Utilizing Church's Type Theory as a Universal Logic[1]

Christoph Benzmüller

Freie Universität Berlin

Presentation at TU Wien, October 31, 2012

[1]This work has been funded by the DFG under grants BE 2501/6-1, BE 2501/8-1 and BE 2501/9-1

for not visiting earlier!

Presentation Overview

Core questions of my current research:

1. Classical Higher-order Logic (HOL) as Universal Logic?
2. HOL Provers & Model Finders as Generic Reasoning Tools?
3. Integration of Specialist Reasoners (if available)?

Presentation Overview

Core questions of my current research:

1. Classical Higher-order Logic (HOL) as Universal Logic?
2. HOL Provers & Model Finders as Generic Reasoning Tools?
3. Integration of Specialist Reasoners (if available)?

Talk Outline:

- Classical Higher-order Logic HOL & HOL-ATP
- Examples of Natural Fragments of HOL:
  - Quantified Multimodal Logics (QMLs)
  - Quantified Conditional Logics (QCLs) – if time permits –
- Reasoning *about* Logics (and their Combinations)
- Evaluation of HOL-ATPs for reasoning *within* QMLs
- Short Demonstration
- Conclusion

My Motivation

Automated Reasoning *within* and *about* Expressive Ontologies?

- Expressive Ontologies: SUMO (Adam Pease) or Cyc (Doug Lenat)
- They have often been advertised as "first-order" ontologies, but they are not!
  - They contain higher-order constructs
  - They contain modal operators

`holdsDuring` — `knows` — `believes` — ...

→ Limited automation with traditional FOL-ATPs

Hypothesis: We can do better with HOL-ATPs

[BenzmüllerPease, J. Web Semantics, 2012]

HOL & HOL-ATP

(Classical Higher-order Logic/Church's Type Theory)

Expressivity	FOL	HOL	Example
Quantification over			
- Individuals	✓	✓	$\forall X.\, p(f(X))$
- Functions	–	✓	$\forall F.\, p(F(a))$
- Predicates/Sets/Rels	–	✓	$\forall P.\, P(f(a))$
Unnamed			
- Functions	–	✓	$(\lambda X.\, X)$
- Predicates/Sets/Rels	–	✓	$(\lambda X.\, X \neq a)$
Statements about			
- Functions	–	✓	$\mathit{continuous}(\lambda X.\, X)$
- Predicates/Sets/Rels	–	✓	$\mathit{reflexive}(=)$
Powerful abbreviations	–	✓	$\mathit{reflexive} = \lambda R.\, \lambda X.\, R(X, X)$

What is HOL? (Church's Type Theory, Alonzo Church, 1940)

Expressivity	FOL	HOL	Example
Quantification over			
- Individuals	✓	✓	$\forall X_{\iota}.\, p_{\iota\to o}(f_{\iota\to\iota}(X_{\iota}))$
- Functions	–	✓	$\forall F_{\iota\to\iota}.\, p_{\iota\to o}(F_{\iota\to o}(a_{\iota}))$
- Predicates/Sets/Rels	–	✓	$\forall P_{\iota\to o}.\, P_{\iota\to o}(f_{\iota\to\iota}(a_{\iota}))$
Unnamed			
- Functions	–	✓	$(\lambda X_{\iota}.\, X_{\iota})$
- Predicates/Sets/Rels	–	✓	$(\lambda X_{\iota\to\iota}.\, X_{\iota\to\iota} \neq_{\iota\to\iota\to p} a)_{\iota})$
Statements about			
- Functions	–	✓	$continuous_{(\iota\to\iota)\to o}(\lambda X_{\iota}.\, X_{\iota})$
- Predicates/Sets/Rels	–	✓	$reflexive_{(\iota\to\iota\to o)\to o}(=_{\iota\to\iota\to o})$
Powerful abbreviations	–	✓	$reflexive_{(\iota\to\iota\to o)\to o} = \lambda R_{(\iota\to\iota\to o)}.\, \lambda X_{\iota}.\, R$

Simple Types: Prevent Paradoxes and Inconsistencies

What is HOL? (Alonzo Church, 1940)

- Simple Types $\alpha ::= \iota \mid o \mid \alpha_1 \rightarrow \alpha_2$

What is HOL? (Alonzo Church, 1940)

- Simple Types $\alpha ::= \iota \mid o \mid \alpha_1 \rightarrow \alpha_2$

Individuals — ι

Booleans (True and False) — o

Functions — $\alpha_1 \rightarrow \alpha_2$

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

Possible worlds

Individuals

Booleans (True and False)

Functions

- Simple Types $\alpha ::= \iota \mid \mu \mid o \mid \alpha_1 \rightarrow \alpha_2$
- HOL Syntax

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

Constant Symbols

Variable Symbols

- Simple Types $\alpha ::= \iota \mid \mu \mid o \mid \alpha_1 \rightarrow \alpha_2$
- HOL Syntax

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

Constant Symbols
Variable Symbols
Abstraction
Application

- Simple Types $\alpha ::= \iota \mid \mu \mid o \mid \alpha_1 \rightarrow \alpha_2$
- HOL Syntax

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

Constant Symbols
Variable Symbols
Abstraction
Application
Logical Connectives

- Simple Types $\alpha ::= \iota \mid \mu \mid o \mid \alpha_1 \rightarrow \alpha_2$
- HOL Syntax

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

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

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

- HOL is (meanwhile) 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 with Henkin-Semantics: semi-decidable & compact (like FOL)

EU Project THFTPTP: An Infrastructure for HOL-ATP

Results of the EU Project THFTPTP

- Collaboration with Geoff Sutcliffe, Chad Brown and others
- Results
  - THF0 syntax for HOL
  - Online access to provers
  - Library with example problems (e.g. entire TPS library) and results
  - Ontology and syntax for proof results
  - International CASC competition for HOL-ATP
  - Various tools

Improved availability and robustness of HOL-ATPs: TPS, LEO-II, Isabelle, Satallax, Refute, Nitpick, agsyHOL `http://www.tptp.org/cgi-bin/SystemOnTPTP`

[SutcliffeBenzmüller, J. Formalized Reasoning, 2010]

[BenzmüllerRabeSutcliffe, IJCAR, 2008]

CASC Competitions in THF (HOL) Category since 2009

2009

THF	TPS 3.20080227G1d	LEO-II 1.0	LEO-IIP 1.0	IsabelleP 2009
Attempted	200	200	200	200
Solved	170	146	146	124
Av. Time	23.18	2.27	3.44	55.92
Solutions	0	0	146	124

CASC Competitions in THF (HOL) Category since 2009

2009

THF	TPS 3.20080227G1d	LEO-II 1.0	LEO-IIP 1.0	IsabelleP 2009
Attempted	200	200	200	200
Solved	170	146	146	124
Av. Time	23.18	2.27	3.44	55.92
Solutions	0	0	146	124

2010

THF	LEO-II 1.2	Satallax 1.4	IsabelleP 2009-2	TPS 3.20080227G1d
Solved	125/200	120/200	101/200	80/200
Av. CPU Time	16.65	55.24	100.75	36.15
Solutions	125/200	120/200	0/200	0/200

LEO-II 1.2 solved 56% more than previous winner

CASC Competitions in THF (HOL) Category since 2009

2009

THF	TPS 3.20080227G1d	LEO-II 1.0	LEO-IIP 1.0	IsabelleP 2009
Attempted	200	200	200	200
Solved	170	146	146	124
Av. Time	23.18	2.27	3.44	55.92
Solutions	0	0	146	124

2010

THF	LEO-II 1.2	Satallax 1.4	IsabelleP 2009-2	TPS 3.20080227G1d
Solved	125/200	120/200	101/200	80/200
Av. CPU Time	16.65	55.24	100.75	36.15
Solutions	125/200	120/200	0/200	0/200

LEO-II 1.2 solved 56% more than previous winner

2011

THF/300	Satallax 2.1	LEO-II 1.2.8	LEO-II 1.2	Isabelle 2011	TPS 3.110228S1a
Solved	246/300	208/300	204/300	201/300	190/300
Av. CPU Time	12.04	8.97	4.95	36.55	18.69

Satallax 2.1 solved 21% more than previous winner

CASC Competitions in THF (HOL) Category since 2009

2010

THF	LEO-II 1.2	Satallax 1.4	IsabelleP 2009-2	TPS 3.20080227G1d
Solved	125/200	120/200	101/200	80/200
Av. CPU Time	16.65	55.24	100.75	36.15
Solutions	125/200	120/200	0/200	0/200

LEO-II 1.2 solved 56% more than previous winner

2011

THF/300	Satallax 2.1	LEO-II 1.2.8	LEO-II 1.2	Isabelle 2011	TPS 3.110228S1a
Solved	246/300	208/300	204/300	201/300	190/300
Av. CPU Time	12.04	8.97	4.95	36.55	18.69

Satallax 2.1 solved 21% more than previous winner

2012

Higher-order Theorems	Isabelle-H 2012	Isabelle 2012	Satallax 2.4	Satallax 2.1	LEO-II 1.4.0	TPS 3.120601S1b
Solved/200	166/200	135/200	132/200	123/200	81/200	66/200
Av. CPU Time	88.44	70.13	16.20	19.57	11.38	25.23

Isabelle-HOT solved 35% more than previous winner

CASC Competitions in THF (HOL) Category since 2009

2012

Higher-order Theorems	Isabelle-H 2012	Isabelle 2012	Satallax 2.4	*Satallax* *2.1*	LEO-II 1.4.0	TPS 3.120601S1b
Solved/200	166/200	135/200	132/200	123/200	81/200	66/200
Av. CPU Time	88.44	70.13	16.20	19.57	11.38	25.23

2012

Higher-order Theorems	Isabelle-H 2012	Isabelle 2012	Satallax 2.4	*Satallax* 2.1	LEO-II 1.4.0	TPS 3.120601S1b
Solved/200	166/200	135/200	132/200	123/200	81/200	66/200
Av. CPU Time	88.44	70.13	16.20	19.57	11.38	25.23

LEO-II cooperates with FOL prover E

CASC Competitions in THF (HOL) Category since 2009

2012

Higher-order Theorems	Isabelle-H 2012	Isabelle 2012	Satallax 2.4	*Satallax* *2.1*	LEO-II 1.4.0	TPS 3.120601S1b
Solved/200	166/200	135/200	132/200	123/200	81/200	66/200
Av. CPU Time	88.44	70.13	16.20	19.57	11.38	25.23

