}}{\equiv} \mathbf{MB5} \overset{.2\text{-L}}{\equiv} \mathbf{M4B5}$
$\overset{.2\text{-L}}{\equiv} \mathbf{M45} \overset{.1\text{-L}}{\equiv} \mathbf{M4B} \overset{.1\text{-L}}{\equiv} \mathbf{D4B}$
$\overset{.2\text{-L}}{\equiv} \mathbf{D4B5} \overset{.1\text{-L}}{\equiv} \mathbf{DB5}$

$\overset{.2\text{-L}}{\equiv} \mathbf{K4B5} \overset{.1\text{-L}}{\equiv} \mathbf{K4B}$

best proving time by

n-L: LEO-II

n-S: Satallax

n-T: TPS

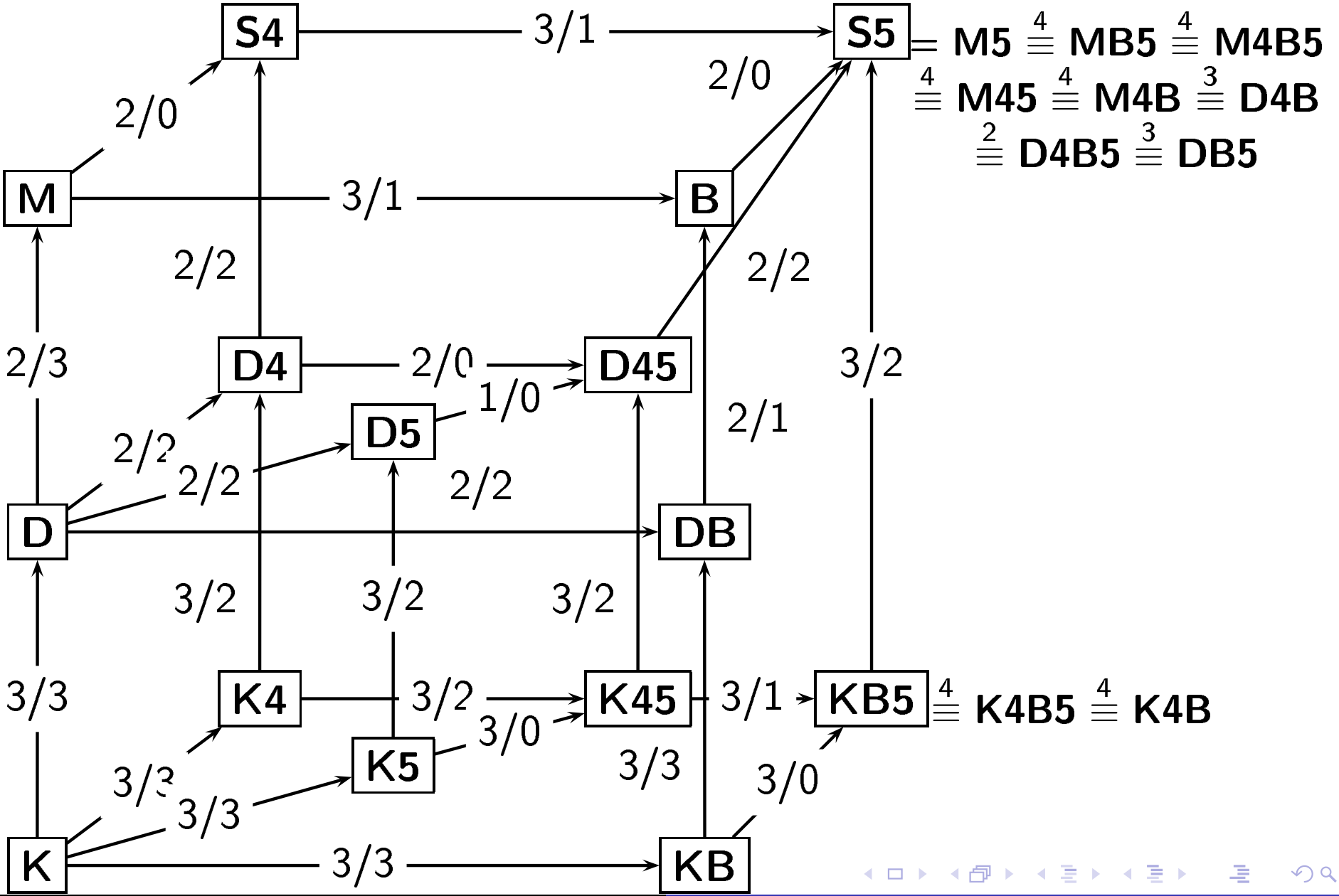
n-$\overline{\text{S}}$: Satallax

n-$\overline{\text{N}}$: IsabelleN

n-$\overline{\text{M}}$: IsabelleM

$\sum < 40\text{sec.}$

Reasoning *about* Combinations of Logics: Cube Verification



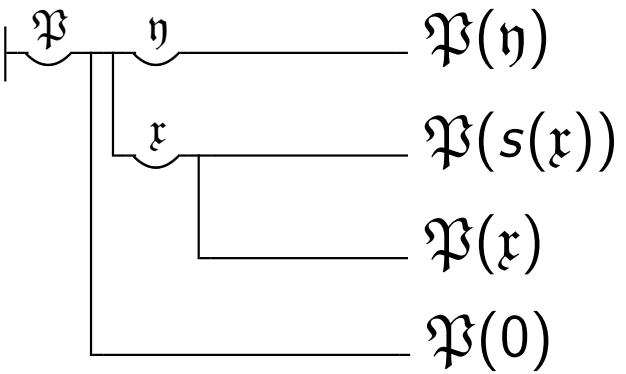
HOL based Universal Reasoning

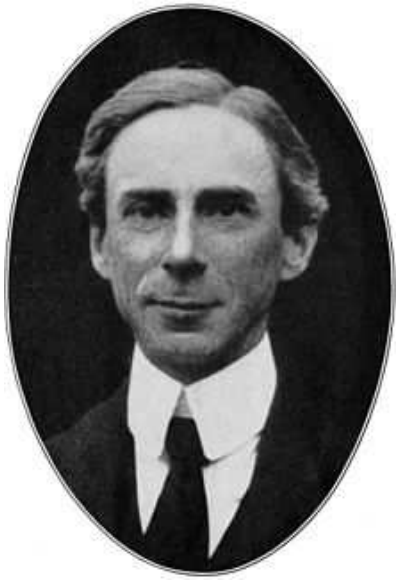
Christoph Benzmüller

Freie Universität Berlin

UNILOG-2013, Rio de Janeiro, Brasil, April 2013

HOL: Church's STT with Henkin Semantics





Automated Reasoners for HOL

TPS ... (Peter Andrews) ?

LEO-I/LEO-II (myself)

Isabelle (Nipkow/Paulson/Blanchette)

Satallax (Brown)

Nitpick (Blanchette)

agsyHOL (Lindblatt)

- all accept TPTP THF Syntax [SutcliffeBenzmüller, J.Form.Reas, 2009]
  - can be called remotely via SystemOnTPTP at Miami
  - they significantly gained in strength over the last years
  - they can be bundled into a combined prover HOL-P

Exploit HOL with Henkin semantics as metalogic
Automate other logics (& combinations) via semantic embeddings
— HOL-P becomes a Universal Reasoner —

Short Demonstration of HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

encoding in HOL: $\text{valid}\,(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

. . . in THF Syntax: . . . not here . . .

Short Demonstration of HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

encoding in HOL: valid $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

. . . in THF Syntax: . . . not here . . .

```
%> ./HOL-P example.thf -timeout 20 -logic s4 -domain varying
```

Short Demonstration of HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

encoding in HOL: valid $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

. . . in THF Syntax: . . . not here . . .

```
%> ./HOL-P example.thf -timeout 20 -logic s4 -domain varying
```

Calling HOL Resoners remotely in Miami . . . thanks to Geoff Sutcliffe

— LEO-II says Theorem — CPU 0.08s
— Satallax says Theorem — CPU 0.03s
— Isabelle says Unknown — CPU 11.93s
— Nitpick says Unknown — CPU 10.62s
— agsyHOL says Theorem — CPU 0.55s

Short Demonstration of HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

encoding in HOL: valid $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

. . . in THF Syntax: . . . not here . . .

```
%> ./HOL-P example.thf -timeout 20 -logic s4 -domain varying
Calling HOL Resoners remotely in Miami . . . thanks to Geoff Sutcliffe
— LEO-II says Theorem — CPU 0.08s
— Satallax says Theorem — CPU 0.03s
— Isabelle says Unknown — CPU 11.93s
— Nitpick says Unknown — CPU 10.62s
— agsyHOL says Theorem — CPU 0.55s
```

```
%> ./HOL-P example.thf -timeout 20 -logic k -domain constant
```

Short Demonstration of HOL-P

FO Modal Logic example: $(\Diamond\exists x Pfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists z Qz$

encoding in HOL: $\text{valid}\,(\Diamond\exists x Pfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists z Qz$

. . . in THF Syntax: . . . not here . . .

```
%> ./HOL-P example.thf -timeout 20 -logic s4 -domain varying
Calling HOL Resoners remotely in Miami . . . thanks to Geoff Sutcliffe
— LEO-II says Theorem — CPU 0.08s
— Satallax says Theorem — CPU 0.03s
— Isabelle says Unknown — CPU 11.93s
— Nitpick says Unknown — CPU 10.62s
— agsyHOL says Theorem — CPU 0.55s
```

```
%> ./HOL-P example.thf -timeout 20 -logic k -domain constant
Calling HOL Resoners remotely in Miami . . . thanks to Geoff Sutcliffe
— LEO-II says Unknown — CPU 11.93s
— Satallax says CounterSatisfiable — CPU 0.04s
— Isabelle says Unknown — CPU 16.19s
— Nitpick says CounterSatisfiable — CPU 8.19s
— agsyHOL says Unknown — CPU 10.82s
```

HOL — Church's Simple Type Theory [Church, 1940]

Simple Types

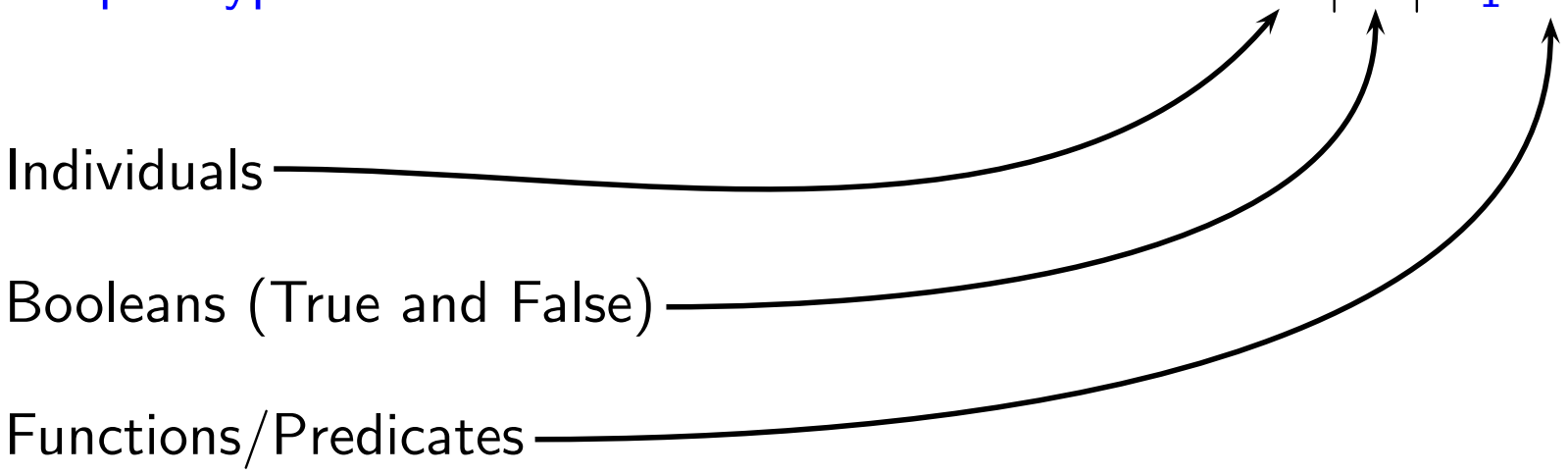
$$\alpha ::= \iota \mid o \mid \alpha_1 \to \alpha_2$$

HOL — Church's Simple Type Theory [Church, 1940]

