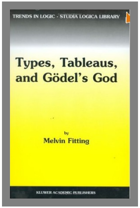
Formalization, Mechanization and Automation of Gödel's Proof of God's Existence

Christoph Benzmüller and **Bruno Woltzenlogel Paleo**

November 1, 2013



$$\dfrac{\dfrac{\text{Axiom 3}}{P(G)} \qquad \dfrac{\dfrac{\text{Theorem 1}}{\forall\varphi.[P(\varphi) \to \Diamond\exists x.\varphi(x)]}}{P(G) \to \Diamond\exists x.G(x)}\,\forall_E}{\Diamond\exists x.G(x)}\,\to_E$$

A gift to **Priest Edvaldo** and his church in Piracicaba, Brazil

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.
- . . .

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

Are we in contact with Steve Jobs? No

Do you really need a MacBook to obtain the results? No

Is Apple sending us money? No
(but maybe they should)

Def: Ontological Argument/Proof

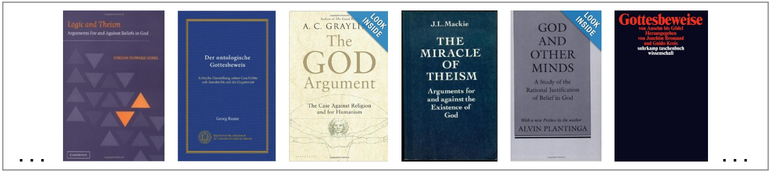
* deductive argument
* for the existence of God
* starting from premises, which are justified by pure reasoning, i.e. they do not depend on observation in the world.

Existence of God: different types of arguments/proofs

- a posteriori (use experience/observation in the world)
  - teleological
  - cosmological
  - moral
  - ...
- a priori (based on pure reasoning, independent)
  - ontological argument
    - definitional
    - modal
    - ...
  - other a priori arguments

Wohl eine jede Philosophie kreist um den ontologischen Gottesbeweis

(Adorno, Th. W.: Negative Dialektik. Frankfurt a. M. 1966, p.378)



Rich history on ontological arguments (pros and cons)

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

Anselm's notion of God:

"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 reasoning:

"(Necessarily) God exists."

Different Interests in Ontological Arguments:

- Philosophical: Boundaries of Metaphysics & Epistemology
  - We talk about a metaphysical concept (God),
  - but we want to draw a conclusion for the real world.

  - Necessary Existence (NE): metaphysical NE vs. logical NE vs. modal NE

- Theistic: Successful argument should convince atheists.

- Our: Can computers (theorem provers) be used
  - to formalize the definitions and axioms?
  - to verify the arguments step-by-step?
  - to fully automate (sub-)arguments?

"Computer-assisted Theoretical Philosophy"

Main challenge: No provers for *Higher-order Modal Logic* (HML)

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

[BenzmüllerPaulson, Logica Universalis, 2013]

What we did (rough outline for remaining presentation!):

A: Pen and paper: detailed natural deduction proof

B: Formalization: in classical higher-order logic (HOL)

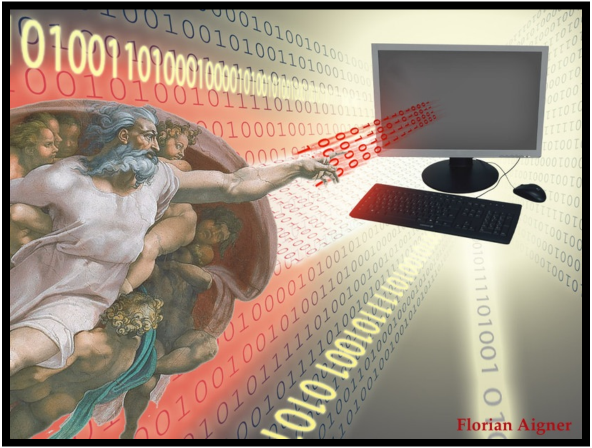
Automation: theorem provers LEO-II and SATALLAX