Satallax cooperates with SAT solver Minisat

CASC Competitions in THF (HOL) Category since 2009

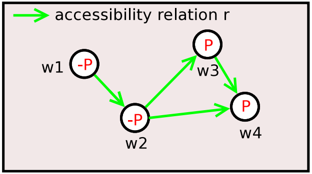
2012

Higher-order Theorems	Isabelle-H 2012	Isabelle 2012	Satallax 2.4	*Satallax* 2.1	LEO-II 1.4.0	TPS 3.120601S1b
Solved/200	166/200	135/200	132/200	123/200	81/200	66/200
Av. CPU Time	88.44	70.13	16.20	19.57	11.38	25.23

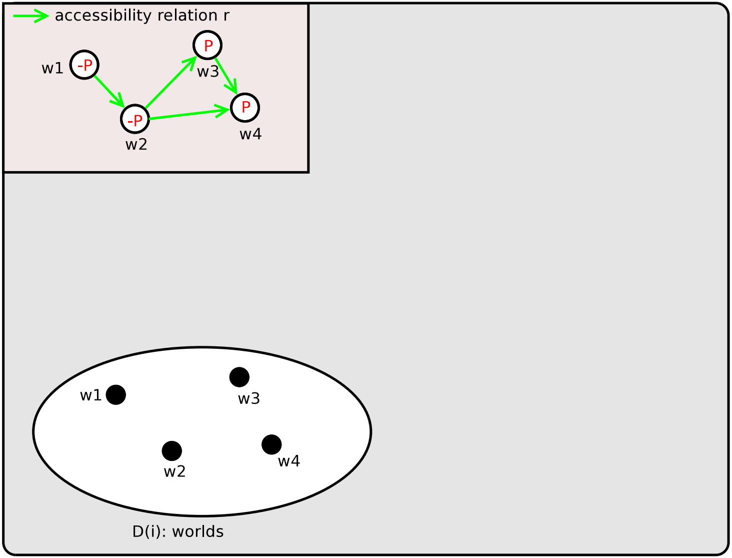
Isabelle-HOT cooperates with various FOL provers (sledgehammer) and SMT solvers (smt) and even with LEO-II and Satallax

Natural Fragments of HOL:
Quantified Multimodal Logics

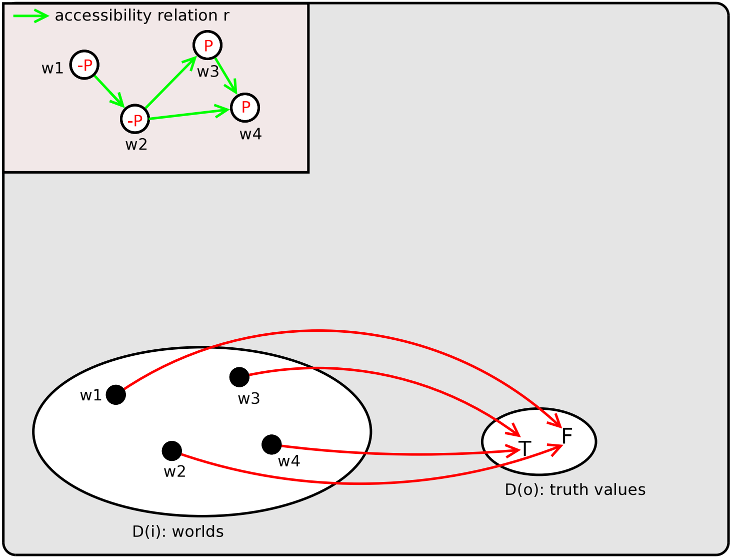
Combining the Kripke View and the Tarski View on Logics



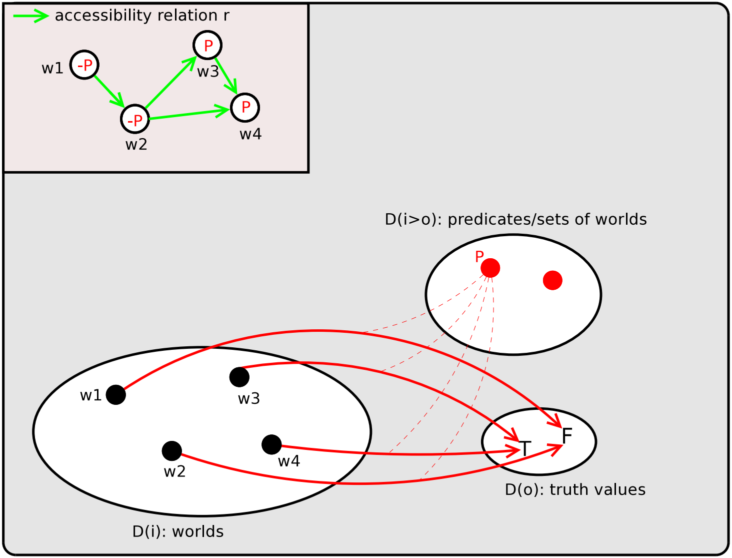
Combining the Kripke View and the Tarski View on Logics



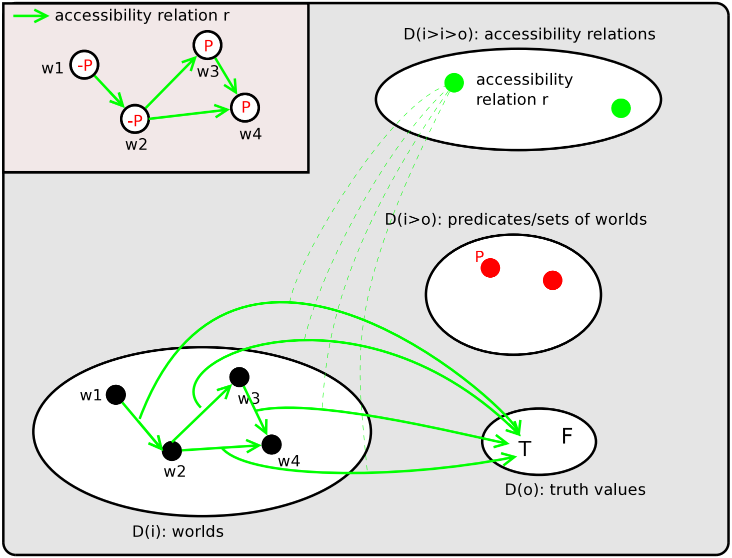
Combining the Kripke View and the Tarski View on Logics



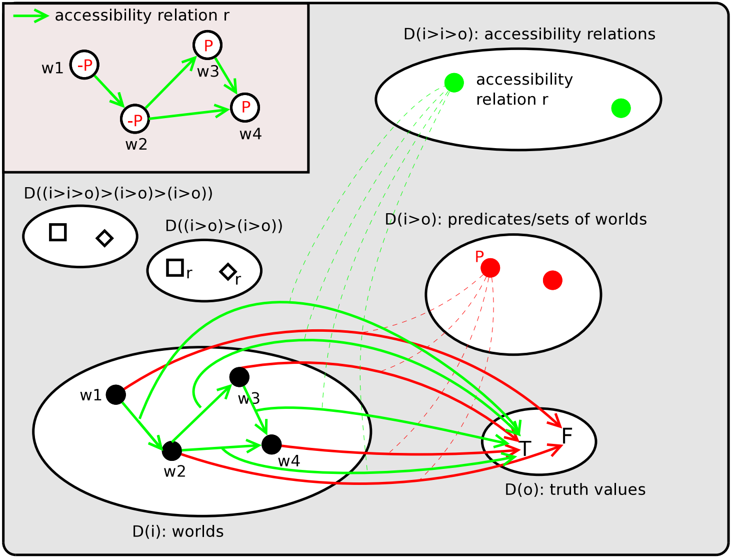
Combining the Kripke View and the Tarski View on Logics



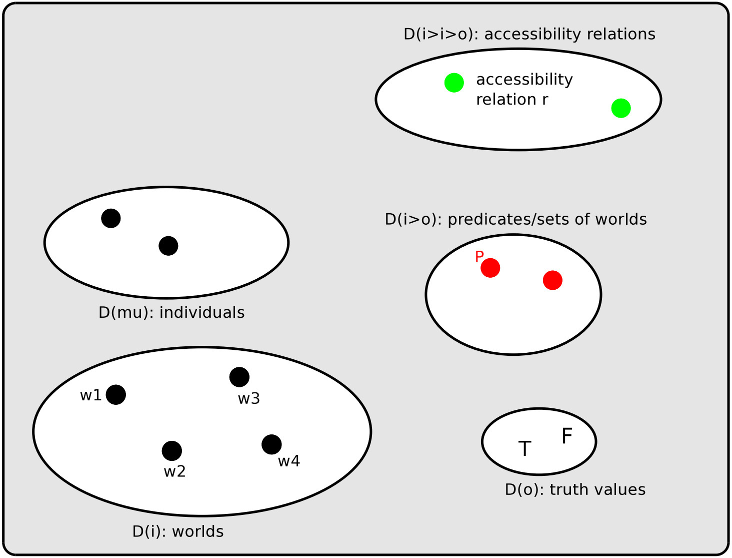
Combining the Kripke View and the Tarski View on Logics



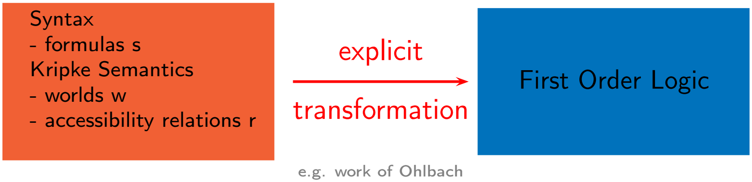
Combining the Kripke View and the Tarski View on Logics



Combining the Kripke View and the Tarski View on Logics



- Syntax: $s, t \quad ::= \quad P \mid \neg\, s \mid s \vee t \mid \Box_r\, s \mid \ldots$

- 

Not Needed!