Freie Universität Berlin

Simple Types

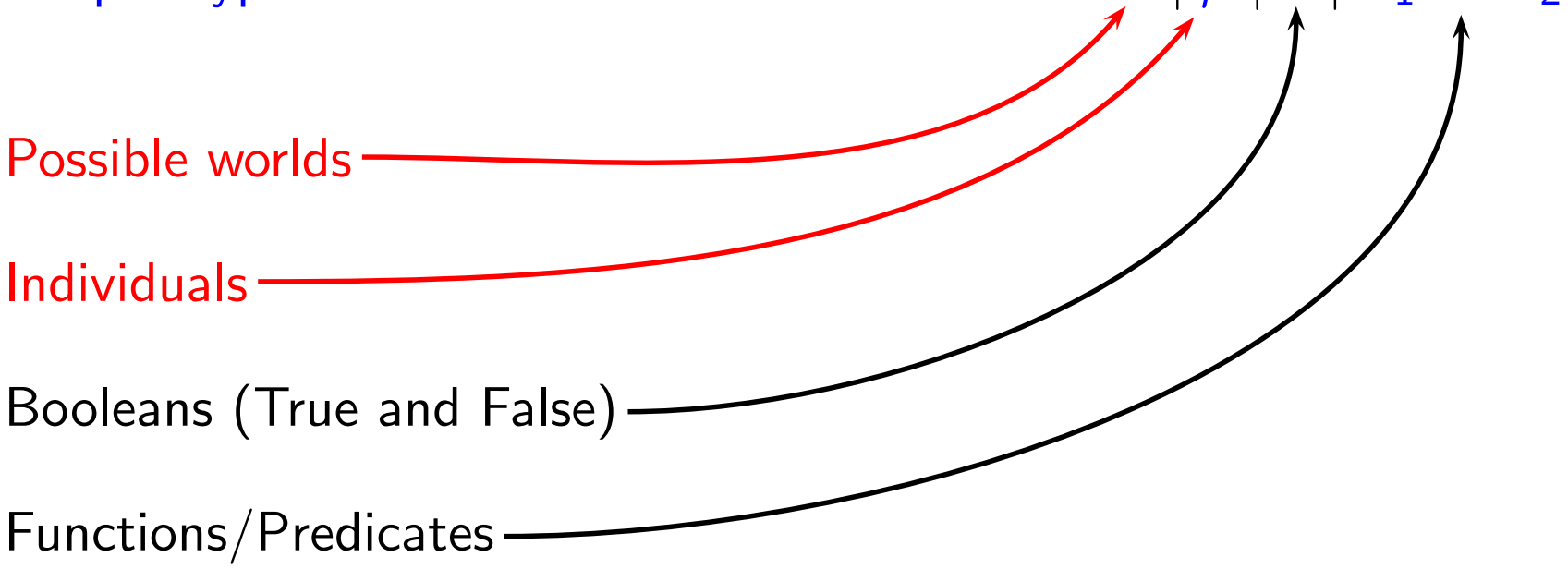
$$\alpha ::= \iota \mid o \mid \alpha_1 \to \alpha_2$$



HOL — Church's Simple Type Theory [Church, 1940]

Freie Universität Berlin

Simple Types $\alpha ::= \iota \mid \mu \mid o \mid \alpha_1 \to \alpha_2$



HOL — Church's Simple Type Theory [Church, 1940]

HOL

$$s, t \quad ::= \quad c_\alpha \mid x_\alpha \mid (\lambda x_\alpha\, s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \mid$$
$$(\neg_{o\to o}\, s_o)_o \mid (s_o \vee_{o\to o\to o}\, t_o)_o \mid (\forall x_\alpha\, t_o)_o$$

HOL — Church's Simple Type Theory [Church, 1940]

Freie Universität Berlin

HOL

$$s, t \quad ::= \quad c_\alpha \mid x_\alpha \mid (\lambda x_\alpha\, s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \mid$$
$$(\neg_{o\to o}\, s_o)_o \mid (s_o \vee_{o\to o\to o}\, t_o)_o \mid (\forall x_\alpha\, t_o)_o$$

$$\Pi_{(\alpha\to o)\to o} \lambda x_\alpha t_o$$

HOL — Church's Simple Type Theory [Church, 1940]

Freie Universität Berlin

$$\text{HOL} \qquad s, t \quad ::= \quad C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$$

HOL — Church's Simple Type Theory [Church, 1940]

HOL $\quad s, t \quad ::= \quad C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$

HOL (with Henkin semantics) is meanwhile very well understood

- Origin [Church, J.Symb.Log., 1940]
- Henkin-Semantics [Henkin, J.Symb.Log., 1950]
  [Andrews, J.Symb.Log., 1971, 1972]
- Extensionality/Intensionality [BenzmüllerBrownKohlhase, J.Symb.Log., 2004]
  [Muskens, J.Symb.Log., 2007]

Embedding of First-order Modal Logic (FML) in HOL

Freie Universität Berlin

$$\text{HOL} \qquad s, t \quad ::= \quad C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$$

Embedding of First-order Modal Logic (FML) in HOL

HOL $\quad s, t \quad ::= \quad C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$

FML $\quad \varphi, \psi \quad ::= \quad P(t_1, \ldots, t_n) \mid (\neg\varphi) \mid (\varphi \vee \psi) \mid \Box\varphi \mid (\forall x\, \varphi)$

$$
\begin{array}{lll}
M, g, s \models \neg\varphi & \text{iff} & \text{not } M, g, s \models \varphi \\
M, g, s \models \varphi \vee \psi & \text{iff} & M, g, s \models \varphi \text{ or } M, g, s \models \psi \\
M, g, s \models \Box\varphi & \text{iff} & M, g, u \models \varphi \text{ for all } u \text{ with } r(s, u) \\
M, g, s \models \forall x\, \varphi & \text{iff} & M, [d/x]g, s \models \varphi \text{ for all } d \in D
\end{array}
$$

Embedding of First-order Modal Logic (FML) in HOL

HOL $\quad s, t \quad ::= \quad C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$

FML $\quad \varphi, \psi \quad ::= \quad P(t_1, \ldots, t_n) \mid (\neg\varphi) \mid (\varphi \vee \psi) \mid \Box\varphi \mid (\forall x\, \varphi)$

$$
\begin{array}{lll}
M, g, s \models \neg\varphi & \text{iff} & \text{not } M, g, s \models \varphi \\
M, g, s \models \varphi \vee \psi & \text{iff} & M, g, s \models \varphi \text{ or } M, g, s \models \psi \\
M, g, s \models \Box\varphi & \text{iff} & M, g, u \models \varphi \text{ for all } u \text{ with } r(s, u) \\
M, g, s \models \forall x\, \varphi & \text{iff} & M, [d/x]g, s \models \varphi \text{ for all } d \in D
\end{array}
$$

FML in HOL:

$$
\begin{array}{rcl}
\neg & = & \lambda\varphi_{\iota\to o}\, \lambda s_\iota\, \neg\varphi s \\
\vee & = & \lambda\varphi_{\iota\to o}\, \lambda\psi_{\iota\to o}\, \lambda s_\iota\, (\varphi s \vee \psi s) \\
\Box_r & = & \lambda\varphi_{\iota\to o}\, \lambda s_\iota\, \forall u_\iota\, (\neg r s u \vee \varphi u) \\
\Pi & = & \lambda h_{\mu\to(\iota\to o)}\, \lambda s_\iota\, \forall d_\mu\, h d s
\end{array}
$$

($\forall x\varphi$ stands for $\Pi\lambda x\varphi$)

Embedding of First-order Modal Logic (FML) in HOL

HOL $\quad s, t \quad ::= \quad C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$

FML $\quad \varphi, \psi \quad ::= \quad P(t_1, \ldots, t_n) \mid (\neg\varphi) \mid (\varphi \vee \psi) \mid \Box\varphi \mid (\forall x\, \varphi)$

$$
\begin{array}{lll}
M, g, s \models \neg\varphi & \text{iff} & \text{not } M, g, s \models \varphi \\
M, g, s \models \varphi \vee \psi & \text{iff} & M, g, s \models \varphi \text{ or } M, g, s \models \psi \\
M, g, s \models \Box\varphi & \text{iff} & M, g, u \models \varphi \text{ for all } u \text{ with } r(s, u) \\
M, g, s \models \forall x\, \varphi & \text{iff} & M, [d/x]g, s \models \varphi \text{ for all } d \in D
\end{array}
$$

FML in HOL:

$$
\begin{array}{rcl}
\neg & = & \lambda\varphi_{\iota\to o}\, \lambda s_\iota\, \neg\varphi s \\
\vee & = & \lambda\varphi_{\iota\to o}\, \lambda\psi_{\iota\to o}\, \lambda s_\iota\, (\varphi s \vee \psi s) \\
\Box & = & \lambda r_{\iota\to\iota\to o}\, \lambda\varphi_{\iota\to o}\, \lambda s_\iota\, \forall u_\iota\, (\neg r s u \vee \varphi u) \\
\Pi & = & \lambda h_{\mu\to(\iota\to o)}\, \lambda s_\iota\, \forall d_\mu\, h d s \\
 & & (\forall x\varphi \text{ stands for } \Pi\lambda x\varphi)
\end{array}
$$

Embedding of First-order Modal Logic (FML) in HOL

Freie Universität Berlin

HOL $\quad s, t \;::=\; C \mid x \mid (\lambda x\, s) \mid (s\, t) \mid (\neg s) \mid (s \vee t) \mid (\forall x\, t)$

FML $\quad \varphi, \psi \;::=\; P(t_1, \ldots, t_n) \mid (\neg\varphi) \mid (\varphi \vee \psi) \mid \Box\varphi \mid (\forall x\, \varphi)$

$$
\begin{array}{lll}
M, g, s \models \neg\varphi & \text{iff} & \text{not } M, g, s \models \varphi \\
M, g, s \models \varphi \vee \psi & \text{iff} & M, g, s \models \varphi \text{ or } M, g, s \models \psi \\
M, g, s \models \Box\varphi & \text{iff} & M, g, u \models \varphi \text{ for all } u \text{ with } r(s, u) \\
M, g, s \models \forall x\, \varphi & \text{iff} & M, [d/x]g, s \models \varphi \text{ for all } d \in D
\end{array}
$$

FML in HOL:

$$
\begin{array}{rcl}
\neg & = & \lambda\varphi_{\iota\to o}\, \lambda s_\iota\, \neg\varphi s \\
\vee & = & \lambda\varphi_{\iota\to o}\, \lambda\psi_{\iota\to o}\, \lambda s_\iota\, (\varphi s \vee \psi s) \\
\Box & = & \lambda r_{\iota\to\iota\to o}\, \lambda\varphi_{\iota\to o}\, \lambda s_\iota\, \forall u_\iota\, (\neg r s u \vee \varphi u) \\
\Pi & = & \lambda h_{\mu\to(\iota\to o)}\, \lambda s_\iota\, \forall d_\mu\, h d s \\
 & & (\forall x\varphi \text{ stands for } \Pi\lambda x\varphi)
\end{array}
$$

Idea: Lifting of modal formulas to predicates on worlds

Metalevel notions: valid $= \lambda\varphi_{\iota\to o}\, \forall s_\iota\, \varphi s$

B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$valid\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

B: Example — Embedding of FML in HOL

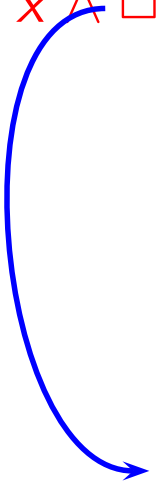
$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$



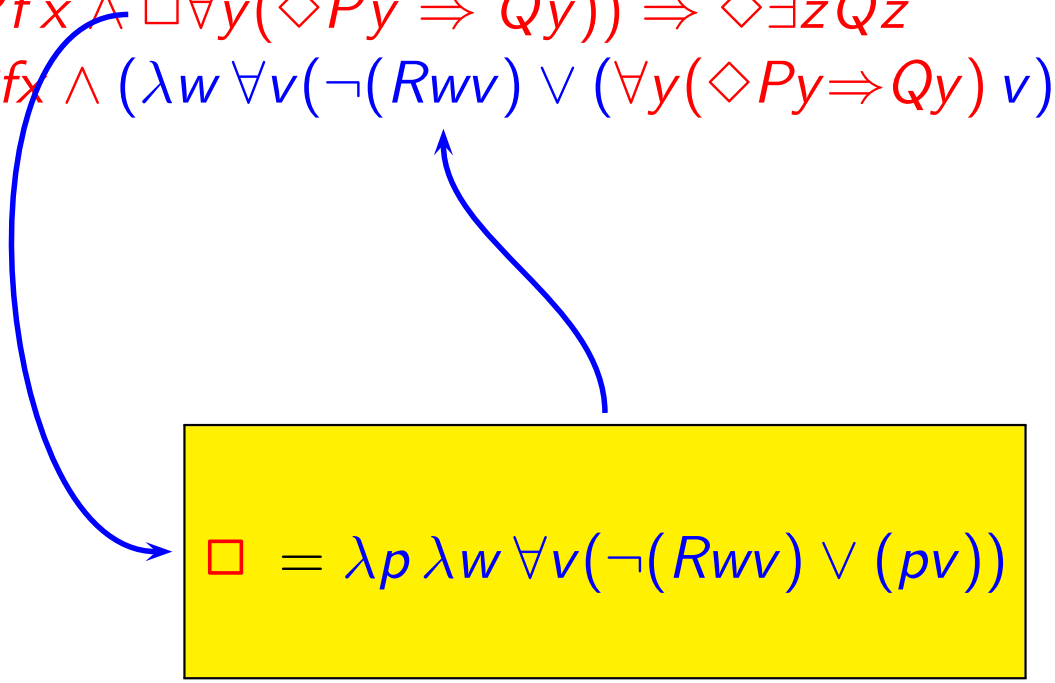
$$\Box = \lambda p\, \lambda w\, \forall v(\neg(Rwv) \vee (pv))$$

B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge (\lambda w\, \forall v(\neg(R w v) \vee (\forall y(\Diamond P y \Rightarrow Q y)\, v)))) \Rightarrow \Diamond\exists z Q z$$



B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge (\lambda w\, \forall v(\neg(R w v) \vee (\forall y(\Diamond P y \Rightarrow Q y)\, v)))) \Rightarrow \Diamond\exists z Q z$$

...

$$\forall w(\neg\neg(\neg\neg\forall v(\neg R w v \vee \neg\neg\forall x \neg P(f x) v) \vee \neg\forall v(\neg R w v \vee \forall y(\neg\neg\forall u(\neg R v u \vee \neg P y u) \vee Q y v))) \vee \neg\forall v(\neg R w v \vee \neg\neg\forall z \neg Q z v))$$

B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge (\lambda w\, \forall v(\neg(R w v) \vee (\forall y(\Diamond P y \Rightarrow Q y)\, v)))) \Rightarrow \Diamond\exists z Q z$$

...

$$\forall w(\neg\neg(\neg\neg\forall v(\neg R w v \vee \neg\neg\forall x \neg P(f x) v) \vee \neg\forall v(\neg R w v \vee \forall y(\neg\neg\forall u(\neg R v u \vee \neg P y u) \vee Q y v))) \vee \neg\forall v(\neg R w v \vee \neg\neg\forall z \neg Q z v))$$

Propositional Quantification

Propositional Quantification [Fitting, J.Symb.Log., 2002]

...