Consistency: model finder NITPICK (NITROX)

C: Step-by-step verification: proof assistant COQ

D: Automation & verification: proof assistant ISABELLE

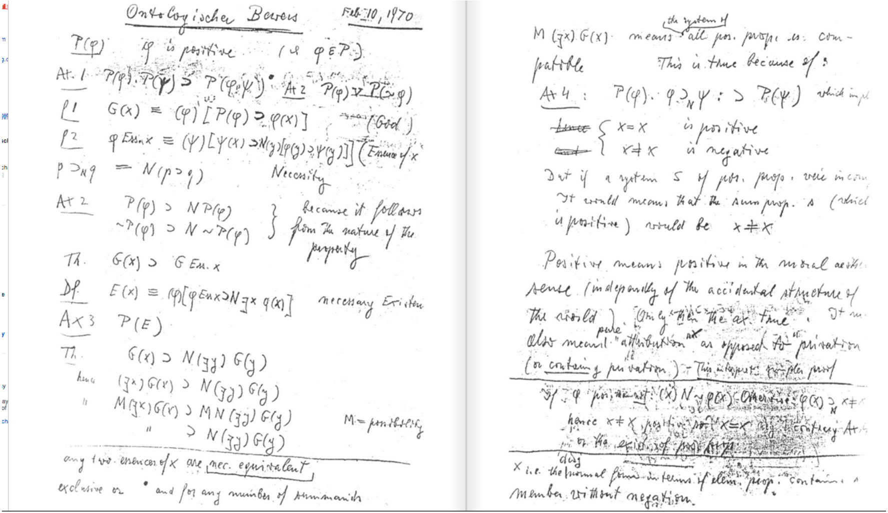
Did we get new results? Yes — let's discuss this later!



Part A:

Informal Proof and Natural Deduction Proof

ToDo: Improve Resolution



Ontologischer Beweis Feb 10, 1970

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

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

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

P2 $\varphi \, Ess. x \equiv (\psi)[\psi(x) \supset N(y)[\varphi(y) \supset \psi(y)]]$ (Essence of x)

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

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

Th. $G(x) \supset G \, Ess. x$

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

Ax 3 $P(E)$

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

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

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

" $\supset N(\exists y) G(y)$

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. prop. is compatible. This is true because of:

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

~~[illegible]~~ { $x = x$ is positive
~~[illegible]~~ { $x \neq x$ is negative

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

Positive means positive in the moral aesth[etic] 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. [illegible] proof

If φ pos. not: $(x) N \sim \varphi(x)$. Otherwise: $\varphi(x) \supset_N x \neq x$ hence $x \neq x$ positive not $x = x$ [illegible] contrary At [illegible] [illegible]

*i.e. the [illegible] disj normal form in terms of elem. prop. contains a member without negation.

Versions

A1 Either a property is positive or its negation is (never both): $\forall\phi[P(\neg\phi) \leftrightarrow \neg P(\phi)]$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$
\dfrac{A \vee B \quad \begin{array}{c}\overline{A}\\ \vdots \\ C\end{array} \quad \begin{array}{c}\overline{B}\\ \vdots \\ C\end{array}}{C}\ \vee_E
\qquad
\dfrac{A \quad B}{A \wedge B}\ \wedge_I
\qquad
\dfrac{\begin{array}{c}\overline{A}^{\,n}\\ \vdots \\ B\end{array}}{A \longrightarrow B}\ \longrightarrow_I^n
$$

$$
\dfrac{A}{A \vee B}\ \vee_{I_1}
\qquad
\dfrac{A \wedge B}{A}\ \wedge_{E_1}
\qquad
\dfrac{B}{A \longrightarrow B}\ \longrightarrow_I
$$

$$
\dfrac{B}{A \vee B}\ \vee_{I_2}
\qquad
\dfrac{A \wedge B}{B}\ \wedge_{E_2}
\qquad
\dfrac{A \quad A \longrightarrow B}{B}\ \longrightarrow_E
$$