(Multi-) Modal Logics in HOL

- Syntax: $s, t \quad ::= \quad P \mid \neg\, s \mid s \vee t \mid \Box_r\, s \mid \ldots$

- **HOL**

Syntax - formulas s	$\longrightarrow$	terms $s_{\iota\to o}$
Kripke Semantics - worlds w	$\longrightarrow$	terms w_ι
- accessibility relations r	$\longrightarrow$	terms $r_{\iota\to\iota\to o}$

- Syntax: $s, t \quad ::= \quad P \mid \neg\, s \mid s \vee t \mid \Box_r\, s \mid \ldots$

- HOL

Syntax - formulas s Kripke Semantics	$\longrightarrow$	terms $s_{\iota\to o}$
- worlds w	$\longrightarrow$	terms w_ι
- accessibility relations r	$\longrightarrow$	terms $r_{\iota\to\iota\to o}$

- Syntax of embedded logic as abbreviations of HOL-terms

$$
\begin{aligned}
P &= P_{\iota\to o} \\
\neg &= \lambda S_{\iota\to o}. \lambda W_\iota. \neg(S\, W) \\
\vee &= \lambda S_{\iota\to o}. \lambda T_{\iota\to o}. \lambda W_\iota. (S\, W) \vee (T\, W) \\
\Box &= \lambda R_{\iota\to\iota\to o}. \lambda S_{\iota\to o}. \lambda W_\iota. \forall V_\iota. \neg(R\, W\, V) \vee (S\, V)
\end{aligned}
$$

[BenzmüllerPaulson, Log.J.IGPL, 2010], [BenzmüllerPaulson, Logica Universalis, 2012]

- Syntax: $s, t \quad ::= \quad P \mid \neg\, s \mid s \vee t \mid \Box_r\, s \mid \ldots$

- HOL

Syntax		
- formulas s	$\longrightarrow$	terms $s_{\iota\to o}$
Kripke Semantics		
- worlds w	$\longrightarrow$	terms w_ι
- accessibility relations r	$\longrightarrow$	terms $r_{\iota\to\iota\to o}$

- Syntax of embedded logic as abbreviations of HOL-terms

$$
\begin{aligned}
P &= P_{\iota\to o}\\
\neg &= \lambda S_{\iota\to o}.\, \lambda W_\iota.\, \neg(S\ W)\\
\vee &= \lambda S_{\iota\to o}.\, \lambda T_{\iota\to o}.\, \lambda W_\iota.\, (S\ W) \vee (T\ W)\\
\Box &= \lambda R_{\iota\to\iota\to o}.\, \lambda S_{\iota\to o}.\, \lambda W_\iota.\, \forall V_\iota.\, \neg(R\ W\ V) \vee (S\ V)\\
(\forall^p), \forall^\mu &= \lambda Q_{\mu\to(\iota\to o)}.\, \lambda W_\iota.\, \forall X_\mu.\, (Q\ X\ W)
\end{aligned}
$$

- Syntax: $s, t \quad ::= \quad P \mid \neg\, s \mid s \vee t \mid \Box_r\, s \mid \ldots$

- **HOL**

Syntax - formulas s	$\longrightarrow$	terms $s_{\iota\to o}$
Kripke Semantics - worlds w	$\longrightarrow$	terms w_ι
- accessibility relations r	$\longrightarrow$	terms $r_{\iota\to\iota\to o}$

- Syntax of embedded logic as abbreviations of HOL-terms

$$
\begin{array}{rcl}
P & = & P_{\iota\to o} \\
\neg & = & \lambda S_{\iota\to o}.\lambda W_\iota.\neg(S\,W) \\
\vee & = & \lambda S_{\iota\to o}.\lambda T_{\iota\to o}.\lambda W_\iota.(S\,W)\vee(T\,W) \\
\Box & = & \lambda R_{\iota\to\iota\to o}.\lambda S_{\iota\to o}.\lambda W_\iota.\forall V_\iota.\neg(R\,W\,V)\vee(S\,V) \\
(\forall^p), \forall^\mu & = & \lambda Q_{\mu\to(\iota\to o)}.\lambda W_\iota.\forall X_\mu.(Q\,X\,W) \\
\Rightarrow_f & = & \lambda S_{\iota\to o}.\lambda T_{\iota\to o}.\lambda W_\iota.\forall V_\iota.\neg(f\,W\,S\,V)\vee(T\,V)
\end{array}
$$

[BenzmüllerGenovese, NCMPL, 2011], [BenzmüllerGabbayGenoveseRispoli, Logica Universalis, 2012]

Embedding Meta-Level Notions

- Validity

$$\text{valid} \quad = \quad \lambda \varphi_{\iota \to o} . \forall W_{\iota} . \varphi\, W$$

Similar: Satisfiability, Countersatisfiability, Unsatisfiability

- Validity

$$\text{valid} \quad = \quad \lambda\varphi_{\iota\to o}\centerdot\forall W_\iota\centerdot\varphi\, W$$

Similar: Satisfiability, Countersatisfiability, Unsatisfiability

Soundness and Completeness Theorem

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

Consequence: Automation for free in HOL-ATPs!

Can Peter retire happy?

- Chris thinks that Peter can retire happy, if he knows that HOL-ATP is fostered by someone

$$\Box_{knowledgeChris}(\Box_{knowledgePeter}\exists X.fosters(X, holatp) \supset canRetireHappy(peter))$$

- Peter knows that Chris fosters HOL-ATP

$$\Box_{knowledgePeter} fosters(chris, holatp)$$

- Peter knows that Chad fosters HOL-ATP

$$\Box_{knowledgePeter} fosters(chad, holatp)$$

- Peter knows that other persons do foster HOL-ATP ...

$$\ldots$$

- Chris thinks that Peter can retire happy

$$\Box_{knowledgeChris} canRetireHappy(peter)$$

- Chris thinks that Peter can retire happy, if he knows that HOL-ATP is fostered by someone

$$\Box_{knowledgeChris}(\Box_{knowledgePeter}\exists X.fosters(X, holatp) \supset canRetireHappy(peter))$$

- Peter knows that Chris fosters HOL-ATP

$$\Box_{knowledgePeter}fosters(chris, holatp)$$

- Peter knows that Chad fosters HOL-ATP

$$\Box_{knowledgePeter}fosters(chad, holatp)$$

- Peter knows that other persons do foster HOL-ATP ...

...

- Chris thinks that Peter can retire happy

$$\Box_{knowledgeChris}canRetireHappy(peter)$$

Ax1 valid $\Box_{knowledgeChris}$ ($\Box_{knowledgePeter}\ \exists X.fosters(X, holatp) \supset canRetireHappy(peter))$

Ax2 valid $\Box_{knowledgePeter}\ fosters(chris, holatp)$

Ax3 valid $\Box_{knowledgePeter}\ fosters(chad, holatp)$

further axioms $\Box_{knowledgeChris}$, $\Box_{knowledgePeter}$ are S4 operators

Conj valid $\Box_{knowledgeChris}\ canRetireHappy(peter)$

Ax1 valid $\Box_{knowledgeChris}$ (
$\Box_{knowledgePeter}\ \exists X. fosters(X, holatp) \supset canRetireHappy(peter))$

Ax2 valid $\Box_{knowledgePeter}\ fosters(chris, holatp)$

Ax3 valid $\Box_{knowledgePeter}\ fosters(chad, holatp)$

further axioms $\Box_{knowledgeChris}$, $\Box_{knowledgePeter}$ are S4 operators

Conj valid $\Box_{knowledgeChris}\ canRetireHappy(peter)$

$\forall W_\iota .\ \Box_{knowledgeChris}\ canRetireHappy(peter)\ W$

expanded abbreviation

Ax1 valid $\Box_{knowledgeChris}$ (
$\Box_{knowledgePeter}\ \exists X.fosters(X, holatp) \supset canRetireHappy(peter))$

Ax2 valid $\Box_{knowledgePeter}\ fosters(chris, holatp)$

Ax3 valid $\Box_{knowledgePeter}\ fosters(chad, holatp)$

further axioms $\Box_{knowledgeChris}$, $\Box_{knowledgePeter}$ are S4 operators

Conj valid $\Box_{knowledgeChris}\ canRetireHappy(peter)$

$\forall W_\iota.\ \Box_{knowledgeChris}\ canRetireHappy(peter)\ W$

$\forall W_\iota.\ \forall V_\iota.\ \neg(knowledgeChris\ W\ V) \vee canRetireHappy(peter)\ W$

expanded abbreviation

Ax1 valid $\Box_{knowledgeChris}$ (
$\Box_{knowledgePeter}\ \exists X.fosters(X, holatp) \supset canRetireHappy(peter))$

Ax2 valid $\Box_{knowledgePeter}\ fosters(chris, holatp)$

Ax3 valid $\Box_{knowledgePeter}\ fosters(chad, holatp)$

further axioms $\Box_{knowledgeChris}$, $\Box_{knowledgePeter}$ are S4 operators

Conj valid $\Box_{knowledgeChris}\ canRetireHappy(peter)$

$\forall W_\iota.\ \Box_{knowledgeChris}\ canRetireHappy(peter)\ W$

$\forall W_\iota.\ \forall V_\iota.\ \neg(knowledgeChris\ W\ V) \vee canRetireHappy(peter)\ W$

$\forall W_\iota.\ \forall V_\iota.\ \neg(knowledgeChris\ W\ V) \vee (canRetireHappy\ peter\ W)$

expanded abbreviation

Translating Kripke style semantics to HOL

- Kripke style semantics

$$
\begin{array}{lll}
M, w \models P & \text{arbitrary} & \\
M, w \models \neg s & \text{iff} & \text{not } M, w \models s \\
M, w \models s \vee t & \text{iff} & M, w \models s \text{ or } M, w \models t
\end{array}
$$

- Semantic embedding in HOL

$$
\begin{array}{lll}
P & = & P_{\iota \to o} \\
\neg & = & \lambda S_{\iota \to o} . \lambda W_{\iota} . \neg (S\, W) \\
\vee & = & \lambda S_{\iota \to o} . \lambda T_{\iota \to o} . \lambda W_{\iota} . (S\, W) \vee (T\, W)
\end{array}
$$

Translating Kripke style semantics to HOL

- Kripke style semantics

$$
\begin{array}{llll}
M, w \models P & \text{arbitrary} & \\
M, w \models \neg s & \text{iff} & \text{not } M, w \models s \\
M, w \models s \vee t & \text{iff} & M, w \models s \text{ or } M, w \models t \\
M, w \models \Box_r\, s & \text{iff} & M, v \models s \text{ for all v such that } r(w, v)
\end{array}
$$