$M, g, s \models \forall^p p\,\varphi$ iff $M, [v/p]g, s \models \varphi$ for all $v \in P$

(P is a non-empty collection of sets of worlds, it includes atom sets)

Embedding in HOL

...

$\Pi^p = \lambda h_{(\iota\to o)\to(\iota\to o)}\, \lambda s_\iota\, \forall v_\mu\, hvs$ ($\forall\varphi\psi$ stands for $\Pi^p\lambda\varphi\psi$)

Modal logic axioms

valid $\forall^p\varphi(\Box\varphi \supset \Diamond\varphi)$

Semantical Condition

$\forall x\exists y(rxy)$

Bridge rules

valid $\forall^p\varphi(\Box_r\varphi \supset \Box_s\varphi)$

Semantical Condition

$\forall x\forall y(rxy \supset sxy)$

We get a wide range of modal logics and combinations for free!

Propositional Quantification

Propositional Quantification [Fitting, J.Symb.Log., 2002]

...

$M, g, s \models \forall^p p\,\varphi$ iff $M, [v/p]g, s \models \varphi$ for all $v \in P$

(P is a non-empty collection of sets of worlds, it includes atom sets)

Embedding in HOL

...

$\Pi^p = \lambda h_{(\iota\to o)\to(\iota\to o)}\, \lambda s_\iota\, \forall v_\mu\, hvs$ ($\forall\varphi\psi$ stands for $\Pi^p\lambda\varphi\psi$)

Modal logic axioms

valid $\forall^p\varphi(\Box\varphi \supset \Diamond\varphi)$

Semantical Condition

$\forall x\exists y(rxy)$

Bridge rules

valid $\forall^p\varphi(\Box_r\varphi \supset \Box_s\varphi)$

Semantical Condition

$\forall x\forall y(rxy \supset sxy)$

We get a wide range of modal logics and combinations for free!

Propositional Quantification

Freie Universität Berlin

Propositional Quantification [Fitting, J.Symb.Log., 2002]

...

$M, g, s \models \forall^p p\,\varphi$ iff $M, [v/p]g, s \models \varphi$ for all $v \in P$

(P is a non-empty collection of sets of worlds, it includes atom sets)

Embedding in HOL

...

$\Pi^p = \lambda h_{(\iota\to o)\to(\iota\to o)}\, \lambda s_\iota\, \forall v_\mu\, hvs$ ($\forall\varphi\psi$ stands for $\Pi^p\lambda\varphi\psi$)

Modal logic axioms

valid $\forall^p\varphi(\Box\varphi \supset \Diamond\varphi)$

Semantical Condition

$\forall x\exists y(rxy)$

Bridge rules

valid $\forall^p\varphi(\Box_r\varphi \supset \Box_s\varphi)$

Semantical Condition

$\forall x\forall y(rxy \supset sxy)$

We get a wide range of modal logics and combinations for free!

Propositional Quantification

Propositional Quantification [Fitting, J.Symb.Log., 2002]

...

$M, g, s \models \forall^p p\,\varphi$ iff $M, [v/p]g, s \models \varphi$ for all $v \in P$

(P is a non-empty collection of sets of worlds, it includes atom sets)

Embedding in HOL

...

$\Pi^p = \lambda h_{(\iota\to o)\to(\iota\to o)}\, \lambda s_\iota\, \forall v_\mu\, hvs$ ($\forall\varphi\psi$ stands for $\Pi^p\lambda\varphi\psi$)

Modal logic axioms

valid $\forall^p\varphi(\Box\varphi \supset \Diamond\varphi)$

Semantical Condition

$\forall x\exists y(rxy)$

Bridge rules

valid $\forall^p\varphi(\Box_r\varphi \supset \Box_s\varphi)$

Semantical Condition

$\forall x\forall y(rxy \supset sxy)$

We get a wide range of modal logics and combinations for free!

Propositional Quantification

Propositional Quantification [Fitting, J.Symb.Log., 2002]

...

$M, g, s \models \forall^p p\,\varphi$ iff $M, [v/p]g, s \models \varphi$ for all $v \in P$

(P is a non-empty collection of sets of worlds, it includes atom sets)

Embedding in HOL

...

$\Pi^p = \lambda h_{(\iota\to o)\to(\iota\to o)}\, \lambda s_\iota\, \forall v_\mu\, hvs$ ($\forall\varphi\psi$ stands for $\Pi^p\lambda\varphi\psi$)

Modal logic axioms

valid $\forall^p\varphi(\Box\varphi \supset \Diamond\varphi)$

Semantical Condition

$\forall x\exists y(rxy)$

Bridge rules

valid $\forall^p\varphi(\Box_r\varphi \supset \Box_s\varphi)$

Semantical Condition

$\forall x\forall y(rxy \supset sxy)$

We get a wide range of modal logics and combinations for free!

B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$valid\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$valid\ (\Diamond\exists x P f x \wedge (\lambda w\, \forall v(\neg(R w v) \vee (\forall y(\Diamond P y \Rightarrow Q y)\, v)))) \Rightarrow \Diamond\exists z Q z$$

...

$$\forall w(\neg\neg(\neg\neg\forall v(\neg R w v \vee \neg\neg\forall x \neg P(f x) v) \vee \neg\forall v(\neg R w v \vee \forall y(\neg\neg\forall u(\neg R v u \vee \neg P y u) \vee Q y v))) \vee \neg\forall v(\neg R w v \vee \neg\neg\forall z \neg Q z v))$$

Axiomatization of properties of accessibility relation R

Logic K: no axioms

Logic T: $(reflexive\ R)$ — which expands into $\forall x\, R x x$

Logic S4: $(reflexive\ R) \wedge (symmetric\ R) \wedge (transitive\ R)$

Logic ... ...

B: Example — Embedding of FML in HOL

$$(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$$

$$\mathit{valid}\ (\Diamond\exists x P f x \wedge (\lambda w\, \forall v(\neg(R w v) \vee (\forall y(\Diamond P y \Rightarrow Q y)\, v)))) \Rightarrow \Diamond\exists z Q z$$

...

$$\forall w(\neg\neg(\neg\neg\forall v(\neg R w v \vee \neg\neg\forall x \neg P(f x) v) \vee \neg\forall v(\neg R w v \vee \forall y(\neg\neg\forall u(\neg R v u \vee \neg P y u) \vee Q y v))) \vee \neg\forall v(\neg R w v \vee \neg\neg\forall z \neg Q z v))$$

Axiomatization of properties of accessibility relation R

Logic K: no axioms

Logic T: $(\mathit{reflexive}\ R)$ — which expands into $\forall x\, R x x$

Logic S4: $(\mathit{reflexive}\ R) \wedge (\mathit{symmetric}\ R) \wedge (\mathit{transitive}\ R)$

Logic ... ...

This automates FML with constant domain semantics in HOL

B: Example — Embedding FML in HOL

Freie Universität Berlin

To obtain varying domain semantics:

- modify quantifier: $\Pi = \lambda q \lambda w \forall x\, \mathsf{ExistsInW}\, x\, w \Rightarrow q\, x\, w$
- add non-emptiness axiom: $\forall w\, \exists x \mathsf{ExistsInW}\, x\, w$
- add designation axioms for constants c: $\forall w\, \mathsf{ExistsInW}\, c\, w$
  (similar for function symbols)

B: Example — Embedding FML in HOL

Freie Universität Berlin

To obtain varying domain semantics:

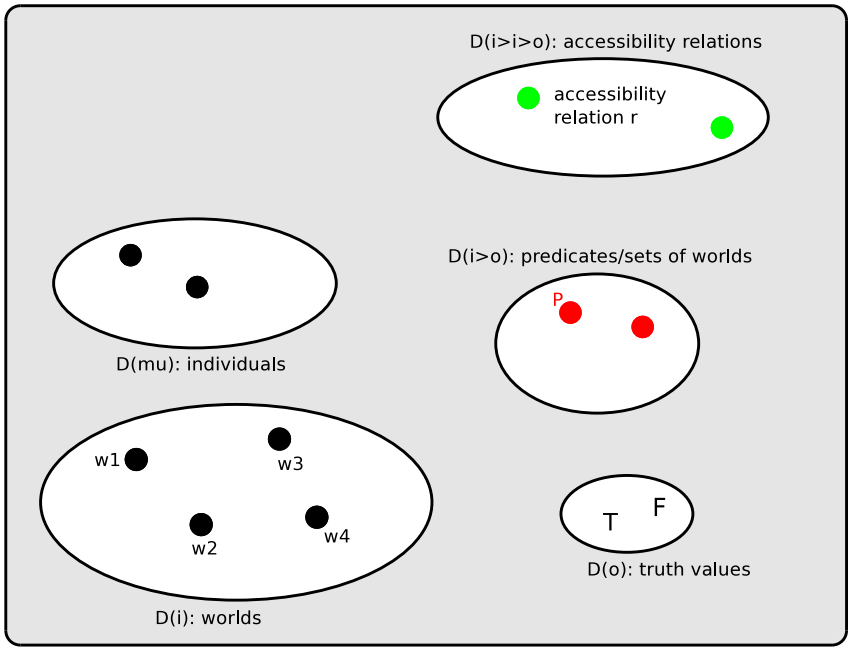
- modify quantifier: $\Pi = \lambda q \lambda w \forall x\, \mathsf{ExistsInW}xw \Rightarrow qxw$
- add non-emptiness axiom: $\forall w\, \exists x \mathsf{ExistsInW}xw$
- add designation axioms for constants c: $\forall w\, \mathsf{ExistsInW}cw$
  (similar for function symbols)

To obtain cumulative domain semantics:

- add axiom: $\forall x \forall v \forall w\, \mathsf{ExistsInW}xv \wedge Rvw \Rightarrow \mathsf{ExistsInW}xw$

Varying and Cumulative Domain Quantification

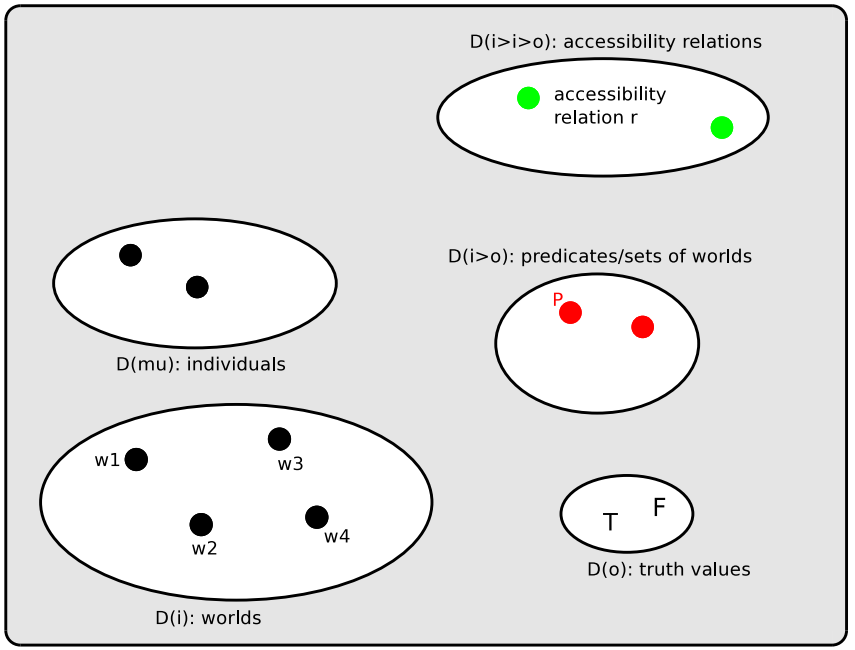
Constant Domain



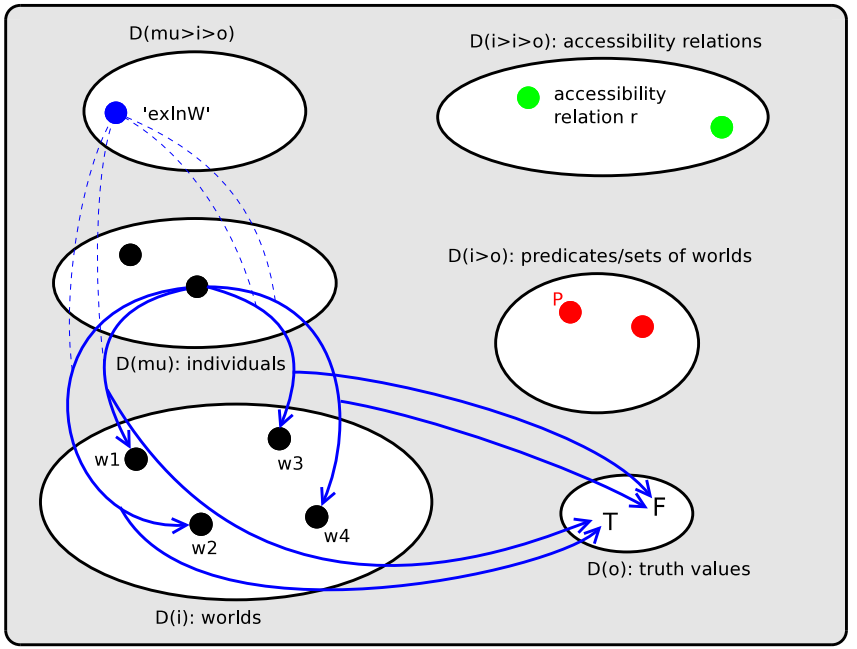
$$\Pi = \lambda h \, \lambda w_\iota \, \forall x_\mu \, hxw$$