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

$$
\neg A \equiv A \longrightarrow \bot
\qquad
\color{red}{\dfrac{\neg\neg A}{A}\ \neg\neg_E}
$$

$$\dfrac{\alpha : \boxed{\begin{array}{c}\vdots\\ A\end{array}}}{\Box A}\ \Box_I \qquad\qquad \dfrac{\Box A}{t : \boxed{\begin{array}{c}A\\ \vdots\end{array}}}\ \Box_E$$

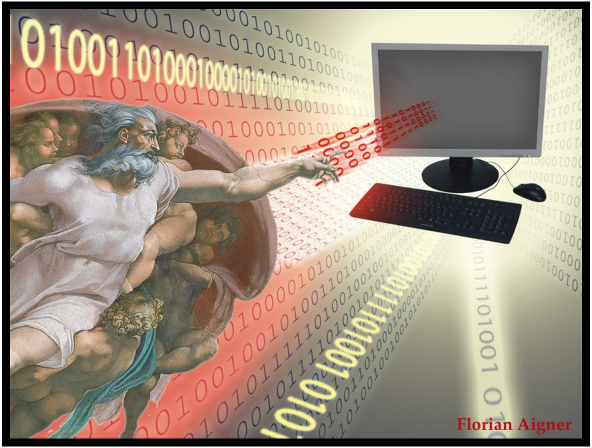
$$\dfrac{t : \boxed{\begin{array}{c}\vdots\\ A\end{array}}}{\Diamond A}\ \Diamond_I \qquad\qquad \dfrac{\Diamond A}{\beta : \boxed{\begin{array}{c}A\\ \vdots\end{array}}}\ \Diamond_E$$

$$\Diamond A \equiv \neg\Box\neg A$$

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

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

$$
\begin{prooftree}
\AxiomC{$\psi(x)^6$}
\AxiomC{Π_2}
\noLine
\UnaryInfC{$\psi(x) \rightarrow \Box P(\psi)$}
\RightLabel{$\rightarrow_E$}
\BinaryInfC{$\Box P(\psi)$}
\AxiomC{$\Box P(\psi)^7$}
\RightLabel{$\Box_E$}
\UnaryInfC{$P(\psi)$}
\AxiomC{Π_3}
\noLine
\UnaryInfC{$P(\psi) \rightarrow \forall x.(G(x) \rightarrow \psi(x))$}
\RightLabel{$\rightarrow_E$}
\BinaryInfC{$\forall x.(G(x) \rightarrow \psi(x))$}
\RightLabel{$\Box_I$}
\UnaryInfC{$\Box \forall x.(G(x) \rightarrow \psi(x))$}
\RightLabel{$\rightarrow_I^7$}
\UnaryInfC{$\Box P(\psi) \rightarrow \Box \forall x.(G(x) \rightarrow \psi(x))$}
\RightLabel{$\rightarrow_E$}
\BinaryInfC{$\Box \forall x.(G(x) \rightarrow \psi(x))$}
\RightLabel{$\rightarrow_I^6$}
\UnaryInfC{$\psi(x) \rightarrow \Box \forall x.(G(x) \rightarrow \psi(x))$}
\end{prooftree}
$$



Part B:

Formalization:	in classical higher-order logic (HOL)
Automation:	theorem provers LEO-II and SATALLAX
Consistency:	model finder NITPICK (NITROX)

Main challenge: No provers for *Higher-order Modal Logic* (HML)

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

Then use existing HOL theorem provers for reasoning in HML

[BenzmüllerPaulson, Logica Universalis, 2013]

Previous empiricial findings:

Embedding of *First-order Modal Logic* in HOL works well

[BenzmüllerOttenRaths, ECAI, 2012]

[Benzmüller, LPAR, 2013]

HML $\varphi, \psi \quad ::= \quad \ldots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x\, \varphi \mid \exists x\, \varphi \mid \forall P\, \varphi$

- Kripke style semantics (possible world semantics)

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

- meanwhile very well understood
- Henkin semantics vs. standard semantics
- various theorem provers do exists

  interactive: Isabelle/HOL, HOL4, Hol Light, Coq/HOL, PVS, ...

  automated: TPS, LEO-II, Satallax, Nitpick, Isabelle/HOL, ...