- Semantic embedding in HOL

$$
\begin{array}{lll}
P & = & P_{\iota\to o} \\
\neg & = & \lambda S_{\iota\to o}.\, \lambda W_{\iota}.\, \neg(S\, W) \\
\vee & = & \lambda S_{\iota\to o}.\, \lambda T_{\iota\to o}.\, \lambda W_{\iota}.\, (S\, W) \vee (T\, W) \\
\Box & = & \lambda R_{\iota\to\iota\to o}.\, \lambda S_{\iota\to o}.\, \lambda W_{\iota}.\, \forall V_{\iota}.\, \neg(R\, W\, V) \vee (S\, V)
\end{array}
$$

- Kripke style semantics

$M, w \models P$	arbitrary	
$M, w \models \neg s$	iff	not $M, w \models s$
$M, w \models s \vee t$	iff	$M, w \models s$ or $M, w \models t$
$M, w \models \Box_r\, s$	iff	$M, v \models s$ for all v such that $r(w, v)$

- Semantic embedding in HOL

$$
\begin{aligned}
P &= P_{\iota\to o} \\
\neg &= \lambda S_{\iota\to o}.\, \lambda W_\iota.\, \neg(S\, W) \\
\vee &= \lambda S_{\iota\to o}.\, \lambda T_{\iota\to o}.\, \lambda W_\iota.\, (S\, W) \vee (T\, W) \\
\Box &= \lambda R_{\iota\to\iota\to o}.\, \lambda S_{\iota\to o}.\, \lambda W_\iota.\, \forall V_\iota.\, \neg(R\, W\, V) \vee (S\, V)
\end{aligned}
$$

- Kripke style semantics

$$
\begin{array}{llll}
M, w \models P & \text{arbitrary} & & \\
M, w \models \neg s & \text{iff} & & \text{not } M, w \models s \\
M, w \models s \vee t & \text{iff} & & M, w \models s \text{ or } M, w \models t \\
M, w \models \Box_r\, s & \text{iff} & & M, v \models s \text{ for all v such that } r(w, v)
\end{array}
$$

- Semantic embedding in HOL

$$
\begin{array}{rcl}
P & = & P_{\iota\rightarrow o} \\
\neg & = & \lambda S_{\iota\rightarrow o}.\, \lambda W_{\iota}.\, \neg(S\; W) \\
\vee & = & \lambda S_{\iota\rightarrow o}.\, \lambda T_{\iota\rightarrow o}.\, \lambda W_{\iota}.\, (S\; W) \vee (T\; W) \\
\Box_r & = & \lambda S_{\iota\rightarrow o}.\, \lambda W_{\iota}.\, \forall V_{\iota}.\, \neg(r\; W\; V) \vee (S\; V)
\end{array}
$$

To model $\Box_r$ as T, S4 operator etc. add axioms like (*reflexive* r), etc.

- Kripke style semantics

higher-order selection function!

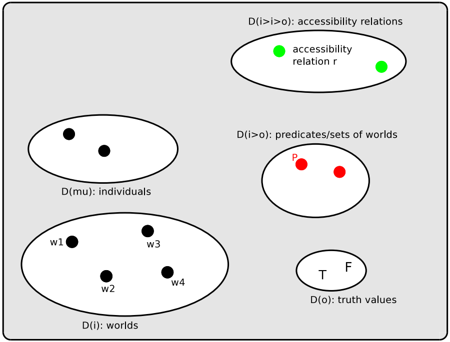
$M, w \models P$	arbitrary	
$M, w \models \neg s$	iff	not $M, w \models s$
$M, w \models s \vee t$	iff	$M, w \models s$ or $M, w \models t$
$M, w \models \Box_r\, s$	iff	$M, v \models s$ for all v such that $r(w, v)$
$M, w \models s \Rightarrow_f t$	iff	$M, v \models t$ for all $v \in f(w, [s])$
		with $[s] = \{u \mid M, u \models s\}$

- Semantic embedding in HOL

$$
\begin{aligned}
P &= P_{\iota\to o} \\
\neg &= \lambda S_{\iota\to o}.\lambda W_{\iota}.\neg(S\,W) \\
\vee &= \lambda S_{\iota\to o}.\lambda T_{\iota\to o}.\lambda W_{\iota}.(S\,W) \vee (T\,W) \\
\Box &= \lambda R_{\iota\to\iota\to o}.\lambda S_{\iota\to o}.\lambda W_{\iota}.\forall V_{\iota}.\neg(R\,W\,V) \vee (S\,V) \\
\Rightarrow_f &= \lambda S_{\iota\to o}.\lambda T_{\iota\to o}.\lambda W_{\iota}.\forall V_{\iota}.\neg(f\,W\,S\,V) \vee (T\,V)
\end{aligned}
$$

Add respective axioms for f

Constant Domain

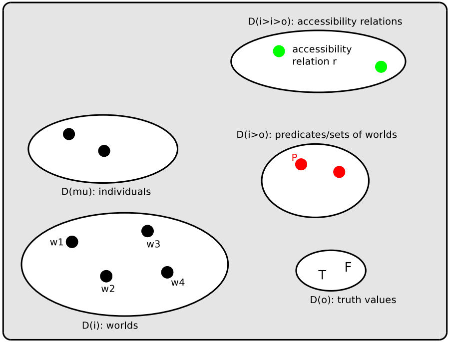


$$\Pi = \lambda Q . \lambda W_\iota . \forall X_\mu . (Q\, X\, W)$$

$$\forall Y . s = \Pi \lambda Y . s$$

Quantified Modal Logics: Varying and Cumulative Domain

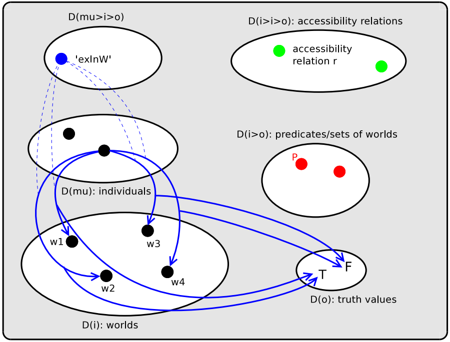
Constant Domain



$$\Pi = \lambda Q . \lambda W_{\iota} . \forall X_{\mu} . (Q\ X\ W)$$

$$\forall Y . s = \Pi \lambda Y . s$$

Varying and Cumulative Domain



$$\Pi_{var} = \lambda Q . \lambda W_{\iota} . \forall X_{\mu} . \neg(\text{exInW}\ X\ W) \vee (Q\ X\ W)$$

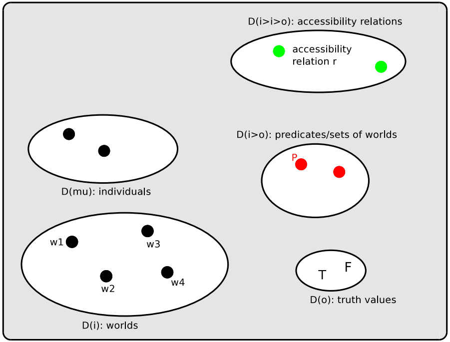
A: $\forall W_{\iota} . \exists X_{\mu} . (\text{exInW}\ X\ W)$

B(c): $\forall W_{\iota} . (\text{exInW}\ c\ W)$

B(f): $\forall W_{\iota} . (\text{exInW}\ t^{1}\ W) \wedge \ldots \wedge (\text{exInW}\ t^{n}\ W) \supset (\text{exInW}\ (f\ t^{1}, \ldots\ t^{n})\ W)$

[BenzmüllerOttenRaths, ECAI'2012]

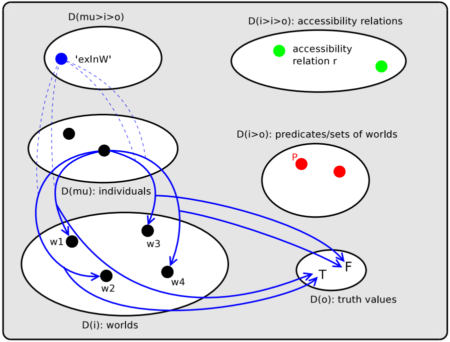
Constant Domain



$$\Pi = \lambda Q.\, \lambda W_\iota.\, \forall X_\mu.\, (Q\, X\, W)$$

$$\forall Y.\, s = \Pi \lambda Y.\, s$$

Varying and Cumulative Domain



$$\Pi_{var} = \lambda Q.\, \lambda W_\iota.\, \forall X_\mu.\, \neg(\text{exInW}\, X\, W) \vee (Q\, X\, W)$$

A: $\forall W_\iota.\, \exists X_\mu.\, (\text{exInW}\, X\, W)$

B(c): $\forall W_\iota.\, (\text{exInW}\, c\, W)$

B(f): $\forall W_\iota.\, (\text{exInW}\, t^1\, W) \wedge \ldots \wedge (\text{exInW}\, t^n\, W) \supset (\text{exInW}\, (f\, t^1, \ldots\, t^n)\, W)$

C: $\forall X_\mu, V_\iota, W_\iota.\, (\text{exInW}\, X\, V) \wedge (r\, V\, W) \supset (\text{exInW}\, X\, W)$

[BenzmüllerOttenRaths, ECAI'2012]

Natural Fragments of HOL: Quantified Conditional Logics

This work extends

[BenzmüllerGenoveseGabbayRispoli, AMAI, 2012 (arXiv:1106.3685v3)]

[BenzmüllerPaulson, Logica Universalis, 2012 (arXiv:0905.2435v1)]

Theory for (Reasoning with) Counterfactual Conditionals

If I had continued with competitive long-distance running in 1992,
I would have won the Olympic Games in 2000.

