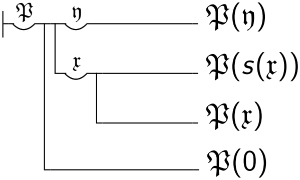
HOL based Universal Reasoning

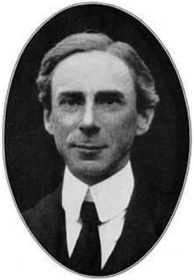
Christoph Benzmüller

Freie Universität Berlin

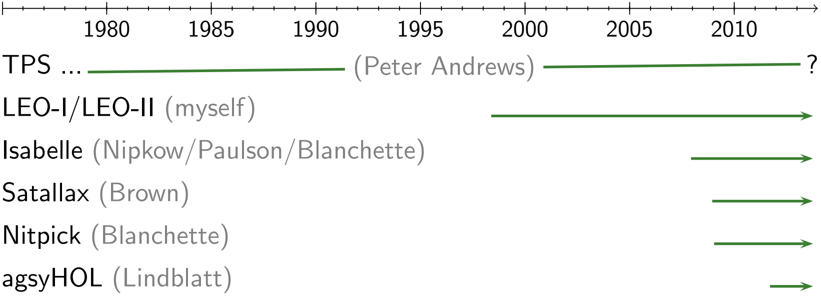
UNILOG-2013, Rio de Janeiro, Brasil, April 2013

HOL: Church's STT with Henkin Semantics





Automated Reasoners for HOL



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

encoding in HOL: $\mathrm{valid}\,(\Diamond\exists x P f x \wedge \Box\forall y(\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond\exists z Q z$

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

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: 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
```

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: 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

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: 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
```

Short Demonstration of HOL-P

FO Modal Logic example: $(\Diamond \exists x P f x \wedge \Box \forall y (\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond \exists z Q z$

encoding in HOL: $\text{valid}\ (\Diamond \exists x P f x \wedge \Box \forall y (\Diamond P y \Rightarrow Q y)) \Rightarrow \Diamond \exists z Q z$

... 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 $\alpha ::= \iota \mid o \mid \alpha_1 \rightarrow \alpha_2$

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

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

Individuals

Booleans (True and False)

Functions/Predicates

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

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

Possible worlds

Individuals

Booleans (True and False)

Functions/Predicates

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

HOL

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

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

HOL

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

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

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

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

Freie Universität Berlin

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 (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

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

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:

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

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

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

Freie Universität Berlin

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:

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

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

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:

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

Idea: Lifting of modal formulas to predicates on worlds

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

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

...

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

Modal logic axioms — Semantical Condition

valid $\forall^p \varphi(\Box \varphi \supset \Diamond \varphi)$ — $\forall x \exists y (rxy)$

Bridge rules — Semantical Condition

valid $\forall^p \varphi(\Box_r \varphi \supset \Box_s \varphi)$ — $\forall x \forall y (rxy \supset sxy)$

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

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 | Semantical Condition

valid $\forall^p \varphi (\Box \varphi \supset \Diamond \varphi)$ | $\forall x \exists y (rxy)$

Bridge rules | Semantical Condition

valid $\forall^p \varphi (\Box_r \varphi \supset \Box_s \varphi)$ | $\forall x \forall y (rxy \supset sxy)$

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

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

. . .

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

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

Embedding in HOL

. . .

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

Modal logic axioms | Semantical Condition

valid $\forall^p \varphi(\Box\varphi \supset \Diamond\varphi)$ | $\forall x \exists y (rxy)$

Bridge rules | Semantical Condition

valid $\forall^p \varphi(\Box_r\varphi \supset \Box_s\varphi)$ | $\forall x \forall y (rxy \supset sxy)$

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

Selection Function Semantics [Stalnaker, 1968]

...

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

Embedding in HOL

...

$$\Rightarrow_f \quad = \quad \lambda\varphi_{\iota\to o}\, \lambda\psi_{\iota\to o}\, \lambda s_\iota\, \forall v_\iota\, (\neg f s \varphi v \vee \psi v)$$

Selection Function Semantics [Stalnaker, 1968]

...

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

Embedding in HOL

...

$$\Rightarrow \quad = \quad \lambda f_{\iota \to (\iota \to o) \to (\iota \to o)} \lambda \varphi_{\iota \to o} \lambda \psi_{\iota \to o} \lambda s_\iota \forall v_\iota (\neg f s \varphi v \vee \psi v)$$

Conditional Logics

Selection Function Semantics [Stalnaker, 1968]

...

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

Embedding in HOL

...

$$\Rightarrow \quad = \quad \lambda f_{\iota\to(\iota\to o)\to(\iota\to o)}\, \lambda\varphi_{\iota\to o}\, \lambda\psi_{\iota\to o}\, \lambda s_\iota\, \forall v_\iota\, (\neg f\, s\varphi v \vee \psi v)$$

Interesting, since selection function semantics is a generalization of Kripke semantics which cannot be naturally translated to FOL.