HML $\varphi, \psi \quad ::= \quad \ldots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x\, \varphi \mid \exists x\, \varphi \mid \forall P\, \varphi$

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

HML in HOL: HML formulas φ are mapped to HOL predicates $\varphi_{\iota\rightarrow o}$

$$
\begin{array}{lcl}
\neg & = & \lambda\varphi_{\iota\rightarrow o}\lambda s_\iota \neg\varphi s \\
\wedge & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda s_\iota(\varphi s \wedge \psi s) \\
\rightarrow & = & \lambda\varphi_{\iota\rightarrow o}\lambda\psi_{\iota\rightarrow o}\lambda s_\iota(\neg\varphi s \vee \psi s) \\
\Box & = & \lambda\varphi_{\iota\rightarrow o}\lambda s_\iota \forall u_\iota\, (\neg rsu \vee \varphi u) \\
\Diamond & = & \lambda\varphi_{\iota\rightarrow o}\lambda s_\iota \exists u_\iota\, (rsu \wedge \varphi u) \\
\forall & = & \lambda h_{\mu\rightarrow(\iota\rightarrow o)}\lambda s_\iota \forall d_\mu\, hds \\
\exists & = & \lambda h_{\mu\rightarrow(\iota\rightarrow o)}\lambda s_\iota \exists d_\mu\, hds \\
\forall & = & \lambda H_{(\mu\rightarrow(\iota\rightarrow o))\rightarrow(\iota\rightarrow o)}\lambda s_\iota \forall d_\mu\, Hds \\
 & & \\
\text{valid} & = & \lambda\varphi_{\iota\rightarrow o}\forall w_\iota \varphi w
\end{array}
\qquad \text{Ax}
$$

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

(Remark: We are here dealing with constant domain quantification.)

Example

HML formula	$\Diamond \exists x G(x)$
HML 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	$\forall w_\iota \exists u_\iota (rwu \wedge \exists x Gxu)$

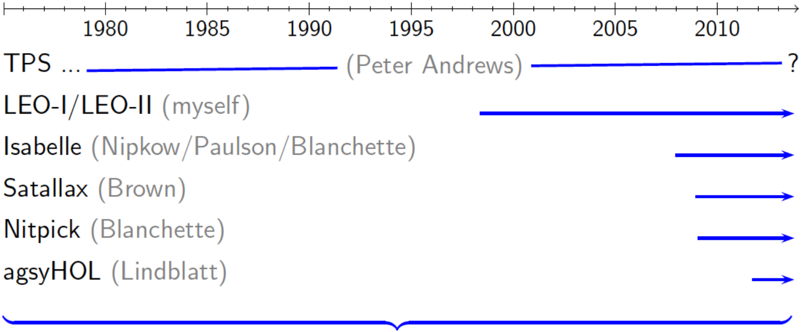
What are we doing?

In order to prove that φ is valid in HML,
–> 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.

Expansion: user or prover may flexibly choose expansion depth

Automated Theorem Provers and Model Finders for HOL



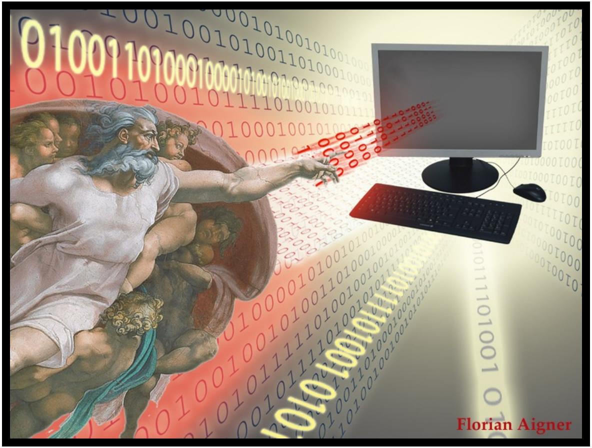
- all accept TPTP THF Syntax [SutcliffeBenzmüller, J.Form.Reas, 2009]
- can be called remotely via SystemOnTPTP at Miami
- they significantly gained in strength over the last years
- they can be bundled into a combined prover HOL-P