Varying and Cumulative Domain Quantification

Constant Domain



Varying and Cumulative Domain



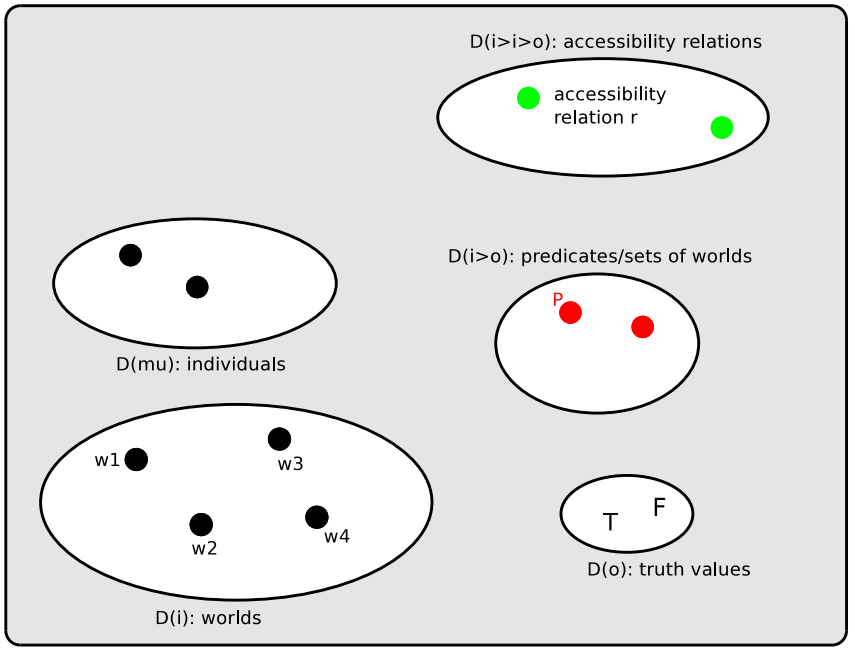
$$\Pi = \lambda h\, \lambda w_\iota\, \forall x_\mu\, hxw$$

$$\Pi^{va} = \lambda h\, \lambda w_\iota\, \forall x_\mu\, (\neg \mathbf{exInW} xw \vee hxw)$$

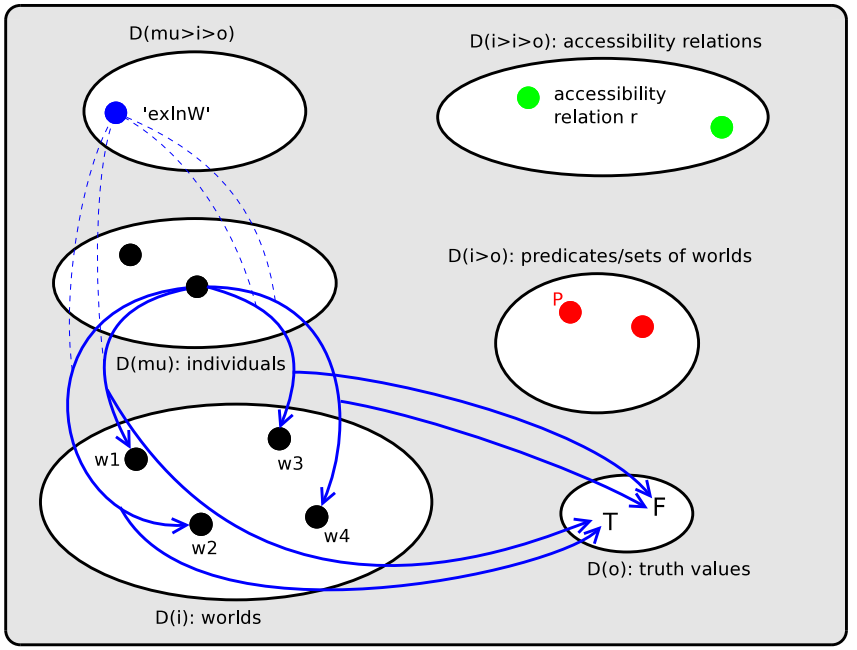
Varying and Cumulative Domain Quantification

Freie Universität Berlin

Constant Domain



Varying and Cumulative Domain



$$\Pi = \lambda h\, \lambda w_\iota\, \forall x_\mu\, hxw$$

$$\Pi^{va} = \lambda h\, \lambda w_\iota\, \forall x_\mu\, (\neg \mathbf{exInW} xw \vee hxw)$$

domains are non-empty $\rightarrow$ $\forall w_\iota\, \exists x_\mu\, \mathbf{exInW} xw$

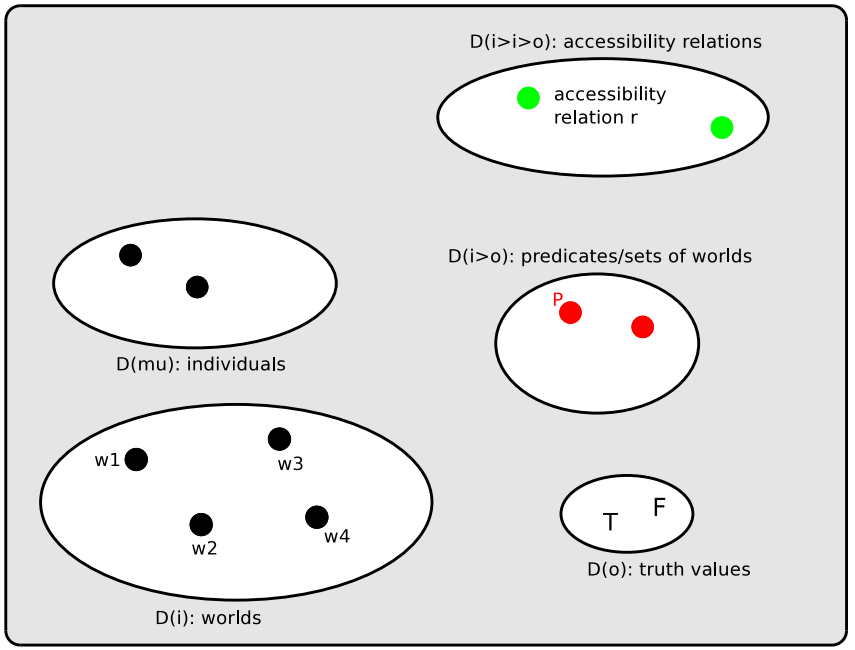
denotation (constants & functions) $\rightarrow$ $\forall w_\iota\, \mathbf{exInW} cw$

$\rightarrow$ $\forall w_\iota\, (\mathbf{exInW} t^1 w \wedge \ldots \wedge \mathbf{exInW} t^n w \supset \mathbf{exInW}(f\, t^1 \ldots t^n) w)$

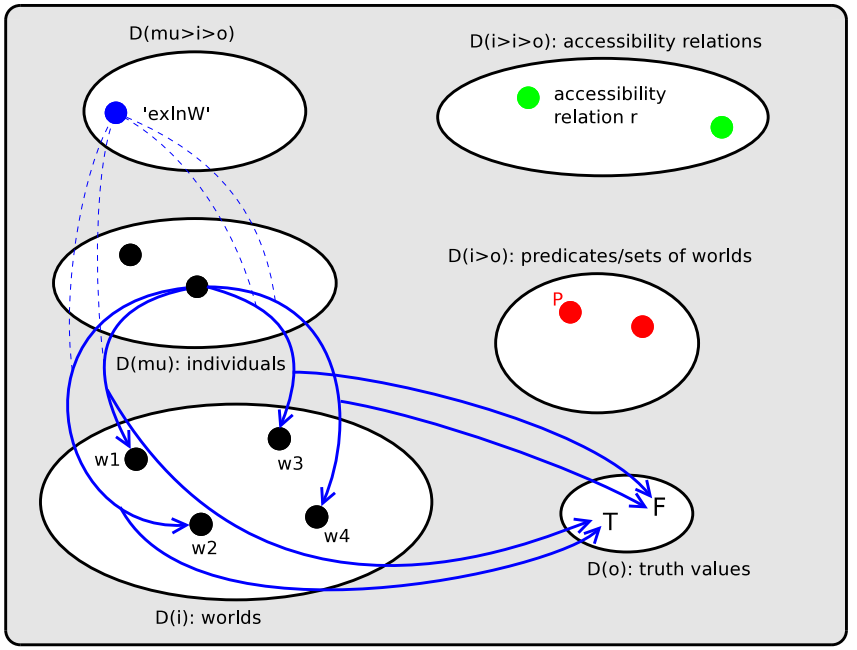
Varying and Cumulative Domain Quantification

Freie Universität Berlin

Constant Domain



Varying and Cumulative Domain



$$\Pi = \lambda h\, \lambda w_\iota\, \forall x_\mu\, hxw$$

$$\Pi^{va} = \lambda h\, \lambda w_\iota\, \forall x_\mu\, (\neg \mathbf{exInW} xw \vee hxw)$$

domains are non-empty → $\forall w_\iota\, \exists x_\mu\, \mathbf{exInW} xw$

denotation (constants & functions) → $\forall w_\iota\, \mathbf{exInW} c w$

→ $\forall w_\iota\, (\mathbf{exInW} t^1 w \wedge \ldots \wedge \mathbf{exInW} t^n w \supset \mathbf{exInW}(f\, t^1 \ldots t^n) w)$

cumulative domains → $\forall x, v, w\, (\mathbf{exInW} xv \wedge rvw \supset \mathbf{exInW} xw)$

Soundness and Completeness (and Cut-elimination)

$$\models^{L} \varphi \quad \text{iff} \quad \models^{HOL}_{\text{Henkin}} \text{valid}\, \varphi_{\iota\to o}$$

Logics L studied so far:

- Propositional Multimodal Logics [BenzmüllerPaulson, Log.J.IGPL, 2010]
- Quantified Multimodal Logics [BenzmüllerPaulson, Log.Univ., 2012]
- Intuitionistic Logics [BenzmüllerPaulson, Log.J.IGPL, 2010]
- Access Control Logics [Benzmüller, IFIP SEC, 2009]
- Propositional Conditional Logics [BenzmüllerEtAl., AMAI, 2012]
- Quantified Conditional Logics [Benzmüller, IJCAI, 2013]
- . . . more is on the way . . .

Soundness and Completeness (and Cut-elimination)

$$\models^{L} \varphi \quad \text{iff} \quad \models^{HOL}_{\text{Henkin}} \text{valid}\, \varphi_{\iota\to o} \quad \text{iff} \quad \vdash^{\text{seq}^{\text{HOL}}}_{\text{cut-free}} \text{valid}\, \varphi_{\iota\to o}$$

Logics L studied so far:

- Propositional Multimodal Logics [BenzmüllerPaulson, Log.J.IGPL, 2010]
- Quantified Multimodal Logics [BenzmüllerPaulson, Log.Univ., 2012]
- Intuitionistic Logics [BenzmüllerPaulson, Log.J.IGPL, 2010]
- Access Control Logics [Benzmüller, IFIP SEC, 2009]
- Propositional Conditional Logics [BenzmüllerEtAl., AMAI, 2012]
- Quantified Conditional Logics [Benzmüller, IJCAI, 2013]
- . . . more is on the way . . .

Evaluation: What Systems are there to compare with?