Problem: non-truth-functionality of counterfactual conditional statements

Solution (Stalnaker and Thomason)

- selection function semantics (a possible world semantics, extension of modal logics) [Stalnaker68]

$$\underbrace{\text{'If } A \text{ then } B\text{'}}_{(A \Rightarrow B)} \textit{ is true in world } w \quad \textit{iff} \quad B \textit{ is true for all } v \in f(w, A)$$

- idea: f selects worlds that are very *similar/close* to the actual world w
- many closely related theories: [Lewis73, Pollock76, Chellas75]

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi$$

Quantified Conditional Logic – Syntax

Propositional Variables (PV)   Individual Variables (IV)   Constants (Sym)

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Quantified Conditional Logic – Syntax

Propositional Variables (PV) Individual Variables (IV) Constants (Sym)

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Quantified Conditional Logic – Syntax

Propositional Variables (PV) Individual Variables (IV) Constants (Sym)

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Quantified Conditional Logic – Syntax

Propositional Variables (PV)   Individual Variables (IV)   Constants (Sym)

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Quantified Conditional Logic – Syntax

Propositional Variables (PV) Individual Variables (IV) Constants (Sym)

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Logical Connectives and Quantifiers (others may be defined as usual)

Quantified Conditional Logic – Syntax

Propositional Variables (PV)   Individual Variables (IV)   Constants (Sym)

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Conditional (modal) operator

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Interpretation

- is a structure $M = \langle S, f, D, Q, I \rangle$ with
  - S set of possible worlds
  - $f : S \times 2^S \mapsto 2^S$ is the selection function
  - D is a non-empty set of individuals (the first-order domain)
  - Q is a non-empty collection of subsets of S (the propositional domain)
  - I is a classical interpretation function where for each n-ary predicate symbol k, $I(k, w) \subseteq D^n$

Variable Assignment

- $g = \langle g^{iv}, g^{pv} \rangle$
  - $g^{iv} : IV \mapsto D$ maps individual variables to objects in D
  - $g^{pv} : PV \mapsto Q$ maps propositional variables to sets of worlds in Q

$$\varphi, \psi ::= P \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \Rightarrow \psi \mid \forall P.\varphi \mid \forall X.\varphi \mid k(X^1, \ldots, X^n)$$

Interpretation

- is a structure $M = \langle S, f, D, Q, I \rangle$ with
  - S set of possible worlds
  - $f : S \times 2^S \mapsto 2^S$ is the selection function
  - D is a non-empty set of individuals (the first-order domain)
  - Q is a non-empty collection of subsets of S (the propositional domain)
  - I is a classical interpretation function where for each n-ary predicate symbol k, $I(k, w) \subseteq D^n$

Variable Assignment

- $g = \langle g^{iv}, g^{pv} \rangle$
  - $g^{iv} : IV \mapsto D$ maps individual variables to objects in D
  - $g^{pv} : PV \mapsto Q$ maps propositional variables to sets of worlds in Q

Satisfiability $M, g, s \models \varphi$ defined as:

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X . \varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P . \varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Validity

- $M \models \varphi$ iff for all worlds s and assignments g holds $M, g, s \models \varphi$
- $\models \varphi$ iff φ is valid in every model M

Satisfiability $M, g, s \models \varphi$ defined as:

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X. \varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P. \varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Validity

- $M \models \varphi$ iff for all worlds s and assignments g holds $M, g, s \models \varphi$
- $\models \varphi$ iff φ is valid in every model M

Satisfiability $M, g, s \models \varphi$ defined as:

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X. \varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P. \varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Validity

- $M \models \varphi$ iff for all worlds s and assignments g holds $M, g, s \models \varphi$
- $\models \varphi$ iff φ is valid in every model M

Satisfiability $M, g, s \models \varphi$ defined as:

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X.\varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P.\varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Validity

- $M \models \varphi$ iff for all worlds s and assignments g holds $M, g, s \models \varphi$
- $\models \varphi$ iff φ is valid in every model M

Satisfiability $M, g, s \models \varphi$ defined as:

$$
\begin{array}{lll}
M, g, s \models P & \text{iff} & s \in g(P) \\
M, g, s \models k(X^1, \ldots, X^n) & \text{iff} & s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w) \\
M, g, s \models \neg\varphi & \text{iff} & \text{not } M, g, s \models \varphi \\
M, g, s \models \varphi \vee \psi & \text{iff} & M, g, s \models \varphi \text{ or } M, g, s \models \psi \\
M, g, s \models \varphi \Rightarrow \psi & \text{iff} & M, g, v \models \psi \text{ for all } v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]}) \\
M, g, s \models \forall X.\varphi & \text{iff} & M, [d/X]g, s \models \varphi \text{ for all } d \in D \\
M, g, s \models \forall P.\varphi & \text{iff} & M, [p/P]g, s \models \varphi \text{ for all } p \in Q
\end{array}
$$

Validity

- $M \models \varphi$ iff for all worlds s and assignments g holds $M, g, s \models \varphi$
- $\models \varphi$ iff φ is valid in every model M

Quantified Conditional Logic – Normality

Above semantics of $\Rightarrow$ enforces normality property:

> if φ and φ' are equivalent, then they index the same formulas wrt. $\Rightarrow$

Quantified Conditional Logic – Normality

Above semantics of $\Rightarrow$ enforces normality property:

if φ and φ' are equivalent, then they index the same formulas wrt. $\Rightarrow$

The axiomatic counterpart of the normality condition given by rule (RCEA)

$$\frac{\varphi \leftrightarrow \varphi'}{(\varphi \Rightarrow \psi) \leftrightarrow (\varphi' \Rightarrow \psi)} \ (RCEA)$$

Above semantics forces also the following rules to hold:

$$\frac{(\varphi_1 \wedge \ldots \wedge \varphi_n) \leftrightarrow \psi}{(\varphi_0 \Rightarrow \varphi_1 \wedge \ldots \wedge \varphi_0 \Rightarrow \varphi_n) \rightarrow (\varphi_0 \Rightarrow \psi)} \ (RCK) \qquad \frac{\varphi \leftrightarrow \varphi'}{(\psi \Rightarrow \varphi) \leftrightarrow (\psi \Rightarrow \varphi')} \ (RCEC)$$

Quantified Conditional Logic – Normality

Above semantics of $\Rightarrow$ enforces normality property:

if φ and φ' are equivalent, then they index the same formulas wrt. $\Rightarrow$

The axiomatic counterpart of the normality condition given by rule (RCEA)

$$\frac{\varphi \leftrightarrow \varphi'}{(\varphi \Rightarrow \psi) \leftrightarrow (\varphi' \Rightarrow \psi)}\ (RCEA)$$

Above semantics forces also the following rules to hold:

$$\frac{(\varphi_1 \wedge \ldots \wedge \varphi_n) \leftrightarrow \psi}{(\varphi_0 \Rightarrow \varphi_1 \wedge \ldots \wedge \varphi_0 \Rightarrow \varphi_n) \rightarrow (\varphi_0 \Rightarrow \psi)}\ (RCK) \qquad \frac{\varphi \leftrightarrow \varphi'}{(\psi \Rightarrow \varphi) \leftrightarrow (\psi \Rightarrow \varphi')}\ (RCEC)$$

Logic CK: minimal logic closed under rules RCEA, RCEC and RCK.
In what follows only logic CK and its extensions are considered.

Quantified Conditional Logics as Fragments of HOL

Kripke style semantics (higher-order) selection function!

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X. \varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P. \varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \rightarrow o$

$$P = \lambda W_\iota . (P_{\iota\rightarrow o}\, W) = P_{\iota\rightarrow o}$$

$$k(X^1, \ldots, X^n) = \lambda W_\iota . (k_{\mu^n \rightarrow (\iota\rightarrow o)}\, X^1_\mu \ldots X^n_\mu)\, W$$

$$\neg = \lambda \varphi_{\iota\rightarrow o} . \lambda W_\iota . \neg(\varphi\, W)$$

$$\vee = \lambda \varphi_{\iota\rightarrow o} . \lambda \psi_{\iota\rightarrow o} . \lambda W_\iota . (\varphi\, W) \vee (\psi\, W)$$

$$\Rightarrow = \lambda \varphi_{\iota\rightarrow o} . \lambda \psi_{\iota\rightarrow o} . \lambda W_\iota . \forall V_\iota . \neg(f\, W\, \varphi\, V) \vee (\psi\, V)$$

$$\forall^\mu(\Pi^\mu) = \lambda Q_{\mu\rightarrow(\iota\rightarrow o)} . \lambda W_\iota . \forall X_\mu . (Q\, X\, W)$$

$$\forall^p(\Pi^p) = \lambda Q_{(\iota\rightarrow o)\rightarrow(\iota\rightarrow o)} . \lambda W_\iota . \forall P_{\iota\rightarrow o} . (Q\, P\, W)$$

Kripke style semantics (higher-order) selection function!

$$
\begin{array}{lll}
M,g,s \models P & \text{iff} & s \in g(P) \\
M,g,s \models k(X^1,\ldots,X^n) & \text{iff} & s \in \langle g(X^1),\ldots,g(X^n)\rangle \in I(k,w) \\
M,g,s \models \neg\varphi & \text{iff} & \text{not } M,g,s \models \varphi \\
M,g,s \models \varphi \vee \psi & \text{iff} & M,g,s \models \varphi \text{ or } M,g,s \models \psi \\
M,g,s \models \varphi \Rightarrow \psi & \text{iff} & M,g,v \models \psi \text{ for all } v \in f(s,\overbrace{\{u \mid M,g,u \models \varphi\}}^{[\varphi]}) \\
M,g,s \models \forall X.\varphi & \text{iff} & M,[d/X]g,s \models \varphi \text{ for all } d \in D \\
M,g,s \models \forall P.\varphi & \text{iff} & M,[p/P]g,s \models \varphi \text{ for all } p \in Q
\end{array}
$$

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \rightarrow o$

$$
\begin{array}{rcl}
P & = & \lambda W_\iota.(P_{\iota\rightarrow o}\, W) = P_{\iota\rightarrow o} \\
k(X^1,\ldots,X^n) & = & \lambda W_\iota.(k_{\mu^n\rightarrow(\iota\rightarrow o)}\, X^1_\mu \ldots X^n_\mu)\, W \\
\neg & = & \lambda\varphi_{\iota\rightarrow o}.\lambda W_\iota.\neg(\varphi\, W) \\
\vee & = & \lambda\varphi_{\iota\rightarrow o}.\lambda\psi_{\iota\rightarrow o}.\lambda W_\iota.(\varphi\, W) \vee (\psi\, W) \\
\Rightarrow & = & \lambda\varphi_{\iota\rightarrow o}.\lambda\psi_{\iota\rightarrow o}.\lambda W_\iota.\forall V_\iota.\neg(f\, W\, \varphi\, V) \vee (\psi\, V) \\
\forall^\mu(\Pi^\mu) & = & \lambda Q_{\mu\rightarrow(\iota\rightarrow o)}.\lambda W_\iota.\forall X_\mu.(Q\, X\, W) \\
\forall^p(\Pi^p) & = & \lambda Q_{(\iota\rightarrow o)\rightarrow(\iota\rightarrow o)}.\lambda W_\iota.\forall P_{\iota\rightarrow o}.(Q\, P\, W)
\end{array}
$$

Kripke style semantics (higher-order) selection function!