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

Proof Automation and Consistencey Checking: Demo!

```
Terminal — bash — 125×32
MacBook-Chris %
MacBook-Chris %
MacBook-Chris % ./call_tptp.sh T3.p

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

MacBook-Chris % agsyHOL---1.0 : T3.p ++++++ RESULT: SOT_7L4x_Y - agsyHOL---1.0 says Unknown - CPU = 0.00 WC = 0.02
LEO-II---1.6.0 : T3.p ++++++ RESULT: SOT_E4SCha - LEO-II---1.6.0 says Theorem - CPU = 0.03 WC = 0.09
Satallax---2.7 : T3.p ++++++ RESULT: SOT_kVZ1cB - Satallax---2.7 says Theorem - CPU = 0.00 WC = 0.14
Isabelle---2013 : T3.p ++++++ RESULT: SOT_xa0gEp - Isabelle---2013 says Theorem - CPU = 14.06 WC = 17.73 SolvedBy = auto
TPS---3.120601S1b : T3.p ++++++ RESULT: SOT_R0Egsg - TPS---3.120601S1b says Unknown - CPU = 33.56 WC = 41.57
Nitrox---2013 : T3.p ++++++ RESULT: SOT_WGY1Tx - Nitrox---2013 says Unknown - CPU = 75.55 WC = 49.24

MacBook-Chris %
MacBook-Chris % ./call_tptp.sh Consistency.p

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

MacBook-Chris % agsyHOL---1.0 : Consistency.p ++++++ RESULT: SOT_ZtY_7o - agsyHOL---1.0 says Unknown - CPU = 0.00 WC = 0.00
Nitrox---2013 : Consistency.p ++++++ RESULT: SOT_HUz10C - Nitrox---2013 says Satisfiable - CPU = 6.56 WC = 8.50
TPS---3.120601S1b : Consistency.p ++++++ RESULT: SOT_fpJxTM - TPS---3.120601S1b says Unknown - CPU = 43.00 WC = 49.42
Isabelle---2013 : Consistency.p ++++++ RESULT: SOT_6Tpp9i - Isabelle---2013 says Unknown - CPU = 69.96 WC = 72.62
LEO-II---1.6.0 : Consistency.p ++++++ RESULT: SOT_dY10sj - LEO-II---1.6.0 says Timeout - CPU = 90 WC = 89.86
Satallax---2.7 : Consistency.p ++++++ RESULT: SOT_Q9WSLf - Satallax---2.7 says Timeout - CPU = 90 WC = 90.50

MacBook-Chris %
```

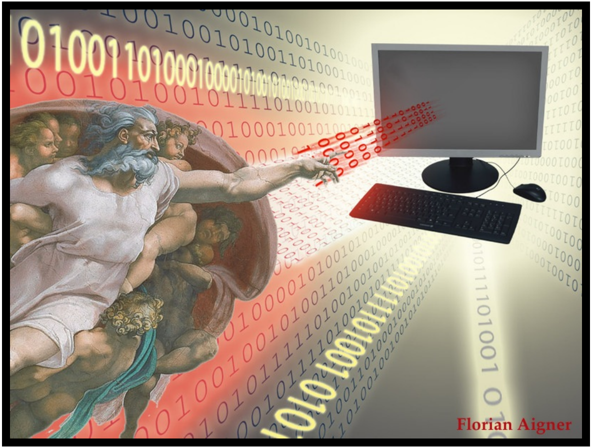
Provers are called remotely in Miami — no local installation needed!



