Higher-Order Modal Logics: Automation and Applications

Christoph Benzmüller[1] and Bruno Woltzenlogel-Paleo

[1] Supported by DFG Heisenberg Fellowship BE 2501/9-1/2

Introduction

Introduction

This Tutorial ...

... is about (narrow picture)

- higher-order modal logic (HOML)
- classical higher-order logic (HOL)
- embedding of HOML in HOL
- mechanisation and automation with HOL ATPs
- various applications, including metaphysics

... is about (wider picture)

- a quite universal approach to automate reasoning for a wide range of classical and non-classical logics
- a very broad range of possible applications
- including meta-reasoning about logical systems

Introduction

Our primary interest . . .

- is in expressive logics:
  quantification (first- and higher-order) and lambda-expressions;
- propositional fragments are trivially covered

Introduction: Outline of Tutorial

Part I

- Intro (15 min) (motivating examples) — Christoph
- HOL (20 min) — Christoph
- HOML (10 min) — Christoph
- Automation of HOML with LEO-II and Isabelle (15 min) — Christoph
- Ontological Argument (30 min) — Bruno

Part II

- HOML in the Coq Proof Assistant (25 min) — Bruno
- SUMO Ontology and HOML (20 min) — Christoph
- Meta-Reasoning (10 min) — Christoph
- Flexibility and Rigidity (10 min) — Bruno
- Paraconsistency (10 min) — Bruno
- Cut-Elimination (15 min) — Christoph

Introduction: Expressivity Matters — Cantor's Theorem

Cantor's theorem: The set of all subsets of A, that is, the power set of A, has a strictly greater cardinality than A itself.

In HOL Cantor's theorem (surjective version) can be encoded as

$$\neg \exists F \forall G \exists X . F X = G$$

HO ATPs can solve this problem very efficiently.

Their solution includes the detection and application of the diagonalisation argument

Today: basic test example for new higher-order theorem provers.

Further reading: [AndrewsEtAl., Automating higher-order logic, 1984]

Introduction: Expressivity Matters — Cantor's Theorem

Cantor's theorem: The set of all subsets of A, that is, the power set of A, has a strictly greater cardinality than A itself.

In HOL Cantor's theorem (surjective version) can be encoded as

$$\neg \exists F_{\iota \to (\iota \to o)} \forall G_{\iota \to o} \exists X_{\iota} . FX = G$$

HO ATPs can solve this problem very efficiently.

Their solution includes the detection and application of the diagonalisation argument

Today: basic test example for new higher-order theorem provers.

Further reading: [AndrewsEtAl., Automating higher-order logic, 1984]

Introduction: Expressivity Matters — Boolos' Example

[George Boolos, A curious inference, J. Philosophical Logic, 16:1-12, 1987]

1. $\forall n\ f(n,1) = s(1)$

2. $\forall x\ f(1,s(x)) = s(s(f(1,x)))$

3. $\forall n \forall x\ f(s(n),s(x)) = f(n,f(s(n),x))$

4. $D(1)$

5. $\forall x\ D(x) \rightarrow D(s(x)))$

$\therefore$

6. $D(f(s(s(s(s(1)))),s(s(s(s(1))))))$

Induction proof: from (4) and (5), we get $\forall x\, D(x)$, hence $D(f(s(s(s(s(1)))),s(s(s(s(1))))))$ by $\forall$-elimination.

But induction is not given, hence the first order proof consists of brute force modus ponens applications: infeasible number of single steps ($2^{(2^{\cdot^{\cdot^{\cdot 2}}})}$ with 64K '2s')

Introduction: Expressivity Matters — Boolos' Example

[Boolos, A curious inference, J. Philosophical Logic, 16:1-12, 1987]

Comprehension axioms $\exists N_{\overline{\alpha^n} \to \beta} \; \forall \overline{z^n} \; N(z^1, \ldots, z^n) = B_\beta$

Can be avoided: use λ-binding construct to denote N

Instances of comprehension axioms in Boolos' proof:

$$\exists N \forall z \, N(z) \leftrightarrow (\forall X \, X(1) \wedge \forall y \, (X(y) \to X(s(y))) \to X(z))$$

$$\exists E \forall z \, E(z) \leftrightarrow (N(z) \wedge D(z))$$

Central idea: "assume the induction principle holds for number z – corresponding to $N(z)$ – then we can show for any predicate X a property $X(z)$ by induction."

The proof employs the following lemmata:

Lemma 1: $N(1)$, $\forall y \, (N(y) \to N(s(y)))$, $N(s(s(s(s(1)))))$, $E(1)$, $\forall y \, (E(y) \to E(s(y)))$, $E(s(1))$

Lemma 2: $\forall n \, N(n) \to \forall x \, (N(x) \to E(f(n,x)))$

The theorem itself is then an easy application of the two lemmata.

Introduction: Expressivity Matters — Boolos' Example

[1]By the comprehension principle of second order logic, $\exists N \forall z(Nz \leftrightarrow \forall X[X1 \& \forall y(Xy \rightarrow Xsy) \rightarrow Xz])$, [2]and then for some N, $\exists E \forall z(Ez \leftrightarrow Nz \& Dz)$.

[3]**Lemma 1:** [3.1]$N1$; [3.2]$\forall y(Ny \rightarrow Nsy)$; [3.3]$Nssss1$; [3.4]$E1$; [3.5]$\forall y(Ey \rightarrow Esy)$; [3.6]$Es1$;

[4]**Lemma 2:** $\forall n(Nn \rightarrow (\forall x(Nx \rightarrow Efnx)))$
Proof: [4.1]By comprehension, $\exists M \forall n(Mn \leftrightarrow \forall x(Nx \Rightarrow Efnx))$. [4.2]We want $\forall n(Nn \rightarrow Mn)$. [4.3]Enough to show [4.3.1]$M1$ and [4.3.2]$\forall n(Mn \rightarrow Msn)$, for then if [4.4]$Nn$, [4.5]$Mn$.
[4.3.1]$M1$: [4.3.1.1]Want $\forall x(Nx \rightarrow Ef1x)$. [4.3.1.2]By comprehension, $\exists Q \forall x(Qx \leftrightarrow Ef1x)$. [4.3.1.3]Want $\forall x(Nx \rightarrow Qx)$. [4.3.1.4]Enough to show [4.3.1.4.1]$Q1$ and [4.3.1.4.2]$\forall x(Qx \rightarrow Qsx)$.
[4.3.1.4.1]$Q1$: [4.3.1.4.1.1]Want $Ef11$. [4.3.1.4.1.2]But $f11 = s1$ by (1) and [4.3.1.4.1.3]$Es1$ by Lemma 1.
[4.3.1.4.2]$\forall x(Qx \rightarrow Qsx)$: [4.3.1.4.2.1]Suppose Qx, [4.3.1.4.2.2]i.e. Ef1x. [4.3.1.4.2.3]By (2) $f1sx = ssf1x$; [4.3.1.4.2.4]by Lemma 1 twice, $Ef1sx$. [4.3.1.4.2.5]Thus Qsx and [4.3.1.4.2.6]$M1$.
[4.3.2]$\forall n(Mn \rightarrow Msn)$: [4.3.2.1]Suppose Mn, [4.3.2.2]i.e. $\forall x(Nx \rightarrow Efnx)$. [4.3.2.3]Want Msn, [4.3.2.4]i.e. $\forall x(Nx \rightarrow Efsnx)$. [4.3.2.5]By comprehension, $\exists P \forall x(Px \leftrightarrow Efsnx)$. [4.3.2.6]Want $\forall x(Nx \rightarrow Px)$. [4.3.2.7]Enough to show [4.3.2.7.1]$P1$ and [4.3.2.7.2]$\forall x(Px \rightarrow Psx)$.
[4.3.2.7.1]$P1$: [4.3.2.7.1.1]Want $Efsn1$. [4.3.2.7.1.2]But $fsn1 = s1$ by (1) and [4.3.2.7.1.3]$Es1$ by Lemma 1.
[4.3.2.7.2]$\forall x(Px \rightarrow Psx)$: [4.3.2.7.2.1]Suppose Px, [4.3.2.7.2.2]i.e. $Efsnx$; [4.3.2.7.2.3]thus $Nfsnx$. [4.3.2.7.2.4]Want $Efsnsx$. [4.3.2.7.2.5]Since $Nfsnx$ and Mn, $Efnfsnx$. [4.3.2.7.2.6]But by (3) $fnfsnx = fsnsx$; [4.3.2.7.2.7]thus $Efsnsx$.
[5]By Lemma 1, $Nssss1$. [6]By Lemma 2, $Efssss1ssss1$. [7]Thus, $Dfssss1ssss1$, as desired.

Formalization in OMEGA and Mizar: see [BenzmüllerBrown, The curious inference of Boolos in MIZAR and OMEGA, Studies in Logic, Grammar, and Rhetoric, volume 10(23), pp. 299-388, 2007.]
Earlier paper: [BenzmüllerKerber, A Lost Proof, 2001].
Earlier poster: `http://christoph-benzmueller.de/papers/poster-tphols01.pdf`

Inroduction: Higher-Order Modal Logics

$\Box P$: P is necessary, P is obligatory, P is known, P is believed, always P ...

$\Diamond P$: P is possible, P is permissible, P is epistemically possible, P is doxastically possible, eventually P ...

$\Box$ and $\Diamond$ are not truth-functional

HOL can be extended by $\Box$ and $\Diamond$ to obtain HOML

There are many interesting applications of such logics, e.g. the Ontological Argument

Introduction: Embedding Approach — Idea

Your-logic (object-logic) ψ ::=

HOL (meta-logic) φ ::=

Embedding of in

$=$

$=$

$=$

$=$

Embedding of meta-logical notions on in

$\mathit{valid} =$

$\mathit{satisfiable} =$

$\ldots =$

Pass this set of equations to a higher-order automated theorem prover

Introduction: Embedding Approach — HOML in HOL

HOML $\varphi, \psi \quad ::= \quad \ldots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x_\gamma\, \varphi \mid \exists x_\gamma\, \varphi$

HOL $s, t \quad ::= \quad C_\alpha \mid x_\alpha \mid (\lambda x_\alpha s_\beta)_{\alpha\rightarrow\beta} \mid (s_{\alpha\rightarrow\beta}\, t_\alpha)_\beta \mid \neg s_o \mid s_o \vee t_o \mid \forall x_\alpha\, t_o$

HOML in HOL: HOML formulas φ are mapped to HOL predicates $\varphi_{\iota\rightarrow o}$
(explicit representation of labelled formulas)

$$
\begin{array}{lcl}
\neg & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \neg\varphi w \\
\wedge & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda w_\iota(\varphi w \wedge \psi w) \\
\rightarrow & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda w_\iota(\neg\varphi w \vee \psi w) \\
\forall & = & \lambda h_{\gamma\rightarrow(\iota\rightarrow o)}\lambda w_\iota \forall d_\gamma\, hdw \\
\exists & = & \lambda h_{\gamma\rightarrow(\iota\rightarrow o)}\lambda w_\iota \exists d_\gamma\, hdw \\
\Box & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \forall u_\iota\, (\neg rwu \vee \varphi u) \\
\Diamond & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \exists u_\iota\, (rwu \wedge \varphi u) \\
\text{valid} & = & \lambda\varphi_{\iota\rightarrow o}\forall w_\iota\, \varphi w
\end{array}
$$

Ax (polymorphic over γ)

The equations in Ax are given as axioms to the HOL provers!

Introduction: Embedding Approach — A “Lean” Prover for HOML KB

```
%----The base type $i (already built-in) stands here for worlds and
%----mu for individuals; $o (also built-in) is the type of Booleans
thf(mu_type,type,(mu:$tType)).
%----Reserved constant r for accessibility relation
thf(r,type,(r:$i>$i>$o)).
%----Modal logic operators not, or, and, implies, box, diamond
thf(mnot_type,type,(mnot:($i>$o)>$i>$o)).
thf(mnot,definition,(mnot = (^[A:$i>$o,W:$i]:~(A@W)))).
thf(mor_type,type,(mor:($i>$o)>($i>$o)>$i>$o)).
thf(mor,definition,(mor = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)|(Psi@W))))).
thf(mand_type,type,(mand:($i>$o)>($i>$o)>$i>$o)).
thf(mand,definition,(mand = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)&(Psi@W))))).
thf(mimplies_type,type,(mimplies:($i>$o)>($i>$o)>$i>$o)).
thf(mimplies,definition,(mimplies = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)&(Psi@W))))).
thf(mbox_type,type,(mbox:($i>$o)>$i>$o)).
thf(mbox,definition,(mbox = (^[A:$i>$o,W:$i]:![V:$i]:(~(r@W@V)|(A@V))))).
thf(mdia_type,type,(mdia:($i>$o)>$i>$o)).
thf(mdia,definition,(mdia = (^[A:$i>$o,W:$i]:?[V:$i]:((r@W@V)&(A@V))))).
%----Quantifiers (constant domains) for individuals and propositions
thf(mforall_ind_type,type,(mforall_ind:(mu>$i>$o)>$i>$o)).
thf(mforall_ind,definition,(mforall_ind = (^[A:mu>$i>$o,W:$i]:![X:mu]:(A@X@W)))).
thf(mforall_indset_type,type,(mforall_indset:((mu>$i>$o)>$i>$o)>$i>$o)).
thf(mforall_indset,definition,(mforall_indset = (^[A:(mu>$i>$o)>$i>$o,W:$i]:![X:mu>$i>$o]:(A@X@W)))).
thf(mexists_ind_type,type,(mexists_ind:(mu>$i>$o)>$i>$o)).
thf(mexists_ind,definition,(mexists_ind = (^[A:mu>$i>$o,W:$i]:?[X:mu]:(A@X@W)))).
thf(mexists_indset_type,type,(mexists_indset:((mu>$i>$o)>$i>$o)>$i>$o)).
thf(mexists_indset,definition,(mexists_indset = (^[A:(mu>$i>$o)>$i>$o,W:$i]:?[X:mu>$i>$o]:(A@X@W)))).
%----Definition of validity (grounding of lifted modal formulas)
thf(v_type,type,(v:($i>$o)>$o)).
thf(mvalid,definition,(v = (^[A:$i>$o]:![W:$i]:(A@W)))).
%----Properties of accessibility relations: symmetry
thf(msymmetric_type,type,(msymmetric:($i>$i>$o)>$o)).
thf(msymmetric,definition,(msymmetric = (^[R:$i>$i>$o]:![S:$i,T:$i]:((R@S@T)=>(R@T@S))))).
%----Here we work with logic KB, i.e., we postulate symmetry for r
thf(sym,axiom,(msymmetric@r)).
```

Reading on THF0 syntax: [SutcliffeBenzmüller, J.Formalized Reasoning, 2010]

Advantages of the Approach

[BenzmüllerWoltzenlogelPaleo, CSR 2015]
[BenzmüllerWoltzenlogelPaleo, ECAI 2014]
[BenzmüllerPaulson, Logica Universalis 2013]
[BenzmüllerWoltzenlogelPaleo, AFP 2013]
[Benzmüller, IJCAI 2013]

Pragmatics and convenience.
* 'Implementing' a new theorem prover imade very simple
* even for very challenging quantified non-classical logics

Flexibility
* Rapid experimentations with logic variations: quantifiers for constant, varying and cumulative domains; rigid or non-rigid terms; etc.
* Sahlqvist axioms may be postulated or (preferably) the corresponding conditions of the accessibility relation
* prominent connections between logics can be verified and exploited

Availability
* Option 1: Reuse and adapt our TPTP THF0 encodings
* Option 2: Reuse and adapt our Isabelle encodings
* Option 3: Reuse and adapt our Coq encodings
* Option X: Adapt our encodings for your preferred HO prover
* Options can be employed simultaneously.

Advantages of the Approach

Relation to labelled deductive systems
* Labelled deductive systems: meta-level (world-)labelling techniques
* Embedding approach: labels are encoded directly in HOL logic
* No extra-logical annotations in embedding approach

Relation to the standard translation
* Intra-logical formalisation and implementation of the standard translation
* Extension: various other logics
* Extension: quantifiers (different domain conditions)
* Future work: functional translation as an alternative

Soundness and completeness
* Sound and complete (Henkin semantics)

Meta-reasoning
* Example: Verification of the modal logic cube in Isabelle (PxTP@CADE talk)
* Example: Soundness of the usual ALC tableaux rules
* Example: Correspondence between ALC and base modal logic K
* Example: Some meta-level results for conditional logics
* Future work: Completeness results via abstract consistency

Advantages of the Approach

Direct calculi and user intuition
* Implementation of ‘direct’ proof calculi on top of logic embeddings
* Example: ND calculus for HOML as tactics in Coq
* Human intuitive proofs enabled at the interaction layer
* Automation via embedding and ATPs for HOL
* Interesting perspective for mixed proof developments
* Future work: proof planning to automate the abstract-level direct proof calculi; proof assistants in the style of Ωmega could eventually be adapted for this (support for 3-dimensional proof objects!)

In this Tutorial we will address and illustrate many of the above advantages!

Cut-elimination
* Very generic result: combine soundness and completeness of the embedding with the fact that HOL already enjoys cut-elimination for Henkin semantics

λ

Higher-order Logic (HOL)

Classical Higher-Order Logic (HOL)

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	-	✓	$continuous(\lambda X\, X)$
- Predicates/Sets/Rels	-	✓	$reflexive(=)$
Powerful abbreviations	-	✓	$reflexive = \lambda R\, \forall X R(X, X)$

Classical Higher-Order Logic (HOL)

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	-	✓	$reflexive_{(\iota\to\iota\to o)\to o} = \lambda R_{(\iota\to\iota\to o)} \, \forall X_\iota R(X, X)$

Simple Types: Prevent Some Paradoxes and Inconsistencies

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \rightarrow \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$
(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \rightarrow \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg \forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists Xs$ stands for $\neg\forall X\neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

$$s,t \quad ::= \quad p_\alpha \mid X_\alpha \mid (\lambda X_\alpha\, s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \mid$$

$$(\neg_{o\to o}\, s_o) \mid \boxed{((\vee_{o\to o\to o}\, s_o)\, t_o)} \mid \forall_{(\alpha\to o)\to o}(\lambda X_\alpha\, s_o)$$

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction (we may use infix notation, omit types and brackets: $s \vee t$)
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \rightarrow \beta)$

(we may add further base types, e.g. μ)

HOL Language:

$$s, t \quad ::= \quad p_\alpha \mid X_\alpha \mid (\lambda X_\alpha\, s_\beta)_{\alpha\rightarrow\beta} \mid (s_{\alpha\rightarrow\beta}\, t_\alpha)_\beta \mid$$
$$(\neg_{o\rightarrow o}\, s_o) \mid ((\vee_{o\rightarrow o\rightarrow o}\, s_o)\, t_o) \mid \boxed{\forall_{(\alpha\rightarrow o)\rightarrow o}(\lambda X_\alpha\, s_o)}$$

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification (we may use syntactical sugar: $\forall X_\alpha s$)

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg \forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists X s$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists Xs$ stands for $\neg\forall X \neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$
(we may add further base types, e.g. μ)

HOL Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\exists Xs$ stands for $\neg\forall X\neg s$

Equality may also be defined: $s \doteq t$ stands for $\forall P(Ps \Rightarrow Pt)$
(but it is stongly recommended to add primitive equality!)

HOL: Syntax

α-conversion

is considered implicitly, e.g. $(\lambda X\, pX)$ is identified with $(\lambda Y\, pY)$.

Substitution

of a term s_α for X_α in t_β is denoted by $[s/X]t$.
We assume the bound variables of t avoid variable capture.

β-reduction and η-reduction

β-redex has the form $(\lambda X\, s)t$ and β-reduces to $[t/X]s$.
η-redex has the form $(\lambda X\, sX)$ where X is not free in s; it η-reduces to s.
$s \equiv_\beta t$ means s can be converted to t by β-reductions and expansions.
$s \equiv_{\beta\eta} t$ means s can be converted to t using both β and η.
For each $s_\alpha \in \mathrm{HOML}$ there is a unique β-normal form and a unique $\beta\eta$-normal form.

HOL: Syntax

α-conversion
is considered implicitly, e.g. $(\lambda X\, pX)$ is identified with $(\lambda Y\, pY)$.

Substitution
of a term s_α for X_α in t_β is denoted by $[s/X]t$.
We assume the bound variables of t avoid variable capture.

β-reduction and η-reduction
β-redex has the form $(\lambda X\, s)t$ and β-reduces to $[t/X]s$.
η-redex has the form $(\lambda X\, sX)$ where X is not free in s; it η-reduces to s.
$s \equiv_\beta t$ means s can be converted to t by β-reductions and expansions.
$s \equiv_{\beta\eta} t$ means s can be converted to t using both β and η.
For each $s_\alpha \in \mathrm{HOML}$ there is a unique β-normal form and a unique $\beta\eta$-normal form.

HOL: Syntax

α-conversion

is considered implicitly, e.g. $(\lambda X \, pX)$ is identified with $(\lambda Y \, pY)$.

Substitution

of a term s_α for X_α in t_β is denoted by $[s/X]t$.
We assume the bound variables of t avoid variable capture.

β-reduction and η-reduction

β-redex has the form $(\lambda X \, s)t$ and β-reduces to $[t/X]s$.
η-redex has the form $(\lambda X \, sX)$ where X is not free in s; it η-reduces to s.
$s \equiv_\beta t$ means s can be converted to t by β-reductions and expansions.
$s \equiv_{\beta\eta} t$ means s can be converted to t using both β and η.
For each $s_\alpha \in \mathrm{HOML}$ there is a unique β-normal form and a unique $\beta\eta$-normal form.

HOL: Semantics

Semantics of HOL is (meanwhile) well understood

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

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

Recommended Reading: [BenzmüllerMiller, HandbookHistoryOfLogic, Vol.9, 2014]

HOL: Semantics

A Frame D
is a collection $\{D_\alpha\}_{\alpha \in \mathrm{T}}$ of nonempty sets D_α, such that

- D_ι can be chosen freely
- $D_o = \{T, F\}$ (for truth and falsehood), and
- $D_{\alpha\to\beta}$ are collections of functions mapping D_α into D_β

A Model M
for HOL is a tuple $\boldsymbol{M} = \langle \boldsymbol{D}, \boldsymbol{I} \rangle$, where

- $\boldsymbol{D}$ is a frame;
- $\boldsymbol{I}$ is a family of typed interpretation functions mapping constant symbols $\boldsymbol{p_\alpha}$ to appropriate elements of $\boldsymbol{D_\alpha}$, called the <u>denotation of $\boldsymbol{p_\alpha}$</u>;
- the logical connectives $\neg$, $\vee$, and $\forall$ are always given the standard denotations;
- moreover, we assume that the domains $\boldsymbol{D_{\alpha\to\alpha\to o}}$ contain the respective identity relations.

HOL: Semantics

Variable Assignment

A variable assignment g maps variables X_α to elements in D_α.
$g[d/W]$ denotes the assignment that is identical to g, except for variable W, which is now mapped to d.

Interpretation/Value of a HOL term

The value $\|s_\alpha\|^{M,g}$ of a HOL term s_α on a model $M = \langle D, I\rangle$ under assignment g is an element $d \in D_\alpha$ defined in the following way:

1. $\|p_\alpha\|^{M,g} = I(p_\alpha)$
2. $\|X_\alpha\|^{M,g} = g(X_\alpha)$
3. $\|(s_{\alpha\to\beta}\, t_\alpha)_\beta\|^{M,g} = \|s_{\alpha\to\beta}\|^{M,g}(\|t_\alpha\|^{M,g})$
4. $\|(\lambda X_\alpha\, s_\beta)_{\alpha\to\beta}\|^{M,g}$ = the function f from D_α to D_β such that $f(d) = \|s_\beta\|^{M,g[d/X_\alpha]}$ for all $d \in D_\alpha$
5. $\|(\neg_{o\to o}\, s_o)_o\|^{M,g} = T$ iff $\|s_o\|^{M,g} = F$
6. $\|((\vee_{o\to o\to o}\, s_o)\, t_o)_o\|^{M,g} = T$ iff $\|s_o\|^{M,g} = T$ or $\|t_o\|^{M,g} = T$
7. $\|(\forall_{(\alpha\to o)\to o}(\lambda X_\alpha\, s_o))_o\|^{M,g} = T$ iff for all $d \in D_\alpha$ we have $\|s_o\|^{M,g[d/X_\alpha]} = T$

Standard and Henkin Models

In a standard model $\boldsymbol{M} = \langle \boldsymbol{D}, \boldsymbol{I} \rangle$ we have

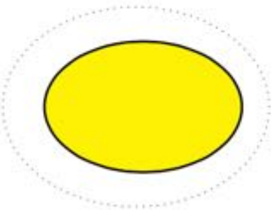
- $\boldsymbol{D_{\alpha \to \beta}} = \{f \mid f : D_\alpha \longrightarrow D_\beta\}$ (for all types α, β)

In a Henkin model $\boldsymbol{M} = \langle \boldsymbol{D}, \boldsymbol{I} \rangle$ we only require

- $\boldsymbol{D_{\alpha \to \beta}} \subseteq \{f \mid f : \boldsymbol{D_\alpha} \longrightarrow \boldsymbol{D_\beta}\}$ (for all types α, β)
- the valuation function $\| \cdot \|^{M,g}$ from above is total (every term denotes)

Any standard model is obviously also a Henkin model.
We consider Henkin models in the remainder.

HOL: Semantics
(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



Standard Models $\mathfrak{ST}(\Sigma)$

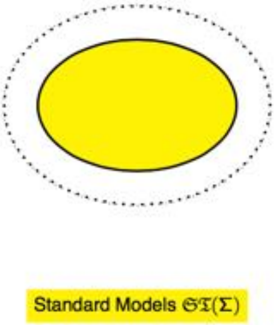
- Idea of Standard Semantics:

  $\iota \longrightarrow \mathcal{D}_\iota$ (choose)

  $o \longrightarrow \mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ (fixed)

  $(\alpha \to \beta) \longrightarrow$
  $\mathcal{D}_{\alpha\to\beta} = \mathcal{F}(\mathcal{D}_\alpha, \mathcal{D}_\beta)$ (fixed)

HOL: Semantics
(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



- Idea of Standard Semantics:

  $\iota \longrightarrow \mathcal{D}_\iota$ (choose)

  $o \longrightarrow \mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ (fixed)

  $(\alpha \to \beta) \longrightarrow$
  $\mathcal{D}_{\alpha\to\beta} = \mathcal{F}(\mathcal{D}_\alpha, \mathcal{D}_\beta)$ (fixed)

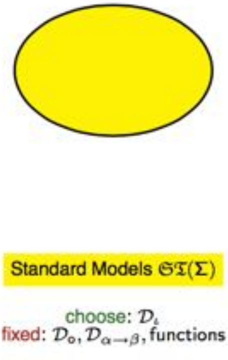
- Henkin's Generalization:

  $\mathcal{D}_{\alpha\to\beta} \subseteq \mathcal{F}(\mathcal{D}_\alpha, \mathcal{D}_\beta)$ (choose)
  but elements are still functions!

[Henkin-50]

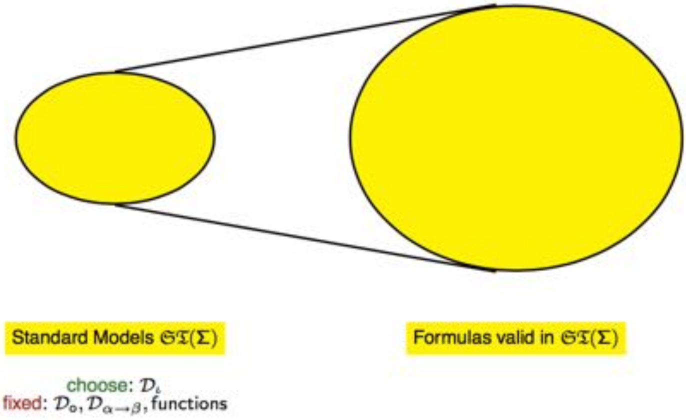
HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



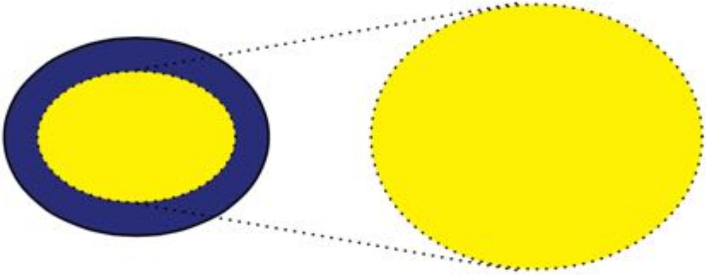
HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



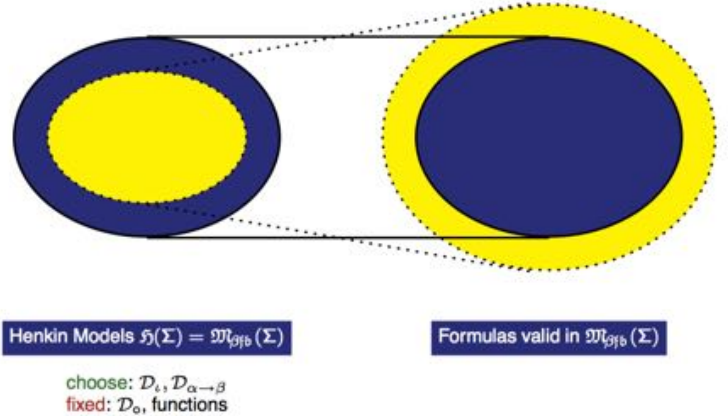
Henkin Models $\mathfrak{H}(\Sigma) = \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}(\Sigma)$

Formulas valid in $\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}(\Sigma)$?