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X.\varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P.\varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \rightarrow o$

$$P = \lambda W_\iota.(P_{\iota\rightarrow o}\, W) = P_{\iota\rightarrow o}$$
$$k(X^1, \ldots, X^n) = \lambda W_\iota.(k_{\mu^n\rightarrow(\iota\rightarrow o)}\, X^1_\mu \ldots X^n_\mu)\, W$$
$$\neg = \lambda\varphi_{\iota\rightarrow o}.\lambda W_\iota.\neg(\varphi\, W)$$
$$\vee = \lambda\varphi_{\iota\rightarrow o}.\lambda\psi_{\iota\rightarrow o}.\lambda W_\iota.(\varphi\, W) \vee (\psi\, W)$$
$$\Rightarrow = \lambda\varphi_{\iota\rightarrow o}.\lambda\psi_{\iota\rightarrow o}.\lambda W_\iota.\forall V_\iota.\neg(f\, W\, \varphi\, V) \vee (\psi\, V)$$
$$\forall^\mu(\Pi^\mu) = \lambda Q_{\mu\rightarrow(\iota\rightarrow o)}.\lambda W_\iota.\forall X_\mu.(Q\, X\, W)$$
$$\forall^p(\Pi^p) = \lambda Q_{(\iota\rightarrow o)\rightarrow(\iota\rightarrow o)}.\lambda W_\iota.\forall P_{\iota\rightarrow o}.(Q\, P\, W)$$

Quantified Conditional Logics as Fragments of HOL

Kripke style semantics (higher-order) selection function!

$M, g, s \models P$	iff	$s \in g(P)$
$M, g, s \models k(X^1, \ldots, X^n)$	iff	$s \in \langle g(X^1), \ldots, g(X^n)\rangle \in I(k, w)$
$M, g, s \models \neg\varphi$	iff	not $M, g, s \models \varphi$
$M, g, s \models \varphi \vee \psi$	iff	$M, g, s \models \varphi$ or $M, g, s \models \psi$
$M, g, s \models \varphi \Rightarrow \psi$	iff	$M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$
$M, g, s \models \forall X. \varphi$	iff	$M, [d/X]g, s \models \varphi$ for all $d \in D$
$M, g, s \models \forall P. \varphi$	iff	$M, [p/P]g, s \models \varphi$ for all $p \in Q$

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \rightarrow o$

$$
\begin{aligned}
P &= \lambda W_\iota. (P_{\iota\rightarrow o}\, W) = P_{\iota\rightarrow o} \\
k(X^1, \ldots, X^n) &= \lambda W_\iota. (k_{\mu^n\rightarrow(\iota\rightarrow o)}\, X^1_\mu \ldots X^n_\mu)\, W \\
\neg &= \lambda \varphi_{\iota\rightarrow o}. \lambda W_\iota. \neg(\varphi\, W) \\
\vee &= \lambda \varphi_{\iota\rightarrow o}. \lambda \psi_{\iota\rightarrow o}. \lambda W_\iota. (\varphi\, W) \vee (\psi\, W) \\
\Rightarrow &= \lambda \varphi_{\iota\rightarrow o}. \lambda \psi_{\iota\rightarrow o}. \lambda W_\iota. \forall V_\iota. \neg(f\, W\, \varphi\, V) \vee (\psi\, V) \\
\forall^\mu(\Pi^\mu) &= \lambda Q_{\mu\rightarrow(\iota\rightarrow o)}. \lambda W_\iota. \forall X_\mu. (Q\, X\, W) \\
\forall^p(\Pi^p) &= \lambda Q_{(\iota\rightarrow o)\rightarrow(\iota\rightarrow o)}. \lambda W_\iota. \forall P_{\iota\rightarrow o}. (Q\, P\, W)
\end{aligned}
$$

Quantified Conditional Logics as Fragments of HOL

Kripke style semantics (higher-order) selection function!

$M, g, s \models P$ iff $s \in g(P)$

$M, g, s \models k(X^1, \ldots, X^n)$ iff $s \in \langle g(X^1), \ldots, g(X^n) \rangle \in I(k, w)$

$M, g, s \models \neg\varphi$ iff not $M, g, s \models \varphi$

$M, g, s \models \varphi \vee \psi$ iff $M, g, s \models \varphi$ or $M, g, s \models \psi$

$M, g, s \models \varphi \Rightarrow \psi$ iff $M, g, v \models \psi$ for all $v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]})$

$M, g, s \models \forall X . \varphi$ iff $M, [d/X]g, s \models \varphi$ for all $d \in D$

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

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \to o$

$$P = \lambda W_\iota . (P_{\iota\to o}\, W) = P_{\iota\to o}$$

$$k(X^1, \ldots, X^n) = \lambda W_\iota . (k_{\mu^n \to (\iota\to o)}\, X^1_\mu \ldots X^n_\mu)\, W$$

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

$$\vee = \lambda \varphi_{\iota\to o} . \lambda \psi_{\iota\to o} . \lambda W_\iota . (\varphi\, W) \vee (\psi\, W)$$

$$\Rightarrow = \lambda \varphi_{\iota\to o} . \lambda \psi_{\iota\to o} . \lambda W_\iota . \forall V_\iota . \neg(f\, W\, \varphi\, V) \vee (\psi\, V)$$

$$\forall^\mu(\Pi^\mu) = \lambda Q_{\mu\to(\iota\to o)} . \lambda W_\iota . \forall X_\mu . (Q\, X\, W)$$

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

Kripke style semantics (higher-order) selection function!

$$
\begin{array}{lll}
M,g,s \models P & \text{iff} & s \in g(P) \\
M,g,s \models k(X^1,\ldots,X^n) & \text{iff} & s \in \langle g(X^1),\ldots,g(X^n)\rangle \in I(k,w) \\
M,g,s \models \neg\varphi & \text{iff} & \text{not } M,g,s \models \varphi \\
M,g,s \models \varphi \vee \psi & \text{iff} & M,g,s \models \varphi \text{ or } M,g,s \models \psi \\
M,g,s \models \varphi \Rightarrow \psi & \text{iff} & M,g,v \models \psi \text{ for all } v \in f(s,\overbrace{\{u \mid M,g,u \models \varphi\}}^{[\varphi]}) \\
M,g,s \models \forall X_\centerdot \varphi & \text{iff} & M,[d/X]g,s \models \varphi \text{ for all } d \in D \\
M,g,s \models \forall P_\centerdot \varphi & \text{iff} & M,[p/P]g,s \models \varphi \text{ for all } p \in Q
\end{array}
$$

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \to o$

$$
\begin{array}{rcl}
P & = & \lambda W_{\iota\centerdot}(P_{\iota\to o}\, W) = P_{\iota\to o} \\
k(X^1,\ldots,X^n) & = & \lambda W_{\iota\centerdot}(k_{\mu^n\to(\iota\to o)}\, X^1_\mu \ldots X^n_\mu)\, W \\
\neg & = & \lambda \varphi_{\iota\to o\centerdot}\lambda W_{\iota\centerdot}\neg(\varphi\, W) \\
\vee & = & \lambda \varphi_{\iota\to o\centerdot}\lambda \psi_{\iota\to o\centerdot}\lambda W_{\iota\centerdot}(\varphi\, W) \vee (\psi\, W) \\
\Rightarrow & = & \lambda \varphi_{\iota\to o\centerdot}\lambda \psi_{\iota\to o\centerdot}\lambda W_{\iota\centerdot}\forall V_{\iota\centerdot}\neg(f\, W\, \varphi\, V) \vee (\psi\, V) \\
\forall^\mu(\Pi^\mu) & = & \lambda Q_{\mu\to(\iota\to o)\centerdot}\lambda W_{\iota\centerdot}\forall X_{\mu\centerdot}(Q\, X\, W) \\
\forall^p(\Pi^p) & = & \lambda Q_{(\iota\to o)\to(\iota\to o)\centerdot}\lambda W_{\iota\centerdot}\forall P_{\iota\to o\centerdot}(Q\, P\, W)
\end{array}
$$

Kripke style semantics (higher-order) selection function!

$$
\begin{array}{lll}
M, g, s \models P & \text{iff} & s \in g(P) \\
M, g, s \models k(X^1, \ldots, X^n) & \text{iff} & s \in \langle g(X^1), \ldots, g(X^n)\rangle \in I(k, w) \\
M, g, s \models \neg\varphi & \text{iff} & \text{not } M, g, s \models \varphi \\
M, g, s \models \varphi \vee \psi & \text{iff} & M, g, s \models \varphi \text{ or } M, g, s \models \psi \\
M, g, s \models \varphi \Rightarrow \psi & \text{iff} & M, g, v \models \psi \text{ for all } v \in f(s, \overbrace{\{u \mid M, g, u \models \varphi\}}^{[\varphi]}) \\
M, g, s \models \forall X \mathbin{.} \varphi & \text{iff} & M, [d/X]g, s \models \varphi \text{ for all } d \in D \\
M, g, s \models \forall P \mathbin{.} \varphi & \text{iff} & M, [p/P]g, s \models \varphi \text{ for all } p \in Q
\end{array}
$$

Semantic embedding: ML $\longrightarrow$ HOL terms of type $\iota \to o$

$$
\begin{array}{rcl}
P & = & \lambda W_\iota \mathbin{.} (P_{\iota\to o}\, W) = P_{\iota\to o} \\
k(X^1, \ldots, X^n) & = & \lambda W_\iota \mathbin{.} (k_{\mu^n\to(\iota\to o)}\, X^1_\mu \ldots X^n_\mu)\, W \\
\neg & = & \lambda\varphi_{\iota\to o} \mathbin{.} \lambda W_\iota \mathbin{.} \neg(\varphi\, W) \\
\vee & = & \lambda\varphi_{\iota\to o} \mathbin{.} \lambda\psi_{\iota\to o} \mathbin{.} \lambda W_\iota \mathbin{.} (\varphi\, W) \vee (\psi\, W) \\
\Rightarrow & = & \lambda\varphi_{\iota\to o} \mathbin{.} \lambda\psi_{\iota\to o} \mathbin{.} \lambda W_\iota \mathbin{.} \forall V_\iota \mathbin{.} \neg(f\, W\, \varphi\, V) \vee (\psi\, V) \\
\forall^\mu(\Pi^\mu) & = & \lambda Q_{\mu\to(\iota\to o)} \mathbin{.} \lambda W_\iota \mathbin{.} \forall X_\mu \mathbin{.} (Q\, X\, W) \\
\forall^p(\Pi^p) & = & \lambda Q_{(\iota\to o)\to(\iota\to o)} \mathbin{.} \lambda W_\iota \mathbin{.} \forall P_{\iota\to o} \mathbin{.} (Q\, P\, W)
\end{array}
$$

Validity defined as before

$$\text{valid} \quad = \quad \lambda \varphi_{\iota \to o} . \forall W_\iota . \varphi\, W$$

Soundness and Completeness Theorem

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

Proof Idea:

Explicate and analyze the relation between selection functions semantics and corresponding Henkin models; see paper for details.