Freie Universität Berlin

- **Combinations of Quantified Logics** — no systems available
- **Quantified Conditional Logics** — no systems available
- **Quantified Multimodal Logics** — no systems available
- ... — ...
- **First-order Monomodal Logics** — yes, some systems exist

  There is now even a benchmark library:

  QMLTP-lib (580 Problems): `http://www.iltp.de/qmltp/`

  Earlier experiments (see [BenzmüllerOttenRaths, ECAI, 2012]) already showed that the HOL approach performs quite well.

Evaluation: What Systems are there to compare with?

Freie Universität Berlin

- **Combinations of Quantified Logics** — no systems available
- **Quantified Conditional Logics** — no systems available
- **Quantified Multimodal Logics** — no systems available
- ... — ...
- **First-order Monomodal Logics** — yes, some systems exist

  There is now even a benchmark library:

  QMLTP-lib (580 Problems): `http://www.iltp.de/qmltp/`

  Earlier experiments (see [BenzmüllerOttenRaths, ECAI, 2012]) already showed that the HOL approach performs quite well.

FMLtoHOL

- implemented as part of Sutcliffe's TPTP2X tool
- included in the QMLTP—v1.1 package available at: `http://www.iltp.de/qmltp/problems.html`
- written in Prolog, can be easily modified and extended
- invoked as

  `./tptp2X -f thf:<logic>:<domain> <qmf-file>`

  where `<logic>` $\in \{$k,k4,d,d4,t,s4,s5$\}$ and
  `<domain>` $\in \{$*const*, *vary*, *cumul*$\}$.
- generates TPTP thf0-files; employs `include`-mechanism
- can easily be combined (shell script) with HOL-P metaprover

FMLtoHOL combined with HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

FMLtoHOL combined with HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

```
%> ./FMLtoHOL-P example.thf -timeout 20 -logic s4 -domain varying
```

FMLtoHOL combined with HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

```
%> ./FMLtoHOL-P example.thf -timeout 20 -logic s4 -domain varying
```

Calling HOL Resoners remotely in Miami . . . thanks to Geoff Sutcliffe

— LEO-II says Theorem — CPU 0.08s
— Satallax says Theorem — CPU 0.03s
— Isabelle says Unknown — CPU 11.93s
— Nitpick says Unknown — CPU 10.62s
— agsyHOL says Theorem — CPU 0.55s

FMLtoHOL combined with HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

```
%> ./FMLtoHOL-P example.thf -timeout 20 -logic s4 -domain varying
```

Calling HOL Resoners remotely in Miami . . . thanks to Geoff Sutcliffe
— LEO-II says Theorem — CPU 0.08s
— Satallax says Theorem — CPU 0.03s
— Isabelle says Unknown — CPU 11.93s
— Nitpick says Unknown — CPU 10.62s
— agsyHOL says Theorem — CPU 0.55s

```
%> ./FMLtHOL-P example.thf -timeout 20 -logic k -domain constant
```

FMLtoHOL combined with HOL-P

FO Modal Logic example: $(\Diamond\exists xPfx \wedge \Box\forall y(\Diamond Py \Rightarrow Qy)) \Rightarrow \Diamond\exists zQz$

```
%> ./FMLtoHOL-P example.thf -timeout 20 -logic s4 -domain varying
```

Calling HOL Resoners remotely in Miami ... thanks to Geoff Sutcliffe
— LEO-II says Theorem — CPU 0.08s
— Satallax says Theorem — CPU 0.03s
— Isabelle says Unknown — CPU 11.93s
— Nitpick says Unknown — CPU 10.62s
— agsyHOL says Theorem — CPU 0.55s

```
%> ./FMLtHOL-P example.thf -timeout 20 -logic k -domain constant
```

Calling HOL Resoners remotely in Miami ... thanks to Geoff Sutcliffe
— LEO-II says Unknown — CPU 11.93s
— Satallax says CounterSatisfiable — CPU 0.04s
— Isabelle says Unknown — CPU 16.19s
— Nitpick says CounterSatisfiable — CPU 8.19s
— agsyHOL says Unknown — CPU 10.82s

Evaluation: FML's (D — constant/varying/cumulative)

Freie Universität Berlin

No. of solved monomodal problems (out of 580 problems, 600sec timeout, inHOL-P a timeout of 120s was given to each of the 5 subprovers.)

	MleanSeP labelled sequents	**MleanTAP** labelled tableaux	**f2p-MSPASS** instant. & transform.	**MleanCoP** labelled connections	**HOL-P**
Logic D, constant domains					
Theorem	135	134	76	**217**	208
Non-Theorem	1	4	107	209	**250**
Solved	136	138	183	426	**458**
Logic D, cumulative domains					
Theorem	130	120	79	**200**	184
Non-Theorem	4	4	108	224	**269**
Solved	134	124	187	424	**453**
Logic D, varying domains					
Theorem	-	100	-	**170**	163
Non-Theorem	-	4	-	243	**295**
Solved	-	104	-	413	**458**

Evaluation: FML's (S4— constant/varying/cumulative)

No. of solved monomodal problems (out of 580 problems, 600sec timeout, inHOL-P a timeout of 120s was given to each of the 5 subprovers.)

	MleanSeP labelled sequents	**MleanTAP** labelled tableaux	**f2p-MSPASS** instant. & transform.	**MleanCoP** labelled connections	**HOL-P**
Logic S4, constant domains					
Theorem	197	220	111	**352**	300
Non-Theorem	1	4	36	82	**132**
Solved	198	224	147	**434**	432
Logic S4, cumulative domains					
Theorem	197	205	121	**338**	278
Non-Theorem	4	4	41	94	**146**
Solved	201	209	162	**432**	424
Logic S4, varying domains					
Theorem	-	169	-	**274**	245
Non-Theorem	-	4	-	119	**184**
Solved	-	173	-	393	**429**

Evaluation: Coverage of Logics

ATP system	supported modal logics	supported domain cond.
MleanSeP 1.2	K,K4,D,D4,T,S4	constant,cumulative
MleanTAP 1.3	D,T,S4,S5	constant,cumulative,varying
MleanCoP 1.2	D,T,S4,S5 (meanwhile extended)	constant,cumulative,varying
f2p-MSPASS 3.0	K,K4,K5,B,D,T,S4,S5	constant,cumulative
HOL-P	K,K4,K5,B,D,D4,T,S4,S5,...	constant,cumulative,varying

HOL-P directly applicable also for multi-modal logics.

New Evaluation: First Experiment

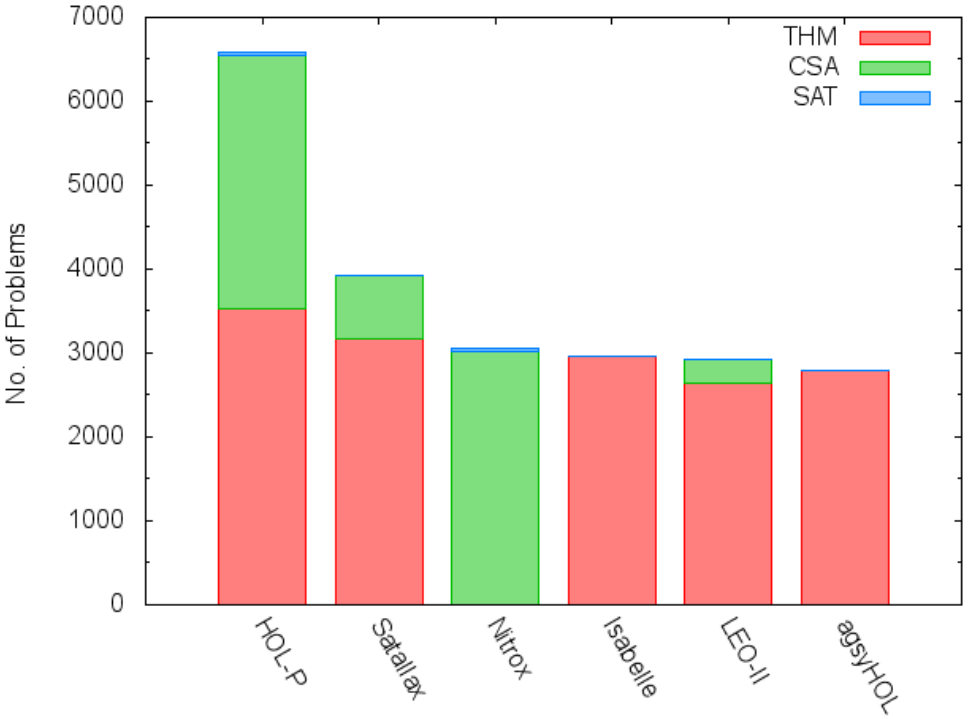
- HOL-P: sequentially schedules LEO-II—1.6.2, Satallax—2.7, Isabelle—2013, Nitrox—2013, agsyHOL—1.0.
- Timeout for each prover 120sec of CPU time (HOL-P 600sec).
- Experiments were run via SystemOnTPTP in Miami

	K			D			T			S4			S5		
Type	co	cu	va	co	cu	va	co	cu	va	co	cu	va	co	cu	va
THM	192	168	149	206	180	159	260	234	211	298	271	242	345	333	282
CSA	259	284	309	253	270	299	177	190	229	132	146	186	77	77	129
SAT	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
Σ	454	455	461	461	452	460	439	426	442	432	419	430	424	412	413
Σ				458	453	458				432	424	429			

- The particular results for logics D and S4 slightly differ from those reported at LPAR 2013
- Conjecture: Differences are related to SystemOnTPTP issues. How can the replication precision of experiments conducted via the SystemOnTPTP be improved?

New Evaluation: First Experiment

Cumulative performance of each prover (with a timeout of 120sec) for all 8700 QMLTP problem variants. The cumulative performance of HOL-P (with a 600sec timeout) is also depicted.



	THM	CSA	SAT	Σ	UNK
HOL-P	3530	3017	33	6580	2120
Satallax	3167	752	0	3919	4781
Nitrox	0	3017	33	3050	5650
Isabelle	2955	0	0	2955	5745
LEO-II	2647	284	0	2931	5769
agsyHOL	2784	0	0	2784	5916

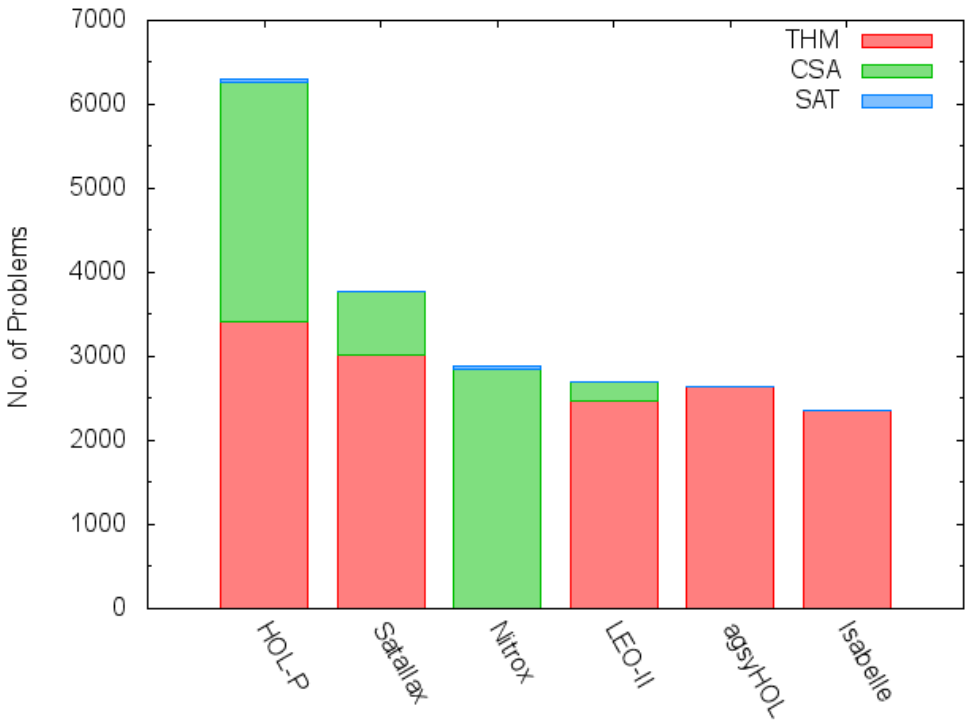
New Evaluation: Second Experiment

- HOL-P: sequentially schedules LEO-II—1.6.2, Satallax—2.7, Isabelle—2013, Nitrox—2013, agsyHOL—1.0.
- Timeout for each prover 20sec of CPU time (HOL-P 100sec).

	K			D			T			S4			S5		
Type	co	cu	va	co	cu	va	co	cu	va	co	cu	va	co	cu	va
THM	186	162	141	201	175	154	252	223	205	289	261	233	345	319	270
CSA	263	275	298	233	245	268	159	180	211	128	140	179	77	74	126
SAT	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
Σ	452	440	442	436	422	424	413	405	418	419	403	414	424	395	398

New Evaluation: Second Experiment

Cumulative performance of each prover (with a timeout of 20sec) for all 8700 QMLTP problem variants. The cumulative performance of HOL-P (with a 100sec timeout) is also depicted.