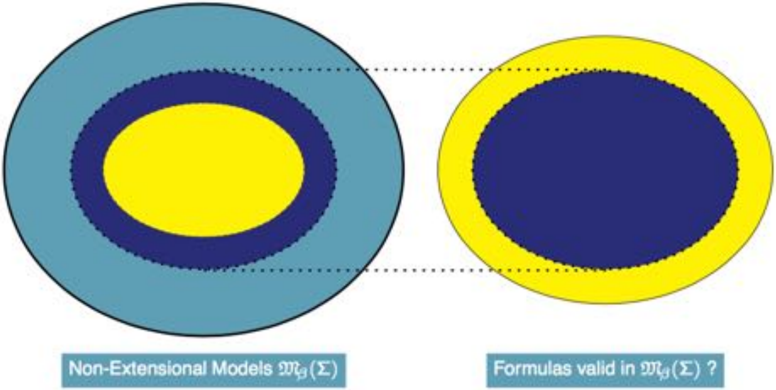
choose: $\mathcal{D}_\iota, \mathcal{D}_{\alpha\to\beta}$
fixed: $\mathcal{D}_o$, functions

HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



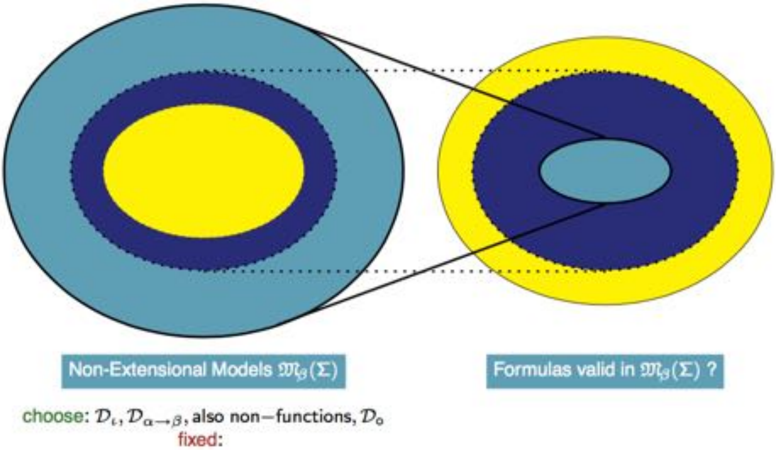
HOL: Semantics
(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



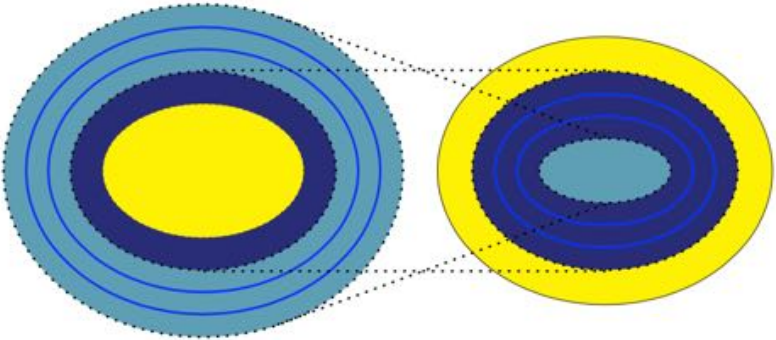
choose: $\mathcal{D}_\iota, \mathcal{D}_{\alpha\to\beta}$, also non−functions, $\mathcal{D}_o$
fixed:

HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



HOL: Semantics
(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



We additionally studied different model classes with 'varying degrees of extensionality'

HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009])

$\mathfrak{M}_\beta(\Sigma)$ — non-extensional models

$\mathfrak{b}$: Boolean extensionality, $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$
$\mathfrak{f}(= \eta + \xi)$: functional extensionality
η: η-functional
ξ: ξ-functionality

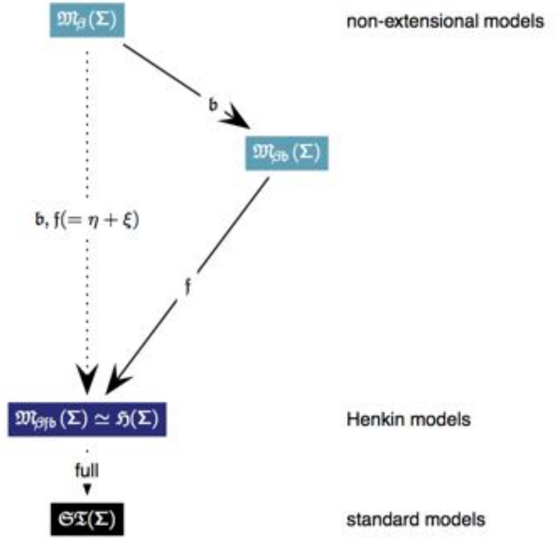
$\mathfrak{M}_{\beta\mathfrak{fb}}(\Sigma) \simeq \mathfrak{H}(\Sigma)$ — Henkin models

full

$\mathfrak{ST}(\Sigma)$ — standard models

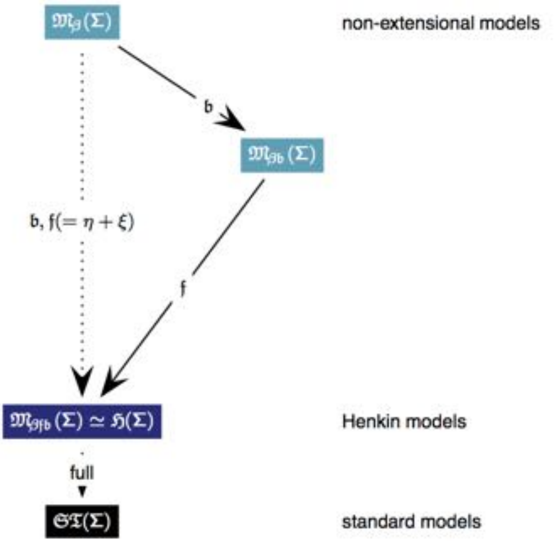
HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



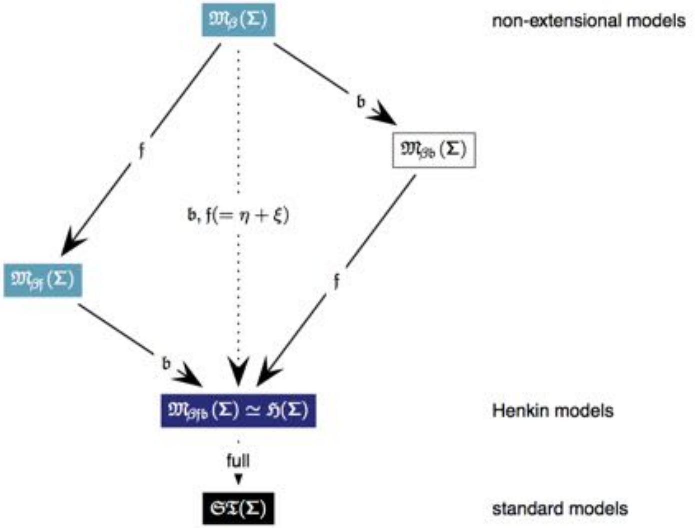
HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]

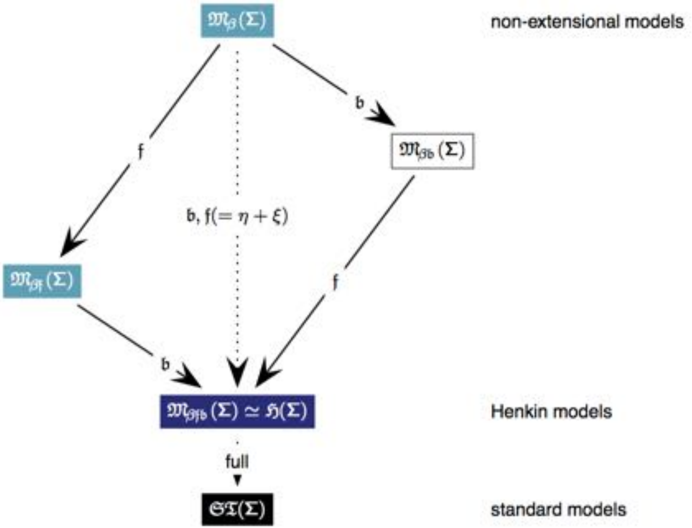


HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]

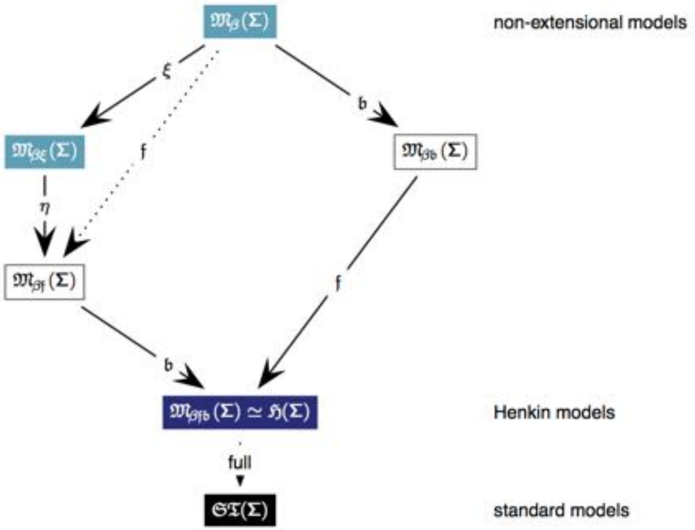


HOL: Semantics
(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



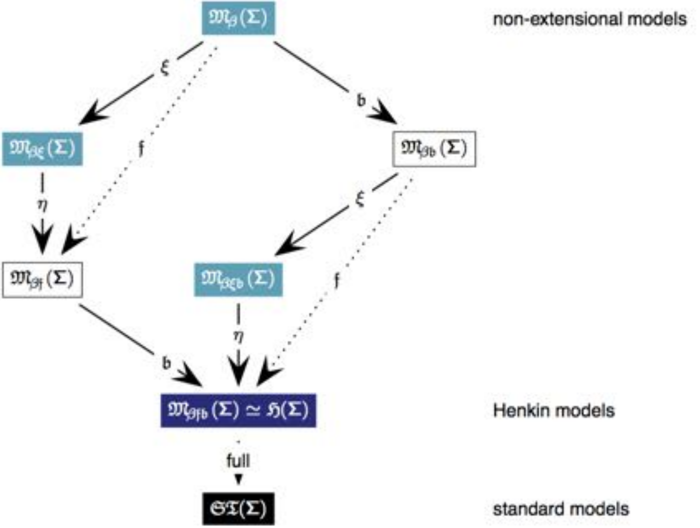
HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



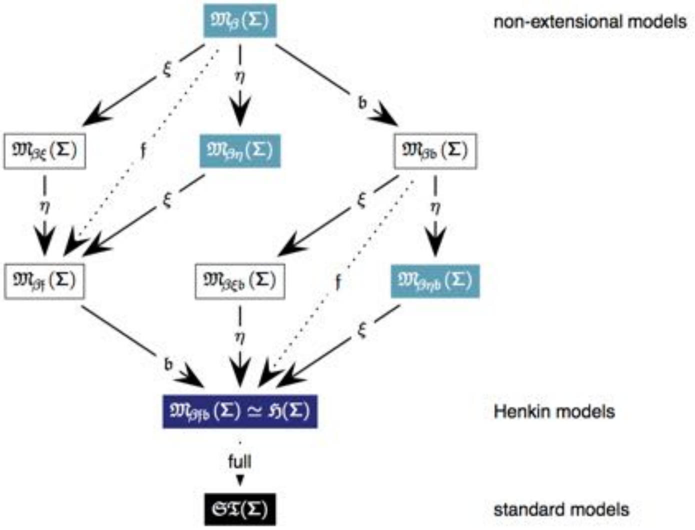
HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]

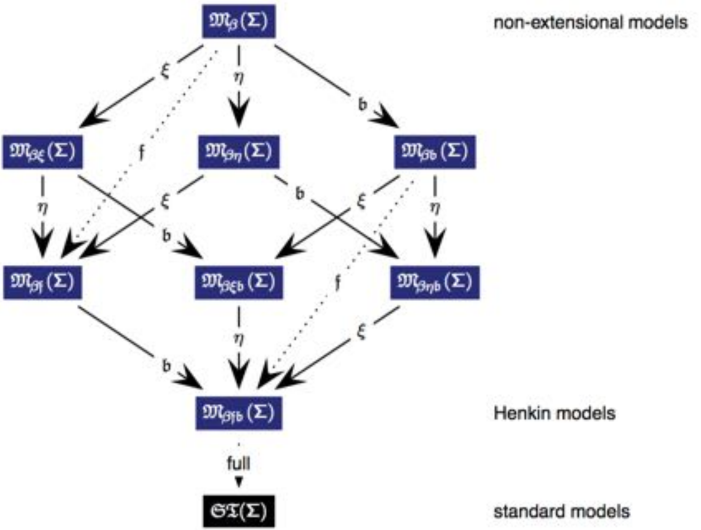


HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]

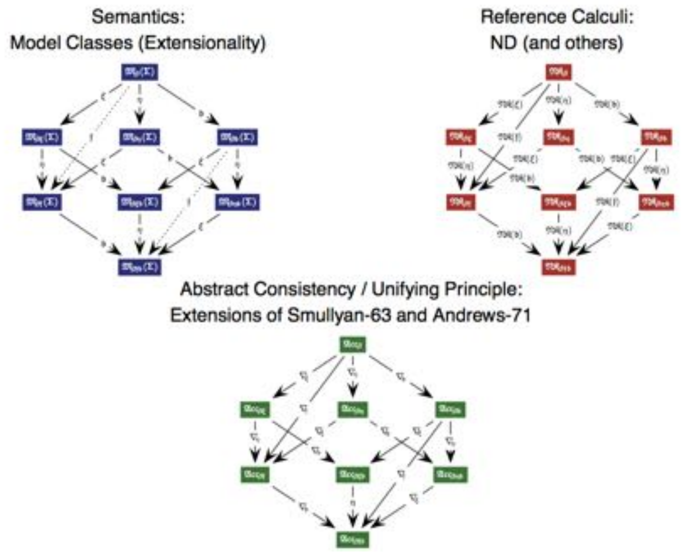


HOL: Semantics
(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009])



HOL: Semantics

(see [BenzmüllerBrownKohlhase, J.Symb.Log., 2004] and [BenzmüllerBrownKohlhase, LMCS, 2009]



Higher-order Modal Logic (HOML)

HOML: Motivation

$\Box P$

P is necessary, P is obligatory, P is known,
P is believed, always P ...

$\Diamond P$

P is possible, P is permissible, P is epistemically possible,
P is doxastically possible, eventually P ...

$\Box$ and $\Diamond$ are not truth-functional

HOL can be extended by $\Box P$ and $\Diamond P$ to obtain HOML

HOML: Motivation

$$\Box P$$

P is necessary, P is obligatory, P is known,
P is believed, always P . . .

$$\Diamond P$$

P is possible, P is permissible, P is epistemically possible,
P is doxastically possible, eventually P . . .

$\Box$ and $\Diamond$ are not truth-functional

HOL can be extended by $\Box P$ and $\Diamond P$ to obtain HOML

HOML: Motivation

$$\Box P$$

P is necessary, P is obligatory, P is known,
P is believed, always P . . .

$$\Diamond P$$

P is possible, P is permissible, P is epistemically possible,
P is doxastically possible, eventually P . . .

$\Box$ and $\Diamond$ are not truth-functional

HOL can be extended by $\Box P$ and $\Diamond P$ to obtain HOML

HOML: Motivation

$$\Box P$$

P is necessary, P is obligatory, P is known,
P is believed, always P . . .

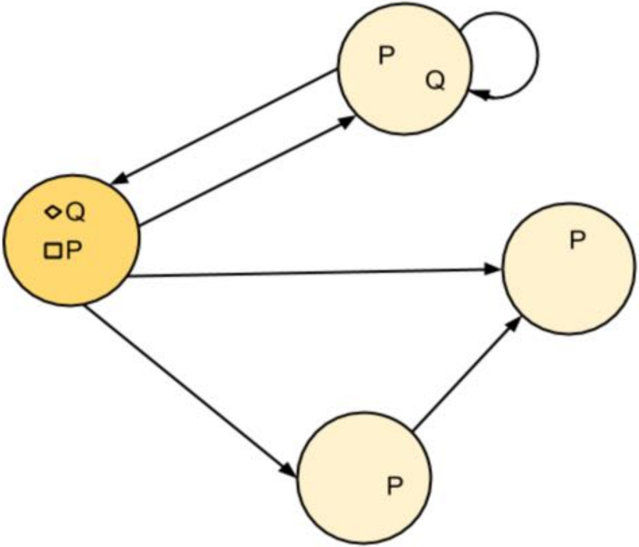
$$\Diamond P$$

P is possible, P is permissible, P is epistemically possible,
P is doxastically possible, eventually P . . .

$\Box$ and $\Diamond$ are not truth-functional

HOL can be extended by $\Box P$ and $\Diamond P$ to obtain HOML

HOML: Motivation
Kripke Semantics - Possible Worlds



HOML: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

HOML Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification
modal box operator

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\Diamond s$ stands for $\neg\Box\neg s$

HOML: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

HOML Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification
modal box operator

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\Diamond s$ stands for $\neg\Box\neg s$

HOML: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

HOML Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification
modal box operator

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\Diamond s$ stands for $\neg\Box\neg s$

HOML: Syntax

Simple Types: $\alpha, \beta \quad ::= \quad \iota \mid o \mid (\alpha \to \beta)$

HOML Language:

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

constant symbols
variable symbols
lambda abstraction
application
negation
disjunction
universal quantification
modal box operator

Terms of type o: formulas

Other logical connectives can be defined, e.g. $\Diamond s$ stands for $\neg\Box\neg s$

HOML: Syntax

α-conversion

... as before ...

Substitution

... as before ...

β-reduction and η-reduction

... as before ...

HOML: Semantics

Semantics of HOML investigated in

- [D. Gallin, Instensional and Higher-Order Modal Logic, North Holland, 1975]
- [R. Muskens, Higher Order Modal Logic, Handbook of Modal Logic, 2006]
- [Benzmüller and Woltzenlogel Paleo, Automating Gödel's Ontological Proof . . . , ECAI, 2014]
- our interest: combination of Kripke style models and Henkin semantics

HOML: Semantics

A Frame D

... as before ...

A Model M

for HOML is a quadruple $M = \langle W, R, D, \{I_w\}_{w \in W} \rangle$, where

- W is a set of worlds (or states);
- R is an accessibility relation between the worlds in W;
- D is a frame;
- for each $w \in W$, $\{I_w\}_{w \in W}$ is a family of typed interpretation functions mapping constant symbols p_α to appropriate elements of D_α, called the denotation of p_α in world w;
- the logical connectives $\neg$, $\vee$, $\forall$, and $\Box$ are always given the standard denotations;
- moreover, it is assumed that the domains $D_{\alpha \to \alpha \to o}$ contain the respective identity relations on objects of type α.

HOML: Semantics

Variable Assignment

. . . as before . . .

Interpretation/Value of a HOML term

The value $\|s_\alpha\|^{M,g,w}$ of a HOML term s_α on a model $M = \langle W, R, D, \{I_w\}_{w\in W}\rangle$ in a world $w \in W$ under variable assignment g is an element $d \in D_\alpha$ defined in the following way:

1. $\|p_\alpha\|^{M,g,w} = I_w(p_\alpha)$
2. $\|X_\alpha\|^{M,g,w} = g(X_\alpha)$
3. $\|(s_{\alpha\to\beta}\, t_\alpha)_\beta\|^{M,g,w} = \|s_{\alpha\to\beta}\|^{M,g,w}(\|t_\alpha\|^{M,g,w})$
4. $\|(\lambda X_\alpha\, s_\beta)_{\alpha\to\beta}\|^{M,g,w} =$ the function f from D_α to D_β such that $f(d) = \|s_\beta\|^{M,g[d/X_\alpha],w}$ for all $d \in D_\alpha$
5. $\|(\neg_{o\to o}\, s_o)_o\|^{M,g,w} = T$ iff $\|s_o\|^{M,g,w} = F$
6. $\|((\vee_{o\to o\to o}\, s_o)\, t_o)_o\|^{M,g,w} = T$ iff $\|s_o\|^{M,g,w} = T$ or $\|t_o\|^{M,g,w} = T$
7. $\|(\forall_{(\alpha\to o)\to o}(\lambda X_\alpha\, s_o))_o\|^{M,g,w} = T$ iff for all $d \in D_\alpha$ we have $\|s_o\|^{M,g[d/X_\alpha],w} = T$
8. $\|(\Box_{o\to o}\, s_o)_o\|^{M,g,w} = T$ **iff for all** $v \in W$ **with** wRv **we have** $\|s_o\|^{M,g,v} = T$

HOML: Semantics

Standard and Henkin Models (as before)

In a standard model $M = \langle W, R, D, \{I_w\}_{w \in W} \rangle$ we have

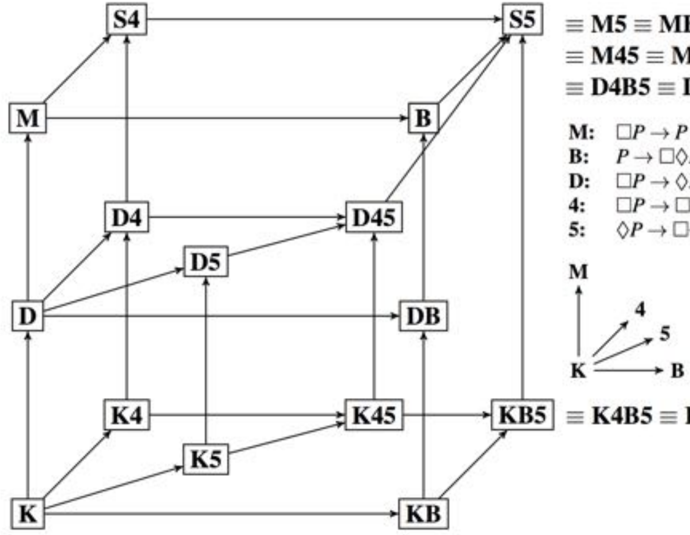
- $D_{\alpha \to \beta} = \{f \mid f : D_\alpha \longrightarrow D_\beta\}$ (for all types α, β)

In a Henkin model $M = \langle W, R, D, \{I_w\}_{w \in W} \rangle$ we only require

- $D_{\alpha \to \beta} \subseteq \{f \mid f : D_\alpha \longrightarrow D_\beta\}$ (for all types α, β)
- the valuation function $\| \cdot \|^{M,g,w}$ from above is total (every term denotes)

Any standard model is obviously also a Henkin model.
We consider Henkin models in the remainder.

HOML: The Modal Logic Cube



$\equiv$ **M5** $\equiv$ **MB5** $\equiv$ **M4B5**
$\equiv$ **M45** $\equiv$ **M4B** $\equiv$ **D4B**
$\equiv$ **D4B5** $\equiv$ **DB5**

M: $\Box P \rightarrow P$
B: $P \rightarrow \Box\Diamond P$
D: $\Box P \rightarrow \Diamond P$
4: $\Box P \rightarrow \Box\Box P$
5: $\Diamond P \rightarrow \Box\Diamond P$

$\equiv$ **K4B5** $\equiv$ **K4B**

How to automate HOML?

Embedding HOML in HOL

Challenge: No provers for Higher-order Modal Logic (HOML)

Church's Simple Type Theory

Our solution: Embedding in **Higher-order Classical Logic** (HOL)

Then use existing HOL theorem provers for reasoning in HOML

[BenzmüllerPaulson, Logica Universalis, 2013]

Assumption: Henkin semantics for both HOML and HOL

Previous empirical findings:

Embedding of First-order Modal Logic in HOL works well

[BenzmüllerOttenRaths, ECAI, 2012]

[Benzmüller, LPAR, 2013]

Embedding HOML in HOL

HOML $\quad \varphi, \psi \quad ::= \quad \ldots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x_\gamma\, \varphi \mid \exists x_\gamma\, \varphi$

HOL $\quad s, t \quad ::= \quad C_\alpha \mid x_\alpha \mid (\lambda x_\alpha s_\beta)_{\alpha\rightarrow\beta} \mid (s_{\alpha\rightarrow\beta}\, t_\alpha)_\beta \mid \neg s_o \mid s_o \vee t_o \mid \forall x_\alpha\, t_o$

HOML in HOL: HOML formulas φ are mapped to HOL predicates $\varphi_{\iota\rightarrow o}$
(explicit representation of labelled formulas)

$$
\begin{array}{lcl}
\neg & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \neg\varphi w \\
\wedge & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda w_\iota(\varphi w \wedge \psi w) \\
\rightarrow & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda w_\iota(\neg\varphi w \vee \psi w) \\
\forall & = & \lambda h_{\gamma\rightarrow(\iota\rightarrow o)}\lambda w_\iota \forall d_\gamma\, hdw \\
\exists & = & \lambda h_{\gamma\rightarrow(\iota\rightarrow o)}\lambda w_\iota \exists d_\gamma\, hdw \\
\Box & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \forall u_\iota\, (\neg rwu \vee \varphi u) \\
\Diamond & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \exists u_\iota\, (rwu \wedge \varphi u) \\
\text{valid} & = & \lambda\varphi_{\iota\rightarrow o}\forall w_\iota\, \varphi w
\end{array}
$$

Ax (polymorphic over γ)

The equations in Ax are given as axioms to the HOL provers!

Embedding HOML in HOL

HOML $\quad \varphi, \psi \quad ::= \quad \ldots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x_\gamma\, \varphi \mid \exists x_\gamma\, \varphi$

HOL $\quad s, t \quad ::= \quad C_\alpha \mid x_\alpha \mid (\lambda x_\alpha s_\beta)_{\alpha\rightarrow\beta} \mid (s_{\alpha\rightarrow\beta}\, t_\alpha)_\beta \mid \neg s_o \mid s_o \vee t_o \mid \forall x_\alpha\, t_o$

HOML in HOL: HOML formulas φ are mapped to HOL predicates $\varphi_{\iota\rightarrow o}$
(explicit representation of labelled formulas)

$$
\begin{array}{lcl}
\neg & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \neg\varphi w \\
\wedge & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda w_\iota(\varphi w \wedge \psi w) \\
\rightarrow & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda w_\iota(\neg\varphi w \vee \psi w) \\
\forall & = & \lambda h_{\gamma\rightarrow(\iota\rightarrow o)}\lambda w_\iota \forall d_\gamma\, h d w \\
\exists & = & \lambda h_{\gamma\rightarrow(\iota\rightarrow o)}\lambda w_\iota \exists d_\gamma\, h d w \\
\Box & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \forall u_\iota\, (\neg r w u \vee \varphi u) \\
\Diamond & = & \lambda\varphi_{\iota\rightarrow o}\lambda w_\iota \exists u_\iota\, (r w u \wedge \varphi u) \\
\text{valid} & = & \lambda\varphi_{\iota\rightarrow o}\forall w_\iota\, \varphi w
\end{array}
$$

Ax (polymorphic over γ)

The equations in Ax are given as axioms to the HOL provers!

Embedding HOML in HOL

HOML $\quad \varphi, \psi \quad ::= \quad \ldots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x_\gamma\, \varphi \mid \exists x_\gamma\, \varphi$

HOL $\quad s, t \quad ::= \quad C_\alpha \mid x_\alpha \mid (\lambda x_\alpha s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \mid \neg s_o \mid s_o \vee t_o \mid \forall x_\alpha\, t_o$

HOML in HOL: HOML formulas φ are mapped to HOL predicates $\varphi_{\iota\to o}$ (explicit representation of labelled formulas)

$$
\begin{array}{lcl}
\neg & = & \lambda\varphi_{\iota\to o}\lambda w_\iota \neg\varphi w \\
\wedge & = & \lambda\varphi_{\iota\to o}\lambda\psi_{\iota\to o}\lambda w_\iota(\varphi w \wedge \psi w) \\
\rightarrow & = & \lambda\varphi_{\iota\to o}\lambda\psi_{\iota\to o}\lambda w_\iota(\neg\varphi w \vee \psi w) \\
\forall & = & \lambda h_{\gamma\to(\iota\to o)}\lambda w_\iota \forall d_\gamma\, hdw \\
\exists & = & \lambda h_{\gamma\to(\iota\to o)}\lambda w_\iota \exists d_\gamma\, hdw \\
\Box & = & \lambda\varphi_{\iota\to o}\lambda w_\iota \forall u_\iota\, (\neg rwu \vee \varphi u) \\
\Diamond & = & \lambda\varphi_{\iota\to o}\lambda w_\iota \exists u_\iota\, (rwu \wedge \varphi u) \\
\text{valid} & = & \lambda\varphi_{\iota\to o}\forall w_\iota\, \varphi w
\end{array}
$$

Ax (polymorphic over γ)

The equations in Ax are given as axioms to the HOL provers!

Embedding HOML in HOL

Example

HOML formula $\Diamond \exists x G(x)$

HOML formula in HOL valid $(\Diamond \exists x G(x))_{\iota \rightarrow o}$

expansion, $\beta\eta$-conversion $\forall w_\iota (\Diamond \exists x G(x))_{\iota \rightarrow o}\, w$

expansion, $\beta\eta$-conversion $\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota \rightarrow o} u)$

expansion, $\beta\eta$-conversion $\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota \rightarrow o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL

Example

HOML formula	$\Diamond\exists x G(x)$
HOML formula in HOL	valid $(\Diamond\exists x G(x))_{\iota\to o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond\exists x G(x))_{\iota\to o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota\to o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota\to o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL

Example

HOML formula	$\Diamond \exists x G(x)$
HOML formula in HOL	valid $(\Diamond \exists x G(x))_{\iota \to o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond \exists x G(x))_{\iota \to o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota \to o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota \to o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL

Example

HOML formula	$\Diamond \exists x G(x)$
HOML formula in HOL	valid $(\Diamond \exists x G(x))_{\iota \to o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond \exists x G(x))_{\iota \to o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota \to o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota \to o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL

Example

HOML formula	$\Diamond \exists x G(x)$
HOML formula in HOL	valid $(\Diamond \exists x G(x))_{\iota \to o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond \exists x G(x))_{\iota \to o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota \to o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota \to o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL

Example

HOML formula	$\Diamond \exists x G(x)$
HOML formula in HOL	valid $(\Diamond \exists x G(x))_{\iota\rightarrow o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond \exists x G(x))_{\iota\rightarrow o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota\rightarrow o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota\rightarrow o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL

Example

HOML formula	$\Diamond \exists x G(x)$
HOML formula in HOL	valid $(\Diamond \exists x G(x))_{\iota\to o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond \exists x G(x))_{\iota\to o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota\to o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Expansion: user or prover may flexibly choose expansion depth

What are we doing?

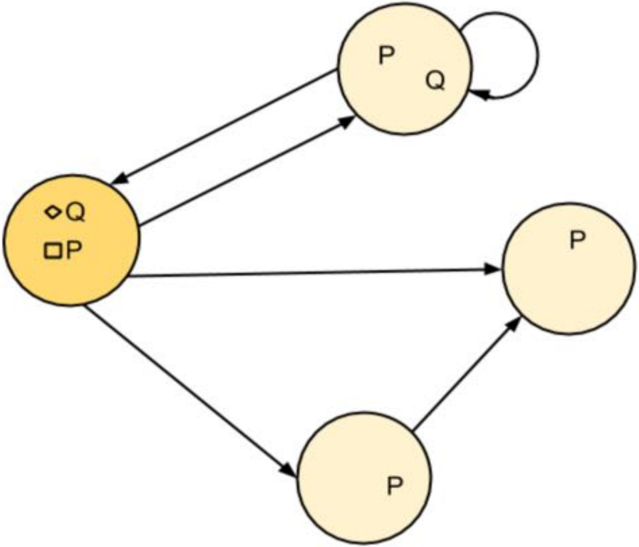
In order to prove that φ is valid in HOML,
–> we instead prove that valid $\varphi_{\iota\to o}$ can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Embedding HOML in HOL: Logics beyond K

Modal logic axioms	Semantical conditions
M: valid $\forall\varphi(\Box\varphi \rightarrow \varphi)$	$\forall x(rxy)$
B: valid $\forall\varphi(\varphi \rightarrow \Box\Diamond\varphi)$	$\forall x \forall y(rxy \rightarrow ryx)$
D: valid $\forall\varphi(\Box\varphi \rightarrow \Diamond\varphi)$	$\forall x \exists y(rxy)$
4: valid $\forall\varphi(\Box\varphi \rightarrow \Box\Box\varphi)$	$\forall x \forall y \forall z(rxy \wedge ryz \rightarrow rxz)$
5: valid $\forall\varphi(\Diamond\varphi \rightarrow \Box\Diamond\varphi)$	$\forall x \forall y \forall z(rxy \wedge rxz \rightarrow ryz)$

Possibilist vs. Actualist Quantification



Embedding HOML in HOL: Possibilist vs. Actualist Quantification

$\forall \quad = \lambda h_{\gamma \to (\iota \to o)} \lambda w_\iota \forall d_\gamma \; hdw$ (possibilist / constant dom.)

becomes

$\forall^{va} = \lambda h_{\gamma \to (\iota \to o)} \lambda w_\iota \forall d_\gamma \; (\mathbf{ExInW} dw \to hdw)$ (actualist / varying dom.)

where **ExInW** is an existence predicate.