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

Automation Quantified Conditional Logics in HOL ATPs

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

valid $(\forall X.\varphi \Rightarrow (\psi\, X)) \rightarrow (\varphi \Rightarrow \forall X.(\psi\, X))$

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

$$\text{valid}\, \neg(\Pi^{\mu}\lambda X.\,\varphi \Rightarrow (\psi\, X)) \vee (\varphi \Rightarrow \Pi^{\mu}\lambda X.\,(\psi\, X))$$

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

$$\forall W_\iota\mathbf{.}(\neg(\Pi^\mu \lambda X\mathbf{.}\varphi \Rightarrow (\psi\, X)) \vee (\varphi \Rightarrow \Pi^\mu \lambda X\mathbf{.}(\psi\, X)))W$$

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

$$\forall W_\iota\textbf{.}(\lambda V_\iota\textbf{.}((\neg(\Pi^\mu \lambda X\textbf{.}\varphi \Rightarrow (\psi\, X))\, V) \vee ((\varphi \Rightarrow \Pi^\mu \lambda X\textbf{.}(\psi\, X))\, V)))\, W$$

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

$$\forall W_\iota.(\neg(\Pi^\mu \lambda X.\varphi \Rightarrow (\psi\, X))\, W \vee (\varphi \Rightarrow \Pi^\mu \lambda X.(\psi\, X))\, W)$$

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

$$\ldots$$

by LEO-II or Satallax in 0.01 seconds

Proof of the Barcan formula (confirms constant domain)

$$(\forall X.\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall X.\psi(X))$$

$$\ldots$$

by LEO-II or Satallax in 0.01 seconds

Proof of the Converse Barcan formula

$$(\varphi \Rightarrow \forall X.\psi(x)) \rightarrow (\forall X.\varphi \Rightarrow \psi(x))$$

by LEO-II or Satallax in 0.01 seconds

Natural Fragments of HOL

Soundness and Completeness Results

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

- Propositional Multimodal Logics [BenzmüllerPaulson, Log.J.IGPL, 2010]
- Quantified Multimodal Logics [BenzmüllerPaulson, Logica Universalis, 2012]
- Propositional Conditional Logics [BenzmüllerEtAl., AMAI, 2012]
- Quantified Conditional Logics [BenzmüllerGenovese, NCMPL, 2011]
- Intuitionistic Logics: [BenzmüllerPaulson, Log.J.IGPL, 2010]
- Access Control Logics: [Benzmüller, IFIP SEC, 2009]

- Combinations of Logics: [Benzmüller, AMAI, 2011]

Why not throwing things together?

Terms: $m ::= c \mid X \mid (f\ m^1 \dots m^n)$

Formulas:

$$s, t ::= P \mid (k\ m^1 \dots m^n) \mid \neg\, s \mid s \vee t \mid \Box_r\, s \mid s \Rightarrow_f\ t \mid \forall X.s \mid \forall_{var} X.s \mid \forall P.s$$

Embedding in HOL:

$$c = c_\mu \qquad X = X_\mu \qquad f = f_{\mu^n \to \mu}$$

$$P = P_{\iota\to o} \qquad k = k_{\mu^n \to \iota \to o}$$

$$r = k_{\iota\to\iota\to o} \quad (+axioms\ for\ r) \qquad f = f_{\iota\to\iota\to o} \quad (+axioms\ for\ f)$$

$$\neg = \lambda S_{\iota\to o}.\lambda W_\iota.\neg(S\ W)$$

$$\vee = \lambda S_{\iota\to o}.\lambda T_{\iota\to o}.\lambda W_\iota.(S\ W) \vee (T\ W)$$

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

$$\Rightarrow = \lambda F_{\iota\to(\iota\to o)\iota\to o}.\lambda S_{\iota\to o}.\lambda T_{\iota\to o}.\lambda W_\iota.\forall V_\iota.\neg(F\ W\ S\ V) \vee (T\ V)$$

$$\Pi = \lambda Q_{\mu\to(\iota\to o)}.\lambda W_\iota.\forall X_\mu.(Q\ X\ W)$$

$$\Pi^p = \lambda Q_{(\iota\to o)\to(\iota\to o)}.\lambda W_\iota.\forall P_{\iota\to o}.(Q\ P\ W)$$

$$\Pi_{var/cumul} = \lambda Q.\lambda W_\iota.\forall X_\mu.\neg(\mathrm{exInW}\ X\ W) \vee (Q\ X\ W)$$

$\dots$

further non-classical connectives, quantification over higher types, predicate abstraction, definite description ...

Reasoning *about* Logics (and their Combinations)

[Benzmüller, Festschrift Walther, 2010]

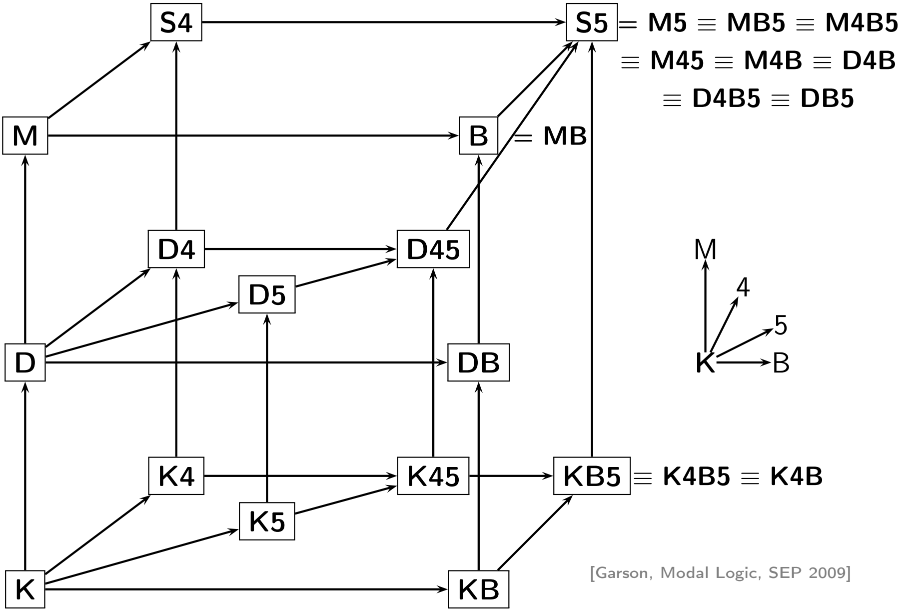
[Benzmüller, AMAI, 2012]

- Correspondences between axioms and semantic properties
  valid $\forall\phi.\ \Box_r\,\phi \supset \Box_r\,\Box_r\,\phi$
  $\Leftrightarrow$ (transitive r)
- Dependence/independence of axioms
  base modal logic K $\not\models$ axiom 4?
- Inclusion/non-inclusion relations between logics
  Is logic K45 (K+M+5) included in logic S4 (K+M+4)?
- (Relative) Consistency of logics and logic combinations
  Is logic S4 (K+M+4) consistent?

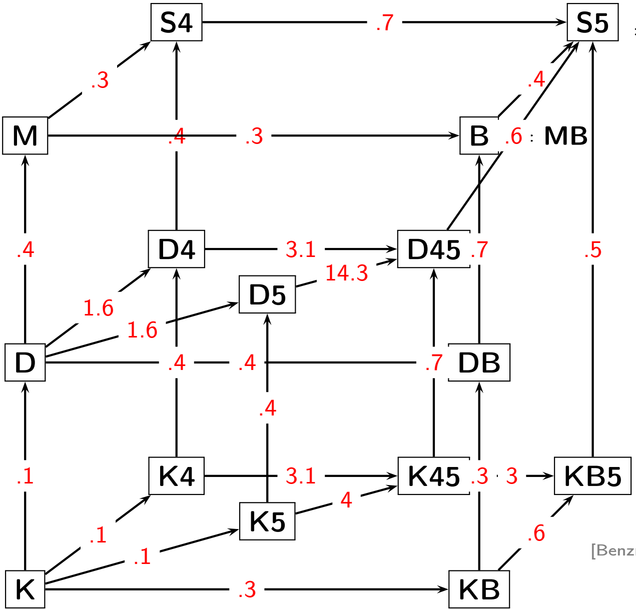
Experiments:

- Modal Logics [Benzmüller, Festschrift Walther, 2010]
- Conditional Logics [Benzmüller, AMAI, 2012]

Automating Meta-Properties of Logics in HOL



[Garson, Modal Logic, SEP 2009]



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

fastest results by:

LEO-II
Satallax
TPS
IsabelleP
IsabelleN
IsabelleM

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

[Benzmüller, Festschrift Walther, 2010]

Semantic Conditions for Conditional Logic Axioms

			TPS
ID	Axiom	$A \Rightarrow_f A$	
	Condition	$f(w,[A]) \subseteq [A]$	✓
MP	Axiom	$(A \Rightarrow_f B) \supset (A \supset B)$	
	Condition	$[A] \subseteq f(w,[A])$	✓
CS	Axiom	$(A \wedge B) \supset (A \Rightarrow_f B)$	
	Condition	$w \in [A] \supset f(w,[A]) \subseteq \{w\}$	✓
CEM	Axiom	$(A \Rightarrow_f B) \vee (A \Rightarrow_f \neg B)$	
	Condition	$\|f(w,[A])\| \leq 1$	✓
AC	Axiom	$(A \Rightarrow_f B) \wedge (A \Rightarrow_f C) \supset (A \wedge C \Rightarrow_f B)$	
	Condition	$f(w,[A]) \subseteq [B] \supset f(w,[A \wedge B]) \subseteq f(w,[A])$	✓
RT	Axiom	$(A \wedge B \Rightarrow_f C) \supset ((A \Rightarrow_f B) \supset (A \Rightarrow_f C))$	
	Condition	$f(w,[A]) \subseteq [B] \supset f(w,[A]) \subseteq f(w,[A \wedge B])$	✓
CV	Axiom	$(A \Rightarrow_f B) \wedge \neg(A \Rightarrow_f \neg C) \supset (A \wedge C \Rightarrow_f B)$	
	Condition	$(f(w,[A]) \subseteq [B]$ and $f(w,[A]) \cap [C] \neq \emptyset) \supset f(w,[A \wedge C]) \subseteq [B]$	✓
CA	Axiom	$(A \Rightarrow_f B) \wedge (C \Rightarrow_f B) \supset (A \vee C \Rightarrow_f B)$	
	Condition	$f(w,[A \vee B]) \subseteq f(w,[A]) \cup f(w,[B])$	✓

[BenzmüllerEtAl., AMAI, 2012]

Proofs and Countermodels at Meta-Level

The correct interpretation of the proof task for MP is

$$[\forall A, B.(A \Rightarrow_f B) \supset (A \supset B)] \leftrightarrow [\forall A, W.A \subseteq (f\ W\ A)]$$

versus (incorrect statement for MP)

$$\forall A, B.[((A \Rightarrow_f B) \supset (A \supset B)) \leftrightarrow \forall W.A \subseteq (f\ W\ A)]$$

The former is provable.