	THM	CSA	SAT	Σ	UNK
HOL-P	3408	2856	33	6297	2403
Satallax	3024	749	0	3773	4927
Nitrox	0	2856	33	2889	5811
LEO-II	2472	231	0	2703	5997
agsyHOL	2644	0	0	2644	6056
Isabelle	2354	0	0	2354	6346

New Evaluation: Third Experiment

- HOL-P: sequentially schedules only Satallax—2.7 and Nitrox—2013.
- Timeout for each prover 50sec of CPU time (HOL-P 100sec).

	K			D			T			S4			S5		
Type	co	cu	va	co	cu	va	co	cu	va	co	cu	va	co	cu	va
THM	162	150	132	175	161	141	225	212	190	262	246	219	305	305	258
THM	186	162	141	201	175	154	252	223	205	289	261	233	345	319	270
CSA	266	280	308	251	267	298	176	190	223	132	146	186	77	77	129
CSA	263	275	298	233	245	268	159	180	211	128	140	179	77	74	126
SAT	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
SAT	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
Σ	431	433	443	428	430	441	403	404	415	396	394	407	384	384	389
Σ	452	440	442	436	422	424	413	405	418	419	403	414	424	395	398

- performance of HOL-P in this experiment is weaker than in the second experiment
- illustrates the complementary strength of the HOL proveres for proving theorems
- however, the performance for finding countermodels (mainly by Nitrox) has slightly improved now for HOL-P

Conclusion

HOL based universal reasoning

- many quantified non-classical logics are fragments of HOL
- logic combinations: bridge rules as axioms
- cut-elimination and automation for free
- applications: expressive ontologies (SUMO, Cyc, Dolce, ...)

Other (implemented) approaches to compare with?

- Institutions are great — but not helpful for automation

Future work

- more embeddings (eg. multi-valued, paraconsistent)
- other combinations (eg. fibrings)
- range of embeddable logics
- scalability to real world applications

Conclusion

HOL based universal reasoning

- many quantified non-classical logics are fragments of HOL
- logic combinations: bridge rules as axioms
- cut-elimination and automation for free
- applications: expressive ontologies (SUMO, Cyc, Dolce, ...)

Other (implemented) approaches to compare with?

- Institutions are great — but not helpful for automation

Future work

- more embeddings (eg. multi-valued, paraconsistent)
- other combinations (eg. fibrings)
- range of embeddable logics
- scalability to real world applications

Conclusion

HOL based universal reasoning

- many quantified non-classical logics are fragments of HOL
- logic combinations: bridge rules as axioms
- cut-elimination and automation for free
- applications: expressive ontologies (SUMO, Cyc, Dolce, ...)

Other (implemented) approaches to compare with?

- Institutions are great — but not helpful for automation

Future work

- more embeddings (eg. multi-valued, paraconsistent)
- other combinations (eg. fibrings)
- range of embeddable logics
- scalability to real world applications

Automating Access Control Logics in Simple Type Theory with LEO-II[1]

Christoph Benzmüller

International University in Germany, Bruchsal, Germany
& Articulate Software, Angwin, CA, U.S.

IFIP/SEC-2009, Paphos, Cyprus, May 18-20, 2009

[1]This work was supported by EU grant PIIF-GA-2008-219982 (THFTPTP)

The Story — on a single slide

Simple Type Theory / HOL – an Expressive Logic

Multimodal Logics as Fragments of HOL

Access Control Logics as Fragments of S4 and hence HOL

Mechanization and Automation in HOL (prover LEO-II)

Simple Type Theory / HOL

Simple Type Theory / HOL

- simple types $\alpha, \beta ::= \iota | o | \alpha \to \beta$ (additional base types μ_i)
- simple type theory / HOL defined by

$$s, t ::= p_\alpha \mid X_\alpha \mid (\lambda X_\alpha . s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \mid (\neg_{o\to o}\, s_o)_o \mid (s_o \vee_{o\to o\to o}\, t_o)_o \mid (\Pi_{(\alpha\to o)\to o}\, t_{\alpha\to o})_o$$

- semantics well understood [Henkin50,Andrews72a/b,BenzmüllerEtAl04] - Henkin semantics

- base logic of many (interactive) proof assistants: Isabelle/HOL, HOL, HOL-light, PVS, OMEGA, ...
- (too) few ATPs so far $\longrightarrow$ EU IIF Project THFTPTP

Simple Type Theory / HOL – Expressivity

Property	FOL	HOL	Example
Quantification over			
- individuals	✓	✓	$\forall x.P(F(x))$
- functions	–	✓	$\forall F.P(F(x))$
- predicates/sets/relations	–	✓	$\forall P.P(F(x))$
Unnamed			
- functions	–	✓	$(\lambda x.x)$
- predicates/sets/relations	–	✓	$(\lambda x.x \neq 2)$
Statements about			
- functions	–	✓	$continuous(\lambda x.x)$
- predicates/sets/relations	–	✓	$reflexive(=)$

Multimodal Logics as Fragments of HOL

Multimodal Logics as Fragments of HOL

$$s, t ::= p \mid \neg s \mid s \vee t \mid \Box_r s$$

Simple, Straightforward Encoding

- base type ι: set of possible worlds
- (certain) terms of type $\iota \to o$: multimodal logic formulas

$$\begin{aligned} \lfloor \neg\ s \rfloor &= \lambda w_\iota . \neg(\lfloor s \rfloor\ w) \\ \lfloor s \vee t \rfloor &= \lambda w_\iota . \lfloor s \rfloor\ w \vee \lfloor t \rfloor\ w \\ \lfloor \Box_r\ s \rfloor &= \lambda w_\iota . \forall y_\iota . \lfloor r \rfloor\ w\ y \Rightarrow \lfloor s \rfloor\ y \\ \lfloor p \rfloor &= p_{\iota \to o} \end{aligned}$$

Related Work: [Gallin73], [Ohlbach88], [Carpenter98], [Merz99], [Brown05], [Hardt&Smolka07], [Kaminski&Smolka07]

Multimodal Logics as Fragments of HOL

$$s, t ::= p | \neg s | s \vee t | \Box_r s$$

Simple, Straightforward Encoding

- base type ι: set of possible worlds
- (certain) terms of type $\iota \to o$: multimodal logic formulas

$$\begin{aligned}
|\neg| &= \lambda s_{\iota\to o}.\, \lambda w_\iota.\, \neg(s\, w) \\
|\vee| &= \lambda s_{\iota\to o}.\, \lambda t_{\iota\to o}.\, \lambda w_\iota.\, s\, w \vee t\, w \\
|\Box| &= \lambda r_{\iota\to\iota\to o}.\, \lambda s_{\iota\to o}.\, \lambda w_\iota.\, \forall y_\iota.\, r\, w\, y \Rightarrow s\, y \\
|p| &= p_{\iota\to o} \\
|r| &= r_{\iota\to\iota\to o}
\end{aligned}$$

Related Work: [Gallin73], [Ohlbach88], [Carpenter98], [Merz99], [Brown05], [Hardt&Smolka07], [Kaminski&Smolka07]

(Normal) Multimodal Logic in HOL

Encoding of Validity

$$\begin{aligned} |\texttt{Mval}\ s_{\iota\to o}| &= \forall w_\iota . s\ w \\ |\texttt{Mval}| &= \lambda s_{\iota\to o} . \forall w_\iota . s\ w \end{aligned}$$

Local Definition Expansion

$$\begin{aligned} |\texttt{Mval}\ \Box_r\ \top| &= |\texttt{Mval}|\,|\Box|\,|r|\,|\top| \\ &=^{\beta\eta} \forall w_\iota . \forall y_\iota . r\ w\ y \Rightarrow \top \end{aligned}$$

(Normal) Multimodal Logic in HOL

Encoding of Validity

$$\begin{aligned} |\texttt{Mval}\ s_{\iota\to o}| &= \forall w_\iota . s\ w \\ |\texttt{Mval}| &= \lambda s_{\iota\to o} . \forall w_\iota . s\ w \end{aligned}$$

Local Definition Expansion

$$\begin{aligned} |\texttt{Mval}\ \Box_r\ \top| &= |\texttt{Mval}|\,|\Box|\,|r|\,|\top| \\ &=^{\beta\eta} \forall w_\iota . \forall y_\iota . r\ w\ y \Rightarrow \top \end{aligned}$$

(Normal) Multimodal Logic in HOL

Encoding of Validity

$$\begin{aligned} |\mathtt{Mval}\, s_{\iota\to o}| &= \forall w_\iota . s\, w \\ |\mathtt{Mval}| &= \lambda s_{\iota\to o} . \forall w_\iota . s\, w \end{aligned}$$

Local Definition Expansion

$$\begin{aligned} |\mathtt{Mval}\ \Box_r\ \top| &= |\mathtt{Mval}|\,|\Box|\,|r|\,|\top| \\ &=^{\beta\eta} \forall w_\iota . \forall y_\iota . r\, w\, y \Rightarrow \top \end{aligned}$$

Even simpler: Reasoning within Multimodal Logics

Problem	LEO-II
$\lvert\texttt{Mval}\ \Box_r\ \top\rvert$	0.025s
$\lvert\texttt{Mval}\ \Box_r\, a \supset \Box_r\, a\rvert$	0.026s
$\lvert\texttt{Mval}\ \Box_r\, a \supset \Box_s\, a\rvert$	–
$\lvert\texttt{Mval}\ \Box_s\,(\Box_r\, a \supset \Box_r\, a)\rvert$	0.026s
$\lvert\texttt{Mval}\ \Box_r\,(a \wedge b) \Leftrightarrow (\Box_r\, a \wedge \Box_r\, b)\rvert$	0.044s
$\lvert\texttt{Mval}\ \Diamond_r\,(a \supset b) \supset \Box_r\, a \supset \Diamond_r\, b\rvert$	0.030s
$\lvert\texttt{Mval}\ \neg\Diamond_r\, a \supset \Box_r\,(a \supset b)\rvert$	0.029s
$\lvert\texttt{Mval}\ \Box_r\, b \supset \Box_r\,(a \supset b)\rvert$	0.026s
$\lvert\texttt{Mval}\ (\Diamond_r\, a \supset \Box_r\, b) \supset \Box_r\,(a \supset b)\rvert$	0.027s
$\lvert\texttt{Mval}\ (\Diamond_r\, a \supset \Box_r\, b) \supset (\Box_r\, a \supset \Box_r\, b)\rvert$	0.029s
$\lvert\texttt{Mval}\ (\Diamond_r\, a \supset \Box_r\, b) \supset (\Diamond_r\, a \supset \Diamond_r\, b)\rvert$	0.030s

Example Proof: $|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$

Initialization of problem

$$\neg|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$$

Definition expansion

$$\neg(\forall x_\iota.\forall y_\iota.\neg s\,x\,y \vee ((\neg(\forall u_\iota.\neg r\,y\,u \vee a\,u)) \vee (\forall v_\iota.\neg r\,y\,v \vee a\,v)))$$

Normalization (x, y, u are now Skolem constants, V is a free variable)

$$\begin{array}{ll} s\,x\,y & \neg a\,u \\ r\,y\,u & a\,V \vee \neg r\,y\,V \end{array}$$

Translation to FOL [Kerber94], [Hurd02], [MengPaulson04]

$$\begin{array}{ll} [@^{\cdots}(@^{\cdots}(s,x),y)]^T & [@^{\cdots}(a,u)]^F \\ [@^{\cdots}(@^{\cdots}(r,y),u)]^T & [@^{\cdots}(a,V)]^T \vee [@^{\cdots}(@^{\cdots}(r,y),V)]^F \end{array}$$

Example Proof: $|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$

Initialization of problem

$$\neg|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$$

Definition expansion

$$\neg(\forall x_\iota.\forall y_\iota.\neg s\,x\,y \vee ((\neg(\forall u_\iota.\neg r\,y\,u \vee a\,u)) \vee (\forall v_\iota.\neg r\,y\,v \vee a\,v)))$$

Normalization (x, y, u are now Skolem constants, V is a free variable)

$$s\,x\,y \qquad \neg a\,u$$
$$r\,y\,u \qquad a\,V \vee \neg r\,y\,V$$

Translation to FOL [Kerber94], [Hurd02], [MengPaulson04]

$$[@^{\cdots}(@^{\cdots}(s,x),y)]^T \qquad [@^{\cdots}(a,u)]^F$$
$$[@^{\cdots}(@^{\cdots}(r,y),u)]^T \qquad [@^{\cdots}(a,V)]^T \vee [@^{\cdots}(@^{\cdots}(r,y),V)]^F$$

Example Proof: $|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$

Initialization of problem

$$\neg|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$$

Definition expansion

$$\neg(\forall x_\iota.\forall y_\iota.\neg s\,x\,y \vee ((\neg(\forall u_\iota.\neg r\,y\,u \vee a\,u)) \vee (\forall v_\iota.\neg r\,y\,v \vee a\,v)))$$

Normalization (x, y, u are now Skolem constants, V is a free variable)

$$\begin{array}{ll} s\,x\,y & \neg a\,u \\ r\,y\,u & a\,V \vee \neg r\,y\,V \end{array}$$

Translation to FOL [Kerber94], [Hurd02], [MengPaulson04]

$$\begin{array}{ll} [@^{\cdots}(@^{\cdots}(s,x),y)]^T & [@^{\cdots}(a,u)]^F \\ [@^{\cdots}(@^{\cdots}(r,y),u)]^T & [@^{\cdots}(a,V)]^T \vee [@^{\cdots}(@^{\cdots}(r,y),V)]^F \end{array}$$