Additional axioms (optional):

- 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: $\forall x \forall v \forall w (\mathbf{exInW} xv \wedge rvw \supset \mathbf{exInW} xw)$

Embedding HOML in HOL: Theoretical Results

Soundness and Completeness

$$\models^{HOML}_{\text{Henkin}} s_o \quad \text{iff} \quad \text{Ax} \models^{HOL}_{\text{Henkin}} \mathit{valid}\, s_{\iota\to o} \quad (\text{iff} \quad \text{Ax} \vdash^{\text{HOL}}_{\text{cut-free}} \mathit{valid}\, \varphi_{\iota\to o})$$

Embedding HOML in HOL: Theoretical Results

Soundness and Completeness

$$\models^{HOML}_{\text{Henkin}} s_o \quad \text{iff} \quad \text{Ax} \models^{HOL}_{\text{Henkin}} \mathit{valid}\, s_{\iota\to o}$$

Proof sketch (adapts [Benzmüller and Paulson, Logica Universalis, 2013]):
By contraposition it is sufficient to show

$$\not\models^{HOML}_{\text{Henkin}} s_o \text{ iff Ax} \not\models^{HOL}_{\text{Henkin}} \mathit{valid}\, s_{\iota\to o}$$

One easily gets the proof by choosing the obvious correspondences between D and $\boldsymbol{D}$, W and $\boldsymbol{D_\iota}$, I and $\boldsymbol{I}$, g and $\boldsymbol{g}$, R and $r_{\iota\to\iota\to o}$, and w and $\boldsymbol{w}$. $\Box$

Embedding HOML in HOL: Theoretical Results

Soundness and Completeness (and Cut-elimination)

$\models^{HOML}_{\text{Henkin}} s_o$ iff $Ax \models^{HOL}_{\text{Henkin}} valid\ s_{\iota\to o}$

Proof sketch (adapts [Benzmüller and Paulson, Logica Universalis, 2013]):
By contraposition it is sufficient to show

$$\not\models^{HOML}_{\text{Henkin}} s_o \text{ iff } Ax \not\models^{HOL}_{\text{Henkin}} valid\ s_{\iota\to o}$$

One easily gets the proof by choosing the obvious correspondences between D and $\boldsymbol{D}$, W and $\boldsymbol{D_\iota}$, I and $\boldsymbol{I}$, g and $\boldsymbol{g}$, R and $r_{\iota\to\iota\to o}$, and w and $\boldsymbol{w}$. □

Embedding of Other Logics in HOL: Theoretical Results

Soundness and Completeness (and Cut-elimination)

$$\models^{L} s_o \quad \text{iff} \quad \mathsf{Ax} \models^{HOL}_{\mathsf{Henkin}} \mathit{valid}\, s_{\iota\to o} \quad (\text{iff} \quad \mathsf{Ax} \vdash^{\mathsf{HOL}}_{\mathsf{cut\text{-}free}} \mathit{valid}\, \varphi_{\iota\to o})$$

Logic L:

- Higher-order Modal Logics
- First-order Multimodal Logics
- Propositional Multimodal Logics
- Quantified Conditional Logics
- Propositional Conditional Logics
- Intuitionistic Logics
- Access Control Logics
- Logic Combinations
- . . . more is on the way . . . including:
  - Description Logics
  - Nominal Logics
  - Multivalued Logics (SIXTEEN)
  - Logics based on Neighborhood Semantics
  - (Mathematical) Fuzzy Logics
  - Paraconsistent Logics

Embedding HOML in HOL: TPTP THF

```
%----The base type $i (already built-in) stands here for worlds and
%----mu for individuals; $o (also built-in) is the type of Booleans
thf(mu_type,type,(mu:$tType)).
%----Reserved constant r for accessibility relation
thf(r,type,(r:$i>$i>$o)).
%----Modal logic operators not, or, and, implies, box, diamond
thf(mnot_type,type,(mnot:($i>$o)>$i>$o)).
thf(mnot,definition,(mnot = (^[A:$i>$o,W:$i]:~(A@W)))).
thf(mor_type,type,(mor:($i>$o)>($i>$o)>$i>$o)).
thf(mor,definition,(mor = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)|(Psi@W))))).
thf(mand_type,type,(mand:($i>$o)>($i>$o)>$i>$o)).
thf(mand,definition,(mand = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)&(Psi@W))))).
thf(mimplies_type,type,(mimplies:($i>$o)>($i>$o)>$i>$o)).
thf(mimplies,definition,(mimplies = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)&(Psi@W))))).
thf(mbox_type,type,(mbox:($i>$o)>$i>$o)).
thf(mbox,definition,(mbox = (^[A:$i>$o,W:$i]:![V:$i]:(~(r@W@V)|(A@V))))).
thf(mdia_type,type,(mdia:($i>$o)>$i>$o)).
thf(mdia,definition,(mdia = (^[A:$i>$o,W:$i]:?[V:$i]:((r@W@V)&(A@V))))).
%----Quantifiers (constant domains) for individuals and propositions
thf(mforall_ind_type,type,(mforall_ind:(mu>$i>$o)>$i>$o)).
thf(mforall_ind,definition,(mforall_ind = (^[A:mu>$i>$o,W:$i]:![X:mu]:(A@X@W)))).
thf(mforall_indset_type,type,(mforall_indset:((mu>$i>$o)>$i>$o)>$i>$o)).
thf(mforall_indset,definition,(mforall_indset = (^[A:(mu>$i>$o)>$i>$o,W:$i]:![X:mu>$i>$o]:(A@X@W)))).
thf(mexists_ind_type,type,(mexists_ind:(mu>$i>$o)>$i>$o)).
thf(mexists_ind,definition,(mexists_ind = (^[A:mu>$i>$o,W:$i]:?[X:mu]:(A@X@W)))).
thf(mexists_indset_type,type,(mexists_indset:((mu>$i>$o)>$i>$o)>$i>$o)).
thf(mexists_indset,definition,(mexists_indset = (^[A:(mu>$i>$o)>$i>$o,W:$i]:?[X:mu>$i>$o]:(A@X@W)))).
%----Definition of validity (grounding of lifted modal formulas)
thf(v_type,type,(v:($i>$o)>$o)).
thf(mvalid,definition,(v = (^[A:$i>$o]:![W:$i]:(A@W)))).
%----Properties of accessibility relations: symmetry
thf(msymmetric_type,type,(msymmetric:($i>$i>$o)>$o)).
thf(msymmetric,definition,(msymmetric = (^[R:$i>$i>$o]:![S:$i,T:$i]:((R@S@T)=>(R@T@S))))).
%----Here we work with logic KB, i.e., we postulate symmetry for r
thf(sym,axiom,(msymmetric@r)).
```

Reading on THF0 syntax: [SutcliffeBenzmüller, J.Formalized Reasoning, 2010]

Proof Automation with LEO-II

```
>
>
> Beweise-mit-Leo2 Notwendigerweise-existiert-Gott.p

Leo-II tries to prove
=====================

  Goedel's Theorem T3: "Necessarily, God exists"
  thf(thmT3,conjecture,
        ( v
        @ ( mbox
          @ ( mexists_ind
            @ ^ [X: mu] :
                ( g @ X ) ) ) )).

  Assumptions: D1, C, T2, D3, A5

 . searching for proof ..

 *******************
 *   Proof found   *
 *******************
 % SZS status Theorem for Notwendigerweise-existiert-Gott.p

 . generating proof object
```

Provers can be called remotely in Miami — no local installation needed!

Download our experiments from

`https://github.com/FormalTheology/GoedelGod/tree/master/Formalizations/THF`

Embedding HOML in HOL: Evaluation — How good is the approach?

- There are no other provers for HOML
- There are some provers for first-order modal logic (FML)
- Comparative evaluation done in 2014 for the proof problems in the QMLTP (v1.1) library [Otten and Raths, `http://www.iltp.de/qmltp/`]
- This library contains 580 FML problems (in fact, 5 x 3 x 580 = 8700 problems).
- Metaprover HOL-P: sequentially schedules LEO-II—1.6.2, Satallax—2.7, Isabelle—2013, Nitrox—2013, agsyHOl—1.0
- Timeout for each HOL prover 120sec of CPU time (HOL-P: 600sec)
- Timeout for competitor systems: 600sec

Embedding HOML in HOL: Evaluation — FML's (D — constant/varying/cumulative)

No. of solved problems in the QMLTP library

	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**

Embedding HOML in HOL: Evaluation — FML's (S4 — constant/varying/cumulative)

No. of solved problems in the QMLTP library

	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**

The Ontological Argument

A Long History of Ontological Arguments

pros and cons

… Anselm v. C. Gaunilo … Th. Aquinas …… Descartes Spinoza Leibniz … Hume Kant … Hegel … Frege … Hartshorne Malcolm Lewis Plantinga Gödel …

Anselm's notion of God (Proslogion, 1078):

"God is that, than which nothing greater can be conceived."

Gödel's notion of God:

"A God-like being possesses all 'positive' properties."

To show by logical, deductive reasoning:

"God exists."

$$\exists x G(x)$$

A Long History of Ontological Arguments

pros and cons

... Anselm v. C. Gaunilo ... Th. Aquinas ...... Descartes Spinoza Leibniz ... Hume Kant ... Hegel ... Frege ... Hartshorne Malcolm Lewis Plantinga Gödel ...

Anselm's notion of God (Proslogion, 1078):

"God is that, than which nothing greater can be conceived."

Gödel's notion of God:

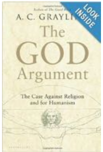
"A God-like being possesses all 'positive' properties."

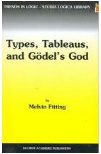
To show by logical, deductive reasoning:

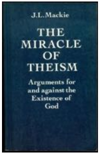
"Necessarily, God exists."

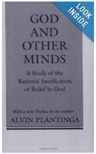
$$\Box \exists x G(x)$$

The Ontological Proof Today









See also our collection of recent papers:

`https://github.com/FormalTheology/GoedelGod/tree/master/Literature`

Various Different Interests in Ontological Arguments

- Philosophical: Boundaries of Metaphysics & Epistemology
- Theistic: Successful argument could convince atheists?
- Ours: Can computers (theorem provers) be used ...

  ... to formalize the definitions, axioms and theorems?
  ... to verify/falsify the arguments step-by-step?
  ... to automate (sub-)arguments?
  ... to discover new philosophical knowledge?

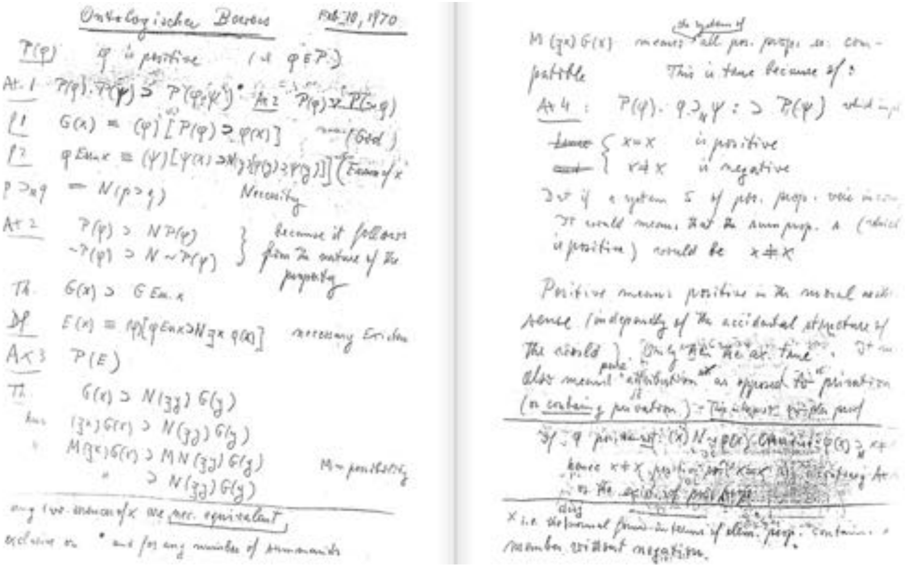
Vision of Leibniz (1646–1716): *Calculemus!*

Quo facto, quando orientur controversiae, non magis disputatione opus erit inter duos philosophos, quam inter duos Computistas. Sufficiet enim calamos in manus sumere sedereque ad abacos, et sibi mutuo . . . dicere: calculemus. (Leibniz, 1684)

If controversies were to arise, there would be no more need of disputation between two philosophers than between two accountants. For it would suffice to take their pencils in their hands, to sit down to their slates, and to say to each other . . . : Let us calculate.
(Translation by Russell)

Required:
characteristica universalis and **calculus ratiocinator**

Gödel's Manuscript: 1930's, 1941, 1946-1955, 1970



Ontologischer Beweis Feb. 10, 1970

$P(\varphi)$ φ is positive (or $\varphi \in P$)

Ax. 1 $P(\varphi).P(\psi) \supset P(\varphi.\psi)$ Ax 2 $P(\varphi) \vee P(\sim\varphi)$

P1 $G(x) \equiv (\varphi)[P(\varphi) \supset \varphi(x)]$ (God)

P2 $\varphi \operatorname{Ess} x \equiv (\psi)[\psi(x) \supset N(y)[\varphi(y) \supset \psi(y)]]$ (Essence of x)

$p \supset_N q = N(p \supset q)$ Necessity

Ax 2 $P(\varphi) \supset NP(\varphi)$, $\sim P(\varphi) \supset N\sim P(\varphi)$ } because it follows from the nature of the property

Th. $G(x) \supset G \operatorname{Ess.} x$

Df. $E(x) \equiv (\varphi)[\varphi \operatorname{Ess} x \supset N \exists x\, \varphi(x)]$ necessary Existence

Ax 3 $P(E)$

Th. $G(x) \supset N(\exists y)G(y)$

hence $(\exists x)G(x) \supset N(\exists y)G(y)$

" $M(\exists x)G(x) \supset MN(\exists y)G(y)$

" $\supset N(\exists y)G(y)$ M = possibility

any two essences of x are nec. equivalent

exclusive or * and for any number of summands

$M(\exists x)G(x)$ means the system of all pos. props. is compatible. This is true because of:

Ax 4: $P(\varphi).\varphi \supset_N \psi :\supset P(\psi)$ which impl

$x = x$ is positive

$x \neq x$ is negative

But if a system S of pos. props. were incons. it would mean that the sum prop. s (which is positive) would be $x \neq x$

Positive means positive in the moral aesthetic sense (independently of the accidental structure of the world). Only then the ax. true. It may also mean pure "attribution" as opposed to "privation" (or containing privation). This interpret. simpler proof

If φ pos. not: $(x) N\sim\varphi(x)$. Otherwise: $\varphi(x) \supset_N x \neq x$

hence $x \neq x$ positive not $x = x$ negative ...

X i.e. the normal form in terms of elem. props. contains a member without negation.

Proof Overview

T3: $\Box\exists x.G(x)$

Proof Overview

$$\frac{\textbf{C1: } \Diamond \exists z. G(z)}{\textbf{T3: } \Box \exists x. G(x)}$$

Proof Overview

$$\frac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

L2: $\Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)$

$$\frac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\dfrac{\dfrac{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \to \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\dfrac{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x) \qquad \dfrac{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\dfrac{\dfrac{\textbf{L1: } \exists z.G(z) \to \Box\exists x.G(x)}{\Diamond\exists z.G(z) \to \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall \xi.[\Diamond\Box\xi \to \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1: } G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\dfrac{\dfrac{\textbf{L1: } \exists z.G(z) \rightarrow \Box\exists x.G(x)}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1:}\ G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D3*:}\ E(x) \equiv \Box\exists y.G(y)$$

$$\dfrac{\dfrac{\dfrac{P(E)}{\textbf{L1:}\ \exists z.G(z) \to \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \to \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \to \Box\xi]}}{\textbf{L2:}\ \Diamond\exists z.G(z) \to \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1:}\ \Diamond\exists z.G(z) \qquad \textbf{L2:}\ \Diamond\exists z.G(z) \to \Box\exists x.G(x)}{\textbf{T3:}\ \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1: } G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$\textbf{D3*: } E(x) \equiv \Box\exists y.G(y)$ (cheating!)

$$\dfrac{\dfrac{\dfrac{P(E)}{\textbf{L1: } \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1:}\ G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D3*:}\ E(x) \equiv \Box\exists y.G(y) \qquad\qquad \textbf{D3:}\ E(x) \equiv \forall\varphi\,.[\varphi\ ess\ x \rightarrow \Box\exists y.\varphi(y)]$$

$$\dfrac{\dfrac{\dfrac{\textbf{T2:}\ \forall y\,.[G(y) \rightarrow G\ ess\ y] \qquad\qquad P(E)}{\textbf{L1:}\ \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi\,.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1:}\ \Diamond\exists z.G(z) \qquad \textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3:}\ \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1:}\ G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D3*:}\ E(x) \equiv \Box\exists y.G(y) \qquad\qquad \textbf{D3:}\ E(x) \equiv \forall\varphi\,.[\varphi\ ess\ x \rightarrow \Box\exists y.\varphi(y)]$$

$$\dfrac{\dfrac{\dfrac{\textbf{T2:}\ \forall y\,.[G(y) \rightarrow G\ ess\ y] \qquad \dfrac{\textbf{A5}}{P(E)}}{\textbf{L1:}\ \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi\,.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1:}\ \Diamond\exists z.G(z) \qquad \textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3:}\ \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1:}\ G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\color{red}{\textbf{D3*:}\ E(x) \equiv \Box\exists y.G(y)} \qquad\qquad \textbf{D3:}\ E(x) \equiv \forall\varphi\,.[\varphi\ ess\ x \rightarrow \Box\exists y.\varphi(y)]$$

$$\dfrac{\dfrac{\dfrac{\textbf{T2:}\ \forall y\,.[G(y) \rightarrow G\ ess\ y] \qquad \dfrac{\textbf{A5}}{P(E)}}{\textbf{L1:}\ \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi\,.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1:}\ \Diamond\exists z.G(z) \qquad \textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3:}\ \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1: } G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D2: } \varphi\ ess\ x \equiv \varphi(x) \wedge \forall\psi.(\psi(x) \to \Box\forall x.(\varphi(x) \to \psi(x)))$$

$$\textbf{D3*: } E(x) \equiv \Box\exists y.G(y) \qquad\qquad \textbf{D3: } E(x) \equiv \forall\varphi.[\varphi\ ess\ x \to \Box\exists y.\varphi(y)]$$

$$\dfrac{\dfrac{\dfrac{\overset{\textbf{A1b}}{\forall\varphi.[\neg P(\varphi) \to P(\neg\varphi)]} \qquad \overset{\textbf{A4}}{\forall\varphi.[P(\varphi) \longrightarrow \Box P(\varphi)]}}{\textbf{T2: } \forall y.[G(y) \to G\ ess\ y]} \qquad \overset{\textbf{A5}}{P(E)}}{\dfrac{\textbf{L1: } \exists z.G(z) \to \Box\exists x.G(x)}{\Diamond\exists z.G(z) \to \Diamond\Box\exists x.G(x)}} \qquad \overset{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \to \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1: } G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D2: } \varphi \; ess \; x \equiv \varphi(x) \wedge \forall\psi.(\psi(x) \rightarrow \Box\forall x.(\varphi(x) \rightarrow \psi(x)))$$

$$\textbf{D3*: } E(x) \equiv \Box\exists y.G(y) \qquad \textbf{D3: } E(x) \equiv \forall\varphi.[\varphi \; ess \; x \rightarrow \Box\exists y.\varphi(y)]$$

$$\textbf{C1: } \Diamond\exists z.G(z)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{\textbf{A1b}}{\forall\varphi.[\neg P(\varphi) \rightarrow P(\neg\varphi)]} \qquad \overset{\textbf{A4}}{\forall\varphi.[P(\varphi) \longrightarrow \Box P(\varphi)]}}{\textbf{T2: } \forall y.[G(y) \rightarrow G \; ess \; y]} \qquad \overset{\textbf{A5}}{P(E)}}{\textbf{L1: } \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \overset{\textbf{S5}}{\forall\xi.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1:}\ G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D2:}\ \varphi\ ess\ x \equiv \varphi(x) \wedge \forall\psi\,.(\psi(x) \rightarrow \Box\forall x\,.(\varphi(x) \rightarrow \psi(x)))$$

$$\textbf{D3*:}\ E(x) \equiv \Box\exists y.G(y) \qquad\qquad \textbf{D3:}\ E(x) \equiv \forall\varphi\,.[\varphi\ ess\ x \rightarrow \Box\exists y.\varphi(y)]$$

$$\frac{P(G)}{\textbf{C1:}\ \Diamond\exists z.G(z)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{\textbf{A1b}}{\forall\varphi\,.[\neg P(\varphi) \rightarrow P(\neg\varphi)]} \qquad \overset{\textbf{A4}}{\forall\varphi\,.[P(\varphi) \longrightarrow \Box\, P(\varphi)]}}{\textbf{T2:}\ \forall y\,.[G(y) \rightarrow G\ ess\ y]} \qquad \overset{\textbf{A5}}{P(E)}}{\textbf{L1:}\ \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \overset{\textbf{S5}}{\forall\xi\,.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\frac{\textbf{C1:}\ \Diamond\exists z.G(z) \qquad \textbf{L2:}\ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3:}\ \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1: } G(x) \equiv \forall \varphi . [P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D2: } \varphi \; ess \; x \equiv \varphi(x) \wedge \forall \psi \,.(\psi(x) \rightarrow \Box \forall x \,.(\varphi(x) \rightarrow \psi(x)))$$

$$\textbf{D3*: } E(x) \equiv \Box \exists y . G(y) \qquad \textbf{D3: } E(x) \equiv \forall \varphi \,.[\varphi \; ess \; x \rightarrow \Box \exists y . \varphi(y)]$$

$$\dfrac{\dfrac{\textbf{A3}}{P(G)}}{\textbf{C1: } \Diamond \exists z . G(z)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\textbf{A1b}}{\forall \varphi \,.[\neg P(\varphi) \rightarrow P(\neg \varphi)]} \quad \dfrac{\textbf{A4}}{\forall \varphi \,.[P(\varphi) \longrightarrow \Box \; P(\varphi)]}}{\textbf{T2: } \forall y \,.[G(y) \rightarrow G \; ess \; y]} \quad \dfrac{\textbf{A5}}{P(E)}}{\dfrac{\textbf{L1: } \exists z . G(z) \rightarrow \Box \exists x . G(x)}{\Diamond \exists z . G(z) \rightarrow \Diamond \Box \exists x . G(x)}} \quad \dfrac{\textbf{S5}}{\forall \xi \,.[\Diamond \Box \xi \rightarrow \Box \xi]}}{\textbf{L2: } \Diamond \exists z . G(z) \rightarrow \Box \exists x . G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond \exists z . G(z) \qquad \textbf{L2: } \Diamond \exists z . G(z) \rightarrow \Box \exists x . G(x)}{\textbf{T3: } \Box \exists x . G(x)}$$

Proof Overview

$$\textbf{D1:}\ G(x) \equiv \forall \varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D2:}\ \varphi\ ess\ x \equiv \varphi(x) \wedge \forall \psi\,.(\psi(x) \rightarrow \Box \forall x\,.(\varphi(x) \rightarrow \psi(x)))$$

$$\textbf{D3*:}\ E(x) \equiv \Box \exists y.G(y) \qquad\qquad \textbf{D3:}\ E(x) \equiv \forall \varphi\,.[\varphi\ ess\ x \rightarrow \Box \exists y.\varphi(y)]$$

$$\dfrac{\dfrac{\textbf{A3}}{P(G)} \qquad \textbf{T1:}\ \forall \varphi\,.[P(\varphi) \rightarrow \Diamond \exists x.\varphi(x)]}{\textbf{C1:}\ \Diamond \exists z.G(z)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\textbf{A1b}}{\forall \varphi\,.[\neg P(\varphi) \rightarrow P(\neg \varphi)]} \qquad \dfrac{\textbf{A4}}{\forall \varphi\,.[P(\varphi) \longrightarrow \Box\ P(\varphi)]}}{\textbf{T2:}\ \forall y\,.[G(y) \rightarrow G\ ess\ y]} \qquad \dfrac{\textbf{A5}}{P(E)}}{\textbf{L1:}\ \exists z.G(z) \rightarrow \Box \exists x.G(x)}}{\Diamond \exists z.G(z) \rightarrow \Diamond \Box \exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall \xi\,.[\Diamond \Box \xi \rightarrow \Box \xi]}}{\textbf{L2:}\ \Diamond \exists z.G(z) \rightarrow \Box \exists x.G(x)}$$

$$\dfrac{\textbf{C1:}\ \Diamond \exists z.G(z) \qquad \textbf{L2:}\ \Diamond \exists z.G(z) \rightarrow \Box \exists x.G(x)}{\textbf{T3:}\ \Box \exists x.G(x)}$$

Proof Overview

D1: $G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$

D2: $\varphi\ ess\ x \equiv \varphi(x) \wedge \forall\psi\,.(\psi(x) \rightarrow \Box\forall x\,.(\varphi(x) \rightarrow \psi(x)))$

D3*: $E(x) \equiv \Box\exists y.G(y)$ **D3:** $E(x) \equiv \forall\varphi\,.[\varphi\ ess\ x \rightarrow \Box\exists y.\varphi(y)]$

$$\dfrac{\dfrac{\textbf{A3}}{P(G)} \qquad \dfrac{\dfrac{\textbf{A2}}{\forall\varphi\,.\forall\psi\,.[(P(\varphi) \wedge \Box\forall x\,.[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]} \qquad \dfrac{\textbf{A1a}}{\forall\varphi\,.[P(\neg\varphi) \rightarrow \neg P(\varphi)]}}{\textbf{T1: } \forall\varphi\,.[P(\varphi) \rightarrow \Diamond\exists x.\varphi(x)]}}{\textbf{C1: } \Diamond\exists z.G(z)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\textbf{A1b}}{\forall\varphi\,.[\neg P(\varphi) \rightarrow P(\neg\varphi)]} \qquad \dfrac{\textbf{A4}}{\forall\varphi\,.[P(\varphi) \longrightarrow \Box\ P(\varphi)]}}{\textbf{T2: } \forall y\,.[G(y) \rightarrow G\ ess\ y]} \qquad \dfrac{\textbf{A5}}{P(E)}}{\textbf{L1: } \exists z.G(z) \rightarrow \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \rightarrow \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi\,.[\Diamond\Box\xi \rightarrow \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Proof Overview

$$\textbf{D1: } G(x) \equiv \forall\varphi.[P(\varphi) \longrightarrow \varphi(x)]$$

$$\textbf{D2: } \varphi\ ess\ x \equiv \varphi(x) \land \forall\psi\,.(\psi(x) \to \Box\forall x\,.(\varphi(x) \to \psi(x)))$$

$$\textbf{D3: } NE(x) \equiv \forall\varphi\,.[\varphi\ ess\ x \to \Box\exists y.\varphi(y)]$$

$$\dfrac{\dfrac{\textbf{A3}}{P(G)} \qquad \dfrac{\dfrac{\textbf{A2}}{\forall\varphi\,.\forall\psi\,.[(P(\varphi) \land \Box\forall x\,.[\varphi(x) \to \psi(x)]) \to P(\psi)]} \qquad \dfrac{\textbf{A1a}}{\forall\varphi\,.[P(\neg\varphi) \to \neg P(\varphi)]}}{\textbf{T1: } \forall\varphi\,.[P(\varphi) \to \Diamond\exists x.\varphi(x)]}}{\textbf{C1: } \Diamond\exists z.G(z)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\textbf{A1b}}{\forall\varphi\,.[\neg P(\varphi) \to P(\neg\varphi)]} \qquad \dfrac{\textbf{A4}}{\forall\varphi\,.[P(\varphi) \longrightarrow \Box\, P(\varphi)]}}{\textbf{T2: } \forall y\,.[G(y) \to G\ ess\ y]} \qquad \dfrac{\textbf{A5}}{P(NE)}}{\textbf{L1: } \exists z.G(z) \to \Box\exists x.G(x)}}{\Diamond\exists z.G(z) \to \Diamond\Box\exists x.G(x)} \qquad \dfrac{\textbf{S5}}{\forall\xi\,.[\Diamond\Box\xi \to \Box\xi]}}{\textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}$$

$$\dfrac{\textbf{C1: } \Diamond\exists z.G(z) \qquad \textbf{L2: } \Diamond\exists z.G(z) \to \Box\exists x.G(x)}{\textbf{T3: } \Box\exists x.G(x)}$$

Scott's Version of Gödel's Axioms, Definitions and Theorems

Axiom A1 Either a property or its negation is positive, but not both: $\forall\varphi[P(\neg\varphi) \leftrightarrow \neg P(\varphi)]$

Axiom A2 A property necessarily implied by a positive property is positive:
$$\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$$

Thm. T1 Positive properties are possibly exemplified: $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

Def. D1 A God-like being possesses all positive properties: $G(x) \leftrightarrow \forall\varphi[P(\varphi) \rightarrow \varphi(x)]$

Axiom A3 The property of being God-like is positive: $P(G)$

Cor. C Possibly, God exists: $\Diamond\exists x G(x)$

Axiom A4 Positive properties are necessarily positive: $\forall\varphi[P(\varphi) \rightarrow \Box P(\varphi)]$

Def. D2 An essence of an individual is a property possessed by it and necessarily implying any of its properties: $\varphi\ ess\ x \leftrightarrow \varphi(x) \wedge \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Thm. T2 Being God-like is an essence of any God-like being: $\forall x[G(x) \rightarrow G\ ess\ x]$

Def. D3 Necessary existence of an individual is the necessary exemplification of all its essences: $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \rightarrow \Box\exists y\varphi(y)]$

Axiom A5 Necessary existence is a positive property: $P(NE)$

Thm. T3 Necessarily, God exists: $\Box\exists x G(x)$

Scott's Version of Gödel's Axioms, Definitions and Theorems

Axiom A1 Either a property or its negation is positive, but not both: $\forall\varphi[P(\neg\varphi) \leftrightarrow \neg P(\varphi)]$

Axiom A2 A property necessarily implied by a positive property is positive: $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Thm. T1 Positive properties are possibly exemplified: $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

Def. D1 A <u>God-like</u> being possesses all positive properties: $G(x) \leftrightarrow \forall\varphi[P(\varphi) \rightarrow \varphi(x)]$

Axiom A3 The property of being God-like is positive: $P(G)$

Cor. C Possibly, God exists: $\Diamond\exists x G(x)$

Axiom A4 Positive properties are necessarily positive: $\forall\varphi[P(\varphi) \rightarrow \Box P(\varphi)]$

Def. D2 An <u>essence</u> of an individual is a property possessed by it and necessarily implying any of its properties: $\varphi \; ess \; x \leftrightarrow \boxed{\varphi(x) \wedge} \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Thm. T2 Being God-like is an essence of any God-like being: $\forall x[G(x) \rightarrow G \; ess \; x]$

Def. D3 <u>Necessary existence</u> of an individual is the necessary exemplification of all its essences: $NE(x) \leftrightarrow \forall\varphi[\varphi \; ess \; x \rightarrow \Box\exists y\varphi(y)]$

Axiom A5 Necessary existence is a positive property: $P(NE)$

Thm. T3 Necessarily, God exists: $\Box\exists x G(x)$

Difference to Gödel (who omits this conjunct)