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)

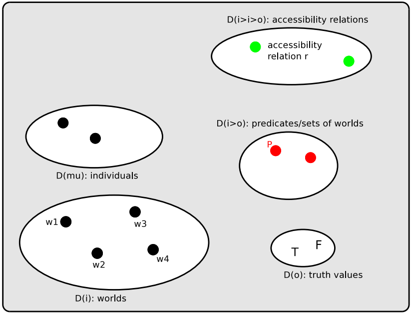
$$\models^{L} \varphi \quad \text{iff} \quad \models^{HOL}_{\text{Henkin}} \text{valid}\ \varphi_{\iota\to o} \quad \text{iff} \quad \vdash^{\text{seq}^{\text{HOL}}}_{\text{cut-free}} \text{valid}\ \varphi_{\iota\to o}$$

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 . . .

Varying and Cumulative Domain Quantification

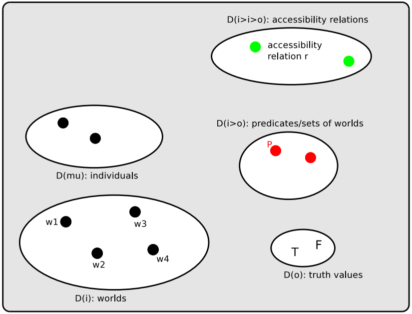
Constant Domain



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

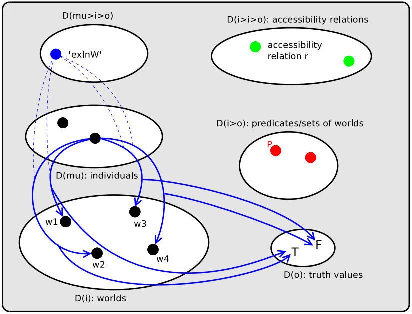
Varying and Cumulative Domain Quantification

Constant Domain



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

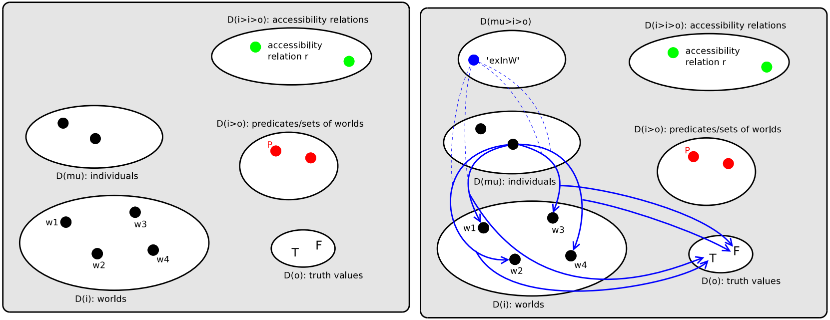
Varying and Cumulative Domain



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

Varying and Cumulative Domain Quantification

Constant Domain — Varying and Cumulative Domain



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

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

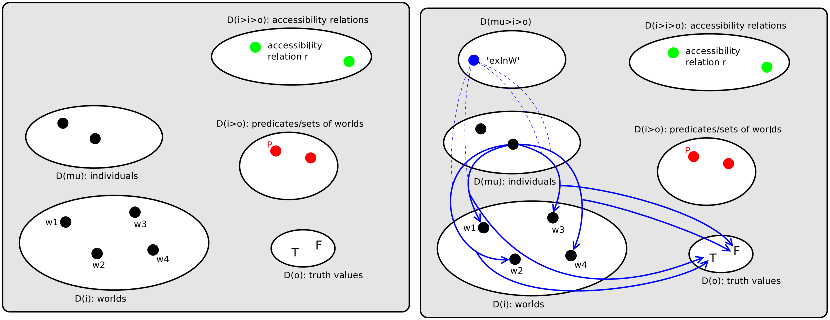
domains are non-empty → $\forall w_\iota\, \exists x_\mu\, \mathbf{exInW} xw$

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

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

Varying and Cumulative Domain Quantification

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) → $\forall w_\iota\, \mathbf{exInW} cw$

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

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

Instances of (Converse) Barcan Formula:

$$\text{valid } \forall^* x(\varphi \Rightarrow \psi(x)) \to (\varphi \Rightarrow \forall^* x \psi(x)) \qquad \text{(BF)}$$

$$\text{valid } (\varphi \Rightarrow \forall^* x \psi(x)) \to \forall^* x(\varphi \Rightarrow \psi(x)) \qquad \text{(CBF)}$$

BF:

if $*$ = varying domain then HOL-P: CounterSatisfiable
if $*$ = constant domain then HOL-P: Theorem

CBF:

if $*$ = varying domain then HOL-P: CounterSatisfiable
if $*$ = constant domain then HOL-P: Theorem

Automating Quantified Conditional Logics

The following examples are taken from [Delgrande, Artif.Intell., 1998]

$\phi \Rightarrow_x \psi$ stands for $(\exists^{va} x \phi) \Rightarrow \forall^{va} x(\phi \rightarrow \psi)$