Example Proof: $|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$

Initialization of problem

$$\neg|\texttt{Mval}\ \Box_s\,(\Box_r\,a \supset \Box_r\,a)|$$

Definition expansion

$$\neg(\forall x_\iota.\forall y_\iota.\neg s\,x\,y \vee ((\neg(\forall u_\iota.\neg r\,y\,u \vee a\,u)) \vee (\forall v_\iota.\neg r\,y\,v \vee a\,v)))$$

Normalization (x, y, u are now Skolem constants, V is a free variable)

$$\begin{array}{ll} s\,x\,y & \neg a\,u \\ r\,y\,u & a\,V \vee \neg r\,y\,V \end{array}$$

Translation to FOL [Kerber94], [Hurd02], [MengPaulson04]

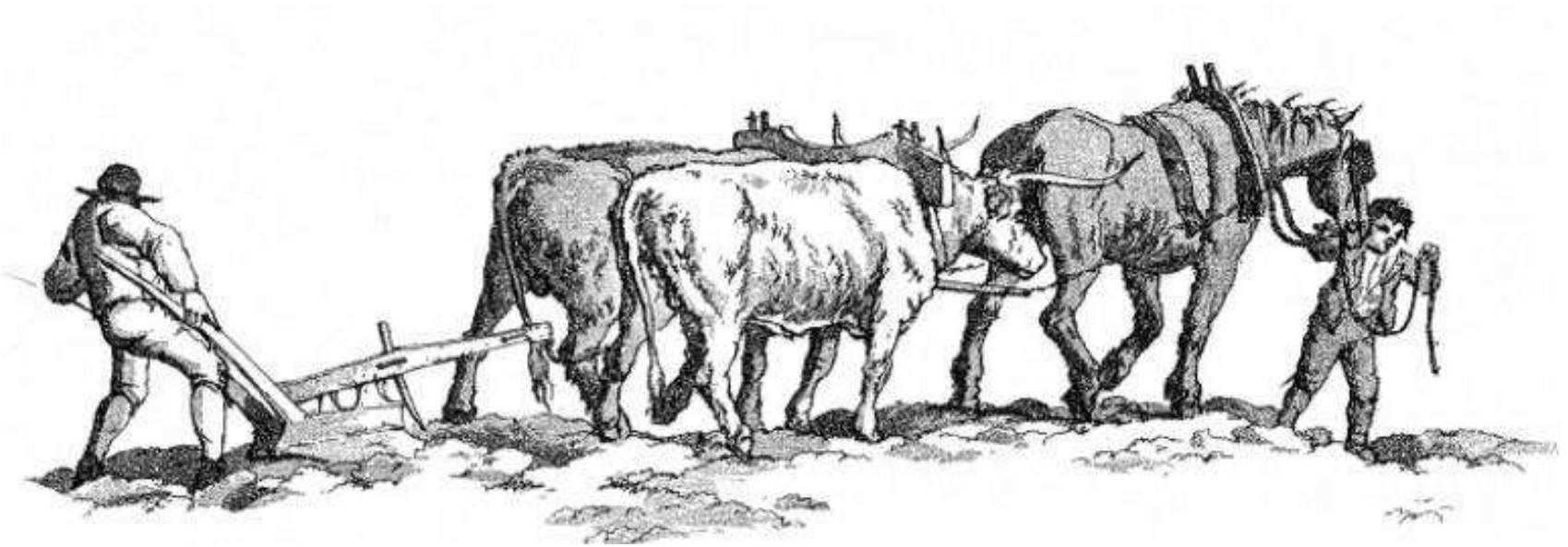
$$\begin{array}{ll} [@^{\cdots}(@^{\cdots}(s,x),y)]^T & [@^{\cdots}(a,u)]^F \\ [@^{\cdots}(@^{\cdots}(r,y),u)]^T & [@^{\cdots}(a,V)]^T \vee [@^{\cdots}(@^{\cdots}(r,y),V)]^F \end{array}$$

LEO-II

UNIVERSITY OF CAMBRIDGE

UNIVERSITÄT DES SAARLANDES

An Effective Higher-Order Theorem Prover



LEO-II employs FO-ATPs: E, Spass, Vampire

`www.leoprover.org`

Access Control Logics are fragments of S4 and hence HOL

[GargAbadi08]:
A Modal Deconstruction of Access Control Logics

- ICL: Propositional Intuitionistic Logic + "says"

$(\texttt{Admin says deletefile1}) \supset \texttt{deletefile1}$
If Admin says that file1 should be deleted, then this must be the case.

$\texttt{Admin says } ((\texttt{Bob says deletefile1}) \supset \texttt{deletefile1})$
Admin trusts Bob to decide whether file1 should be deleted.

$\texttt{Bob says deletefile1}$
Bob wants to delete file1.

$\texttt{deletefile1}$
Is deletion permitted?

Example I

[GargAbadi08]:
A Modal Deconstruction of Access Control Logics

- ICL: Propositional Intuitionistic Logic + "says"
- $\text{ICL}^{\Rightarrow}$: ICL + $\Longrightarrow$ (speaks for)

$(\texttt{Admin says deletefile1}) \supset \texttt{deletefile1}$
If Admin says that file1 should be deleted, then this must be the case.

$\texttt{Admin says } ((\texttt{Bob says deletefile1}) \supset \texttt{deletefile1})$
Admin trusts Bob to decide whether file1 should be deleted.

$\texttt{Bob says } (\texttt{Alice} \Longrightarrow \texttt{Bob})$
Bob delegates his authority to delete file1 to Alice

$\texttt{Alice says deletefile1}$
Alics wants to delete file1.

$\texttt{deletefile1}$
Is deletion permitted?

Example II

[GargAbadi08]:
A Modal Deconstruction of Access Control Logics

- ICL: Propositional Intuitionistic Logic + "says"
- $\text{ICL}^{\Rightarrow}$: ICL + $\Longrightarrow$ (speaks for)
- ICL^{B}: ICL + Boolean combinations of principals

$(\texttt{Admin says } \bot) \supset \texttt{deletefile1}$
Admin is trusted on deletefile1 and its consequences.

$\texttt{Admin says } ((\texttt{Bob} \supset \texttt{Admin}) \texttt{ says deletefile1})$
Admin further delegates this authority to Bob.

$\texttt{Bob says deletefile1}$
Bob wants to delete file1.

$\texttt{deletefile1}$
Is deletion permitted?

Example III

Access Control Logics

[GargAbadi08]:
A Modal Deconstruction of Access Control Logics

- ICL: Propositional Intuitionistic Logic + "says"
- $\text{ICL}^{\Rightarrow}$: ICL + $\Longrightarrow$ (speaks for)
- ICL^{B}: ICL + Boolean combinations of principals

Access Control Logics

[GargAbadi08]:
A Modal Deconstruction of Access Control Logics

- ICL: Propositional Intuitionistic Logic + "says"
- $\text{ICL}^{\Rightarrow}$: ICL + $\Longrightarrow$ (speaks for)
- ICL^{B}: ICL + Boolean combinations of principals

Sound and Complete Translations to Modal Logic S4

Access Control Logics

[GargAbadi08]:
A Modal Deconstruction of Access Control Logics

- ICL: Propositional Intuitionistic Logic + "says"
- $\text{ICL}^{\Rightarrow}$: ICL + $\Longrightarrow$ (speaks for)
- ICL^{B}: ICL + Boolean combinations of principals

Sound and Complete Translations to Modal Logic S4

So, let's combine this with our previous work . . . and apply LEO-II

Access Control Logics as Fragments of S4 and HOL

$$s, t ::= p \mid s \wedge t \mid s \vee t \mid s \supset t \mid \bot \mid \top \mid A \text{ says } s$$

Translation $\lceil . \rceil$ (of Garg and Abadi) into S4

$$\begin{aligned}
\lceil p \rceil &= \Box p \\
\lceil s \wedge t \rceil &= \lceil s \rceil \wedge \lceil t \rceil \\
\lceil s \vee t \rceil &= \lceil s \rceil \vee \lceil t \rceil \\
\lceil s \supset t \rceil &= \Box(\lceil s \rceil \supset \lceil t \rceil) \\
\lceil \top \rceil &= \top \\
\lceil \bot \rceil &= \bot \\
\lceil A \text{ says } s \rceil &= \Box(A \vee \lceil s \rceil)
\end{aligned}$$

Access Control Logics as Fragments of S4 and HOL

$$s, t ::= p \mid s \wedge t \mid s \vee t \mid s \supset t \mid \bot \mid \top \mid A \text{ says } s \mid s \Longrightarrow t$$

Translation $\lceil . \rceil$ (of Garg and Abadi) into S4

$$\begin{aligned}
\lceil p \rceil &= \Box p \\
\lceil s \wedge t \rceil &= \lceil s \rceil \wedge \lceil t \rceil \\
\lceil s \vee t \rceil &= \lceil s \rceil \vee \lceil t \rceil \\
\lceil s \supset t \rceil &= \Box(\lceil s \rceil \supset \lceil t \rceil) \\
\lceil \top \rceil &= \top \\
\lceil \bot \rceil &= \bot \\
\lceil A \text{ says } s \rceil &= \Box(A \vee \lceil s \rceil) \\
\lceil s \Longrightarrow t \rceil &= \Box(\lceil s \rceil \supset \lceil t \rceil)
\end{aligned}$$

Access Control Logics as Fragments of S4 and HOL

$$s, t ::= p \mid s \wedge t \mid s \vee t \mid s \supset t \mid \bot \mid \top \mid A \ \mathtt{says}\ s \mid s \Longrightarrow t$$

Translation $\|.\|$ to HOL

$$
\begin{array}{lcl}
 & & |r| \quad \text{(we fix one single } r\text{!!!)} \\
\|p\| & = & |\Box_r\, p| \\
\|A\| & = & |A| \\
\|\wedge\| & = & \lambda s.\, \lambda t.\, |s \wedge t| \\
\|\vee\| & = & \lambda s.\, \lambda t.\, |s \vee t| \\
\|\supset\| & = & \lambda s.\, \lambda t.\, |\Box(s \supset t)| \\
\|\top\| & = & |\top| \\
\|\bot\| & = & |\bot| \\
\|\mathtt{says}\| & = & \lambda A.\, \lambda s.\, |\Box_r\, (A \vee s)| \\
\|\Longrightarrow\| & = & \lambda s.\, \lambda t.\, |\Box_r\, (s \supset t)|
\end{array}
$$

Access Control Logics as Fragments of S4 and HOL

$$s, t ::= p \mid s \wedge t \mid s \vee t \mid s \supset t \mid \bot \mid \top \mid A \texttt{ says } s \mid s \Longrightarrow t$$

Translation $\|.\|$ to HOL

$$
\begin{array}{lcl}
 & & r_{\iota\to\iota\to o} \quad \text{(we fix one single } r\text{!!!)} \\
\|p\| & = & \lambda x_\iota.\, \forall y_\iota.\, r_{\iota\to\iota\to o}\, x\, y \Rightarrow p_{\iota\to o}\, Y \\
\|A\| & = & a_{\iota\to o} \quad \texttt{(distinct from the } p_{\iota\to o}\texttt{)} \\
\|\wedge\| & = & \lambda s_{\iota\to o}.\, \lambda t_{\iota\to o}.\, \lambda w_\iota.\, s\, w \wedge t\, w \\
\|\vee\| & = & \lambda s_{\iota\to o}.\, \lambda t_{\iota\to o}.\, \lambda w_\iota.\, s\, w \vee t\, w \\
\|\supset\| & = & \lambda s_{\iota\to o}.\, \lambda t_{\iota\to o}.\, \lambda w_\iota.\, \forall y_\iota.\, r\, w\, y \Rightarrow (s\, y \Rightarrow t\, y) \\
\|\top\| & = & \lambda s_{\iota\to o}.\, \top \\
\|\bot\| & = & \lambda s_{\iota\to o}.\, \bot \\
\|\texttt{says}\| & = & \lambda A_{\iota\to o}.\, \lambda s_{\iota\to o}.\, \lambda w_\iota.\, \forall y_\iota.\, r\, w\, y \Rightarrow (A\, y \vee s\, y) \\
\|\Longrightarrow\| & = & \lambda s_{\iota\to o}.\, \lambda t_{\iota\to o}.\, \lambda w_\iota.\, \forall y_\iota.\, r\, w\, y \Rightarrow (s\, y \Rightarrow t\, y)
\end{array}
$$

Access Control Logics as Fragments of S4 and HOL

Notion of Validity

$$\texttt{ICLval} = \texttt{Mval}$$

Addition of Modal Logic Axioms for S4

$$\forall p_{\iota\to o}.|\texttt{Mval}\ \Box_r\, p \supset p|$$

$$\forall p_{\iota\to o}.|\texttt{Mval}\ \Box_r\, p \supset\ \Box_r \Box_r\, p|$$

Soundness and Completeness of Embedding

Proof: see paper; employs transformation from Kripke models into corresponding Henkin models and vice versa; combines this with results of [GargAbadi08]

Access Control Logics as Fragments of S4 and HOL

Notion of Validity

$$\texttt{ICLval} = \texttt{Mval}$$

Addition of Modal Logic Axioms for S4

$$\forall p_{\iota\to o}.\lfloor \texttt{Mval}\ \Box_r\, p \supset p \rfloor$$

$$\forall p_{\iota\to o}.\lfloor \texttt{Mval}\ \Box_r\, p \supset \Box_r \Box_r\, p \rfloor$$

Soundness and Completeness of Embedding

Proof: see paper; employs transformation from Kripke models into corresponding Henkin models and vice versa; combines this with results of [GargAbadi08]

Access Control Logics as Fragments of S4 and HOL

Notion of Validity

$$\texttt{ICLval} = \texttt{Mval}$$

Addition of Modal Logic Axioms for S4

$$\forall p_{\iota\to o}.|\texttt{Mval}\ \Box_r\ p \supset p|$$

$$\forall p_{\iota\to o}.|\texttt{Mval}\ \Box_r\ p \supset\ \Box_r\ \Box_r\ p|$$

Soundness and Completeness of Embedding

Proof: see paper; employs transformation from Kripke models into corresponding Henkin models and vice versa; combines this with results of [GargAbadi08]

Access Control Logics as Fragments of S4 and HOL

Example I (from [GargAbadi08]):

$\texttt{ICLval (Admin says deletefile1)} \supset \texttt{deletefile1}$
If Admin says that file1 should be deleted, then this must be the case.