Scott's Version of Gödel's Axioms, Definitions and Theorems

Axiom A1 Either a property or its negation is positive, but not both: $\forall\varphi[P(\neg\varphi) \leftrightarrow \neg P(\varphi)]$

Axiom A2 A property necessarily implied by a positive property is positive: $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Thm. T1 Positive properties are possibly exemplified: $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

Def. D1 A God-like being possesses all positive properties: $G(x) \leftrightarrow \forall\varphi[P(\varphi) \rightarrow \varphi(x)]$

Axiom A3 The property of being God-like is positive: $P(G)$

Cor. C Possibly, God exists: $\Diamond\exists x G(x)$

Axiom A4 Positive properties are necessarily positive: $\forall\varphi[P(\varphi) \rightarrow \Box P(\varphi)]$

Def. D2 An essence of an individual is a property possessed by it and necessarily implying any of its properties: $\varphi\ ess\ x \leftrightarrow \varphi(x) \wedge \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Thm. T2 Being God-like is an essence of any God-like being: $\forall x[G(x) \rightarrow G\ ess\ x]$

Def. D3 Necessary existence of an individual is the necessary exemplification of all its essences: $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \rightarrow \Box\exists y\varphi(y)]$

Axiom A5 Necessary existence is a positive property: $P(NE)$

Thm. T3 Necessarily, God exists: $\Box\exists x G(x)$

Modal operators are used

Scott's Version of Gödel's Axioms, Definitions and Theorems

Axiom A1 Either a property or its negation is positive, but not both: $\forall\varphi[P(\neg\varphi) \leftrightarrow \neg P(\varphi)]$

Axiom A2 A property necessarily implied by a positive property is positive:
$\boxed{\forall\varphi\forall\psi}[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Thm. T1 Positive properties are possibly exemplified: $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

Def. D1 A God-like being possesses all positive properties: $G(x) \leftrightarrow \boxed{\forall\varphi}[P(\varphi) \rightarrow \varphi(x)]$

Axiom A3 The property of being God-like is positive: $P(G)$

Cor. C Possibly, God exists: $\Diamond\exists x G(x)$

Axiom A4 Positive properties are necessarily positive: $\forall\varphi[P(\varphi) \rightarrow \Box P(\varphi)]$

Def. D2 An essence of an individual is a property possessed by it and necessarily implying any of its properties: $\varphi\ ess\ x \leftrightarrow \varphi(x) \wedge \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Thm. T2 Being God-like is an essence of any God-like being: $\forall x[G(x) \rightarrow G\ ess\ x]$

Def. D3 Necessary existence of an individual is the necessary exemplification of all its essences: $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \rightarrow \Box\exists y\varphi(y)]$

Axiom A5 Necessary existence is a positive property: $P(NE)$

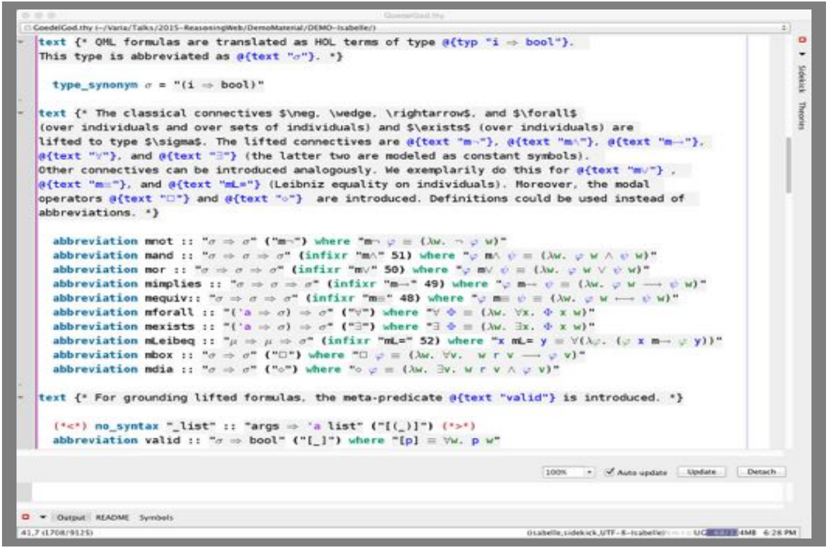
Thm. T3 Necessarily, God exists: $\Box\exists x G(x)$

second-order quantifiers

The Ontological Argument in TPTP THF0

```
%------------------------------------------------------------------------------
%----Axioms for Quantified Modal Logic KB.
include('Quantified_KB.ax').
%------------------------------------------------------------------------------
%----constant symbol for positive (p), God-like (g), essence (ess), necessary existence (ne)
thf(p_tp,type,(p:(mu>$i>$o)>$i>$o)).
thf(g_tp,type,(g:mu>$i>$o)).
thf(ess_tp,type,(ess:(mu>$i>$o)>mu>$i>$o)).
thf(ne_tp,type,(ne:mu>$i>$o)).
%----D1:A God-like being possesses all positive properties.
thf(defD1,definition,(g = (^[X:mu]:(mforall_indset@^[Phi:mu>$i>$o]:(mimplies@(p@Phi)@(Phi@X)))))).
%----C: Possibly, God exists. (Proved in C.p)
thf(corC,axiom,(v@(mdia@(mexists_ind@^[X:mu]:(g@X))))).
%----T2: Being God-like is an essence of any God-like being. (Proved in T2.p)
thf(thmT2,axiom,(v@(mforall_ind@^[X:mu]:(mimplies@(g@X)@(ess@g@X))))).
%----D3: Necessary existence of an individual is the necessary exemplification of all its essences
thf(defD3,definition,(ne = (^[X:mu]:(mforall_indset@^[Phi:mu>$i>$o]:
     (mimplies@(ess@Phi@X)@(mbox@(mexists_ind@^[Y:mu]:(Phi@Y)))))))).
%----A5:Necessary existence is positive.
thf(axA5,axiom,(v@(p@ne))).
%----T3: Necessarily God exists.
thf(thmT3,conjecture,(v@(mbox@(mexists_ind@^[X:mu]:(g@X))))).
```

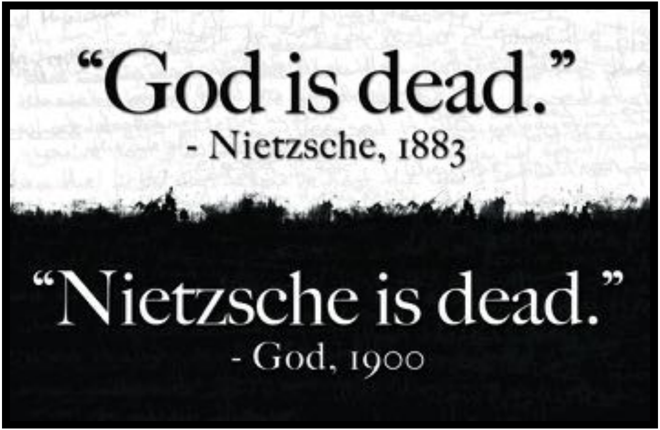
The Ontological Argument in Isabelle/HOL




```
text {* QML formulas are translated as HOL terms of type @{typ "i ⇒ bool"}.
This type is abbreviated as @{text "σ"}. *}

  type_synonym σ = "(i ⇒ bool)"

text {* The classical connectives $\neg, \wedge, \rightarrow$, and $\forall$
(over individuals and over sets of individuals) and $\exists$ (over individuals) are
lifted to type $\sigma$. The lifted connectives are @{text "m¬"}, @{text "m∧"}, @{text "m→"},
@{text "∀"}, and @{text "∃"} (the latter two are modeled as constant symbols).
Other connectives can be introduced analogously. We exemplarily do this for @{text "m∨"} ,
@{text "m≡"}, and @{text "mL="} (Leibniz equality on individuals). Moreover, the modal
operators @{text "□"} and @{text "◇"}  are introduced. Definitions could be used instead of
abbreviations. *}

  abbreviation mnot :: "σ ⇒ σ" ("m¬") where "m¬ φ ≡ (λw. ¬ φ w)"
  abbreviation mand :: "σ ⇒ σ ⇒ σ" (infixr "m∧" 51) where "φ m∧ ψ ≡ (λw. φ w ∧ ψ w)"
  abbreviation mor :: "σ ⇒ σ ⇒ σ" (infixr "m∨" 50) where "φ m∨ ψ ≡ (λw. φ w ∨ ψ w)"
  abbreviation mimplies :: "σ ⇒ σ ⇒ σ" (infixr "m→" 49) where "φ m→ ψ ≡ (λw. φ w ⟶ ψ w)"
  abbreviation mequiv:: "σ ⇒ σ ⇒ σ" (infixr "m≡" 48) where "φ m≡ ψ ≡ (λw. φ w ⟷ ψ w)"
  abbreviation mforall :: "('a ⇒ σ) ⇒ σ" ("∀") where "∀ Φ ≡ (λw. ∀x. Φ x w)"
  abbreviation mexists :: "('a ⇒ σ) ⇒ σ" ("∃") where "∃ Φ ≡ (λw. ∃x. Φ x w)"
  abbreviation mLeibeq :: "μ ⇒ μ ⇒ σ" (infixr "mL=" 52) where "x mL= y ≡ ∀(λφ. (φ x m→ φ y))"
  abbreviation mbox :: "σ ⇒ σ" ("□") where "□ φ ≡ (λw. ∀v.  w r v ⟶ φ v)"
  abbreviation mdia :: "σ ⇒ σ" ("◇") where "◇ φ ≡ (λw. ∃v. w r v ∧ φ v)"

text {* For grounding lifted formulas, the meta-predicate @{text "valid"} is introduced. *}

  (*<*) no_syntax "_list" :: "args ⇒ 'a list" ("[(_)]") (*>*)
  abbreviation valid :: "σ ⇒ bool" ("[_]") where "[p] ≡ ∀w. p w"
```

See verifiable Isabelle/HOL document (Archive of Formal Proofs) at:

`http://afp.sourceforge.net/entries/GoedelGod.shtml`



Experimental Results and Philosophical Discoveries

[BenzmüllerWoltzenlogelPaleo, ECAI, 2014]

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot\neg(\phi X)) \dot\equiv \dot\neg(p\phi)]$						
A2	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall\psi_{\mu\to\sigma}.\, (p_{(\mu\to\sigma)\to\sigma}\phi \,\dot\wedge\, \dot\Box\dot\forall X_\mu.\, (\phi X \dot\supset \psi X)) \dot\supset p\psi]$						
T1	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Diamond\dot\exists X_\mu.\, \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma}\, g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu.\, g_{\mu\to\sigma} X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Box p\phi]$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \phi X \,\dot\wedge\, \dot\forall\psi_{\mu\to\sigma}.\, (\psi X \dot\supset \dot\Box\dot\forall Y_\mu.\, (\phi Y \dot\supset \psi Y))$						
T2	$[\dot\forall X_\mu.\, g_{\mu\to\sigma} X \dot\supset (\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma}\, g X)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, (\mathrm{ess}\,\phi X \dot\supset \dot\Box\dot\exists Y_\mu.\, \phi Y)$						
A5	$[p_{(\mu\to\sigma)\to\sigma}\, \mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot\Box\dot\exists X_\mu.\, g_{\mu\to\sigma} X]$	D1, C, T2, D3, A5	K	CSA	—/—	—/—	3.8/6.2
		A1, A2, D1, A3, A4, D2, D3, A5	K	CSA	—/—	—/—	8.2/7.5
		D1, C, T2, D3, A5	KB	THM	0.0/0.1	0.1/5.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
MC	$[s_\sigma \dot\supset \dot\Box s_\sigma]$	D2, T2, T3	KB	THM	17.9/—	3.3/3.2	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
FG	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall X_\mu.\, (g_{\mu\to\sigma} X \dot\supset (\dot\neg(p_{(\mu\to\sigma)\to\sigma}\phi) \dot\supset \dot\neg(\phi X)))]$	A1, D1	KB	THM	16.5/—	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	12.8/15.1	0.0/5.4	—/—
MT	$[\dot\forall X_\mu.\, \dot\forall Y_\mu.\, (g_{\mu\to\sigma} X \dot\supset (g_{\mu\to\sigma} Y \dot\supset X \dot= Y))]$	D1, FG	KB	THM	—/—	0.0/3.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
CO	$\emptyset$ (no goal, check for consistency)	A1, A2, D1, A3, A4, D2, D3, A5	KB	SAT	—/—	—/—	7.3/7.4
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \dot\forall\psi_{\mu\to\sigma}.\, (\psi X \dot\supset \dot\Box\dot\forall Y_\mu.\, (\phi Y \dot\supset \psi Y))$						
CO'	$\emptyset$ (no goal, check for consistency)	A1(⊃), A2, D2', D3, A5	KB	UNS	7.5/7.8	—/—	—/—
		A1, A2, D1, A3, A4, D2', D3, A5	KB	UNS	—/—	—/—	—/—

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot{\neg}(\phi X)) \dot{\equiv} \dot{\neg}(p\phi)]$						
A2	$[\dot{\forall}\phi_{\mu\to\sigma}.\, \dot{\forall}\psi_{\mu\to\sigma}.\, (p_{(\mu\to\sigma)\to\sigma}\phi \,\dot{\wedge}\, \dot{\Box}\dot{\forall}X_\mu.\, (\phi X \dot{\supset} \psi X)) \dot{\supset} p\psi]$						
T1	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Diamond}\dot{\exists}X_\mu.\, \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma}\, g_{\mu\to\sigma}]$						
C	$[\dot{\Diamond}\dot{\exists}X_\mu.\, g_{\mu\to\sigma}X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Box}p\phi]$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \phi X \,\dot{\wedge}\, \dot{\forall}\psi_{\mu\to\sigma}.\, (\psi X \dot{\supset} \dot{\Box}\dot{\forall}Y_\mu.\, (\phi Y \dot{\supset} \psi Y))$						
T2	$[\dot{\forall}X_\mu.\, g_{\mu\to\sigma}X \dot{\supset} (\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma}\, g X)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu.\, \dot{\forall}\phi_{\mu\to\sigma}.\, (\mathrm{ess}\,\phi X \dot{\supset} \dot{\Box}\dot{\exists}Y_\mu.\, \phi Y)$						
A5	$[p_{(\mu\to\sigma)\to\sigma}\, \mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot{\Box}\dot{\exists}X_\mu.\, g_{\mu\to\sigma}X]$	D1, C, T2, D3, A5	K	CSA	—/—	—/—	3.8/6.2
		A1, A2, D1, A3, A4, D2, D3, A5	K	CSA	—/—	—/—	8.2/7.5
		D1, C, T2, D3, A5	KB	THM	0.0/0.1	0.1/5.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
MC	$[s_\sigma \dot{\supset} \dot{\Box}s_\sigma]$	D2, T2, T3	KB	THM	17.9/—	3.3/3.2	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
FG	$[\dot{\forall}\phi_{\mu\to\sigma}.\, \dot{\forall}X_\mu.\, (g_{\mu\to\sigma}X \dot{\supset} (\dot{\neg}(p_{(\mu\to\sigma)\to\sigma}\phi) \dot{\supset} \dot{\neg}(\phi X)))]$	A1, D1	KB	THM	16.5/—	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	12.8/15.1	0.0/5.4	—/—
MT	$[\dot{\forall}X_\mu.\, \dot{\forall}Y_\mu.\, (g_{\mu\to\sigma}X \dot{\supset} (g_{\mu\to\sigma}Y \dot{\supset} X \dot{=} Y))]$	D1, FG	KB	THM	—/—	0.0/3.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
CO	$\emptyset$ (no goal, check for consistency)	A1, A2, D1, A3, A4, D2, D3, A5	KB	SAT	—/—	—/—	7.3/7.4
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \dot{\forall}\psi_{\mu\to\sigma}.\, (\psi X \dot{\supset} \dot{\Box}\dot{\forall}Y_\mu.\, (\phi Y \dot{\supset} \psi Y))$						
CO'	$\emptyset$ (no goal, check for consistency)	A1(⊃), A2, D2', D3, A5	KB	UNS	7.5/7.8	—/—	—/—
		A1, A2, D1, A3, A4, D2', D3, A5	KB	UNS	—/—	—/—	—/—

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot{\neg}(\phi X)) \dot{\equiv} \dot{\neg}(p\phi)]$						
A2	$[\dot{\forall}\phi_{\mu\to\sigma}.\, \dot{\forall}\psi_{\mu\to\sigma}.\, (p_{(\mu\to\sigma)\to\sigma}\phi \dot{\wedge} \dot{\Box}\dot{\forall}X_\mu.\, (\phi X \dot{\supset} \psi X)) \dot{\supset} p\psi]$						
T1	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Diamond}\dot{\exists}X_\mu.\, \phi X]$	A1($\supset$), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma} g_{\mu\to\sigma}]$						
C	$[\dot{\Diamond}\dot{\exists}X_\mu.\, g_{\mu\to\sigma}X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Box}p\phi]$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \phi X \dot{\wedge} \dot{\forall}\psi_{\mu\to\sigma}.\, (\psi X \dot{\supset} \dot{\Box}\dot{\forall}Y_\mu.\, (\phi Y \dot{\supset} \psi Y))$						
T2	$[\dot{\forall}X_\mu.\, g_{\mu\to\sigma}X \dot{\supset} (\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma}\, g X)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu.\, \dot{\forall}\phi_{\mu\to\sigma}.\, ($e ...						
A5	$[p_{(\mu\to\sigma)\to\sigma}\mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot{\Box}\dot{\exists}X_\mu.\, g_{\mu\to\sigma}X]$						
MC	$[s_\sigma \dot{\supset} \dot{\Box}s_\sigma]$						
FG	$[\dot{\forall}\phi_{\mu\to\sigma}.\, \dot{\forall}X_\mu.\, (g_{\mu\to\sigma} X \dot{\supset}$ ...						
MT	$[\dot{\forall}X_\mu.\, \dot{\forall}Y_\mu.\, (g_{\mu\to\sigma}X \dot{\supset} (g$ ...						
CO	$\emptyset$ (no goal, check for cons...						
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda$ ...						
CO'	$\emptyset$ (no goal, check for cons...						

Automating Scott's proof script

T1: "Positive properties are possibly exemplified" proved by LEO-II and Satallax

- **in logic: K**
- **from assumptions:**
  - **A1 and A2**
  - **A1($\supset$) and A2**
- **notion of quantification**
  - **possibilist quantifiers (constant dom.)**
  - **actualist quantifiers for individuals (varying dom.)**

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot\neg(\phi X)) \dot\equiv \dot\neg(p\phi)]$						
A2	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall\psi_{\mu\to\sigma}.\, (p_{(\mu\to\sigma)\to\sigma}\phi \dot\wedge \dot\Box\dot\forall X_\mu.\, (\phi X \dot\supset \psi X)) \dot\supset p\psi]$						
T1	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Diamond\dot\exists X_\mu.\, \phi X]$	A1($\supset$), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma}\, g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu.\, g_{\mu\to\sigma} X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Box p\phi]$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \phi X \dot\wedge \dot\forall\psi_{\mu\to\sigma}.\, (\psi X \dot\supset \dot\Box\dot\forall Y_\mu.\, (\phi Y \dot\supset \psi Y))$						
T2	$[\dot\forall X_\mu.\, g_{\mu\to\sigma} X \dot\supset (\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma}\, g X)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, ($						
A5	$[p_{(\mu\to\sigma)\to\sigma}\mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot\Box\dot\exists X_\mu.\, g_{\mu\to\sigma} X]$						
MC	$[s_\sigma \dot\supset \dot\Box s_\sigma]$						
FG	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall X_\mu.\, (g_{\mu\to\sigma} X \dot\supset ($						
MT	$[\dot\forall X_\mu.\, \dot\forall Y_\mu.\, (g_{\mu\to\sigma} X \dot\supset (g_\mu$						
CO	$\emptyset$ (no goal, check for cons						
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda$						
CO'	$\emptyset$ (no goal, check for cons						

Automating Scott's proof script

C: "Possibly, God exists" proved by LEO-II and Satallax

- **in logic: K**
- **from assumptions:**
  - **T1, D1, A3**
- **for domain conditions:**
  - **possibilist quantifiers (constant dom.)**
  - **actualist quantifiers for individuals (varying dom.)**

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot{\forall}\phi_{\mu\to\sigma}\text{.}\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu\text{.}\, \dot{\neg}(\phi X)) \dot{\equiv} \dot{\neg}(p\phi)]$						
A2	$[\dot{\forall}\phi_{\mu\to\sigma}\text{.}\, \dot{\forall}\psi_{\mu\to\sigma}\text{.}\, (p_{(\mu\to\sigma)\to\sigma}\phi \dot{\wedge} \dot{\Box}\dot{\forall}X_\mu\text{.}\, (\phi X \dot{\supset} \psi X)) \dot{\supset} p\psi]$						
T1	$[\dot{\forall}\phi_{\mu\to\sigma}\text{.}\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Diamond}\dot{\exists}X_\mu\text{.}\, \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu\text{.}\, \dot{\forall}\phi_{\mu\to\sigma}\text{.}\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma}g_{\mu\to\sigma}]$						
C	$[\dot{\Diamond}\dot{\exists}X_\mu\text{.}\, g_{\mu\to\sigma}X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot{\forall}\phi_{\mu\to\sigma}\text{.}\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Box}p\phi]$						
D2	$\text{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}\text{.}\, \lambda X_\mu\text{.}\, \phi X \dot{\wedge} \dot{\forall}\psi_{\mu\to\sigma}\text{.}\, (\psi X \dot{\supset} \dot{\Box}\dot{\forall}Y_\mu\text{.}\, (\phi Y \dot{\supset} \psi Y))$						
T2	$[\dot{\forall}X_\mu\text{.}\, g_{\mu\to\sigma}X \dot{\supset} (\text{ess}_{(\mu\to\sigma)\to\mu\to\sigma}gX)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$\text{NE}_{\mu\to\sigma} = \lambda X_\mu\text{.}\, \dot{\forall}\phi_{\mu\to\sigma}\text{.}\, ($e						
A5	$[p_{(\mu\to\sigma)\to\sigma}\text{NE}_{\mu\to\sigma}]$						
T3	$[\dot{\Box}\dot{\exists}X_\mu\text{.}\, g_{\mu\to\sigma}X]$						
MC	$[s_\sigma \dot{\supset} \dot{\Box}s_\sigma]$						
FG	$[\dot{\forall}\phi_{\mu\to\sigma}\text{.}\, \dot{\forall}X_\mu\text{.}\, (g_{\mu\to\sigma}X \dot{\supset}$ (						
MT	$[\dot{\forall}X_\mu\text{.}\, \dot{\forall}Y_\mu\text{.}\, (g_{\mu\to\sigma}X \dot{\supset} (g_\mu$						
CO	∅ (no goal, check for cons						
D2'	$\text{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}\text{.}\, \lambda$						
CO'	∅ (no goal, check for cons						

Automating Scott's proof script

T2: "Being God-like is an ess. of any God-like being" proved by LEO-II and Satallax

- **in logic: K**
- **from assumptions:**
  - **A1, D1, A4, D2**
- **for domain conditions:**
  - **possibilist quantifiers (constant dom.)**
  - **actualist quantifiers for individuals (varying dom.)**

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu\centerdot \dot\neg(\phi X)) \dot\equiv \dot\neg(p\phi)]$						
A2	$[\dot\forall\phi_{\mu\to\sigma}\centerdot \dot\forall\psi_{\mu\to\sigma}\centerdot (p_{(\mu\to\sigma)\to\sigma}\phi \mathbin{\dot\wedge} \dot\Box\dot\forall X_\mu\centerdot(\phi X \mathbin{\dot\supset} \psi X)) \mathbin{\dot\supset} p\psi]$						
T1	$[\dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}\phi \mathbin{\dot\supset} \dot\Diamond\dot\exists X_\mu\centerdot \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu\centerdot \dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}\phi \mathbin{\dot\supset} \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma}g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu\centerdot g_{\mu\to\sigma}X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}\phi \mathbin{\dot\supset} \dot\Box p\phi]$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}\centerdot\lambda X_\mu\centerdot \phi X \mathbin{\dot\wedge} \dot\forall\psi_{\mu\to\sigma}\centerdot(\psi X \mathbin{\dot\supset} \dot\Box\dot\forall Y_\mu\centerdot(\phi Y \mathbin{\dot\supset} \psi Y))$						
T2	$[\dot\forall X_\mu\centerdot g_{\mu\to\sigma}X \mathbin{\dot\supset} (\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma}gX)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu\centerdot \dot\forall\phi_{\mu\to\sigma}\centerdot ($e						
T3	$[\dot\Box\dot\exists X_\mu\centerdot g_{\mu\to\sigma}X]$						
MC	$[s_\sigma \mathbin{\dot\supset} \dot\Box s_\sigma]$						
FG	$[\dot\forall\phi_{\mu\to\sigma}\centerdot$ ... $X \mathbin{\dot\supset}$ (						
MT	$[\dot\forall X_\mu\centerdot \dot\forall Y_\mu\centerdot (g_{\mu\to\sigma}X \mathbin{\dot\supset} (g$						
CO	∅ (no goal, check for cons						
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}\centerdot\lambda$						
CO'	∅ (no goal, check for cons						

<u>Automating Scott's proof script</u>

T3: "Necessarily, God exists" proved by LEO-II and Satallax

- **in logic: KB**
- **from assumptions:**
  - **D1, C, T2, D3, A5**
- **for domain conditions:**
  - **possibilist quantifiers (constant dom.)**
  - **actualist quantifiers for individuals (varying dom.)**

For logic K we got a countermodel by Nitpick

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot{\neg}(\phi X)) \dot{\equiv} \dot{\neg}(p\phi)]$						
A2	$[\dot{\forall}\phi_{\mu\to\sigma}.\, \dot{\forall}\psi_{\mu\to\sigma}.\, (p_{(\mu\to\sigma)\to\sigma}\phi \,\dot{\wedge}\, \dot{\Box}\dot{\forall}X_\mu.\, (\phi X \dot{\supset} \psi X)) \dot{\supset} p\psi]$						
T1	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Diamond}\dot{\exists}X_\mu.\, \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \phi X$						
A3	$[p_{(\mu\to\sigma)\to\sigma}\, g_{\mu\to\sigma}]$						
C	$[\dot{\Diamond}\dot{\exists}X_\mu.\, g_{\mu\to\sigma}X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot{\forall}\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot{\supset} \dot{\Box}p\phi]$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \phi X \,\dot{\wedge}\, \dot{\forall}\psi_{\mu\to\sigma}.\, (\psi X \dot{\supset} \dot{\Box}\dot{\forall}Y_\mu.\, (\phi Y \dot{\supset} \psi Y))$						
T2	$[\dot{\forall}X_\mu.\, g_{\mu\to\sigma}X \dot{\supset} (\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma}\, gX)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu.\, \dot{\forall}\phi_{\mu\to\sigma}.\, ($e						
A5	$[p_{(\mu\to\sigma)\to\sigma}\, \mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot{\Box}\dot{\exists}X_\mu.\, g_{\mu\to\sigma}X]$						
MC	$[s_\sigma \dot{\supset} \dot{\Box}s_\sigma]$						
FG	$[\dot{\forall}\phi_{\mu\to\sigma}.\, \dot{\forall}X_\mu.\, (g_{\mu\to\sigma}X \dot{\supset}$ (						
MT	$[\dot{\forall}X_\mu.\, \dot{\forall}Y_\mu.\, (g_{\mu\to\sigma}X \dot{\supset} (g$						
CO	∅ (no goal, check for cons						
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda$						
CO'	∅ (no goal, check for cons						

Automating Scott's proof script

Summary

- proof verified and automated
- KB is sufficient (criticized logic S5 not needed!)
- possibilist and actualist quantifiers (individuals)
- exact dependencies determined experimentally
- ATPs have found alternative proofs
  e.g. self-identity $\lambda x(x = x)$ is not needed

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot\neg(\phi X)) \dot\equiv \dot\neg(p\phi)]$						
A2	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall\psi_{\mu\to\sigma}.\, (p_{(\mu\to\sigma)\to\sigma}\phi \dot\wedge \dot\Box\dot\forall X_\mu.\, (\phi X \dot\supset \psi X)) \dot\supset p\psi]$						
T1	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Diamond\dot\exists X_\mu.\, \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\ldots}$						
A3	$[p_{(\mu\to\sigma)\to\sigma} g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu.\, g_{\mu\to\sigma} X]$						
A4	$[\dot\forall\phi_{\mu\to\sigma}.\, p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Box p\ldots$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda\ldots$						
T2	$[\dot\forall X_\mu.\, g_{\mu\to\sigma} X \dot\supset (\mathrm{ess}_{(\mu\to\sigma)\ldots}$						
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, (\ldots$						
A5	$[p_{(\mu\to\sigma)\to\sigma} \mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot\Box\dot\exists X_\mu.\, g_{\mu\to\sigma} X]$						
		A1, A2, D1, A3, A4, D2, D3, A5	K	CSA	—/—	—/—	8.2/7.5
		D1, C, T2, D3, A5	KB	THM	0.0/0.1	0.1/5.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
MC	$[s_\sigma \dot\supset \dot\Box s_\sigma]$	D2, T2, T3	KB	THM	17.9/—	3.3/3.2	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
FG	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall X_\mu.\, (g_{\mu\to\sigma} X \dot\supset (\dot\neg(p_{(\mu\to\sigma)\to\sigma}\phi) \dot\supset \dot\neg(\phi X)))]$	A1, D1	KB	THM	16.5/—	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	12.8/15.1	0.0/5.4	—/—
MT	$[\dot\forall X_\mu.\, \dot\forall Y_\mu.\, (g_{\mu\to\sigma} X \dot\supset (g_{\mu\to\sigma} Y \dot\supset X \doteq Y))]$	D1, FG	KB	THM	—/—	0.0/3.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
CO	$\emptyset$ (no goal, check for consistency)	A1, A2, D1, A3, A4, D2, D3, A5	KB	SAT	—/—	—/—	7.3/7.4
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\, \lambda X_\mu.\, \dot\forall\psi_{\mu\to\sigma}.\, (\psi X \dot\supset \dot\Box\dot\forall Y_\mu.\, (\phi Y \dot\supset \psi Y))$						
CO'	$\emptyset$ (no goal, check for consistency)	A1(⊃), A2, D2', D3, A5	KB	UNS	7.5/7.8	—/—	—/—
		A1, A2, D1, A3, A4, D2', D3, A5	KB	UNS	—/—	—/—	—/—

Consistency check: Gödel vs. Scott

- Scott's assumptions are consistent; shown by Nitpick
- Gödel's assumptions are inconsistent; shown by LEO-II (new philosophical result?)