Part C:

Formalization and Verification in Coq

- Goal: verification of the natural deduction proof
  - Step-by-step formalization
  - Almost no automation (intentionally!)
- Interesting facts to note:
  - Embedding is transparent to the user
  - Embedding gives labeled calculus for free



Part D:

automation & verification: proof assistant ISABELLE

Isabelle

http://isabelle.in.tum.de/index.html

Home-FU 2012-Watson Homepage 2012-FOL SPIEGEL 2012-FOL-Home GMail Google Maps M&M SigmaOnline Kita Sigma Kalender Beliebt Google Maps

Isabelle

UNIVERSITY OF CAMBRIDGE Computer Laboratory

- Home
- Overview
- Installation
- Documentation
- Community

Site Mirrors:
Cambridge (.uk)
Munich (.de)
Sydney (.au)

What is Isabelle?

Isabelle is a generic proof assistant. It allows mathematical formulas to be expressed in a formal language and provides tools for proving those formulas in a logical calculus. Isabelle is developed at University of Cambridge (Larry Paulson), Technische Universität München (Tobias Nipkow) and Université Paris-Sud (Makarius Wenzel). See the Isabelle overview for a brief introduction.

Now available: Isabelle2013

Download for Mac OS X

Download for Linux - Download for Windows

Some highlights:

- Improvements of Isabelle/Scala and Isabelle/jEdit Prover IDE.
- Advanced build tool based on Isabelle/Scala.
- Updated manuals: isar-ref, implementation, system.
- Pure: improved support for block-structured specification contexts.
- HOL tool enhancements: Sledgehammer, Lifting, Quickcheck.
- HOL library enhancements: HOL-Library, HOL-Probability, HOL-Cardinals.
- HOL: New BNF-based (co)datatype package.
- Improved performance thanks to Poly/ML 5.5.0.

See also the cumulative NEWS.

Distribution & Support

Isabelle is distributed for free under the BSD license. It includes source and binary packages and documentation, see the detailed installation instructions. A vast collection of Isabelle examples and applications is available from the Archive of Formal Proofs.

Support is available by ample documentation, the Isabelle community Wiki, and the following mailing lists:

- **isabelle-users@cl.cam.ac.uk** provides a forum for Isabelle users to discuss problems, exchange information, and make announcements. Users of official Isabelle releases should subscribe or see the archive (also available via Google groups and Narkive).
- **isabelle-dev@in.tum.de** covers the Isabelle development process, including intermediate repository versions, and administrative issues concerning the website or testing infrastructure. Early adopters of repository versions should subscribe or see the archive (also available at mail-archive.com or gmane.org).

Last updated: 2013-03-09 12:21:39

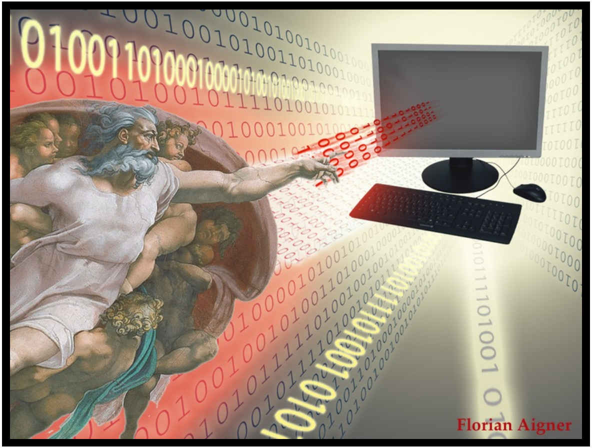
Isabelle/HOL (Cambridge University/TU Munich)

- HOL instance of the generic ISABELLE proof assistant
- User interaction and proof automation
- Automation is supported by SLEDGEHAMMER tool
- Verification of the proofs in ISABELLE/HOL's small proof kernel