$\texttt{ICLval Admin says ((Bob says deletefile1)} \supset \texttt{deletefile1)}$
Admin trusts Bob to decide whether file1 should be deleted.

$\texttt{ICLval Bob says deletefile1}$
Bob wants to delete file1.

$\texttt{ICLval deletefile1}$
Is deletion permitted?

Access Control Logics as Fragments of S4 and HOL

Example I (from [GargAbadi08]):

$\|\texttt{ICLval (Admin says deletefile1)} \supset \texttt{deletefile1}\|$
If Admin says that file1 should be deleted, then this must be the case.

$\|\texttt{ICLval Admin says ((Bob says deletefile1)} \supset \texttt{deletefile1)}\|$
Admin trusts Bob to decide whether file1 should be deleted.

$\|\texttt{ICLval Bob says deletefile1}\|$
Bob wants to delete file1.

$\|\texttt{ICLval deletefile1}\|$
Is deletion permitted?

Access Control Logics as Fragments of S4 and HOL

Example I (from [GargAbadi08]):

$\|\texttt{ICLval (Admin says deletefile1)} \supset \texttt{deletefile1}\|$
If Admin says that file1 should be deleted, then this must be the case.

$\|\texttt{ICLval Admin says ((Bob says deletefile1)} \supset \texttt{deletefile1)}\|$
Admin trusts Bob to decide whether file1 should be deleted.

$|\texttt{Mval } \Box_r \texttt{(Bob} \vee \Box_r \texttt{deletefile1)}|$
Bob wants to delete file1.

$\|\texttt{ICLval deletefile1}\|$
Is deletion permitted?

Access Control Logics as Fragments of S4 and HOL

Example I (from [GargAbadi08]):

$\|\texttt{ICLval (Admin says deletefile1)} \supset \texttt{deletefile1}\|$
If Admin says that file1 should be deleted, then this must be the case.

$\|\texttt{ICLval Admin says ((Bob says deletefile1)} \supset \texttt{deletefile1)}\|$
Admin trusts Bob to decide whether file1 should be deleted.

$\forall w_\iota . \forall y_\iota . r\ w\ y \Rightarrow (Bob\ y \vee \forall u_\iota . r\ w\ u \Rightarrow deletefile1\ u)$
Bob wants to delete file1.

$\|\texttt{ICLval deletefile1}\|$
Is deletion permitted?

LEO-II: 0.301 seconds

More Examples from [GargAbadi08]

- Example I: 0.301 seconds
- Example II (ICL$^{\Rightarrow}$): 0.503 seconds
- Example III (ICLB): 0.077 seconds

Also possible: reasoning about meta-properties

- ICL$^{\Rightarrow}$ can be expressed in ICL^{B}: 0.073 seconds

Exp.: Access Control Logic in HOL

ICL:

Name	Problem	LEO (s)
unit	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \|\mathtt{ICLval}\ s \supset (A\ \mathtt{says}\ s)\|$	0.053
cuc	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \|\mathtt{ICLval}$ $(A\ \mathtt{says}\ (s \supset t)) \supset (A\ \mathtt{says}\ s) \supset (A\ \mathtt{says}\ t)\|$	0.167
idem	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \|\mathtt{ICLval}\ (A\ \mathtt{says}\ A\ \mathtt{says}\ s) \supset (A\ \mathtt{says}\ s)\|$	0.058
unit^K	$\models^{HOL} \|\mathtt{ICLval}\ s \supset (A\ \mathtt{says}\ s)\|$	–
cuc^K	$\models^{HOL} \|\mathtt{ICLval}\ (A\ \mathtt{says}\ (s \supset t)) \supset (A\ \mathtt{says}\ s) \supset (A\ \mathtt{says}\ t)\|$	–
idem^K	$\models^{HOL} \|\mathtt{ICLval}\ (A\ \mathtt{says}\ A\ \mathtt{says}\ s) \supset (A\ \mathtt{says}\ s)\|$	–

R, T: reflexivity and transitivity axioms for S4 as seen before

Exp.: Access Control Logic in HOL

$\mathsf{ICL}^{\Rightarrow}$:

Name	Problem	LEO (s)
refl	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \lVert \mathtt{ICLval}\ A \implies A \rVert$	0.059
trans	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \lVert \mathtt{ICLval}\ (A \implies B) \supset (B \implies C) \supset (A \implies C) \rVert$	0.083
sp.-for	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \lVert \mathtt{ICLval}\ (A \implies B) \supset (A\ \mathtt{says}\ s) \supset (B\ \mathtt{says}\ s) \rVert$	0.107
handoff	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \lVert \mathtt{ICLval}\ (B\ \mathtt{says}\ (A \implies B)) \supset (A \implies B) \rVert$	0.075
refl^K	$\models^{HOL} \lVert \mathtt{ICLval}\ A \implies A \rVert$	0.034
trans^K	$\models^{HOL} \lVert \mathtt{ICLval}\ (A \implies B) \supset (B \implies C) \supset (A \implies C) \rVert$	–
$\mathsf{sp.\text{-}for}^K$	$\models^{HOL} \lVert \mathtt{ICLval}\ (A \implies B) \supset (A\ \mathtt{says}\ s) \supset (B\ \mathtt{says}\ s) \rVert$	–
$\mathsf{handoff}^K$	$\models^{HOL} \lVert \mathtt{ICLval}\ (B\ \mathtt{says}\ (A \implies B)) \supset (A \implies B) \rVert$	–

R, T: reflexivity and transitivity axioms as for S4 seen before

Exp.: Access Control Logic in HOL

ICL^B:

Name	Problem	LEO (s)
trust	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \|\mathtt{ICLval}\ (\bot\ \mathtt{says}\ s) \supset s\|$	0.058
untrust	$\{\mathtt{R},\mathtt{T},\|\mathtt{ICLval}\ A \equiv \top\|\} \models^{HOL} \|\mathtt{ICLval}\ A\ \mathtt{says}\ \bot\|$	0.046
cuc'	$\{\mathtt{R},\mathtt{T}\} \models^{HOL} \|\mathtt{ICLval}$ $((A \supset B)\ \mathtt{says}\ s) \supset (A\ \mathtt{says}\ s) \supset (B\ \mathtt{says}\ s)\|$	0.200
trust^K	$\models^{HOL} \|\mathtt{ICLval}\ (\bot\ \mathtt{says}\ s) \supset s\|$	–
$\mathrm{untrust}^K$	$\{\|\mathtt{ICLval}\ A \equiv \top\|\} \models^{HOL} \|\mathtt{ICLval}\ A\ \mathtt{says}\ \bot\|$	0.055
$\mathrm{cuc'}^K$	$\models^{HOL} \|\mathtt{ICLval}\ ((A \supset B)\ \mathtt{says}\ s) \supset (A\ \mathtt{says}\ s) \supset (B\ \mathtt{says}\ s)\|$	–

R, T: reflexivity and transitivity axioms for S4 as seen before

Conclusion

- Prominent Access Control Logics are fragments of HOL
- Interactive and automated HOL provers can generally be applied for reasoning in and **about** these logics
- Challenge: How good does approach scale?
- Examples submitted to THFTPTP

Ongoing and Future Research

- THFTPTP infrastructure
- Improvement of LEO-II – make it scale for larger examples
- Combination of different logics
- Formal verification of approach e.g. in Isabelle/HOL

THFTPTP

(EU grant THFTPTP – PIIF-GA-2008-219982)

Thanks to hard working Geoff Sutcliffe

THFTPTP – Progress in ATP for HOL

- THF syntax for HOL
- library for HOL ($>$ 2700 problems)
- tools for HOL
  (parser, type checker, pretty printer, ...)
- integrated HOL ATPs: IsabelleP, TPS, LEO-II
- integrated HOL model generator: IsabelleM
- SystemOnTPTP online interface

THFTPTP – Progress in ATP for HOL

ALG	higher-order abstract syntax
GRA	Ramsey numbers (several open)
LCL	modal logic
NUM	Landau's Grundlagen
PUZ	puzzles
SET/SEU	set theory, dependently typed set theory, binary relations
SWV	security, access control logic
SYN/SYO	simple test problems

	ALG	GRA	LCL	NUM	PUZ	SE?	SWV	SY?	Total	Unique
Problems	50	93	61	221	5	749	37	59	1275	
THM/UNS	50	25	51	221	5	746	25	47	1170	
CSA/SAT	0	0	10	0	0	3	5	11	29	
LEO-II 0.99a	34	0	48	181	3	401	19	42	725	127
IsabelleP 2008	0	0	0	197	5	361	1	30	594	74
TPS 3.0	10	0	40	150	3	285	9	35	532	6
Any	32	0	50	203	5	490	20	52	843	207
All	0	0	0	134	2	214	0	22	372	
None	18	93	12	18	0	259	17	15	432	
IsabelleM 2008	0	0	1	0	0	0	0	8	9	

LEO-II

(EPRSC grant EP/D070511/1 at Cambridge University)

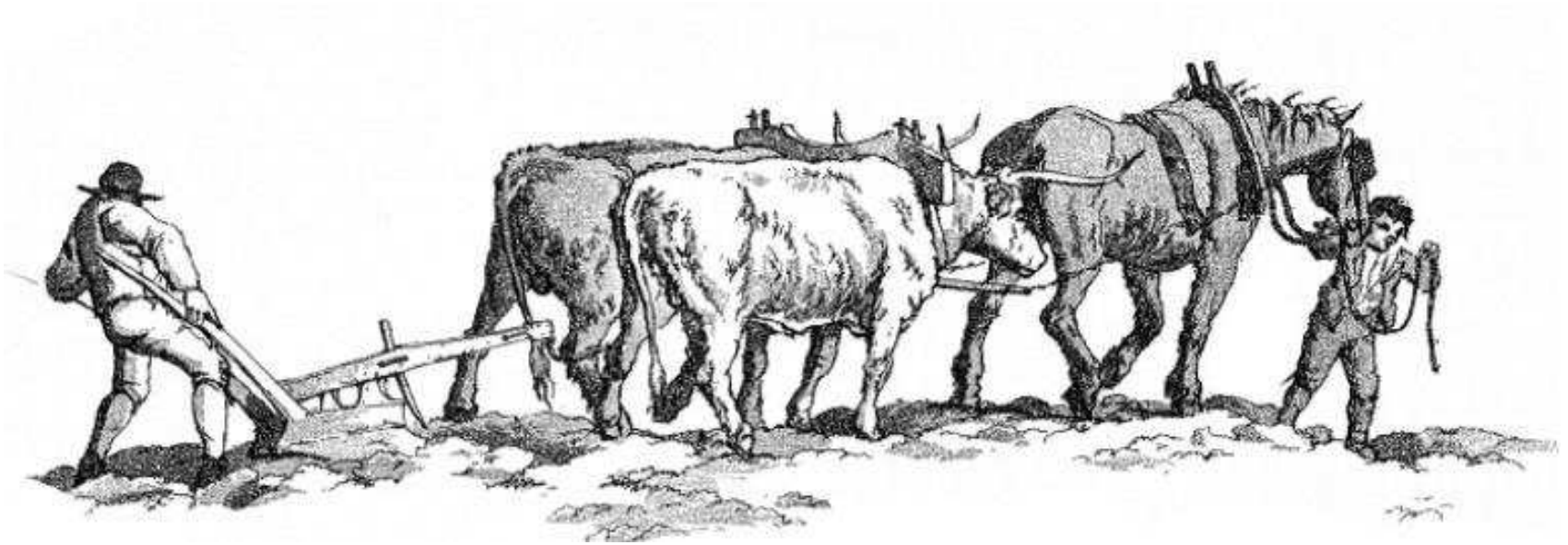
Thanks to Larry Paulson

LEO-II

UNIVERSITY OF CAMBRIDGE

UNIVERSITÄT DES SAARLANDES

An Effective Higher-Order Theorem Prover



LEO-II employs FO-ATPs: E, Spass, Vampire

`http://www.ags.uni-sb.de/~leo`