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu\centerdot \dot\neg(\phi X)) \dot\equiv \dot\neg(p\phi)]$						
A2	$[\dot\forall\phi_{\mu\to\sigma}\centerdot \dot\forall\psi_{\mu\to\sigma}\centerdot (p_{(\mu\to\sigma)\to\sigma}\phi \,\dot\wedge\, \dot\Box\dot\forall X_\mu\centerdot(\phi X \dot\supset \psi X)) \dot\supset p\psi]$						
T1	$[\dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Diamond\dot\exists X_\mu\centerdot \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu\centerdot \dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to$ [illegible]						
A3	$[p_{(\mu\to\sigma)\to\sigma} g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu\centerdot g_{\mu\to\sigma} X]$						
A4	$[\dot\forall\phi_{\mu\to\sigma}\centerdot p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Box p$ [illegible]						
D2	$ess_{(\mu\to\sigma)\to\mu}$ [illegible] $= \lambda\phi_{\mu\to\sigma}\centerdot$ [illegible]						
T2	$[\dot\forall X_\mu\centerdot g_{\mu\to\sigma} X \dot\supset (ess_{(\mu\to\sigma)}$ [illegible]						
D3	$NE_{\mu\to\sigma} = \lambda X_\mu\centerdot \dot\forall\phi_{\mu\to\sigma}\centerdot ($ [illegible]						
A5	$[p_{(\mu\to\sigma)\to\sigma} NE_{\mu\to\sigma}]$						
T3	$[\dot\Box\dot\exists X_\mu\centerdot g_{\mu\to\sigma} X]$	D1, C, T2, D3, A5	K	CSA	—/—	—/—	3.8/6.2
		A1, A2, D1, A3, A4, D2, D3, A5	K	CSA	—/—	—/—	8.2/7.5
		D1, C, T2, D3, A5	KB	THM	0.0/0.1	0.1/5.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
MC	$[s_\sigma \dot\supset \dot\Box s_\sigma]$	D2, T2, T3	KB	THM	17.9/—	3.3/3.2	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
FG	$[\dot\forall\phi_{\mu\to\sigma}\centerdot \dot\forall X_\mu\centerdot (g_{\mu\to\sigma} X \dot\supset (\dot\neg(p_{(\mu\to\sigma)\to\sigma}\phi) \dot\supset \dot\neg(\phi X)))]$	A1, D1	KB	THM	16.5/—	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	12.8/15.1	0.0/5.4	—/—
MT	$[\dot\forall X_\mu\centerdot \dot\forall Y_\mu\centerdot (g_{\mu\to\sigma} X \dot\supset (g_{\mu\to\sigma} Y \dot\supset X \dot= Y))]$	D1, FG	KB	THM	—/—	0.0/3.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
CO	∅ (no goal, check for consistency)	A1, A2, D1, A3, A4, D2, D3, A5	KB	SAT	—/—	—/—	7.3/7.4
D2'	$ess_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}\centerdot \lambda X_\mu\centerdot \dot\forall\psi_{\mu\to\sigma}\centerdot(\psi X \dot\supset \dot\Box\dot\forall Y_\mu\centerdot(\phi Y \dot\supset \psi Y))$						
CO'	∅ (no goal, check for consistency)	A1(⊃), A2, D2', D3, A5	KB	UNS	7.5/7.8	—/—	—/—
		A1, A2, D1, A3, A4, D2', D3, A5	KB	UNS	—/—	—/—	—/—

Further Results

- Monotheism holds
- God is flawless

Main Findings

	HOL encoding						
A1	$[\dot\forall \phi_{\mu\to\sigma}. p_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu. \dot\neg$						
A2	$[\dot\forall \phi_{\mu\to\sigma}. \dot\forall \psi_{\mu\to\sigma}. (p_{(\mu\to\sigma)\to}$						
T1	$[\dot\forall \phi_{\mu\to\sigma}. p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Diamond\dot\exists$						
D1	$g_{\mu\to\sigma} = \lambda X_\mu. \dot\forall \phi_{\mu\to\sigma}. p_{(\mu}$						
A3	$[p_{(\mu\to\sigma)\to\sigma} g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu. g_{\mu\to\sigma} X]$						
A4	$[\dot\forall \phi_{\mu\to\sigma}. p_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Box p$						
D2	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}. \lambda$						
T2	$[\dot\forall X_\mu. g_{\mu\to\sigma} X \dot\supset (\mathrm{ess}_{(\mu\to\sigma)}$						
D3	$\mathrm{NE}_{\mu\to\sigma} = \lambda X_\mu. \dot\forall \phi_{\mu\to\sigma}. ($						
A5	$[p_{(\mu\to\sigma)\to\sigma} \mathrm{NE}_{\mu\to\sigma}]$						
T3	$[\dot\Box\dot\exists X_\mu. g_{\mu\to\sigma} X]$						
MC	$[s_\sigma \dot\supset \dot\Box s_\sigma]$	D2, T2, T3	KB	THM	17.9/—	3.3/3.2	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
FG	$[\dot\forall \phi_{\mu\to\sigma}. \dot\forall X_\mu. (g_{\mu\to\sigma} X \dot\supset (\dot\neg(p_{(\mu\to\sigma)\to\sigma}\phi) \dot\supset \dot\neg(\phi X)))]$	A1, D1	KB	THM	16.5/—	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	12.8/15.1	0.0/5.4	—/—
MT	$[\dot\forall X_\mu. \dot\forall Y_\mu. (g_{\mu\to\sigma} X \dot\supset (g_{\mu\to\sigma} Y \dot\supset X \doteq Y))]$	D1, FG	KB	THM	—/—	0.0/3.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
CO	$\emptyset$ (no goal, check for consistency)	A1, A2, D1, A3, A4, D2, D3, A5	KB	SAT	—/—	—/—	7.3/7.4
D2'	$\mathrm{ess}_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}. \lambda X_\mu. \dot\forall \psi_{\mu\to\sigma}. (\psi X \dot\supset \dot\Box\dot\forall Y_\mu. (\phi Y \dot\supset \psi Y))$						
CO'	$\emptyset$ (no goal, check for consistency)	A1($\supset$), A2, D2', D3, A5	KB	UNS	7.5/7.8	—/—	—/—
		A1, A2, D1, A3, A4, D2', D3, A5	KB	UNS	—/—	—/—	—/—

Modal Collapse (Sobel)

$$\forall\varphi(\varphi \supset \Box\varphi)$$

- **proved by LEO-II and Satallax**
- **for possibilist and actualist quantification (ind.)**

Main critique on Gödel's ontological proof:

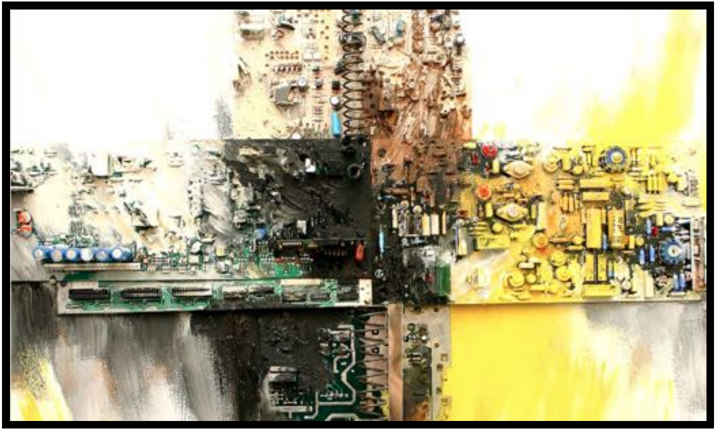
- **there are no contingent truths**
- **everything is determined / no free will**

Main Findings

	HOL encoding	dependencies	logic	status	LEO-II const/vary	Satallax const/vary	Nitpick const/vary
A1	$[\dot\forall\phi_{\mu\to\sigma}.\, P_{(\mu\to\sigma)\to\sigma}(\lambda X_\mu.\, \dot\neg(\phi X)) \dot\equiv \dot\neg(P\phi)]$						
A2	$[\dot\forall\phi_{\mu\to\sigma}.\, \dot\forall\psi_{\mu\to\sigma}.\, (P_{(\mu\to\sigma)\to\sigma}\phi \,\dot\wedge\, \dot\Box\dot\forall X_\mu.(\phi X \dot\supset \psi X)) \dot\supset P\psi]$						
T1	$[\dot\forall\phi_{\mu\to\sigma}.\, P_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Diamond\dot\exists X_\mu.\, \phi X]$	A1(⊃), A2	K	THM	0.1/0.1	0.0/0.0	—/—
		A1, A2	K	THM	0.1/0.1	0.0/5.2	—/—
D1	$g_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.\, P_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \phi X$						
A3	$[P_{(\mu\to\sigma)\to\sigma}\, g_{\mu\to\sigma}]$						
C	$[\dot\Diamond\dot\exists X_\mu.\, g_{\mu\to\sigma}X]$	T1, D1, A3	K	THM	0.0/0.0	0.0/0.0	—/—
		A1, A2, D1, A3	K	THM	0.0/0.0	5.2/31.3	—/—
A4	$[\dot\forall\phi_{\mu\to\sigma}.\, P_{(\mu\to\sigma)\to\sigma}\phi \dot\supset \dot\Box P\phi]$						
D2	$ess_{(\mu\to\sigma)\to\mu\to\sigma} = \lambda\phi_{\mu\to\sigma}.\lambda X_\mu.\, \phi X \,\dot\wedge\, \dot\forall\psi_{\mu\to\sigma}.(\psi X \dot\supset \dot\Box\dot\forall Y_\mu.(\phi Y \dot\supset \psi Y))$						
T2	$[\dot\forall X_\mu.\, g_{\mu\to\sigma}X \dot\supset (ess_{(\mu\to\sigma)\to\mu\to\sigma}\, g X)]$	A1, D1, A4, D2	K	THM	19.1/18.3	0.0/0.0	—/—
		A1, A2, D1, A3, A4, D2	K	THM	12.9/14.0	0.0/0.0	—/—
D3	$NE_{\mu\to\sigma} = \lambda X_\mu.\, \dot\forall\phi_{\mu\to\sigma}.(ess\, \phi X \dot\supset \dot\Box\dot\exists Y_\mu.\, \phi Y)$						
A5	$[P_{(\mu\to\sigma)\to\sigma}\, NE_{\mu\to\sigma}]$						
T3	$[\dot\Box\dot\exists X_\mu.\, g_{\mu\to\sigma}X]$	D1, C, T2, D3, A5	K	CSA	—/—	—/—	3.8/6.2
		A1, A2, D1, A3, A4, D2, D3, A5	K	CSA	—/—	—/—	8.2/7.5
		D1, C, T2, D3, A5	KB	THM	0.0/0.1	0.1/5.3	—/—
		A1, A2, D1, A3, A4, D2, D3, A5	KB	THM	—/—	—/—	—/—
	[illegible]	[illegible]	KB	THM	17.9/—	3.3/3.2	—/—
	[illegible]	3, A5	KB	THM	—/—	—/—	—/—
	[illegible]	[illegible]	KB	THM	16.5/—	0.0/0.0	—/—
	[illegible]	3, A5	KB	THM	12.8/15.1	0.0/5.4	—/—
	[illegible]	[illegible]	KB	THM	—/—	0.0/3.3	—/—
	[illegible]	3, A5	KB	THM	—/—	—/—	—/—
	[illegible]	3, A5	KB	SAT	—/—	—/—	7.3/7.4
	[illegible]	[illegible]	KB	UNS	7.5/7.8	—/—	—/—
	[illegible]	D3, A5	KB	UNS	—/—	—/—	—/—

Observation

- good performance of ATPs
- excellent match between argumentation granularity in papers and the reasoning strength of the ATPs



Reconstruction of the Inconsistency of Gödel's Axioms

Inconsistency (Gödel): Proof by LEO-II in KB

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

by A1($\supset$), A2

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

by A1($\supset$), A2

Def. D2*
$$\varphi\ ess\ x \leftrightarrow \cancel{\varphi(x) \wedge}\ \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$$

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \to \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \to \psi(x)]) \to P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \to \Diamond\exists x\varphi(x)]$

by A1($\supset$), A2

Def. D2* $\varphi\ ess\ x \leftrightarrow \cancel{\varphi(x) \wedge}\ \forall\psi(\psi(x) \to \Box\forall y(\varphi(y) \to \psi(y)))$

Lemma 1 The empty property is an essence of every entity. $\forall x\,(\emptyset\ ess\ x)$

by D2*

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

by A1($\supset$), A2

Def. D2* $\varphi\ ess\ x \leftrightarrow \cancel{\varphi(x) \wedge} \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Lemma 1 The empty property is an essence of every entity. $\forall x\,(\emptyset\ ess\ x)$

by D2*

Def. D3 $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \rightarrow \Box\exists y\varphi(y)]$

Axiom B $\forall\varphi(\varphi \rightarrow \Box\Diamond\varphi)$ (resp. $\forall x\forall y(rxy \rightarrow ryx)$)

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1(⊃) $\forall\varphi[P(\neg\varphi) \to \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \to \psi(x)]) \to P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \to \Diamond\exists x\varphi(x)]$

by A1(⊃), A2

Def. D2* $\varphi\ ess\ x \leftrightarrow \cancel{\varphi(x) \wedge} \forall\psi(\psi(x) \to \Box\forall y(\varphi(y) \to \psi(y)))$

Lemma 1 The empty property is an essence of every entity. $\forall x\,(\emptyset\ ess\ x)$

by D2*

Def. D3 $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \to \Box\exists y\varphi(y)]$

Axiom B $\forall\varphi(\varphi \to \Box\Diamond\varphi)$ (resp. $\forall x\forall y(rxy \to ryx)$)

Lemma 2 Exemplification of necessary existence is not possible. $\neg\Diamond\exists x\,NE(x)$

by B, D3, Lemma1

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

by A1($\supset$), A2

Def. D2* $\varphi\ ess\ x \leftrightarrow \cancel{\varphi(x) \wedge}\ \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Lemma 1 The empty property is an essence of every entity. $\forall x\,(\emptyset\ ess\ x)$

by D2*

Def. D3 $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \rightarrow \Box\exists y\varphi(y)]$

Axiom B $\forall\varphi(\varphi \rightarrow \Box\Diamond\varphi)$ (resp. $\forall x\forall y(rxy \rightarrow ryx)$)

Lemma 2 Exemplification of necessary existence is not possible. $\neg\Diamond\exists x\, NE(x)$

by B, D3, Lemma1

Axiom A5 $P(NE)$

Inconsistency (Gödel): Reconstruction of Informal Argument (KB)

(special thanks to Chad Brown for a fruitful discussion)

Axiom A1($\supset$) $\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$

Axiom A2 $\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$

Theorem 1 Positive Properties are possibly exemplified. $\forall\varphi[P(\varphi) \rightarrow \Diamond\exists x\varphi(x)]$

by A1($\supset$), A2

Def. D2* $\varphi\ ess\ x \leftrightarrow \cancel{\varphi(x) \wedge}\ \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$

Lemma 1 The empty property is an essence of every entity. $\forall x\,(\emptyset\ ess\ x)$

by D2*

Def. D3 $NE(x) \leftrightarrow \forall\varphi[\varphi\ ess\ x \rightarrow \Box\exists y\varphi(y)]$

Axiom B $\forall\varphi(\varphi \rightarrow \Box\Diamond\varphi)$ (resp. $\forall x\forall y(rxy \rightarrow ryx)$)

Lemma 2 Exemplification of necessary existence is not possible. $\neg\Diamond\exists x\,NE(x)$

by B, D3, Lemma1

Axiom A5 $P(NE)$

Inconsistency $\bot$

by A5, T1, Lemma2

Inconsistency (Gödel): Verification in Isabelle/HOL (KB)

GoedelGodWithoutConjunctInEss_KB.thy

GoedelGodWithoutConjunctInEss_KB.thy (~/GoedelGod/Talks/2015/LogicAndReligion/DemoMaterial/)

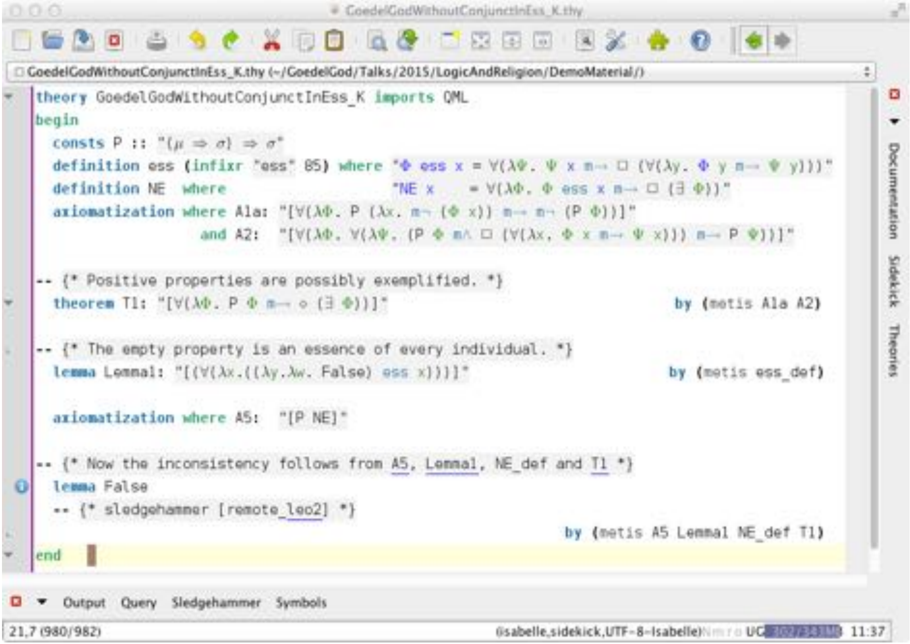
```
theory GoedelGodWithoutConjunctInEss_KB imports QML
begin
  consts P :: "(μ ⇒ σ) ⇒ σ"
  axiomatization where A1a: "[∀(λΦ. P (λx. m¬ (Φ x)) m→ m¬ (P Φ))]"
                   and A2:  "[∀(λΦ. ∀(λΨ. (P Φ m∧ □ (∀(λx. Φ x m→ Ψ x))) m→ P Ψ))]"

-- {* Positive properties are possibly exemplified. *}
  theorem T1: "[∀(λΦ. P Φ m→ ◇ (∃ Φ))]"                                by (metis A1a A2)

  definition ess (infixr "ess" 85) where "Φ ess x = ∀(λΨ. Ψ x m→ □ (∀(λy. Φ y m→ Ψ y)))"

-- {* The empty property is an essence of every individual. *}
  lemma Lemma1: "[(∀(λx.( (λy.λw. False) ess x)))]"                     by (metis ess_def)

  definition NE  where "NE x = ∀(λΦ. Φ ess x m→ □ (∃ Φ))"
  axiomatization where sym: "x r y ⟶ y r x"

-- {* Exemplification of necessary existence is not possible. *}
  lemma Lemma2: "[m¬ (◇ (∃ NE))]"                               by (metis sym Lemma1 NE_def)

  axiomatization where A5:  "[P NE]"

-- {* Now the inconsistency follows from A5, T1 and Lemma2 *}
  lemma False                                                          by (metis A5 T1 Lemma2)
end
```

Documentation Sidekick Theories

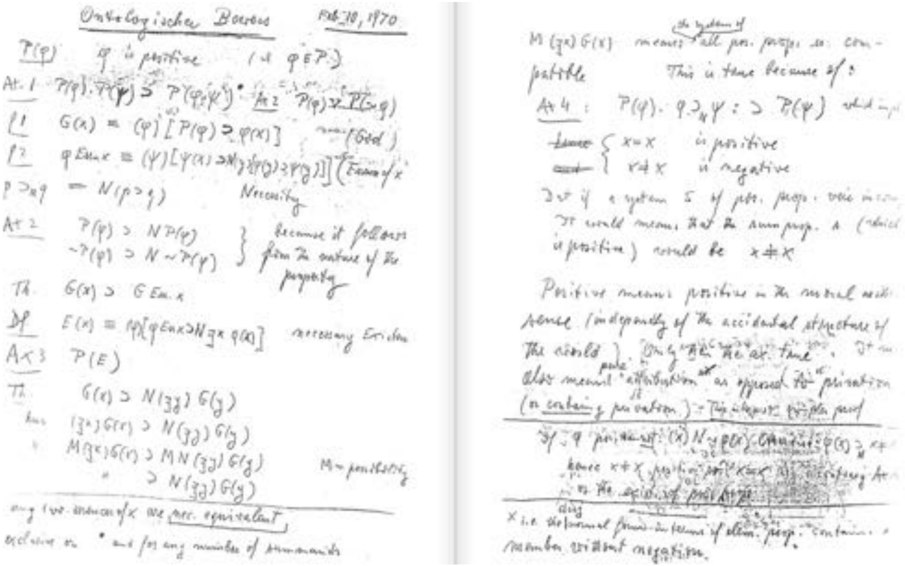
Output Query Sledgehammer Symbols

11,1 (477/1095) (isabelle,sidekick,UTF-8-Isabelle) UG 17:18

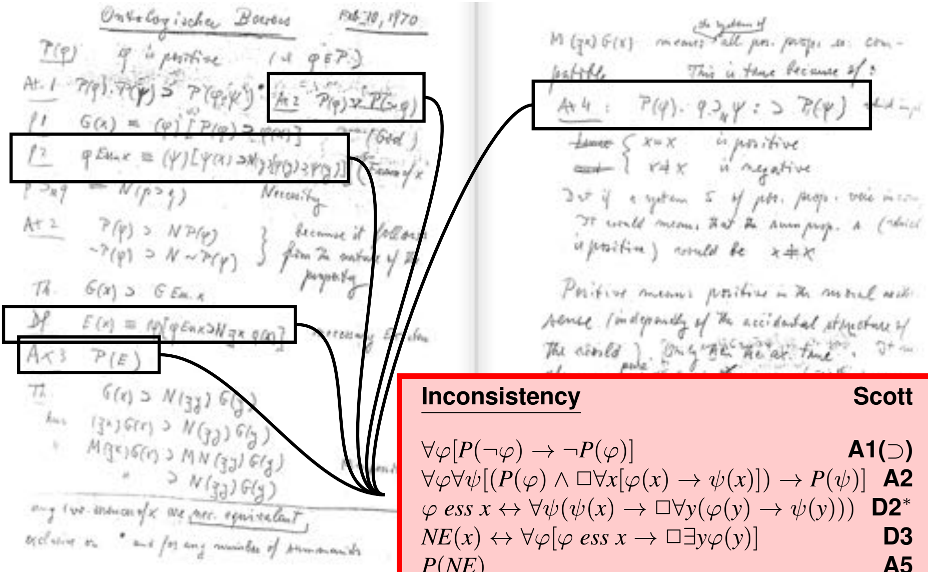
Inconsistency (Gödel): Verification in Isabelle/HOL (K)



Gödel's Manuscript



Gödel's Manuscript



Inconsistency **Scott**

$\forall\varphi[P(\neg\varphi) \rightarrow \neg P(\varphi)]$ **A1(⊃)**

$\forall\varphi\forall\psi[(P(\varphi) \wedge \Box\forall x[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$ **A2**

$\varphi \; ess \; x \leftrightarrow \forall\psi(\psi(x) \rightarrow \Box\forall y(\varphi(y) \rightarrow \psi(y)))$ **D2***

$NE(x) \leftrightarrow \forall\varphi[\varphi \; ess \; x \rightarrow \Box\exists y\varphi(y)]$ **D3**

$P(NE)$ **A5**

Summary: Results of Experiments

Gödel's version	K		KB		S5	
	constant	varying	constant	varying	constant	varying
Consistency	×	×	×	×	×	×
T1	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete
C	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete
T2	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete
T3	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete
Flawless God	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete
Monotheism	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete
Modal Collapse	obsolete	obsolete	obsolete	obsolete	obsolete	obsolete

Further logic details

- Henkin semantics
- full comprehension
- rigid constant symbols

Question: **Has this inconsistency been reported before?**

If not, then LEO-II deserves (part of) the credit!

Summary: Results of Experiments

Scott's version	K		KB		S5	
	constant	varying	constant	varying	constant	varying
Consistency	✓	✓	✓	✓	✓	✓
T1	✓	✓	✓	✓	✓	✓
C	✓	✓	✓	✓	✓	✓
T2	✓	✓	✓	✓	✓	✓
T3	×	×	✓	✓	✓	✓
Flawless God	✓	✓	✓	✓	✓	✓
Monotheism	✓	✓	✓	✓	✓	✓
Modal Collapse	✓	✓	✓	✓	✓	✓

Further logic details

- Henkin semantics
- full comprehension
- rigid constant symbols

Avoiding the Modal Collapse: Recent Variants

SOME EMENDATIONS OF GÖDEL'S ONTOLOGICAL PROOF

C. Anthony Anderson

Kurt Gödel's version of the ontological argument was shown by J. Howard Sobel to be defective, but some plausible modifications in the argument result in a version which is immune to Sobel's objection. A definition is suggested which permits the proof of some of Gödel's axioms.

Gödel's Ontological Proof Revisited *

C. Anthony Anderson and Michael Gettings

University of California, Santa Barbara
Department of Philosophy

Gödel's version of the modal ontological argument for the existence of God has been criticized by J. Howard Sobel [5] and modified by C. Anthony Anderson [1]. In the present paper we consider the extent to which Anderson's emendation is defeated by the type of objection first offered by the Monk Gaunilo to St. Anselm's original Ontological Argument. And we try to push the analysis of this Gödelian argument a bit further to bring it into closer agreement with the details of Gödel's own formulation. Finally, we indicate what seems to be the main weakness of this emendation of Gödel's attempted proof.

Magari and others on Gödel's ontological proof

Petr Hájek
Institute of Computer Science, Academy of Sciences
182 07 Prague, Czech Republic
e-mail: hajek@uivt.cas.cz

1 Introduction

This paper is a continuation of my paper [H] and concentrates almost exclusively to mathematical properties of logical systems underlying Gödel's ontological proof [G] and its variant by Anderson [A], with special care paid to Magari's criticism [M]. Since [H] is written in German, we shall try to summarize its content in such a way that knowledge of [H] will be not obligatory for reading the present paper (even it remains advantageous). Here we describe

Der Mathematiker und die Frage der Existenz Gottes
(betreffend Gödels ontologischen Beweis)

Es ist gut, daß wir nicht wissen,
sondern glauben, daß ein Gott sei.
(Kant, Nachlaß)

1. Einführung

Gödels zu Lebzeiten unveröffentlichter Beweis für die notwendige Existenz eines Gott-ähnlichen Wesens hat sowohl philosophisches als auch mathematisches Interesse geweckt. Zweck der vorliegenden Arbeit ist es, zu einer Deutung des Gödelschen Textes beizutragen, 1. durch Kommentierung der einschlägigen Literatur und 2. durch Bereitstellung von etwas Modelltheorie. Die Arbeit enthält keinen philosophischen Beitrag. Während der letzten Jahre habe ich etliche Male über Gödels Gottesbeweis vorgetragen, insbesondere auf dem Symposium zur Feier von Professor Gert Müller (Heidelberg, Jänner 1991), doch habe ich niemals beabsichtigt, eine Veröffentlichung über das Thema zu machen. Da ich wiederholt um eine schriftliche Version gebeten wurde, entschloß ich mich, schnell eine „erweiterte Kurzfassung"[1] zu schreiben, ohne aus ihr einen

PETR HÁJEK

A New Small Emendation of Gödel's Ontological Proof

Keywords: Ontological proof, Gödel, modal logic, comprehension, positive properties.

1. Introduction

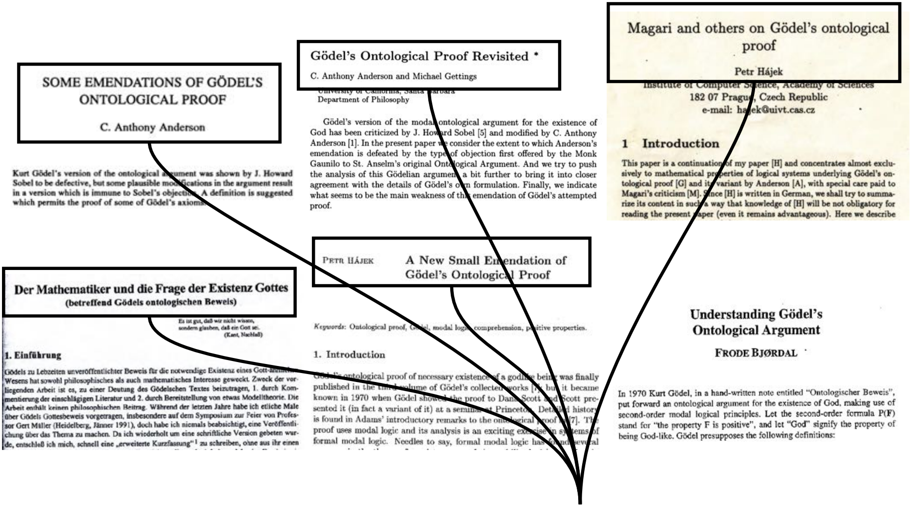
Gödel's ontological proof of necessary existence of a godlike being was finally published in the third volume of Gödel's collected works [7]; but it became known in 1970 when Gödel showed the proof to Dana Scott and Scott presented it (in fact a variant of it) at a seminar at Princeton. Detailed history is found in Adams' introductory remarks to the ontological proof in [7]. The proof uses modal logic and its analysis is an exciting exercise in systems of formal modal logic. Needles to say, formal modal logic has found several

Understanding Gödel's Ontological Argument

FRODE BJØRDAL

In 1970 Kurt Gödel, in a hand-written note entitled "Ontologischer Beweis", put forward an ontological argument for the existence of God, making use of second-order modal logical principles. Let the second-order formula P(F) stand for "the property F is positive", and let "God" signify the property of being God-like. Gödel presupposes the following definitions:

Avoiding the Modal Collapse: Some Emendations



SOME EMENDATIONS OF GÖDEL'S ONTOLOGICAL PROOF

C. Anthony Anderson

Kurt Gödel's version of the ontological argument was shown by J. Howard Sobel to be defective, but some plausible modifications in the argument result in a version which is immune to Sobel's objection. A definition is suggested which permits the proof of some of Gödel's axioms.

Gödel's Ontological Proof Revisited *

C. Anthony Anderson and Michael Gettings

University of California, Santa Barbara
Department of Philosophy

Gödel's version of the modal ontological argument for the existence of God has been criticized by J. Howard Sobel [5] and modified by C. Anthony Anderson [1]. In the present paper we consider the extent to which Anderson's emendation is defeated by the type of objection first offered by the Monk Gaunilo to St. Anselm's original Ontological Argument. And we try to push the analysis of this Gödelian argument a bit further to bring it into closer agreement with the details of Gödel's own formulation. Finally, we indicate what seems to be the main weakness of this emendation of Gödel's attempted proof.

Magari and others on Gödel's ontological proof

Petr Hájek

Institute of Computer Science, Academy of Sciences
182 07 Prague, Czech Republic
e-mail: hajek@uivt.cas.cz

1 Introduction

This paper is a continuation of my paper [H] and concentrates almost exclusively to mathematical properties of logical systems underlying Gödel's ontological proof [G] and its variant by Anderson [A], with special care paid to Magari's criticism [M]. Since [H] is written in German, we shall try to summarize its content in such a way that knowledge of [H] will be not obligatory for reading the present paper (even it remains advantageous). Here we describe

Der Mathematiker und die Frage der Existenz Gottes

(betreffend Gödels ontologischen Beweis)

Es ist gut, daß wir nicht wissen, sondern glauben, daß ein Gott sei.
(Kant, Nachlaß)

1. Einführung

Gödels zu Lebzeiten unveröffentlichter Beweis für die notwendige Existenz eines Gott-ähnlichen Wesens hat sowohl philosophisches als auch mathematisches Interesse geweckt. Zweck der vorliegenden Arbeit ist es, zu einer Deutung des Gödelschen Textes beizutragen, 1. durch Kommentierung der einschlägigen Literatur und 2. durch Bereitstellung von etwas Modelltheorie. Die Arbeit enthält keinen philosophischen Beitrag. Während der letzten Jahre habe ich etliche Male über Gödels Gottesbeweis vorgetragen, insbesondere auf dem Symposium zur Feier von Professor Gert Müller (Heidelberg, Jänner 1991), doch habe ich niemals beabsichtigt, eine Veröffentlichung über das Thema zu machen. Da ich wiederholt um eine schriftliche Version gebeten wurde, entschloß ich mich, schnell eine „erweiterte Kurzfassung"[1] zu schreiben, ohne aus ihr einen

PETR HÁJEK

A New Small Emendation of Gödel's Ontological Proof

Keywords: Ontological proof, Gödel, modal logic, comprehension, positive properties.