The latter is countersatisfiable; the countermodel reported by Nitpick is:

choose $D_i = \{i1\}$, $A = \{i1\}$, $B = \{i1\}$, $W = i1$,
and

$$f = \left\{ \; i1 \quad \longrightarrow \left\{ \begin{array}{ll} \emptyset & \longrightarrow \emptyset \\ \{i1\} & \longrightarrow \emptyset \end{array} \right. \right.$$

Evaluation of HOL-ATPs for First-order Monomodal Logics

[BenzmüllerOttenRaths, ECAI'2012]

QMLTP project: HOL-ATPs perform well for FML

The QMLTP project: see http://www.iltp.de/qmltp/

- Jens Otten and Thomas Raths, University of Potsdam
- infrastructure and benchmark library for testing and evaluating ATP systems for first-order modal logic
- collaborators: myself, Geoff Sutcliffe's TPTP project
- standardized extended TPTP syntax (called 'fml')
- 600 problems in 11 problem domains
- 20 problems in first-order multimodal logic

See our paper at ECAI 2012:

Theory & implementation of new provers for FML:

- embedding into higher-order logic (LEO-II & Satallax)
- a connection calculus based prover (MleanCoP)
- a sequent calculus based prover (MleanSeP)
- a tableau based prover (MleanTAP)
- an instantiation based prover (f2p-MSPASS)

Moreover, we present

- a first comparative prover evaluation
- exploiting the new QMLTP library for FML

Experiment: $\underbrace{\textbf{580 problems} \times \textbf{5 logics} \times \textbf{3 domain conditions}}_{\textbf{8700 problems}} \times$ **6 provers** $\times$ **600s tmo**

Experiment: **580 problems** × **5 logics** × **3 domain conditions** × **6 provers** × **600s tmo**

Logic/ Domain	ATP system f2p-MSPASS v3.0	MleanSeP v1.2	LEO-II v1.4.2	Satallax v2.2	MleanTAP v1.3	MleanCoP v1.2
K/varying	-	-	72	104	-	-
K/cumul.	70	121	89	122	-	-
K/constant	67	124	120	146	-	-
D/varying	-	-	81	113	100	179
D/cumul.	79	130	100	133	120	200
D/constant	76	134	135	160	135	217
T/varying	-	-	120	170	138	224
T/cumul.	105	163	139	192	160	249
T/constant	95	166	173	213	175	269
S4/varying	-	-	140	207	169	274
S4/cumul.	121	197	166	238	205	338
S4/constant	111	197	200	261	220	352
S5/varying	-	-	169	248	219	359
S5/cumul.	140	-	215	297	272	438
S5/constant	131	-	237	305	272	438

Strongest Prover!

Experiment: **580 problems** × **5 logics** × **3 domain conditions** × **6 provers** × **600s tmo**

Logic/ Domain	ATP system f2p-MSPASS v3.0	MleanSeP v1.2	LEO-II v1.4.2	Satallax v2.2	MleanTAP v1.3	MleanCoP v1.2
K/varying	-	-	72	104	-	-
K/cumul.	70	121	89	122	-	-
K/constant	67	124	120	146	-	-
D/varying	-	-	128 81	113	100	179
D/cumul.	79	130	144 100	133	120	200
D/constant	76	134	167 135	160	135	217
T/varying	-	-	170 120	170	138	224
T/cumul.	105	163	190 139	192	160	249
T/constant	95	166	217 173	213	175	269
S4/varying	-	-	140	207	169	274
S4/cumul.	121	197	218 166	238	205	338
S4/constant	111	197	244 200	261	220	352
S5/varying	-	-	169	248	219	359
S5/cumul.	140	-	215	297	272	438
S5/constant	131	-	237	305	272	438

Second best provers; best coverage; strong recent improvement ($\geq$ 25%)

Experiment: **580 problems × 5 logics × 3 domain conditions × 6 provers × 600s tmo**

Logic/ Domain	ATP system					
	f2p-MSPASS v3.0	MleanSeP v1.2	LEO-II v1.4.2	Satallax v2.2	MleanTAP v1.3	MleanCoP v1.2
K/varying	-	-	72	104	-	-
K/cumul.	70	121	89	122	-	-
K/constant	67	124	120	146	-	-
D/varying	-	-	128 81	113	100	179
D/cumul.	79	130	144 100	133	120	200
D/constant	76	134	167 135	160	135	217
T/varying	-	-	170 120	170	138	224
T/cumul.	105	163	190 139	192	160	249
T/constant	95	166	217 173	213	175	269
S4/varying	-	-	140	207	169	274
S4/cumul.	121	197	218 166	238	205	338
S4/constant	111	197	244 200	261	220	352
S5/varying	-	-	169	248	219	359
S5/cumul.	140	-	215	297	272	438
S5/constant	131	-	237	305	272	438

Results for 20 multimodal logic problems: LEO-II 15, Satallax 14

Demo

Analyzing example formula

$$(\Diamond(\exists X.pX) \wedge \Box\forall Y.\Diamond pY \supset qY) \supset \Diamond\exists Z.qZ$$

with HOL-ATPs:

	constant	varying	cumulative
K	CSA	CSA	???
D	CSA	CSA	???
T	THM	THM	THM
S4	THM	THM	THM
S5	THM	THM	THM

CSA means Countersatisfiable, THM means Theorem

Terminal — bash — 98×56

```
z8b8b:2012-CMU-Andrews christophbenzmueller$ more demo2.fml

qmf(con,conjecture,
    ( ( (#dia:  ? [X] : p(X))
        &
        (#box: ! [Y]: ((#dia: p(Y)) => q(Y))) )
      => #dia: ? [X] : q(X) )).
z8b8b:2012-CMU-Andrews christophbenzmueller$
z8b8b:2012-CMU-Andrews christophbenzmueller$
z8b8b:2012-CMU-Andrews christophbenzmueller$ ./universal-reasoner demo2.fml s4 vary
--- Running version 0.1 of the HOL-ATP based universal logic engine ---

INPUT: fml MODALLOGIC: s4 DOMAIN: vary

Converting from fml to thf (thanks to Thomas Raths)

Asking various HOL-ATPs in Miami remotely (thanks to Geoff Sutcliffe)

 TPS---3.110228S1a (20 sec timeout)
   RESULT: SOT_JG2Nm_ - TPS---3.110228S1a says Theorem - CPU = 5.62 WC = 7.50 Mode = MODE-X5202
 LEO-II---1.3.1 (20 sec timeout)
   RESULT: SOT_HyrsXa - LEO-II---1.3.1 says Theorem - CPU = 0.04 WC = 0.12
 Satallax---2.2 (20 sec timeout)
   RESULT: SOT_6YiBu4 - Satallax---2.2 says Theorem - CPU = 0.04 WC = 0.09
 Isabelle---2011 (20 sec timeout)
   RESULT: SOT_AiyQ6a - Isabelle---2011 says Theorem - CPU = 3.87 WC = 3.93 SolvedBy = smt
 Refute---2011 (20 sec timeout)
   RESULT: SOT_kBCM07 - Refute---2011 says Timeout - CPU = 21.75 WC = 22.21
 Nitpick---2011 (20 sec timeout)
   RESULT: SOT_vfrCvq - Nitpick---2011 says Timeout - CPU = 20.58 WC = 22.20

---
z8b8b:2012-CMU-Andrews christophbenzmueller$ ./universal-reasoner demo2.fml k const
--- Running version 0.1 of the HOL-ATP based universal logic engine ---

INPUT: fml MODALLOGIC: k DOMAIN: const

Converting from fml to thf (thanks to Thomas Raths)

Asking various HOL-ATPs in Miami remotely (thanks to Geoff Sutcliffe)

 TPS---3.110228S1a (20 sec timeout)
   RESULT: SOT_18LNWW - TPS---3.110228S1a says Unknown - CPU = 12.70 WC = 13.06
 LEO-II---1.3.1 (20 sec timeout)
   RESULT: SOT_G7CJ5d - LEO-II---1.3.1 says Unknown - CPU = 4.95 WC = 5.03
 Satallax---2.2 (20 sec timeout)
   RESULT: SOT_3cM_9y - Satallax---2.2 says CounterSatisfiable - CPU = 0.00 WC = 0.04
 Isabelle---2011 (20 sec timeout)
   RESULT: SOT_rR3DwX - Isabelle---2011 says Unknown - CPU = 17.87 WC = 17.76
 Refute---2011 (20 sec timeout)
   RESULT: SOT_Bl1n1S - Refute---2011 says CounterSatisfiable - CPU = 3.57 WC = 3.37
 Nitpick---2011 (20 sec timeout)
   RESULT: SOT_C41Wci - Nitpick---2011 says CounterSatisfiable - CPU = 4.76 WC = 4.19

---
```

Core Questions:

1. Classical Higher-order Logic (HOL) as Universal Logic?
2. HOL Provers & Model Finders as Generic Reasoning Tools?
3. Combinations with Specialist Reasoners (if available)?

- (1)&(2) are interesting and relevant: evidence given in talk!?
- (3) not further discussed: ongoing and future work

Discussion

- Practical strength of approach?
  Collaboration with QMLTP project in Potsdam
  http://www.cs.uni-potsdam.de/ti/iltp/qmltp/ [BenzmüllerEtAl, ECAI, 2012]
- Scalability to large knowledge bases? relevance filtering
- How to deal with impredicativity? try to avoid