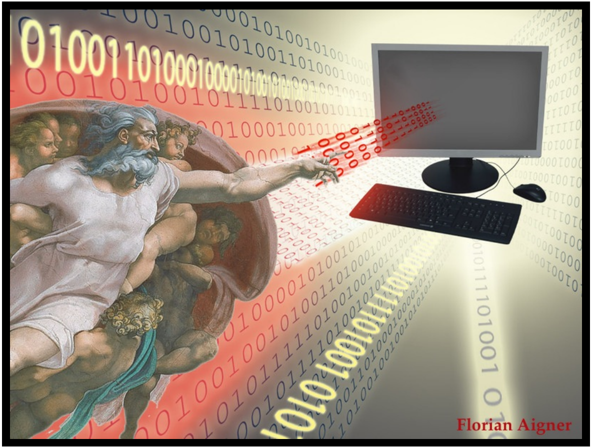
What have we done?

- Proof automation of Gödel's proof script (Scott version)
- SLEDGHAMMER makes calls to remote THF provers in Miami
- These calls the suggest respective calls to the METIS prover
- METIS proofs are verified in ISABELLE/HOL's proof kernel

See the handout (generated from the Isabelle source file).



Part E:
Criticisms



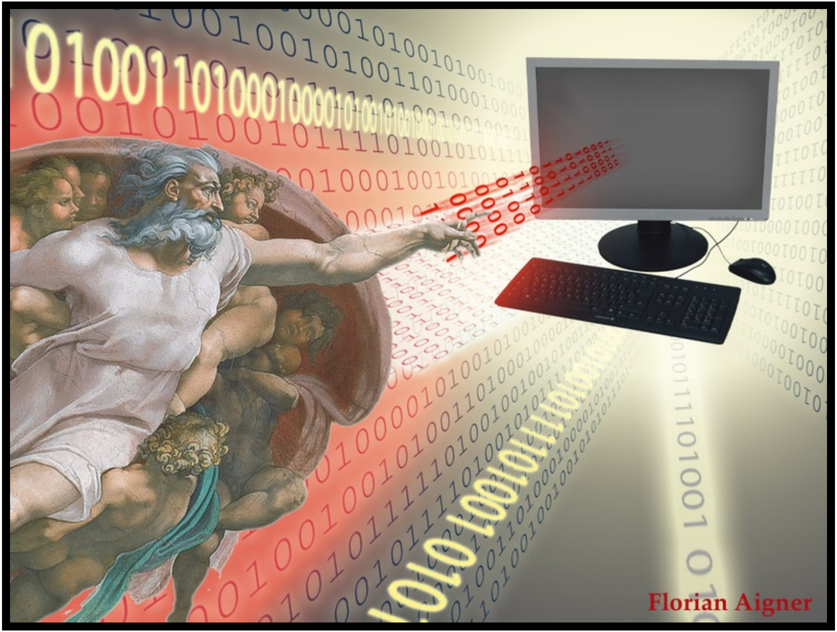
Part F:
Conclusions

Summary of Results about the Ontological Proof

- K sufficient for T1, C1 and T2
- S5 not needed for T3
- KB sufficient for T3
- A simpler new proof of C1
- Gödel's original axioms (without conjunct $\phi(x)$ in D2) are inconsistent
- Scott's axioms are consistent
- For T1, only half of A1 (A1a) is needed
- For T2, the other half (A1b) is needed

Summary of Results for Logic

- Infra-structure for reasoning with modal logic using existing proof assistants and higher-order automated theorem provers
- A new natural deduction calculus for higher-order modal logic
- Difficult benchmarks for higher-order automated theorem provers


Florian Aigner

Conclusion

What have we achieved

- Verification of Gödel's ontological argument with HOL provers
  - exact parameters known: constant domain quantification, Henkin Semantics
  - parameters can be varied and experiments can repeated
- Major step towards Computer-assisted Theoretical Philosophy
  - see also Ed Zalta's *Computational Metaphysics* project at Stanford University
  - remember Leibniz' dictum — *Calculemus!*
- Highly fascinating bridge between CS, Philosophy and Theology
- Major public interest

Future Work

-