1. Introduction

Gödel's ontological proof of necessary existence of a godlike being was finally published in the third volume of Gödel's collected works [7], but it became known in 1970 when Gödel showed the proof to Dana Scott and Scott presented it (in fact a variant of it) at a seminar at Princeton. Detailed history is found in Adams' introductory remarks to the ontological proof [7]. The proof uses modal logic and its analysis is an exciting exercise in systems of formal modal logic. Needless to say, formal modal logic has found several

Understanding Gödel's Ontological Argument

FRODE BJØRDAL

In 1970 Kurt Gödel, in a hand-written note entitled "Ontologischer Beweis", put forward an ontological argument for the existence of God, making use of second-order modal logical principles. Let the second-order formula P(F) stand for "the property F is positive", and let "God" signify the property of being God-like. Gödel presupposes the following definitions:

Computer-supported Clarification of Controversy

Results Obtained with Fully Automated Reasoners

A controversy between Magari, Hájek and Anderson regarding the redundancy of some axioms

Proof	D1'	A:A1'	A2'	A3'	A4	A4'	H:A4	A5	A5'	H:A5	T3	T3'	MC
Scott (const)	-	-	-	-	N/I	-	-	N/I	-	-	P	-	P
Scott (var)	-	-	-	-	N/I	-	-	N/I	-	-	P	-	P
Anderson (const)	-	-	-	-	R (K4B)	-	-	-	R	-	-	P	CS
Anderson (var)	-	-	-	-	R (K4B)	-	-	-	R	-	-	P	CS
Anderson (mix)	-	-	-	-	R (K4B)	-	-	-	I	-	-	CS	CS
Hájek AOE' (var)	-	-	CS		S/I	-	-	-	S/I	-	-	P (KB)	CS
Hájek AOE'_0 (var)	-	-	-	CS	R	-	-	-	S/U	-	-	P (KB)	CS
Hájek AOE'' (var)	-	-			-	-	S/I	-	-	S/I	-	P (KB)	CS
Anderson (simp) (var)	-	R	R			R (K4B)	-	-		-	-		
Bjørdal (const)	R (K4)	-	R	R	-	R (KT)	-	-	N/I	-	-	P (KB)	CS
Bjørdal (var)	CS	-	R	R	-	R (KT)	-	-	N/I	-	-	P (KB)	CS

Results Obtained with Fully Automated Reasoners

A controversy between Magari, Hájek and Anderson regarding the redundancy of some axioms

Proof	D1'	A:A1'	A2'	A3'	A4	A4'	H:A4	A5	A5'	H:A5	T3	T3'	MC
Scott (const)	-	-	-	-	N/I	-	-	N/I	-	-	P	-	P
Scott (var)	-	-	-	-	N/I	-	-	N/I	-	-	P	-	P
Anderson (const)	-	-	-	-	R (K4B)	-	-	-	R	-	-	P	CS
Anderson (var)	-	-	-	-	R (K4B)	-	-	-	R	-	-	P	CS
Anderson (mix)	-	-	-	-	R (K4B)	-	-	-	I	-	-	CS	CS
Hájek AOE' (var)	-	-	CS		S/I	-	-	-	S/I	-	-	P (KB)	CS
Hájek AOE'_0 (var)	-	-	-	CS	R	-	-	-	S/U	-	-	P (KB)	CS
Hájek AOE'' (var)	-	-			-	-	S/I	-	-	S/I	-	P (KB)	CS
Anderson (simp) (var)	-	R	R			R (K4B)	-	-		-	-		
Bjørdal (const)	R (K4)	-	R	R	-	R (KT)	-	-	N/I	-	-	P (KB)	CS
Bjørdal (var)	CS	-	R	R	-	R (KT)	-	-	N/I	-	-	P (KB)	CS

Leibniz (1646–1716)

characteristica universalis and **calculus ratiocinator**

If controversies were to arise, there would be no more need of disputation between two philosophers than between two accountants. For it would suffice to take their pencils in their hands, to sit down to their slates, and to say to each other . . . : Let us calculate.


SPIEGEL ONLINE WISSENSCHAFT

Login | Registrierung

Politik | Wirtschaft | Panorama | Sport | Kultur | Netzwelt | Wissenschaft | Gesundheit | einestages | Karriere | Uni | Schule | Reise | Auto

Nachrichten > Wissenschaft > Mensch > Mathematik > Formel von Kurt Gödel: Mathematiker bestätigen Gottesbeweis

Formel von Kurt Gödel: Mathematiker bestätigen Gottesbeweis

Von Tobias Hürter

picture-alliance/ Imagno/ Wiener Stadt- und Landesbibliothek

Kurt Gödel (um das Jahr 1935): Der Mathematiker hielt seinen Gottesbeweis jahrzehntelang geheim

Ein Wesen existiert, das alle positiven Eigenschaften in sich vereint. Das bewies der legendäre Mathematiker Kurt Gödel mit einem komplizierten Formelgebilde. Zwei Wissenschaftler haben diesen Gottesbeweis nun überprüft - und für gültig befunden.

Montag, 09.09.2013 – 12:03 Uhr

Drucken | Versenden | Merken

Jetzt sind die letzten Zweifel ausgeräumt: Gott existiert tatsächlich. Ein Computer hat es mit kalter Logik bewiesen - das MacBook des Computerwissenschaftlers Christoph Benzmüller von der Freien Universität Berlin.

Germany
- Telepolis & Heise
- Spiegel Online
- FAZ
- Die Welt
- Berliner Morgenpost
- Hamburger Abendpost
- . . .

Austria
- Die Presse
- Wiener Zeitung
- ORF
- . . .

Italy
- Repubblica
- Ilsussidario
- . . .

India
- DNA India
- Delhi Daily News
- India Today
- . . .

US
- ABC News
- . . .

International
- Spiegel International
- Yahoo Finance
- United Press Intl.
- . . .

Home | Video | Themen | Forum | English | DER SPIEGEL | SPIEGEL TV | Abo | Shop   RSS | Mobile | Newsletter

SPIEGEL ONLINE INTERNATIONAL

Sign in | Register

Front Page World Europe Germany Business Zeitgeist Newsletter

English Site > Germany > Science > Scientists Use Computer to Mathematically Prove Gödel God Theorem

Holy Logic: Computer Scientists 'Prove' God Exists

By David Knight

picture-alliance/ Imagno/ Wiener Stadt- und Landesbibliothek

Austrian mathematician Kurt Gödel kept his proof of God's existence a secret for decades. Now two scientists say they have proven it mathematically using a computer.

Two scientists have formalized a theorem regarding the existence of God penned by mathematician Kurt Gödel. But the God angle is somewhat of a red herring -- the real step forward is the example it sets of how computers can make scientific progress simpler.

Germany
- Telepolis & Heise
- Spiegel Online
- FAZ
- Die Welt
- Berliner Morgenpost
- Hamburger Abendpost
- . . .

Austria
- Die Presse
- Wiener Zeitung
- ORF
- . . .

Italy
- Repubblica
- Ilsussidario
- . . .

India
- DNA India
- Delhi Daily News
- India Today
- . . .

US
- ABC News
- . . .

International
- Spiegel International
- Yahoo Finance
- United Press Intl.
- . . .

DIE WELT

Mindestlohn lockt SPD in dritte Ehe mit der Union

Göttliche Mathematik

UNION

Austria
- Die Presse
- Wiener Zeitung
- ORF
- . . .

Italy
- Repubblica
- Ilsussidario
- . . .

India
- DNA India
- Delhi Daily News
- India Today
- . . .

US
- ABC News
- . . .

International
- Spiegel International
- Yahoo Finance
- United Press Intl.
- . . .

SCIENCE NEWS

HOME / SCIENCE NEWS / RESEARCHERS SAY THEY USED MACBOOK TO PROVE GOEDEL'S GOD THEOREM

Researchers say they used MacBook to prove Goedel's God theorem

Oct. 23, 2013 | 8:14 PM | 1 comments

See more serious and funny news links at
https://github.com/FormalTheology/GoedelGod/tree/master/Press

HOML in the Coq Proof Assistant

Embedding HOML in HOL

Example

HOML formula	$\Diamond \exists x G(x)$
HOML formula in HOL	valid $(\Diamond \exists x G(x))_{\iota \to o}$
expansion, $\beta\eta$-conversion	$\forall w_\iota (\Diamond \exists x G(x))_{\iota \to o}\, w$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge (\exists x G(x))_{\iota \to o} u)$
expansion, $\beta\eta$-conversion	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

Does this actually work in practice?

Is it efficient ??

Is it user-friendly ??

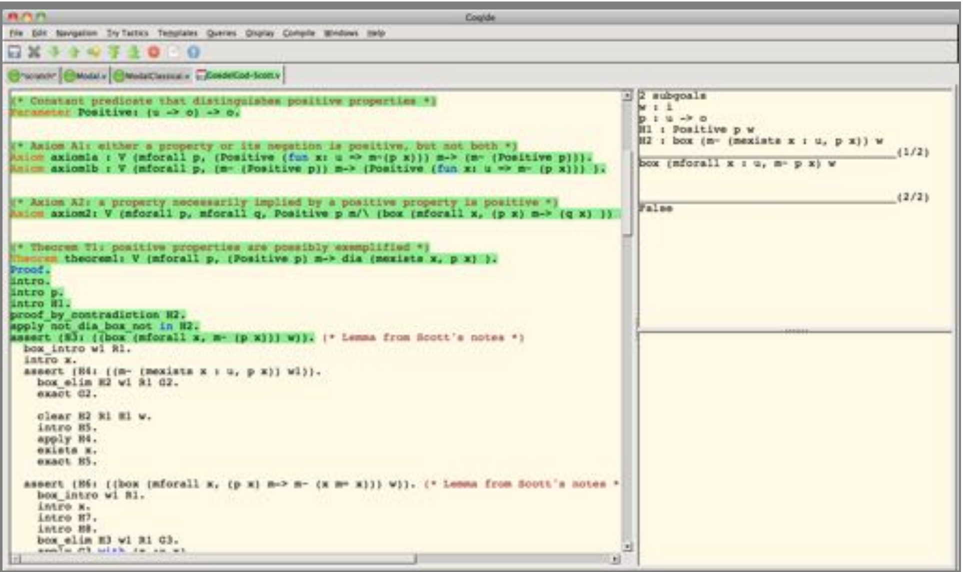
Issues with Fully Automated Reasoning

Proofs are hard to read and do not necessarily correspond to the informal proofs being verified

```
DemoMaterial — bash — 166×52
@SV8)@SV3)=$false) | (((p@(^[SX0:mu,SX1:$i]: $false))@SV3)=$true))),inference(prim_subst,[status(thm)],[66:[bind(SV11,$thf(^[SV23:mu,SV24:$i]: $false))]])).
 thf(84,plain,(![SV22:(mu>($i>$o)),SV3:$i,SV8:(mu>($i>$o))]: ((((SV8@(((sK2_SY33@SV3)@(^[SX0:mu,SX1:$i]: (~ ((SV22@SX0)@SX1))))@SV8))@(((sK1_SY31@(^[SX0:mu,SX1:$i]: (
~ ((SV22@SX0)@SX1))))@SV8)@SV3))=$true) | (((p@SV8)@SV3)=$false) | (((p@(^[SX0:mu,SX1:$i]: (~ ((SV22@SX0)@SX1))))@SV3)=$true))),inference(prim_subst,[status(thm)],[66
:[bind(SV11,$thf(^[SV20:mu,SV21:$i]: (~ ((SV22@SV20)@SV21))))]])).
 thf(85,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4)=$false) | ((((p@SV9)@SV4) = ((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)
@SY28))))@SV4))=$false))),inference(fac_restr,[status(thm)],[56])).
 thf(86,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4)=$true) | ((((p@SV9)@SV4) = ((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@
SY30))))@SV4))=$false))),inference(fac_restr,[status(thm)],[57])).
 thf(87,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((~ (((p@SV9)@SV4) | ((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4))) | (~ ((~ ((p@SV9)@SV4)) | (~ ((p@(^[SY27:mu,
SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4)))))=$false) | (((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4)=$false))),inference(extcnf_equal_neg,[status(thm)],[85])).
 thf(89,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((~ (((p@SV9)@SV4) | ((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4))) | (~ ((~ ((p@SV9)@SV4)) | (~ ((p@(^[SY29:mu,
SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4)))))=$false) | (((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4)=$true))),inference(extcnf_equal_neg,[status(thm)],[86])).
 thf(92,plain,(![SV4:$i,SV9:(mu>($i>$o))]: (((~ ((~ ((p@SV9)@SV4)) | (~ ((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4))))=$false) | (((p@(^[SY27:mu,SY28:$i]: (
~ ((SV9@SY27)@SY28))))@SV4)=$false))),inference(extcnf_or_neg,[status(thm)],[87])).
 thf(93,plain,(![SV4:$i,SV9:(mu>($i>$o))]: (((~ (((p@SV9)@SV4) | ((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4)))=$false) | (((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@
SY29)@SY30))))@SV4)=$true))),inference(extcnf_or_neg,[status(thm)],[89])).
 thf(96,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((~ ((p@SV9)@SV4)) | (~ ((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4)))=$true) | (((p@(^[SY27:mu,SY28:$i]: (~ ((S
V9@SY27)@SY28))))@SV4)=$false))),inference(extcnf_not_neg,[status(thm)],[92])).
 thf(97,plain,(![SV4:$i,SV9:(mu>($i>$o))]: (((((p@SV9)@SV4) | ((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4))=$true) | (((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)
@SY30))))@SV4)=$true))),inference(extcnf_not_neg,[status(thm)],[93])).
 thf(100,plain,(![SV4:$i,SV9:(mu>($i>$o))]: (((~ ((p@SV9)@SV4))=$true) | ((~ ((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4))=$true) | (((p@(^[SY27:mu,SY28:$i]:
 (~ ((SV9@SY27)@SY28))))@SV4)=$false))),inference(extcnf_or_pos,[status(thm)],[96])).
 thf(101,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@SV9)@SV4)=$true) | (((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4)=$true) | (((p@(^[SY29:mu,SY30:$i]: (~ ((SV
9@SY29)@SY30))))@SV4)=$true))),inference(extcnf_or_pos,[status(thm)],[97])).
 thf(103,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@SV9)@SV4)=$false) | ((~ ((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4))=$true) | (((p@(^[SY27:mu,SY28:$i]: (~
 ((SV9@SY27)@SY28))))@SV4)=$false))),inference(extcnf_not_pos,[status(thm)],[100])).
 thf(105,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4)=$false) | (((p@SV9)@SV4)=$false) | (((p@(^[SY27:mu,SY28:$i]: (~ ((
SV9@SY27)@SY28))))@SV4)=$false))),inference(extcnf_not_pos,[status(thm)],[103])).
 thf(107,plain,(![SV8:(mu>($i>$o)),SV3:$i,SV22:(mu>($i>$o))]: ((((SV22@(((sK2_SY33@SV3)@(^[SX0:mu,SX1:$i]: (~ ((SV22@SX0)@SX1))))@SV8))@(((sK1_SY31@(^[SX0:mu,SX1:$i]:
 (~ ((SV22@SX0)@SX1))))@SV8)@SV3))=$true) | (((p@SV8)@SV3)=$false) | (((p@(^[SX0:mu,SX1:$i]: (~ ((SV22@SX0)@SX1))))@SV3)=$true))),inference(extcnf_not_neg,[status(thm
)],[78])).
 thf(108,plain,(![SV11:(mu>($i>$o)),SV3:$i,SV15:(mu>($i>$o))]: ((((SV15@(((sK2_SY33@SV3)@SV11)@(^[SX0:mu,SX1:$i]: (~ ((SV15@SX0)@SX1)))))@(((sK1_SY31@SV11)@(^[SX0:mu,
SX1:$i]: (~ ((SV15@SX0)@SX1))))@SV3))=$false) | (((p@(^[SX0:mu,SX1:$i]: (~ ((SV15@SX0)@SX1))))@SV3)=$false) | (((p@SV11)@SV3)=$true))),inference(extcnf_not_pos,[statu
s(thm)],[81])).
 thf(109,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@(^[SY27:mu,SY28:$i]: (~ ((SV9@SY27)@SY28))))@SV4)=$false) | (((p@SV9)@SV4)=$false))),inference(sim,[status(thm)],[10
5])).
 thf(110,plain,(![SV4:$i,SV9:(mu>($i>$o))]: ((((p@SV9)@SV4)=$true) | (((p@(^[SY29:mu,SY30:$i]: (~ ((SV9@SY29)@SY30))))@SV4)=$true))),inference(sim,[status(thm)],[101]
)).
 thf(111,plain,(![SV3:$i,SV8:(mu>($i>$o))]: ((((p@SV8)@SV3)=$false) | (((p@(^[SX0:mu,SX1:$i]: $true))@SV3)=$true))),inference(sim,[status(thm)],[78])).
 thf(112,plain,(![SV11:(mu>($i>$o)),SV3:$i]: ((((p@(^[SX0:mu,SX1:$i]: $false))@SV3)=$false) | (((p@SV11)@SV3)=$true))),inference(sim,[status(thm)],[80])).
 thf(113,plain,((($false)=$true)),inference(fo_atp_e,[status(thm)],[25,112,111,110,109,108,107,84,83,82,75,74,73,72,71,70,69,68,67,66,65,62,57,56,51,42,29])).
 thf(114,plain,($false),inference(solved_all_splits,[solved_all_splits(join,[])],[113])).
% SZS output end CNFRefutation

%**** End of derivation protocol ****
%**** no. of clauses in derivation: 97 ****
%**** clause counter: 113 ****

% SZS status Unsatisfiable for ConsistencyWithoutFirstConjunctinD2.p : (rf:0,axioms:6,ps:3,u:6,ude:false,rLeibEQ:true,rAndEQ:true,use_choice:true,use_extuni:true,use_
extcnf_combined:true,expand_extuni:false,foatp:e,atp_timeout:25,atp_calls_frequency:10,ordering:none,proof_output:1,clause_count:113,loop_count:0,foatp_calls:2,transl
ation:fof_full)
ontoleo:DemoMaterial cbenzmueller$
```

The Coq Proof Assistant

Winner of the ACM Software System Award in 2013



The Proof/Type System of Coq

- Calculus of Inductive Constructions (CIC)
- Related to CC and λC (cf. next slides).
- A minimalistic higher-order natural deduction calculus.
- Typical natural deduction rules are admissible.

Pure Type Systems

A Pure Type System is a triple $(\mathcal{S}, \mathcal{A}, \mathcal{R})$, where:
$\mathcal{S}$ is a set of sorts; $\mathcal{A} \subseteq \mathcal{S} \times \mathcal{S}$ (axioms); $\mathcal{R} \subseteq \mathcal{S} \times \mathcal{S} \times \mathcal{S}$ ($Prod$ rules).

$$(Ax) \quad \frac{}{\vdash s_1 : s_2} \; (s_1, s_2) \in \mathcal{A}$$

$$(Var) \quad \frac{\Gamma \vdash A : s}{\Gamma x : A \vdash x : A} \; x \notin \mathrm{dom}(\Gamma)$$

$$(Prod) \quad \frac{\Gamma \vdash A : s_1 \quad \Gamma, x : A \vdash B : s_2}{\Gamma \vdash (\Pi x : A.B) : s_3} \; (s_1, s_2, s_3) \in \mathcal{R}$$

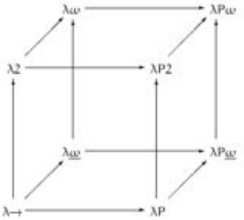
$$(Abs) \quad \frac{\Gamma, x : A \vdash B : C \quad \Gamma \vdash (\Pi x : A.C) : s}{\Gamma \vdash (\lambda x : A.B) : \Pi x : A.C}$$

$$(App) \quad \frac{\Gamma \vdash A : (\Pi x : B.C) \quad \Gamma \vdash D : B}{\Gamma \vdash (A\ D) : C[x := D]}$$

$$(Weak) \quad \frac{\Gamma \vdash A : B \quad \Gamma \vdash C : s}{\Gamma, x : C \vdash A : B}$$

$$(Conv) \quad \frac{\Gamma \vdash A : B \quad \Gamma \vdash B' : s}{\Gamma \vdash A : B'} \; B =_\beta B'$$

Barendregt's Cube



The various type systems in the cube have $\mathcal{A} = \{\ (*, \Box)\ \}$ and differ from each other w.r.t. the sorts allowed in the Prod rule:

$$\text{(Prod)} \qquad \frac{\Gamma \vdash A : s_1 \quad \Gamma, x : A \vdash B : s_2}{\Gamma \vdash (\Pi x : A.B) : s_3} \ (s_1, s_2, s_3) \in \mathcal{R}$$

- $\lambda\!\rightarrow$: $\mathcal{R} = \{(*, *, *)\}$
- λP: $\mathcal{R} = \{(*, *, *), (*, \Box, \Box)\}$
- $\lambda 2$: $\mathcal{R} = \{(*, *, *), (\Box, *, *)\}$
- ...
- λC: $\mathcal{R} = \{(*, *, *), (\Box, *, *), (*, \Box, \Box), (\Box, \Box, \Box)\}$

Typical Natural Deduction Rules

Interactive proving using basic tactics in Coq feels roughly like constructing a natural deduction proof

$$\dfrac{A \lor B \quad \begin{array}{c}\overline{A}\\ \vdots \\ C\end{array} \quad \begin{array}{c}\overline{B}\\ \vdots \\ C\end{array}}{C}\ \lor_E \qquad \dfrac{A \quad B}{A \land B}\ \land_I \qquad \dfrac{\begin{array}{c}\overline{A}^{\,h}\\ \vdots \\ B\end{array}}{A \rightarrow B}\ \rightarrow_I^h$$

$$\dfrac{A}{A \lor B}\ \lor_{I_1} \qquad \dfrac{A \land B}{A}\ \land_{E_1} \qquad \dfrac{B}{A \rightarrow B}\ \rightarrow_I$$

$$\dfrac{B}{A \lor B}\ \lor_{I_2} \qquad \dfrac{A \land B}{B}\ \land_{E_2} \qquad \dfrac{A \quad A \rightarrow B}{B}\ \rightarrow_E$$

$$\dfrac{A[\alpha]}{\forall x.A[x]}\ \forall_I \qquad \dfrac{\forall x.A[x]}{A[t]}\ \forall_E \qquad \dfrac{A[t]}{\exists x.A[x]}\ \exists_I \qquad \dfrac{\exists x.A[x]}{A[\beta]}\ \exists_E$$

$$\neg A \equiv A \rightarrow \bot \qquad \dfrac{\neg\neg A}{A}\ \neg\neg_E$$

- Challenges:
  - Can we hide the semantic embedding from the user?

- Challenges:
  - Can we hide the semantic embedding from the user?
  - Can we provide an interaction experience to the user that differs as little as possible from what he is already used to?

- Challenges:
  - Can we hide the semantic embedding from the user?
  - Can we provide an interaction experience to the user that differs as little as possible from what he is already used to?
  - Can we reconstruct, step-by-step in Coq, precisely Scott's formulation of Gödel's argument?

Modal Logics in the Coq Proof Assistant

```
Parameter i: Type. (* Type for worlds *)
Parameter u: Type. (* Type for individuals *)
Definition o := i -> Prop. (* Type of modal propositions *)

Parameter r: i -> i -> Prop. (* Accessibility relation for worlds *)

Definition mnot (p: o)(w: i) := ~ (p w).
Notation "m~  p" := (mnot p) (at level 74, right associativity).

Definition mand (p q:o)(w: i) := (p w) /\ (q w).
Notation "p m/\ q" := (mand p q) (at level 79, right associativity).

Definition mor (p q:o)(w: i) := (p w) \/ (q w).
Notation "p m\/ q" := (mor p q) (at level 79, right associativity).

Definition mimplies (p q:o)(w:i) := (p w) -> (q w).
Notation "p m-> q" := (mimplies p q) (at level 99, right associativity).

Definition mequiv (p q:o)(w:i) := (p w) <-> (q w).
Notation "p m<-> q" := (mequiv p q) (at level 99, right associativity).
```

Modal Logics in the Coq Proof Assistant

```
Definition A {t: Type}(p: t -> o)(w: i) := forall x, p x w.
Notation "'mforall'  x , p" := (A (fun x => p))
  (at level 200, x ident, right associativity) : type_scope.
Notation "'mforall' x : t , p" := (A (fun x:t => p))
  (at level 200, x ident, right associativity,
   format "'[' 'mforall' '/ '  x  :  t , '/ '  p ']'")
  : type_scope.

Definition E {t: Type}(p: t -> o)(w: i) := exists x, p x w.
Notation "'mexists' x , p" := (E (fun x => p))
  (at level 200, x ident, right associativity) : type_scope.
Notation "'mexists' x : t , p" := (E (fun x:t => p))
  (at level 200, x ident, right associativity,
   format "'[' 'mexists' '/ '  x  :  t , '/ '  p ']'")
  : type_scope.
```

Modal Logics in the Coq Proof Assistant

```
Definition box (p: o) := fun w => forall w1, (r w w1) -> (p w1).
Definition dia (p: o) := fun w => exists w1, (r w w1) /\ (p w1).
```

Modal Logics in the Coq Proof Assistant

```
Lemma mp_dia:
  [mforall p, mforall q, (dia p) m-> (box (p m-> q)) m-> (dia q)].
Proof. mv.
intros p q H1 H2. unfold dia. unfold dia in H1. unfold box in H2.
destruct H1 as [w0 [R1 H1]]. exists w0. split.
  exact R1.
  apply H2.
    exact R1.
    exact H1.
Qed.
```

```
Ltac box_i := let w := fresh "w" in let R := fresh "R"
               in (intro w at top; intro R at top).

Ltac box_elim H w1 H1 := match type of H with
      ((box ?p) ?w) => cut (p w1);
                       [intros H1 | (apply (H w1); try assumption)] end.

Ltac box_e H H1:= match goal with | [ |- (_ ?w) ] => box_elim H w H1 end.

Ltac dia_e H := let w := fresh "w" in let R := fresh "R" in
                (destruct H as [w [R H]]; move w at top; move R at top).

Ltac dia_i w := (exists w; split; [assumption | idtac]).
```

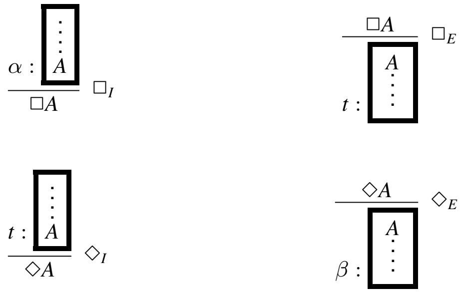
Modal Logics in the Coq Proof Assistant

```
Lemma mp_dia:
  [mforall p, mforall q, (dia p) m-> (box (p m-> q)) m-> (dia q)].
Proof. mv.
intros p q H1 H2. unfold dia. unfold dia in H1. unfold box in H2.
destruct H1 as [w0 [R1 H1]]. exists w0. split.
  exact R1.
  apply H2.
    exact R1.
    exact H1.
Qed.
```

```
Lemma mp_dia:
  [mforall p, mforall q, (dia p) m-> (box (p m-> q)) m-> (dia q)].
Proof. mv.
intros p q H1 H2. dia_e H1. dia_i w0. box_e H2 H3. apply H3. exact H1.
Qed.
```

Natural Deduction Calculus

Rules for Modalities

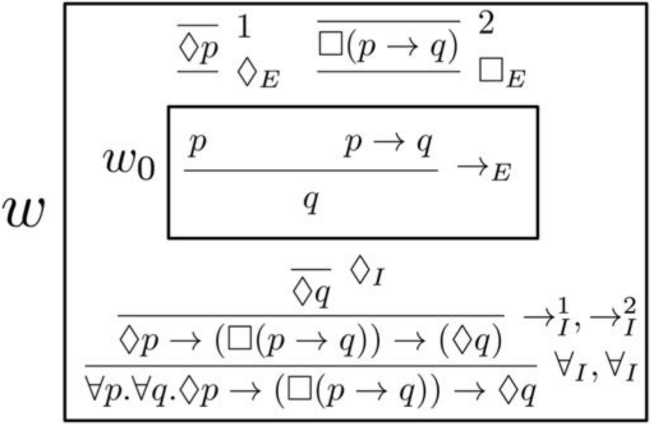


eigen-box condition:
$\Box_I$ and $\Diamond_E$ are strong modal rules:
α and β must be fresh names for the boxes they access
(in analogy to the eigen-variable condition for strong quantifier rules).
Every box must be accessed by exactly one strong modal inference.

boxed assumption condition:
assumptions should be discharged within
the box where they are created.