"Birds (b) normally fly (f), but Opus (o) is a bird that normally does not fly."

$$b(x) \Rightarrow_x f(x), \quad b(o), \quad b(o) \Rightarrow \neg f(o)$$

HOL-P: Satisfiable (constant domain HOL-P: Unsatisfiable)

"Birds normally fly and necessarily Opus the bird does not fly."

$$b(x) \Rightarrow_x f(x), \quad \Box(b(o) \wedge \neg f(o))$$

HOL-P: Satisfiable (constant domain HOL-P: Unsatisfiable)

"Birds normally fly, penguins normally do not fly and all penguins are necessarily birds."

$$b(x) \Rightarrow_x f(x), \quad p(x) \Rightarrow_x \neg f(x), \quad \forall^{va} \Box(p(x) \rightarrow b(x))$$

HOL-P: Satisfiable (constant domain HOL-P: Satisfiable)

for more see [Benzmüller, IJCAI, 2013]

Automating Quantified Conditional Logics

The following examples are taken from [Delgrande, Artif.Intell., 1998]

$\phi \Rightarrow_x \psi$ stands for $(\exists^{va} x \phi) \Rightarrow \forall^{va} x (\phi \rightarrow \psi)$

"Birds (b) normally fly (f), but Opus (o) is a bird that normally does not fly."

$$b(x) \Rightarrow_x f(x), \quad b(o), \quad b(o) \Rightarrow \neg f(o)$$

HOL-P: Satisfiable (constant domain HOL-P: Unsatisfiable)

"Birds normally fly and necessarily Opus the bird does not fly."

$$b(x) \Rightarrow_x f(x), \quad \Box(b(o) \wedge \neg f(o))$$

HOL-P: Satisfiable (constant domain HOL-P: Unsatisfiable)

"Birds normally fly, penguins normally do not fly and all penguins are necessarily birds."

$$b(x) \Rightarrow_x f(x), \quad p(x) \Rightarrow_x \neg f(x), \quad \forall^{va} \Box(p(x) \rightarrow b(x))$$

HOL-P: Satisfiable (constant domain HOL-P: Satisfiable)

for more see [Benzmüller, IJCAI, 2013]

Automating Quantified Conditional Logics

Freie Universität Berlin

The following examples are taken from [Delgrande, Artif.Intell., 1998]

$\phi \Rightarrow_x \psi$ stands for $(\exists^{va} x \phi) \Rightarrow \forall^{va} x (\phi \rightarrow \psi)$

"Birds (b) normally fly (f), but Opus (o) is a bird that normally does not fly."

$$b(x) \Rightarrow_x f(x), \quad b(o), \quad b(o) \Rightarrow \neg f(o)$$

HOL-P: Satisfiable (constant domain HOL-P: Unsatisfiable)

"Birds normally fly and necessarily Opus the bird does not fly."

$$b(x) \Rightarrow_x f(x), \quad \Box(b(o) \wedge \neg f(o))$$

HOL-P: Satisfiable (constant domain HOL-P: Unsatisfiable)

"Birds normally fly, penguins normally do not fly and all penguins are necessarily birds."

$$b(x) \Rightarrow_x f(x), \quad p(x) \Rightarrow_x \neg f(x), \quad \forall^{va} \Box(p(x) \rightarrow b(x))$$

HOL-P: Satisfiable (constant domain HOL-P: Satisfiable)

for more see [Benzmüller, IJCAI, 2013]

Automating Quantified Conditional Logics

The following examples are taken from [Delgrande, Artif.Intell., 1998]

$\phi \Rightarrow_x \psi$ stands for $(\exists^{va} x \phi) \Rightarrow \forall^{va} x(\phi \rightarrow \psi)$

"Birds (b) normally fly (f), but Opus (o) is a bird that normally does not fly."

$$b(x) \Rightarrow_x f(x), \quad b(o), \quad b(o) \Rightarrow \neg f(o)$$

HOL-P: <u>Satisfiable</u> (constant domain HOL-P: <u>Unsatisfiable</u>)

"Birds normally fly and necessarily Opus the bird does not fly."

$$b(x) \Rightarrow_x f(x), \quad \Box(b(o) \wedge \neg f(o))$$

HOL-P: <u>Satisfiable</u> (constant domain HOL-P: <u>Unsatisfiable</u>)

"Birds normally fly, penguins normally do not fly and all penguins are necessarily birds."

$$b(x) \Rightarrow_x f(x), \quad p(x) \Rightarrow_x \neg f(x), \quad \forall^{va} \Box(p(x) \rightarrow b(x))$$

HOL-P: <u>Satisfiable</u> (constant domain HOL-P: <u>Satisfiable</u>)

for more see [Benzmüller, IJCAI, 2013]

Evaluation: What Systems are there to compare with?

- **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.

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

Freie Universität Berlin

No. of solved monomodal problems (out of 580, 600sec timeout)

	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**

Experiments for K, T, S4, S5, . . . (const/vary/cumul) still running.

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