Modal Logics in the Coq Proof Assistant

```
Lemma mp_dia:
  [mforall p, mforall q, (dia p) m-> (box (p m-> q)) m-> (dia q)].
Proof. mv.
intros p q H1 H2. dia_e H1. dia_i w0. box_e H2 H3. apply H3. exact H1.
Qed.
```



Modal Logics in the Coq Proof Assistant

Part of Scott's Formulation of Gödel's Proof

```
(* Theorem T1: positive properties are possibly exemplified *)
Theorem theorem1: [ mforall p, (Positive p) m-> dia (mexists x, p x) ].
Proof. mv.
intro p. intro H1. proof_by_contradiction H2. apply not_dia_box_not in H2.
assert (H3: ((box (mforall x, m~ (p x))) w)). (* Scott *)
  box_i. intro x. assert (H4: ((m~ (mexists x : u, p x)) w0)).
    box_e H2 G2. exact G2.
    clear H2 R H1 w. intro H5. apply H4. exists x. exact H5.
  assert (H6: ((box (mforall x, (p x) m-> m~ (x m= x))) w)). (* Scott *)
    box_i. intro x. intros H7 H8. box_elim H3 w0 G3. eapply G3. exact H7.
    assert (H9: ((Positive (fun x => m~ (x m= x))) w)). (* Scott *)
      apply (axiom2 w p (fun x => m~ (x m= x))). split.
        exact H1.
        exact H6.
      assert (H10: ((box (mforall x, (p x) m-> (x m= x))) w)). (* Scott *)
        box_i. intros x H11. reflexivity.
        assert (H11 : ((Positive (fun x => (x m= x))) w)). (* Scott *)
          apply (axiom2 w p (fun x => x m= x )). split.
            exact H1.
            exact H10.
          apply axiom1a in H9. contradiction.
Qed.
```

Natural Deduction Proofs for the Ontological Argument

T1 and C1

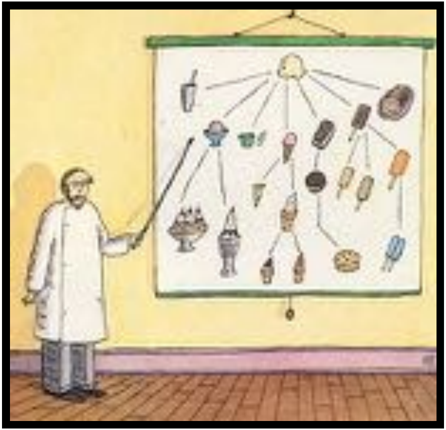
$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{\textbf{A2}}{\forall\varphi.\forall\psi.[(P(\varphi)\wedge\Box\forall x.[\varphi(x)\rightarrow\psi(x)])\rightarrow P(\psi)]}
}{\forall\psi.[(P(\rho)\wedge\Box\forall x.[\rho(x)\rightarrow\psi(x)])\rightarrow P(\psi)]}\forall_E
}{(P(\rho)\wedge\Box\forall x.[\rho(x)\rightarrow\neg\rho(x)])\rightarrow P(\neg\rho)}\forall_E
}{(P(\rho)\wedge\Box\forall x.[\neg\rho(x)])\rightarrow P(\neg\rho)}
\qquad
\dfrac{\dfrac{\textbf{A1a}}{\forall\varphi.[P(\neg\varphi)\rightarrow\neg P(\varphi)]}}{P(\neg\rho)\rightarrow\neg P(\rho)}\forall_E
}{(P(\rho)\wedge\Box\forall x.[\neg\rho(x)])\rightarrow\neg P(\rho)}
}{P(\rho)\rightarrow\Diamond\exists x.\rho(x)}
}{\textbf{T1:}\ \forall\varphi.[P(\varphi)\rightarrow\Diamond\exists x.\varphi(x)]}\forall_I
$$

$$
\dfrac{
\dfrac{\textbf{A3}}{P(G)}
\qquad
\dfrac{\dfrac{\textbf{T1}}{\forall\varphi.[P(\varphi)\rightarrow\Diamond\exists x.\varphi(x)]}}{P(G)\rightarrow\Diamond\exists x.G(x)}\forall_E
}{\Diamond\exists x.G(x)}\rightarrow_E
$$

Natural Deduction Proofs for the Ontological Argument

T2 (Partial)

$$
\dfrac{\dfrac{\dfrac{\psi(x)^6 \qquad \dfrac{\Pi_2}{\psi(x) \to \Box P(\psi)}}{\Box P(\psi)}\,{\to_E} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\Box P(\psi)^7}{P(\psi)}\,\Box_E \qquad \dfrac{\Pi_3}{P(\psi) \to \forall x.(G(x) \to \psi(x))}}{\forall x.(G(x) \to \psi(x))}\,{\to_E}}{\Box \forall x.(G(x) \to \psi(x))}\,\Box_I}{\Box P(\psi) \to \Box \forall x.(G(x) \to \psi(x))}\,{\to_I^7}}{\Box \forall x.(G(x) \to \psi(x))}\,{\to_E}}{\psi(x) \to \Box \forall x.(G(x) \to \psi(x))}\,{\to_I^6}
$$



SUMO Ontology and HOML

SUMO Ontology and HOML

Context

- prominent challenge in AI (CS, Philosophy)
- McCarthy: modeling of contexts as first-class objects

```
ist(context_of("Ben's Knowledge),likes(Sue,Bill))

ist(context_of("Ben's Knowledge),
    ist(context_of(...),...))
```

- McCarthy's [McCarthy, Comm.ACM 1987] [McCarthy, IJCAI 1993] approach has been followed by many others
- Giunchiglia's contextual reasoning [Giunchiglia, Epistemologia 1993] emphasizes the locality aspect; structured knowledge
- McCarthy and Giunchiglia avoid modal logics
- they also avoid a HOL perspective

SUMO Ontology and HOML

Context

- prominent challenge in AI (CS, Philosophy)
- McCarthy: modeling of contexts as first-class objects

```
ist(context_of("Ben's Knowledge),likes(Sue,Bill))

          ist(context_of("Ben's Knowledge),
              ist(context_of(...),...))
```

- McCarthy's [McCarthy, Comm.ACM 1987] [McCarthy, IJCAI 1993] approach has been followed by many others
- Giunchiglia's contextual reasoning [Giunchiglia, Epistemologia 1993] emphasizes the locality aspect; structured knowledge
- McCarthy and Giunchiglia avoid modal logics
- they also avoid a HOL perspective

Our approach is complementary:
takes a HOL perspective and integrates modal logics

SUMO Ontology and HOML

Expressive ontologies

- SUMO and Cyc
- modeling of contexts:

```
(holdsDuring (yearFn 2009) (loves Bill Mary))

    (believes Bill
      (knows Ben
        (forall (?X)
          ((woman ?X) => (loves Bill ?X)))))
```

- relation to McCarthy's approach is obvious
- often a questionable semantics assumed for embedded formulas and modal predicates (also in Common Logic)

SUMO Ontology and HOML

Expressive ontologies

- SUMO and Cyc
- modeling of contexts:

```
(holdsDuring (yearFn 2009) (loves Bill Mary))

    (believes Bill
      (knows Ben
        (forall (?X)
          ((woman ?X) => (loves Bill ?X)))))
```

- relation to McCarthy's approach is obvious
- often a questionable semantics assumed for embedded formulas and modal predicates (also in Common Logic)

Our approach:
HOL-based semantics, but holdsDuring, believes, knows and alike are associated with modal logic connectives

Combining logics

- prominent challenge in AI (CS, Philosophy)
- epistemic, deontic, temporal, intuitionistic, relevant, linear, conditional, security . . .
- wide literature—few implementations
- some propositional systems exists: Logic Workbench, LoTREC, Tableaux Workbench, FaCT, ileanCoP, MSPASS
- no implemented systems for combinations of first-order logics
- combination is typically approached bottom-up

SUMO Ontology and HOML

Combining logics

- prominent challenge in AI (CS, Philosophy)
- epistemic, deontic, temporal, intuitionistic, relevant, linear, conditional, security ...
- wide literature—few implementations
- some propositional systems exists: Logic Workbench, LoTREC, Tableaux Workbench, FaCT, ileanCoP, MSPASS
- no implemented systems for combinations of first-order logics
- combination is typically approached bottom-up

Our approach is complementary:
works top-down starting from classical higher-order logic (HOL)

SUMO Ontology and HOML

SUMO — Suggested Upper Merged Ontology

[NilesPease FOIS 2001, Pease 2011]

- open source, formal ontology: `www.ontologyportal.org`
- has been extended for a number of domain specific ontologies
- altogether approx. 20,000 terms and 70,000 axioms
- employs the SUO-KIF representation language, a simplification of Genesereth's original Knowledge Interchange Format (KIF)

Sigma [PeaseBenzmüller AI Comm. 2013]

- browsing and inference system for ontology development
- integrates KIF-Vampire and SystemOnTPTP

SUMO Ontology and HOML

SUMO — Suggested Upper Merged Ontology

[NilesPease FOIS 2001, Pease 2011]

- open source, formal ontology: `www.ontologyportal.org`
- has been extended for a number of domain specific ontologies
- altogether approx. 20,000 terms and 70,000 axioms
- employs the SUO-KIF representation language, a simplification of Genesereth's original Knowledge Interchange Format (KIF)

Sigma [PeaseBenzmüller AI Comm. 2013]

- browsing and inference system for ontology development
- integrates KIF-Vampire and SystemOnTPTP

SUMO (and similarly Cyc) contains higher-order representations, but there is only very limited automation support so far

Higher-Order Aspects in SUO-KIF and SUMO: Examples

- Embedded formulas

$$\mathit{term} ::= \mathit{variable} | \mathit{word} | \mathit{string} | \mathit{funterm} | \mathit{number} | \mathit{sentence}$$

```
(holdsDuring (YearFn 2009) (likes Mary Bill))
```

Higher-Order Aspects in SUO-KIF and SUMO: Examples

- Embedded formulas
- …often in combination with modal operators such as `holdsDuring`, `knows`, `believes`, etc.

Higher-Order Aspects in SUO-KIF and SUMO: Examples

- Embedded formulas
- …often in combination with modal operators such as `holdsDuring`, `knows`, `believes`, etc.
- Predicate variables, function variables, propositional variables

funterm ::= (funword arg+)   relsent ::= (relword arg+)

funword, relword ::= initialchar wordchar* | variable

```
(<=>
   (instance ?REL TransitiveRelation)
   (forall (?INST1 ?INST2 ?INST3)
      (=>
         (and
            (?REL ?INST1 ?INST2)
            (?REL ?INST2 ?INST3))
         (?REL ?INST1 ?INST3))))
```

Higher-Order Aspects in SUO-KIF and SUMO: Examples

- Embedded formulas
- ...often in combination with modal operators such as `holdsDuring`, `knows`, `believes`, etc.
- Predicate variables, function variables, propositional variables
- Lambda-Abstraction with KappaFN

```
(=>
  (attribute ?X Celebrity)
  (greaterThan
    (CardinalityFn
      (KappaFn ?A
        (knows ?A (exists (?P) (equal ?P ?X)))))
    1000))
```

Higher-Order Aspects in SUO-KIF and SUMO: Examples

- Embedded formulas
- ...often in combination with modal operators such as `holdsDuring`, `knows`, `believes`, etc.
- Predicate variables, function variables, propositional variables
- Lambda-Abstraction with KappaFN

Our focus in the remainder:

embedded formulas and modal operators

Some FO translation 'tricks'

First-order reasoning on a large ontology

[PeaseSutcliffe, CEUR 257, 2007]

- Quoting of embedded formulas

A: `(holdsDuring (YearFn 2009) (likes Mary Bill))`
Q: `(holdsDuring (YearFn ?Y) (likes ?X Bill))`

Some FO translation 'tricks'

First-order reasoning on a large ontology

[PeaseSutcliffe, CEUR 257, 2007]

- Quoting of embedded formulas

A: `(holdsDuring (YearFn 2009) '(likes Mary Bill))`
Q: `(holdsDuring (YearFn ?Y) '(likes ?X Bill))`

Answer with FO-ATPs (`?Y` $\leftarrow$ `2009`, `?X` $\leftarrow$ `Mary`)

Some FO translation 'tricks'

First-order reasoning on a large ontology

[PeaseSutcliffe, CEUR 257, 2007]

- Quoting of embedded formulas

```
A: (holdsDuring (YearFn 2009)
     '(and (likes Mary Bill) (likes Sue Bill)))
Q: (holdsDuring (YearFn ?Y) '(likes ?X Bill))
```

Failure with FO-ATP

Some FO translation 'tricks'

First-order reasoning on a large ontology

[PeaseSutcliffe, CEUR 257, 2007]

- Quoting of embedded formulas
- Expansion of predicate variables

Some FO translation 'tricks'

First-order reasoning on a large ontology

[PeaseSutcliffe, CEUR 257, 2007]

- Quoting of embedded formulas
- Expansion of predicate variables

Why not trying higher-order automated theorem proving directly?

Our focus:

embedded formulas and modal operators

The SUO-KIF to TPTP THF0 Translation

The SUO-KIF to TPTP THF0 Translation

- THF0: TPTP format for simple type theory
  [SutcliffeBenzmüller, J.Formalized Reasoning, 2010]
- THF0 ATPs: LEO-II, TPS, Isabelle, Satallax, . . .
  THF0 (counter-)model finders: Nitpick, Satallax

The SUO-KIF to TPTP THF0 Translation

- THF0: TPTP format for simple type theory
  [SutcliffeBenzmüller, J.Formalized Reasoning, 2010]
- THF0 ATPs: LEO-II, TPS, Isabelle, Satallax, ...
  THF0 (counter-)model finders: Nitpick, Satallax
- achieved:

  SUO-KIF $\longrightarrow$ TPTP THF0

  translation mechanism for SUMO as part of Sigma
- so far only exploits base type ι and o in THF0 ($\rightarrow$ improvable)
- generally applicable to SUO-KIF representations
- translation example (for SUMO) available at:

  `http://christoph-benzmueller.de/papers/SUMO.thf`

The SUO-KIF to TPTP THF Translation

Main challenge: find consistent typing for untyped SUO-KIF

```
(instance instance BinaryPredicate)
```

The SUO-KIF to TPTP THF Translation

Main challenge: find consistent typing for untyped SUO-KIF

```
(p_instance t_instance BinaryPredicate)
```

Higher-Order Automated Theorem Proving in Ontology Reasoning

Embedded Formulas — An Easy Task for HO-ATP

Example (A: Embedded Formulas)

During 2009 Mary liked Bill and Sue liked Bill. Who liked Bill in 2009?

A: `(holdsDuring (YearFn 2009)`
`(and (likes Mary Bill) (likes Sue Bill)))`

Q: `(holdsDuring (YearFn 2009) (likes ?X Bill))`

Proof by LEO-II in milliseconds

Embedded Formulas — An Easy Task for HO-ATP

Example (B: Embedded Formulas)

During 2009 Mary liked Bill and Sue liked Bill. Who liked Bill in 2009?

A:
```
(holdsDuring (YearFn 2009)
  (not (or (not (likes Mary Bill))
           (not (likes Sue Bill)))))
```
Q:
```
(holdsDuring (YearFn 2009) (likes ?X Bill))
```

Proof by LEO-II in milliseconds

Embedded Formulas — An Easy Task for HO-ATP

Example (C: Embedded Formulas)

At all times Mary likes Bill. During 2009 Sue liked whomever Mary liked. Is there a year in which Sue has liked somebody?

A: `(holdsDuring ?Y (likes Mary Bill))`
B: `(holdsDuring (YearFn 2009)`
`    (forall (?X) (=> (likes Mary ?X) (likes Sue ?X))))`
Q: `(holdsDuring (YearFn ?Y) (likes Sue ?X))`

Proof by LEO-II in milliseconds

Embedded Formulas — An Easy Task for HO-ATP

In the above examples we have (silently) assumed that the semantics of the logic underlying SUMO is a classical, bivalent logic, meaning that Boolean extensionality is valid:

```
(<=> (<=> ?P ?Q) (equal ?P ?Q))
```

Example (D: Embedded Formulas)

During 2009 Mary liked Bill and Sue liked Bill. Who liked Bill in 2009?

A:
```
(holdsDuring (YearFn 2009)
  (and (likes Mary Bill) (likes Sue Bill)))
```

Q:
```
(holdsDuring (YearFn 2009)
  (and (likes Sue Bill) (likes Mary Bill)))
```

Proof by LEO-II in milliseconds

Boolean extensionality seems fine for the particular temporal contexts of our previous examples.

However, as we will show next, it quickly leads to counterintuitive inferences in other modal contexts.

Problem: Boolean Extensionality versus Modal Operators

Example (E: Embedded Formulas – Epistemic Contexts)

A: `(knows Chris (equal Chris Chris))`
B: `(likes Mary Bill)`
C: `(knows Chris`
`    (forall (?X) (=> (likes Mary ?X) (likes Sue ?X))))`
Q: `(knows Chris (likes Sue Bill))`

Proof by LEO-II in milliseconds

Problem: Boolean Extensionality versus Modal Operators

Example (E: Embedded Formulas – Epistemic Contexts)

A: `(knows Chris (equal Chris Chris))`
B: `(likes Mary Bill)`
C: `(knows Chris`
`    (forall (?X) (=> (likes Mary ?X) (likes Sue ?X))))`
Q: `(knows Chris (likes Sue Bill))`

Proof by LEO-II in milliseconds

Boolean extensionality is in conflict with (epistemic) modalities!
(Has Boolean extensionality ever been questioned for KIF?)

Proposed Solution: Possible World Semantics for SUMO

SUMO $\longrightarrow$ HOML $\longrightarrow$ TPTP THF

- T-Box like information in SUMO:

  `(instance holdsDuring AsymmetricRelation)` $\longrightarrow$

  $\forall W_\iota$ $(\text{instance holdsDuring AsymmetricRelation})_{\iota\to o}\ W$

Proposed Solution: Possible World Semantics for SUMO

SUMO $\longrightarrow$ HOML $\longrightarrow$ TPTP THF

- T-Box like information in SUMO:

  `(instance holdsDuring AsymmetricRelation)` $\longrightarrow$

  $\forall W_\iota$ $(\text{instance holdsDuring AsymmetricRelation})_{\iota\to o}\ W$

- A-Box like information as in query problem: current world cw_ι

  (likes Mary Bill) $\longrightarrow$ $(\text{likes Mary Bill})_{\iota\to o}\ cw$

  (knows Chris (likes Sue Bill)) $\longrightarrow$

  $(\Box_{\text{Chris}}\ (\text{likes Sue Bill}))_{\iota\to o}\ cw$

Challenge: Embedded Formulas — Epistemic Context

Example (F: Embedded Formulas – Epistemic Contexts)

A: $(\Box_{Chris}\ (equal\ Chris\ Chris))\ cw$
B: $(likes\ Mary\ Bill)\ cw$
C: $(\Box_{Chris}\ (\forall^{i} X_{\mu}\blacksquare((likes\ Mary\ X)\ \supset\ (likes\ Sue\ X))))\ cw$
Q: $(\Box_{Chris}\ (likes\ Sue\ Bill))\ cw$

Challenge: Embedded Formulas — Epistemic Context

Example (F: Embedded Formulas – Epistemic Contexts)

A: $(\Box_{Chris}\,(equal\ Chris\ Chris))\ cw$
B: $(likes\ Mary\ Bill)\ cw$
C: $(\Box_{Chris}\,(\forall^i X_{\mu}.\,((likes\ Mary\ X) \supset (likes\ Sue\ X))))\ cw$
Q: $(\Box_{Chris}\,(likes\ Sue\ Bill))\ cw$

Axioms for $\Box_{Chris}$ can be added:

M: $\forall W_{\iota}\,(\forall^p \varphi_{\iota\to o}.\,\Box_{Chris}\,\varphi \supset \varphi)\ W$
4: $\forall W_{\iota}\,(\forall^p \varphi_{\iota\to o}.\,\Box_{Chris}\,\varphi \supset \Box_{Chris}\,\Box_{Chris}\,\varphi)\ W$
5: $\forall W_{\iota}\,(\forall^p \varphi_{\iota\to o}.\,\Box_{Chris}\,\neg\,\varphi \supset \Box_{Chris}\,\neg\,\Box_{Chris}\,\varphi)\ W$

Challenge: Embedded Formulas — Epistemic Context

Example (F: Embedded Formulas – Epistemic Contexts)

A: $(\Box_{Chris}\ (equal\ Chris\ Chris))\ cw$
B: $(likes\ Mary\ Bill)\ cw$
C: $(\Box_{Chris}\ (\forall^i X_\mu.((likes\ Mary\ X)\ \supset\ (likes\ Sue\ X))))\ cw$
Q: $(\Box_{Chris}\ (likes\ Sue\ Bill))\ cw$

Axioms for $\Box_{Chris}$ can be added:

M: $\forall W_\iota\ (\forall^p \varphi_{\iota\to o}.\Box_{Chris}\,\varphi\ \supset\ \varphi)\ W$
4: $\forall W_\iota\ (\forall^p \varphi_{\iota\to o}.\Box_{Chris}\,\varphi\ \supset\ \Box_{Chris}\,\Box_{Chris}\,\varphi)\ W$
5: $\forall W_\iota\ (\forall^p \varphi_{\iota\to o}.\Box_{Chris}\,\neg\,\varphi\ \supset\ \Box_{Chris}\,\neg\,\Box_{Chris}\,\varphi)\ W$

LEO-II cannot solve this problem anymore! Countermodel exists.

Challenge: Embedded Formulas — Epistemic Context

Example (F: Embedded Formulas – Epistemic Contexts)

A: $(\Box_{Chris}\ (equal\ Chris\ Chris))\ cw$
B: $(\Box_{Chris}\ (likes\ Mary\ Bill))\ cw$
C: $(\Box_{Chris}\ (\forall^i X_{\mu}\blacksquare((likes\ Mary\ X)\ \supset\ (likes\ Sue\ X))))\ cw$
Q: $(\Box_{Chris}\ (likes\ Sue\ Bill))\ cw$

Axioms for $\Box_{Chris}$ can be added:

M: $\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\blacksquare \Box_{Chris}\,\varphi\ \supset\ \varphi)\ W$
4: $\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\blacksquare \Box_{Chris}\,\varphi\ \supset\ \Box_{Chris}\,\Box_{Chris}\,\varphi)\ W$
5: $\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\blacksquare \Box_{Chris}\neg\varphi\ \supset\ \Box_{Chris}\neg\Box_{Chris}\,\varphi)\ W$

But LEO-II(+E) can solve this problem in milliseconds!

Challenge: Embedded Formulas — Epistemic Context

Example (F: Embedded Formulas – Epistemic Contexts)

A: $(\Box_{Chris}\ (equal\ Chris\ Chris))\ cw$
B: $(\Box_{fool}\ (likes\ Mary\ Bill))\ cw$
C: $(\Box_{Chris}\ (\forall^i X_{\mu}\text{▪}((likes\ Mary\ X) \supset (likes\ Sue\ X))))\ cw$
Q: $(\Box_{Chris}\ (likes\ Sue\ Bill))\ cw$

Axioms for $\Box_{Chris}$ can be added:

M: $\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\text{▪}\Box_{Chris}\ \varphi \supset \varphi)\ W$
4: $\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\text{▪}\Box_{Chris}\ \varphi \supset \Box_{Chris}\ \Box_{Chris}\ \varphi)\ W$
5: $\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\text{▪}\Box_{Chris} \neg \varphi \supset \Box_{Chris} \neg \Box_{Chris}\ \varphi)\ W$

Axioms for $\Box_{fool}$ can be added . . .
$\forall W_{\iota}\ (\forall^p \varphi_{\iota\to o}\text{▪}\Box_{fool}\ \varphi \supset \Box_{Chris}\ \varphi)\ W$
. . .

SUMO Ontology — Proposal

Redevelop entire SUMO Ontology in HOML (multimodal, eventually other logics)

Should give a proper semantics for SUMO

Employ HOML embedding in HOL to automated reasoning

Experiments with Large Knowledge Bases (SUMO)

Significant Improvements for Large Theories (PAAR-2010)

LEO-II(+E) version v1.1

Ex.	A	B	C	D	E			F	
local	.19	.19	.13	.16	.08	.34	.18	.04	2642.55
SInE	–	–	–	–	–	–	–	–	–
global	–	–	–	–	–	–	–	–	–

global: all SUMO axioms given to LEO-II
SInE: filters SUMO axioms for problem — ˜400 axioms given to LEO-II
local: only handselected axioms given to LEO-II

Further reading and more experiments [BenzmüllerPease J.WebSemantics 2012]

Significant Improvements for Large Theories (PAAR-2010)

LEO-II(+E) version v1.1

Ex.	A	B	C	D	E			F	
local	.19	.19	.13	.16	.08	.34	.18	.04	2642.55
SInE	–	–	–	–	–	–	–	–	–
global	–	–	–	–	–	–	–	–	–

global: all SUMO axioms given to LEO-II
SInE: filters SUMO axioms for problem — ˜400 axioms given to LEO-II
local: only handselected axioms given to LEO-II

LEO-II(+E) version v1.2.1 (with relevance filtering)

Ex.	A	B	C	D	E			F	
local	.19	.18	.11	.08	.10	.38	.32	.14	.18
SInE	.43	.40	.21	.54	.37	.12	.70	.06	.26
global	2.8	2.7	1.6	4.9	1.4	0.9	4.7	1.3	0.9

Further reading and more experiments [BenzmüllerPease J.WebSemantics 2012]

Meta-Reasoning

Description logic $\mathcal{ALC}$

Syntax	Semantics	Description	Example
A	$A^I \subseteq \Delta^I$	atomic concept	$Human, Female, \ldots$
r	$r^I \subseteq \Delta^I \times \Delta^I$	binary relation	$married, \ldots$
$\bot$	$\emptyset$	empty concept	
$\top$	Δ^I	universal concept	
$\sim A$	$\Delta^I \setminus A^I$	complement	$\sim Female$
$A \sqcup B$	$A^I \cup B^I$	disjunction	$Female \sqcup Male$
$A \sqcap B$	$A^I \cap B^I$	conjunction	$Female \sqcap Human$
$\exists r\ C$	$\{x \mid \exists y. r^I(x,y) \wedge C^I(y)\}$	existential restriction	$\exists married\ Female$
$\forall r\ C$	$\{x \mid \forall y. r^I(x,y) \rightarrow C^I(y)\}$	universal restriction	$\forall married\ Female$

Description logic $\mathcal{ALC}$

Syntax	Semantics	Description	Example
A	$A^I \subseteq \Delta^I$	atomic concept	$Human, Female, \ldots$
r	$r^I \subseteq \Delta^I \times \Delta^I$	binary relation	$married, \ldots$
$\bot$	$\emptyset$	empty concept	
$\top$	Δ^I	universal concept	
$\sim A$	$\Delta^I \setminus A^I$	complement	$\sim Female$
$A \sqcup B$	$A^I \cup B^I$	disjunction	$Female \sqcup Male$
$A \sqcap B$	$A^I \cap B^I$	conjunction	$Female \sqcap Human$
$\exists r\ C$	$\{x \mid \exists y.r^I(x,y) \wedge C^I(y)\}$	existential restriction	$\exists married\ Female$
$\forall r\ C$	$\{x \mid \forall y.r^I(x,y) \rightarrow C^I(y)\}$	universal restriction	$\forall married\ Female$

Simple exercises (useful lemmata)

$$\top = A \sqcup \sim A \qquad (L1)$$

$$\bot = \sim \top \qquad (L2)$$

$$A \sqcap B = \sim(\sim A \sqcup \sim B) \qquad (L3)$$

$$\forall r\ C = \sim(\exists r \sim C) \qquad (L4)$$

Description logic $\mathcal{ALC}$

Syntax	Semantics	Description	Example
A	$A^I \subseteq \Delta^I$	atomic concept	$Human, Female, \ldots$
r	$r^I \subseteq \Delta^I \times \Delta^I$	binary relation	$married, \ldots$
$\sim A$	$\Delta^I \setminus A^I$	complement	$\sim Female$
$A \sqcup B$	$A^I \cup B^I$	disjunction	$Female \sqcup Male$
$\exists r\ C$	$\{x \mid \exists y. r^I(x,y) \wedge C^I(y)\}$	existential restriction	$\exists married\ Female$

Simple exercises (useful lemmata)

$$\top = A \sqcup \sim A \qquad (L1)$$

$$\bot = \sim\top \qquad (L2)$$

$$A \sqcap B = \sim(\sim A \sqcup \sim B) \qquad (L3)$$

$$\forall r\ C = \sim(\exists r \sim C) \qquad (L4)$$

Description logic $\mathcal{ALC}$

Syntax	Semantics	Description	Example
$A \sqsubseteq B$	$A^I \subseteq B^I$	B subsumes A	$Doctor \sqsubseteq Human$
$A \doteq B$	$A^I \sqsubseteq B^I$ *und* $B^I \sqsubseteq A^I$	A defined by B	$Parent \doteq$ $Human \sqcap \exists hasChild\ Human$

Description logic $\mathcal{ALC}$

Syntax	Semantics	Description	Example
$A \sqsubseteq B$	$A^I \subseteq B^I$	B subsumes A	$Doctor \sqsubseteq Human$
$A \doteq B$	$A^I \sqsubseteq B^I$ und $B^I \sqsubseteq A^I$	A defined by B	$Parent \doteq Human \sqcap \exists hasChild\ Human$

Simple exercises:

$$\begin{array}{llll} A \sqsubseteq B & gdw. & \exists {\color{red}C}.A \doteq {\color{red}C} \sqcap B & (L5) \\ A \sqsubseteq B & gdw. & (A \sqcap \sim B) \sqsubseteq \bot & (L6) \\ & gdw. & \exists x.(A \sqcap \sim B)(x) \text{ unerfüllbar} & \end{array}$$

Knowledge representation in $\mathcal{ALC}$

TBox (terminological knowledge, taxonomy)

Example:

$$
\begin{aligned}
\mathit{HappyMan} &\doteq \mathit{Human} \ \sqcap \sim\!\mathit{Female} \sqcap \\
&\quad (\exists \mathit{married}\ \mathit{Doctor}) \sqcap (\forall \mathit{hasChild}(\mathit{Doctor} \sqcup \mathit{Professor})) \\
\mathit{Doctor} &\sqsubseteq \mathit{Human}
\end{aligned}
$$

Knowledge representation in $\mathcal{ALC}$

TBox (terminological knowledge, taxonomy)

Example:

$$\begin{aligned}
\mathit{HappyMan} &\doteq \mathit{Human} \sqcap {\sim}\mathit{Female} \sqcap \\
&\quad (\exists \mathit{married}\ \mathit{Doctor}) \sqcap (\forall \mathit{hasChild}(\mathit{Doctor} \sqcup \mathit{Professor})) \\
\mathit{Doctor} &\sqsubseteq \mathit{Human}
\end{aligned}$$

ABox (assertional knowledge, e.g. assumptions on indiviuals)

Example:

$$\mathit{HappyMan}(\mathit{BOB}), \quad \mathit{hasChild}(\mathit{BOB}, \mathit{MARY}), \quad {\sim}\mathit{Doctor}(\mathit{MARY})$$

EINSCHUB: Knowledge representation & Inference about ALC in HOL

Animation: Max Benzmüller (5 years)

EINSCHUB: Knowledge representation & Inference about ALC in HOL

Animation: Max Benzmüller (5 years)

EINSCHUB: Knowledge representation & Inference about ALC in HOL

Animation: Max Benzmüller (5 years)

EINSCHUB: Knowledge representation & Inference about ALC in HOL

Animation: Max Benzmüller (5 years)

EINSCHUB: Meta theory of ALC in Isabelle/HOL

ALC.thy (~/chris/trunk/tex/talks/2015-Erlangen/)

```
theory ALC  imports Main begin

typedecl i  type_synonym τ = "(i ⇒ bool)" type_synonym σ = "(i ⇒ i ⇒ bool)"

abbreviation bot ::"τ"    ("⊥")              where "⊥     ≡ λx. False"
abbreviation top ::"τ"    ("⊤")              where "⊤     ≡ λx. True"
abbreviation neg          ("∼")              where "∼A    ≡ λx. ¬A(x)"
abbreviation disj         (infixr "⊔" 40)    where "A⊔B   ≡ λx. A(x) ∨ B(x)"
abbreviation conj         (infixr "⊓" 41)    where "A⊓B   ≡ λx. A(x) ∧ B(x)"
abbreviation exi_r        ("∃")              where "∃r A  ≡ λx. ∃y. r x y ∧ A(y)"
abbreviation all_r        ("∀")              where "∀r A  ≡ λx. ∀y. r x y ⟶ A(y)"

abbreviation sub (infixr "⊑" 39)             where "A⊑B   ≡ ∀x. A(x) ⟶ B(x)"
abbreviation eq  (infixr "≜" 38)             where "A≜B   ≡ A ⊑ B ∧ B ⊑ A"

(* Einfaches Beispiele für etwas Meta-Theorie *)
lemma "A⊓B ≜ ∼(∼A⊔∼B)" by metis
lemma "∃r C ≜ ∼(∀r (∼C))" by metis  (* sledgehammer [remote_leo2] *)
lemma "A⊓B ≜ A⊔B" nitpick oops
```

100% | Auto update | Update | Detach

```
Nitpicking formula...
Nitpick found a counterexample for card 'a = 2:

  Free variables:
    A = (λx. _)(a₁ := False, a₂ := False)
    B = (λx. _)(a₁ := False, a₂ := True)
```

Output README Symbols

19,22 (939/2329) (isabelle,sidekick,UTF-8-Isabelle) 9:12 PM

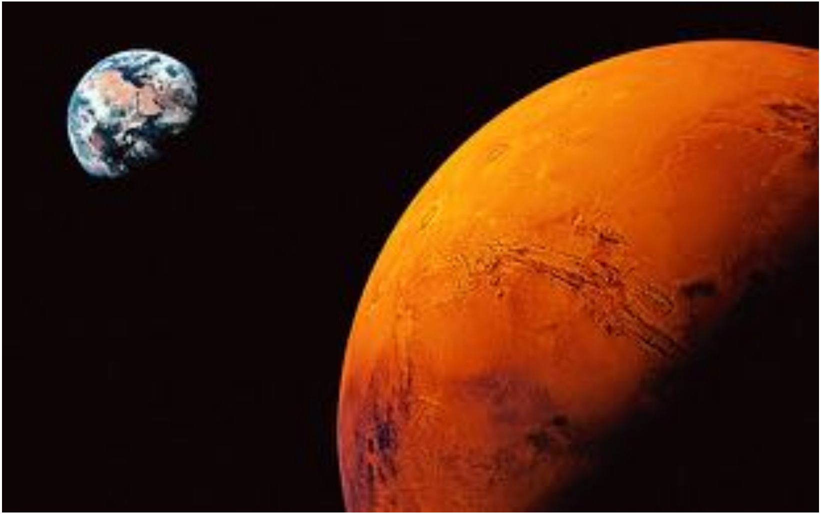
Flexibility in HOML

Many Variations of Higher-Order Modal Logics

- (Meta-)Axioms for the accessibility relation (e.g. reflexivity, transitivity, symmetry)
- Axioms (e.g. $\Box A \rightarrow A$, $\Box A \rightarrow \Box\Box A$, $A \rightarrow \Box\Diamond A$)
- Possibilistic vs. Actualistic Quantifiers
- Constant Domains vs. Varying Domains vs. Cumulative Domains
- Rigidity vs. Flexibility
- Simple Types vs. Dependent Types (e.g. μ vs. $\mu(w)$)
- ...

Flexibility
A Funny Example

$$\Box Blue(sky)$$



Flexibility
A Funny Example

$$\Box Blue(sky)$$

Flexibility
A Funny Example

$$\Box Blue(sky)$$

Human: “Earth’s sky is blue”

Martian: “Mars’ sky is blue”

Flexibility

$$\Box Blue(sky)$$

- Rigid embedding:

$$\lceil \mu \rceil = \mu \qquad \lceil o \rceil = \iota \rightarrow o \qquad \lceil \alpha \rightarrow \beta \rceil = \lceil \alpha \rceil \rightarrow \lceil \beta \rceil$$

$$\lceil sky_{\mu} \rceil = sky_{\mu}$$
$$\lceil Blue_{\mu\rightarrow o} \rceil = Blue_{\mu\rightarrow(\iota\rightarrow o)}$$

- Flexible embedding:

$$\lceil \mu \rceil = \iota \rightarrow \mu \qquad \lceil o \rceil = \iota \rightarrow o \qquad \lceil \alpha \rightarrow \beta \rceil = \iota \rightarrow \lceil \alpha \rceil \rightarrow \lceil \beta \rceil$$

$$\lceil sky_{\mu} \rceil = sky_{\iota\rightarrow\mu}$$
$$\lceil Blue_{\mu\rightarrow o} \rceil = Blue_{\iota\rightarrow(\iota\rightarrow\mu)\rightarrow(\iota\rightarrow o)}$$

Flexibility
Flexible Embedding is Ambiguous

$$[\Box Blue(sky)]$$

$$\forall w.\forall w'.\ Blue\ w\ (sky\ w)\ w'$$

$$\forall w.\forall w'.\ Blue\ w'\ (sky\ w')\ w'$$

$$\forall w.\forall w'.\ Blue\ w'\ (sky\ w)\ w'$$

$$\forall w.\forall w'.\ Blue\ w\ (sky\ w')\ w'$$

Reasoning with Inconsistencies!

Paraconsistency

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$
\begin{array}{l}
Fruit(mango) \\
Fruit(tomato) \\
RichIn(tomato, lycopene) \\
Antioxidant(lycopene)
\end{array}
$$

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$\begin{array}{l} Fruit(mango) \\ Fruit(tomato) \\ RichIn(tomato, lycopene) \\ Antioxidant(lycopene) \end{array}$$

A cook's KB:

$$Fruit(mango)$$

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$
\begin{array}{l}
Fruit(mango) \\
Fruit(tomato) \\
RichIn(tomato, lycopene) \\
Antioxidant(lycopene)
\end{array}
$$

A cook's KB:

$$
\begin{array}{l}
Fruit(mango) \\
\forall x. Fruit(x) \rightarrow Dessert(x)
\end{array}
$$

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$Fruit(mango)$$
$$Fruit(tomato)$$
$$RichIn(tomato, lycopene)$$
$$Antioxidant(lycopene)$$

A cook's KB:

$$Fruit(mango)$$
$$\forall x. Fruit(x) \rightarrow Dessert(x)$$
$$\neg Dessert(tomato)$$

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$\begin{array}{l} Fruit(mango) \\ Fruit(tomato) \\ RichIn(tomato, lycopene) \\ Antioxidant(lycopene) \end{array}$$

A cook's KB:

$$\begin{array}{l} Fruit(mango) \\ \forall x. Fruit(x) \rightarrow Dessert(x) \\ \neg Dessert(tomato) \\ Vegetable(tomato) \end{array}$$

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$\begin{array}{l} Fruit(mango) \\ Fruit(tomato) \\ RichIn(tomato, lycopene) \\ Antioxidant(lycopene) \end{array}$$

A cook's KB:

$$\begin{array}{l} Fruit(mango) \\ \forall x. Fruit(x) \rightarrow Dessert(x) \\ \neg Dessert(tomato) \\ Vegetable(tomato) \\ \forall x. Vegetable(x) \rightarrow SaladIngredient(x) \end{array}$$

Example of Knowledge Bases with Inconsistent Information

A biologist's KB:

$$
\begin{array}{l}
\mathit{Fruit}(\mathit{mango}) \\
\mathit{Fruit}(\mathit{tomato}) \\
\mathit{RichIn}(\mathit{tomato}, \mathit{lycopene}) \\
\mathit{Antioxidant}(\mathit{lycopene})
\end{array}
$$

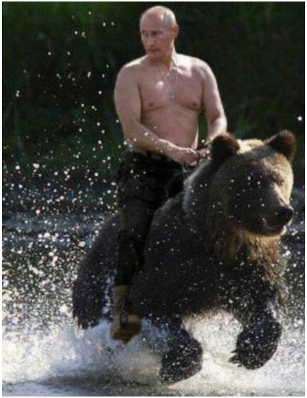
A cook's KB:

$$
\begin{array}{l}
\mathit{Fruit}(\mathit{mango}) \\
\forall x. \mathit{Fruit}(x) \rightarrow \mathit{Dessert}(x) \\
\neg \mathit{Dessert}(\mathit{tomato}) \\
\mathit{Vegetable}(\mathit{tomato}) \\
\forall x. \mathit{Vegetable}(x) \rightarrow \mathit{SaladIngredient}(x) \\
\forall x. \neg(\mathit{Vegetable}(x) \wedge \mathit{Fruit}(x))
\end{array}
$$

Another Example

Information Extracted from CNN's Website:

$Evil(putin)$



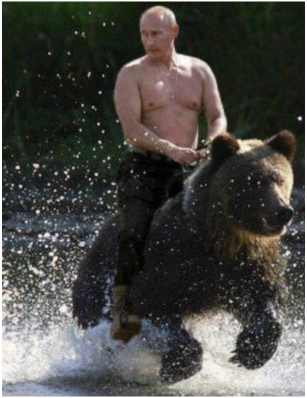
Another Example

Information Extracted from CNN's Website:

$$Evil(putin)$$

Information Extracted from Russia Today's Website:

$$Hero(putin)$$



Another Example

Information Extracted from CNN's Website:

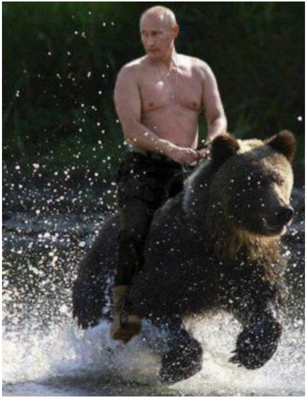
$$Evil(putin)$$

Information Extracted from Russia Today's Website:

$$Hero(putin)$$

Common Knowledge:

$$\forall x. Evil(x) \leftrightarrow \neg Hero(x)$$



In Classical Logic, the Principle of Explosion Holds

Ex Contradictione Quod Libet - From a Contradiction Everything Follows

$$\forall P.\forall Q.(P \land \neg P) \to Q$$

$$\bot \to \forall Q.Q$$

$$(Fruit(tomato) \land \neg Fruit(tomato)) \to \exists x.G(x)$$

Paraconsistent Logics

Logics where the principle of Explosion does not hold

Classification of Paraconsistent Logics:

- Where is the inconsistency?
  - Dialetheism: Reality (or the model) is actually inconsistent. Facts can be simultaneously true and false in the model.
  - Reality is consistent, but our theories about reality may be inconsistent.
- How do we reason properly despite inconsistencies? Several Options:
  - Many-Valued Semantics
  - Relevant Logics
  - Default Logics
  - Belief Revision
  - Jaskowski's Discussive Logics
  - Hybrid Flexible Discussive Logics

Jaskowski's Discussive Logics (1948)

$\Diamond P$ = "someone claims that P"
$\Box P$ = "everybody claims that P"

$$\Box P \rightarrow P$$

$$\Diamond Evil(putin) \wedge \Diamond\neg Evil(putin)$$

$$(\Diamond Fruit(tomato) \wedge \Diamond\neg Fruit(tomato))$$

Problem:

- asumme $\Diamond P$
- assume $\Diamond(P \rightarrow Q)$
- can we derive $\Diamond Q$?
  - not in standard modal logics
  - Jaskowkski attempts to circumvent this problem
  - what we really need:
    a way to refer explicitly to the participants

Jaskowski's Discussive Logics (1948)

$\Diamond P$ = "someone claims that P"
$\Box P$ = "everybody claims that P"

$$\Box P \rightarrow P$$

$$\Diamond Evil(putin) \wedge \Diamond \neg Evil(putin)$$

$$(\Diamond Fruit(tomato) \wedge \Diamond \neg Fruit(tomato))$$

Problem:

- asumme $\Diamond P$ (claimed by discussion's participant S)
- assume $\Diamond (P \rightarrow Q)$ (also claimed by S)
- can we derive $\Diamond Q$?
  - not in standard modal logics
  - Jaskowkski attempts to circumvent this problem
  - what we really need:
    a way to refer explicitly to the participants

Hybrid Logics
Modal Logics with Explicit References to Worlds

$@_w\ P$ = "P holds at world w"

$in\ w$ = "current world is w"

$$@ = \lambda w.\lambda P.\lambda w_0.(Pw)$$

$$in = \lambda w.\lambda w_0.w = w_0$$

Hybrid Discussive Logics

$@_w\, P$ = "source/participant w claims that P"

$in\ w$ = "current source/participant is w"

Hybrid Discussive Logics
Solving The Putin Problem

$$@_{russiatoday} Hero(putin)$$

$$@_{cnn} \neg Hero(putin)$$

Hybrid Discussive Logics
Solving The Putin Problem

$$@_{russiatoday} Hero(putin)$$

$$@_{cnn} \neg Hero(putin)$$

No contradiction!
$\perp$ cannot be derived.

Hybrid Flexible Discussive Logics
Solving The Tomato Problem

$$[@_{w_{cook}} \neg(Fruit\ w_{cook}\ tomato)]$$

$$[@_{w_{biologist}} \neg(Fruit\ w_{biologist}\ tomato)]$$

$$[@_{w_{cook}} \neg(Fruit\ w_{cook}\ tomato)]$$

$$[@_{w_{biologist}} \neg(Fruit\ w_{biologist}\ tomato)]$$

Importing Knowledge from Different Sources:

$$\forall x_\mu. \forall P_{\iota\to\mu\to\iota\to o}. \forall w. (r\ w_{our}\ w) \to (P\ w\ x\ w) \to (P\ w\ x\ w_{our})$$

Hybrid Flexible Discussive Logics

Solving The Tomato Problem

$$[@_{w_{cook}} \neg (Fruit\ w_{cook}\ tomato)]$$

$$[@_{w_{biologist}} \neg (Fruit\ w_{biologist}\ tomato)]$$

Importing Knowledge from Different Sources:

$$\forall x_\mu . \forall P_{\iota \to \mu \to \iota \to o} . \forall w . (r\ w_{our}\ w) \to (P\ w\ x\ w) \to (P\ w\ x\ w_{our})$$

Convenient Renaming of Concepts:

$$ReallyFruit = (Fruit\ w_{biologist})$$

Hybrid Flexible Discussive Logics
Solving The Tomato Problem

$$[@_{w_{cook}} \neg(Fruit\ w_{cook}\ tomato)]$$

$$[@_{w_{biologist}} \neg(Fruit\ w_{biologist}\ tomato)]$$

Importing Knowledge from Different Sources:

$$\forall x_\mu . \forall P_{\iota\to\mu\to\iota\to o} . \forall w . (r\ w_{our}\ w) \to (P\ w\ x\ w) \to (P\ w\ x\ w_{our})$$

Convenient Renaming of Concepts:

$$ReallyFruit = (Fruit\ w_{biologist})$$

$$DessertFruit = (Fruit\ w_{cook})$$

Cut Elimination

Cut-Elimination versus Cut-Simulation

[BenzmüllerBrownKohlhase, Cut-Simulation in Impredicative Logics, LMCS, 2009]

- studies Henkin complete, one-sided sequent calculi for HOL
- cut-elimination proved for a 'naive' calculus
- cut-simulation shown for this calculus
- improved calculi presented that avoid cut-simulation effects

- Why relevant?

  Ideas of the improved calculi are also present in
  LEO-II (resolution) and Satallax (tableaux)

  'Free' cut-elimination result for HOML (and other embedded logics);
  cut-simulation issues

Embedding of Other Logics in HOL: Theoretical Results

Soundness and Completeness (and Cut-elimination)

$$\models^{L} s_o \quad \text{iff} \quad Ax \models^{HOL}_{\text{Henkin}} valid\ s_{\iota\to o} \quad (\text{iff} \quad Ax \vdash^{\text{HOL}}_{\text{cut-free}} valid\ \varphi_{\iota\to o})$$

Logic L:

- Higher-order Modal Logics
- First-order Multimodal Logics
- Propositional Multimodal Logics
- Quantified Conditional Logics
- Propositional Conditional Logics
- Intuitionistic Logics
- Access Control Logics
- Logic Combinations
- ...more is on the way ... including:
  - Description Logics
  - Nominal Logics
  - Multivalued Logics (SIXTEEN)
  - Logics based on Neighborhood Semantics
  - (Mathematical) Fuzzy Logics
  - Paraconsistent Logics

One-sided Sequent Calculus G1

Δ and Δ': finite sets of β-normal closed formulas
$\Delta, \mathbf{A}$ stands for $\Delta \cup \{\mathbf{A}\}$
$l \doteq r$ denotes Leibniz equality: $\Pi(\lambda P_{\alpha\to o}(\neg Pl \vee Pr))$

Basic Rules

$$\frac{\Delta, s}{\Delta, \neg\neg s}\,\mathcal{G}(\neg) \qquad \frac{\Delta, \neg s \quad \Delta, \neg t}{\Delta, \neg(s \vee t)}\,\mathcal{G}(\vee_{-}) \qquad \frac{\Delta, s, t}{\Delta, (s \vee t)}\,\mathcal{G}(\vee_{+})$$

$$\frac{\Delta, \neg\ (sl){\downarrow}_{\beta} \quad l_{\alpha} \text{ closed term}}{\Delta, \neg\Pi^{\alpha} s}\,\mathcal{G}(\Pi^{l}_{-}) \qquad \frac{\Delta,\ (sc){\downarrow}_{\beta} \quad c_{\delta} \text{ new symbol}}{\Delta, \Pi^{\alpha} s}\,\mathcal{G}(\Pi^{c}_{+})$$

Initialization

$$\frac{s \text{ atomic (and } \beta\text{-normal)}}{\Delta, s, \neg s}\,\mathcal{G}(init)$$

One-sided Sequent Calculus G1

Δ and Δ': finite sets of β-normal closed formulas
$\Delta, \mathbf{A}$ stands for $\Delta \cup \{\mathbf{A}\}$
$l \doteq r$ denotes Leibniz equality: $\Pi(\lambda P_{\alpha \to o}(\neg Pl \vee Pr))$

Basic Rules

$$\frac{\Delta, s}{\Delta, \neg\neg s}\,\mathcal{G}(\neg) \qquad \frac{\Delta, \neg s \quad \Delta, \neg t}{\Delta, \neg(s \vee t)}\,\mathcal{G}(\vee_-) \qquad \frac{\Delta, s, t}{\Delta, (s \vee t)}\,\mathcal{G}(\vee_+)$$

$$\frac{\Delta, \neg\ (sl){\downarrow_\beta} \quad l_\alpha \text{ closed term}}{\Delta, \neg\Pi^\alpha s}\,\mathcal{G}(\Pi^l_-) \qquad \frac{\Delta,\ (sc){\downarrow_\beta} \quad c_\delta \text{ new symbol}}{\Delta, \Pi^\alpha s}\,\mathcal{G}(\Pi^c_+)$$

Initialization

$$\frac{s \text{ atomic (and } \beta\text{-normal)}}{\Delta, s, \neg s}\,\mathcal{G}(init)$$

Boolean extensionality axiom ($\mathcal{B}_o$)

$$\forall A_o \forall B_o((A \longleftrightarrow B) \to A \doteq^o B) \qquad \frac{\Delta, \neg\mathcal{B}_o}{\Delta}\,\mathcal{G}(\mathcal{B})$$

Infinitely many functional extensionality axioms ($\mathcal{F}_{\alpha\beta}$)

$$\forall F_{\alpha\to\beta} \forall G_{\alpha\to\beta}(\forall X_\alpha(FX \doteq^\beta GX) \to F \doteq^{\alpha\to\beta} G) \qquad \frac{\Delta, \neg\mathcal{F}_{\alpha\beta} \quad \alpha \to \beta \in \mathcal{T}}{\Delta}\,\mathcal{G}(\mathcal{F}_{\alpha\beta})$$

One-sided Sequent Calculus $G1$

Theorem (Soundness/Completeness [BenzmüllerBrownKohlhase, LMCS, 2009])

G1 is sound and complete for HOL: $\models^{HOL} s$ *iff* $\vdash^{G1} s$

Theorem (Cut-elimination [BenzmüllerBrownKohlhase, LMCS, 2009])

The rule $\mathcal{G}(cut)$

$$\frac{\Delta, s \quad \Delta, \neg s}{\Delta}\ \mathcal{G}(cut)$$

is admissible in G1.

But: G1 supports effective simulation of the cut-rule!

In other words: the above cut-elimination result is meaningless.

One-sided Sequent Calculus $G1$

Cut-simulation with the Boolean extensionality axiom

derivable in 7 steps

$$
\dfrac{
  \dfrac{\begin{array}{c}\vdots\\ \Delta, a \longleftrightarrow a\end{array}}{\Delta, \neg\neg(a \longleftrightarrow a)}\ \mathcal{G}(\neg)
  \qquad
  \begin{array}{c}\Delta, s \quad \Delta, \neg s\\ \vdots\\ \Delta, \neg(a \doteq^o a)\end{array}
}{
  \dfrac{\Delta, \neg(\neg(a \longleftrightarrow a) \vee a \doteq^o a)}{\Delta, \neg\mathcal{B}_o}\ 2 \times \mathcal{G}(''^{a}_{-})
}\ \mathcal{G}(\vee_-)
$$

derivable in 3 steps, see below

One-sided Sequent Calculus $G1$

Cut-simulation with the Boolean extensionality axiom

derivable in 7 steps

derivable in 3 steps, see below

$$
\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\\ \Delta, a \longleftrightarrow a\end{array}}{\Delta, \neg\neg(a \longleftrightarrow a)}\;\mathcal{G}(\neg) \qquad \dfrac{\Delta, s \quad \Delta, \neg s}{\begin{array}{c}\vdots\\ \Delta, \neg(a \doteq^{o} a)\end{array}}}{\Delta, \neg(\neg(a \longleftrightarrow a) \vee a \doteq^{o} a)}\;\mathcal{G}(\vee_{-})}{\Delta, \neg\mathcal{B}_{o}}\;2 \times \mathcal{G}(\stackrel{''a}{-})
$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{\Delta, s}{\Delta, \neg\neg s}\;\mathcal{G}(\neg) \qquad \Delta, \neg s}{\Delta, \neg(\neg s \vee s)}\;\mathcal{G}(\vee_{-})}{\Delta, \neg\forall P_{\alpha\to o}(\neg Pa \vee Pa)}\;\mathcal{G}(\stackrel{''\lambda X\, s}{-})}{\Delta, \neg(a \doteq^{o} a)}\;def.
$$

One-sided Sequent Calculus $G1$

Cut-simulation with functional extensionality axiom

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{\text{derivable in 3 steps}}{\vdots}
\atop
\Delta, fb \doteq^{\beta} fb
}{\Delta, (\forall X_\alpha fX \doteq^{\beta} fX)}\;\mathcal{G}(\doteq^{b}_{+})
}{\Delta, \neg\neg\forall X_\alpha fX \doteq^{\beta} fX}\;\mathcal{G}(\neg)
\qquad
\dfrac{\Delta, {\color{red}s} \quad \Delta, {\color{red}\neg s}}{\vdots\ \text{derivable in 3 steps}}
\atop
\Delta, \neg(f \doteq^{\alpha\to\beta} f)
}{
\dfrac{\Delta, \neg(\neg(\forall X_\alpha fX \doteq^{\beta} fX) \vee f \doteq^{\alpha\to\beta} f)}{\Delta, \neg\mathcal{F}_{\alpha\beta}}\;2\times\mathcal{G}(\doteq^{f}_{-})
}\;\mathcal{G}(\vee_{-})
$$

One-sided Sequent Calculus G2

Basic Rules

$$\frac{\Delta, s}{\Delta, \neg\neg s}\,\mathcal{G}(\neg) \qquad \frac{\Delta, \neg s \quad \Delta, \neg t}{\Delta, \neg(s \vee t)}\,\mathcal{G}(\vee_-) \qquad \frac{\Delta, s, t}{\Delta, (s \vee t)}\,\mathcal{G}(\vee_+)$$

$$\frac{\Delta, \neg\ (sl){\downarrow_\beta} \quad l_\alpha \text{ closed term}}{\Delta, \neg\Pi^\alpha s}\,\mathcal{G}(\Pi^l_-) \qquad \frac{\Delta,\ (sc){\downarrow_\beta} \quad c_\delta \text{ new symbol}}{\Delta, \Pi^\alpha s}\,\mathcal{G}(\Pi^c_+)$$

Initialization

$$\frac{s \text{ atomic (and } \beta\text{-normal)}}{\Delta, s, \neg s}\,\mathcal{G}(init) \qquad \frac{\Delta, (s \doteq^o t) \quad s, t \text{ atomic}}{\Delta, \neg s, t}\,\mathcal{G}(Init^{\doteq})$$

Extensionality Rules

$$\frac{\Delta,\ (\forall X_\alpha\, sX \doteq^\beta tX){\downarrow_\beta}}{\Delta, (s \doteq^{\alpha\to\beta} t)}\,\mathcal{G}(\mathfrak{f}) \qquad \frac{\Delta, \neg s, t \quad \Delta, \neg t, s}{\Delta, (s \doteq^o t)}\,\mathcal{G}(\mathfrak{b})$$

$$\frac{\Delta, (s^1 \doteq^{\alpha_1} t^1) \ \cdots \ \Delta, (s^n \doteq^{\alpha_n} t^n) \qquad \begin{array}{l} n \geq 1, \beta \in \{o, \iota\}, \\ h_{\overline{\alpha^n}\to\beta} \in \Sigma \end{array}}{\Delta, (h\overline{s^n} \doteq^\beta h\overline{t^n})}\,\mathcal{G}(d)$$

Cut-Simulation with Prominent Axioms

▶ Axiom of excluded middle	3 steps
▶ Instances of the comprehension axioms	16 steps
▶ Leibniz equations (axioms/hypotheses)	3 steps
▶ Reflexivity definition of equality (Andrews)	4 steps
▶ Axiom of functional extensionality	11 steps
▶ Axiom of Boolean extensionality	14 steps
▶ Axioms of choice	7 steps
▶ Axiom of description	25 steps
▶ Axiom of Iinduction	18 steps

Consequence: HOL-ATPs should better avoid these axioms!

Problem: Postulating axioms, e.g. in modal logics or conditional logics, may eventually lead to cut-simulation issues.

Conclusion

Conclusion: Wider Perspective

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

Conclusion

Summary of Contributions

- Efficient automated reasoning for HOML
- User-friendly interactive reasoning with HOML (in Coq)
- A new natural deduction calculus for HOML
- Mature Technology

Relevance (wrt foundations and applications)

- Philosophy, AI, Computer Science, Computational Linguistics, Maths

Success Stories

- significant contribution towards a **Computational Metaphysics**
  - Ed Zalta's Computational Metaphysics project at Stanford University
  - John Rushby's formalization of Anselm's proof using PVS
- **novel results** (e.g. inconsistency) contributed by **HOL-ATPs**
- Resolution of philosophical controversies (Leibniz's dream)
- Non-trivial new benchmark problems for HOL provers
- infrastructure can be adapted for **other logics and logic combinations**: SUMO, Paraconsistent Logics, Deontic Logics, . . .

Various Unordered Stuff is following

Theorem Provers for HOL

Higher-Order Automated Theorem Provers (HOL-ATPs)

HOL-ATPs

EU FP7 Project THFTPTP

- Collaboration with Geoff Sutcliffe and others (Chad Brown, Florian Rabe, Nik Sultana, Jasmin Blanchette, Frank Theiss, . . .)
- Results
  - THF0 syntax for HOL (with Choice; Henkin Semantics)
  - library with example problems (e.g. entire TPS library) and results
  - international CASC competition for HOL-ATP
  - online access to provers
  - various tools

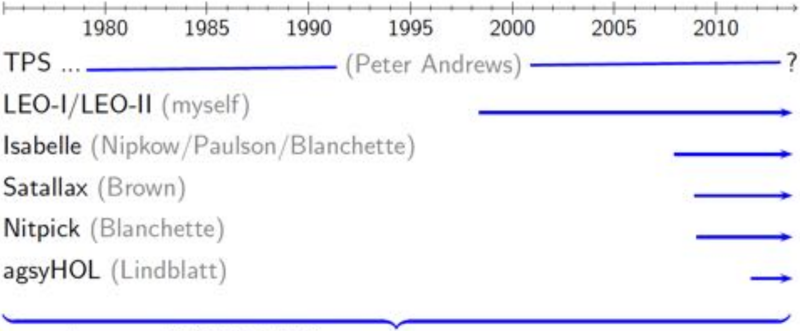
More information: [SutcliffeBenzmüller, J.FormalizedReasoning, 2010]

`http://cordis.europa.eu/result/report/rcn/45614_en.html`

HOL-ATPs: CASC Competitions since 2009

- **2009:** Winner TPS
- **2010:** Winner LEO-II 1.2 solved 56% more (than previous winner)
- **2011:** Winner Satallax 2.1 solved 21% more
- **2012:** Winner Isabelle-HOT-2012 solved 35% more
- **2013:** Winner Satallax-MaLeS solved 21% more

Automated Theorem Provers and Model Finders 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 —

Proof Automation with LEO-II

```
>
>
> Beweise-mit-Leo2 Notwendigerweise-existiert-Gott.p

Leo-II tries to prove
=====================

  Goedel's Theorem T3: "Necessarily, God exists"
  thf(thmT3,conjecture,
      ( v
      @ ( mbox
        @ ( mexists_ind
          @ ^ [X: mu] :
              ( g @ X ) ) ) )).

  Assumptions: D1, C, T2, D3, A5

 . searching for proof ..

 *******************
 *   Proof found   *
 *******************
 % SZS status Theorem for Notwendigerweise-existiert-Gott.p

 . generating proof object
```

Provers can be called remotely in Miami — no local installation needed!

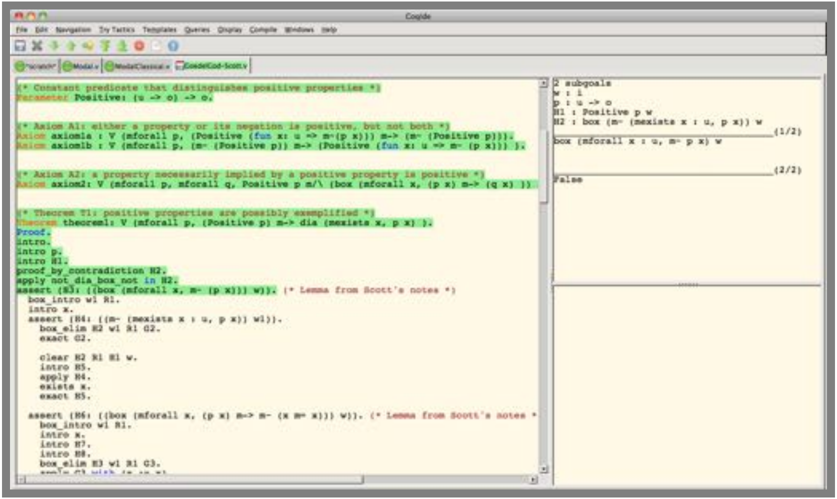
Download our experiments from

`https://github.com/FormalTheology/GoedelGod/tree/master/Formalizations/THF`

Gödel's God in Isabelle/HOL

See verifiable Isabelle/HOL document (Archive of Formal Proofs) at:
`http://afp.sourceforge.net/entries/GoedelGod.shtml`

Gödel’s God in Coq



See verifiable Coq document at:

https://github.com/FormalTheology/GoedelGod/tree/master/Formalizations/Coq

Conclusion

Conclusion

Overall Achievements

- significant contribution towards a **Computational Metaphysics**
- **novel results** contributed by **HOL-ATPs**
- infrastructure can be adapted for **other logics and logic combinations**
- **our technology is sufficiently mature** for use by philosophers

Relevance (wrt foundations and applications)

- Philosophy, AI, Computer Science, Computational Linguistics, Maths

Little related work: only for Anselm's simpler argument

- first-order ATP PROVER9 [OppenheimerZalta, 2011]
- interactive proof assistant PVS [Rushby, 2013]

Ongoing/Future work **see next talk by Bruno Woltzenlogel-P.**

- Landscape of verified/falsified ontological arguments
- You may consider to contribute:
  `https://github.com/FormalTheology/GoedelGod.git`

Personal Statement

(Interim) Culmination of two decades of related own research

- Theory of classical higher-order logic (HOL) (since 1995)
- Automation of HOL / own LEO provers (since 1998)
- Integration of interactive and automated theorem proving (since 1999)
- International TPTP infrastructure for HOL (since 2006)
- HOL as a universal logic via semantic embeddings (since 2008)
- jww Bruno Woltzenlogel-Paleo:
  Application in Metaphysics: Ontological Argument (since 2013)

. . . success story (despite strong criticism/opposition on the way!) . . .
. . . huge media attention . . .

(Interim) Own standpoint

- I am not fully convinced (yet) by the ontological argument.
- However, it seems to me that **the belief in a (God-like) supreme being is at least not necessarily irrational/inconsistent**.

Conclusion

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