On Logic Embeddings and Gödel's God

Christoph Benzmüller[1][*] and Bruno Woltzenlogel Paleo[2]

[1] Freie Universität Berlin, Germany
c.benzmueller@fu-berlin.de
[2] Vienna Technical University, Austria
bruno@logic.at

Abstract. We have applied an elegant and flexible logic embedding approach to verify and automate a prominent philosophical argument: the ontological argument for the existence of God. In our ongoing computer-assisted study, higher-order automated reasoning tools have made some interesting observations, some of which were previously unknown.

Logic embeddings provide an elegant means to formalize sophisticated non-classical logics in classical higher-order logic (HOL, Church's simple type theory [12]). In previous work (cf. [4] and the references therein) the embeddings approach has been successfully applied to automate object-level and meta-level reasoning for a range of logics and logic combinations with off-the-shelf HOL theorem provers. This also includes quantified modal logics (QML) [7] and quantified conditional logics (QCL) [3]. For many of the embedded logics few or no automated theorem provers did exist before. HOL is exploited in this approach to encode the semantics of the logics to be embedded, for example, Kripke semantics for QMLs [13] or selection function semantics for QCLs [24].

The embeddings approach is related to labelled deductive systems [16], which employ meta-level (world-)labeling techniques for the modeling and implementation of non-classical proof systems. In our embeddings approach such labels are instead encoded in the HOL logic.

The embedding approach is flexible, because various modal logics (even with multiple modalities or a mix of varying/cumulative domain quantifiers) can be easily supported by stating their characteristic axioms. Moreover, it is relatively simple to implement, because it does not require any modification in the source code of the higher-order prover. A minimal encoding of second-order modal logic KB in TPTP THF syntax [25] — this syntax is accepted by a range of HOL automated theorem provers (ATPs) — is exemplarily provided in Fig. 1.[3] The

---

[3] Some Notes on THF, which is a concrete syntax for HOL: `$i` and `$o` represent the HOL base types i and o (Booleans). `$i>$o` encodes a function (predicate) type. Predicate application as in $A(X, W)$ is encoded as `((A@X)@W)` or simply as `(A@X@W)`, i.e., function/predicate application is represented by `@`; universal quantification and λ-abstraction as in $\lambda A_{i\to o} \forall W_i (A\, W)$ and are represented as in `^[X:$i>$o]:![W:$i]:(A@W)`; comments begin with `%`.

given set of axioms turns any TPTP THF compliant HOL-ATP in a reasoning tool for second-order modal logic. A Henkin-style semantics is thereby assumed for both logics: HOL and second-order modal logic.

```
1   %----The base type $i (already built-in) stands here for worlds and
2   %----mu for individuals; $o (also built-in) is the type of Booleans
3   thf(mu_type,type,(mu:$tType)).
4   %----Reserved constant r for accessibility relation
5   thf(r,type,(r:$i>$i>$o)).
6   %----Modal operators not, or, box
7   thf(mnot_type,type,(mnot:($i>$o)>$i>$o)).
8   thf(mnot,definition,(mnot = (^[A:$i>$o,W:$i]:~(A@W)))).
9   thf(mor_type,type,(mor:($i>$o)>($i>$o)>$i>$o)).
10  thf(mor,definition,(mor = (^[A:$i>$o,Psi:$i>$o,W:$i]:((A@W)|(Psi@W))))).
11  thf(mbox_type,type,(mbox:($i>$i>$o)>($i>$o)>$i>$o)).
12  thf(mbox,definition,(mbox = (^[A:$i>$o,W:$i]:![V:$i]:(~(r@W@V)|(A@V))))).
13  %----Quantifier (constant domains) for individuals and propositions
14  thf(mall_ind_type,type,(mall_ind:(mu>$i>$o)>$i>$o)).
15  thf(mall_ind,definition,(mall_ind = (^[A:mu>$i>$o,W:$i]:![X:mu]:(A@X@W)))).
16  thf(mall_indset_type,type,(mall_indset:((mu>$i>$o)>$i>$o)>$i>$o)).
17  thf(mall_indset,definition,(
18      mall_indset = (^[A:(mu>$i>$o)>$i>$o,W:$i]:![X:mu>$i>$o]:(A@X@W)))).
19  %----Definition of validity (grounding of lifted modal formulas)
20  thf(v_type,type,(v:($i>$o)>$o)).
21  thf(mvalid,definition,(v = (^[A:$i>$o]:![W:$i]:(A@W)))).
22  %----Properties of accessibility relations: symmetry
23  thf(msymmetric_type,type,(msymmetric:($i>$i>$o)>$o)).
24      msymmetric = (^[R:$i>$i>$o]:![S:$i,T:$i]:((R@S@T)=>(R@T@S))))).
25  %----Here we work with logic KB, i.e., we postulate symmetry for r
26  thf(sym,axiom,(msymmetric@r)).
```

Fig. 1. HOL encoding of second-order modal logic KB in THF syntax. Modal formulas are mapped to HOL predicates (with type `$i>$o`); type `$i` now stands for possible worlds. The modal connectives ¬ (`mnot`), ∨ (`mor`) and □ (`mbox`), universal quantification for individuals (`mall_ind`) and for sets of individuals (`mall_indset`) are introduced in lines 7-18. Validity of lifted modal formulas is defined in the standard way (lines 20-21). Symmetry of accessibility relation r is postulated in lines 23-26. Hence, second-order KB is realized here; for logic K the symmetry axiom can be dropped.

In recent work [6, 5] we have applied the embedding approach to verify and automate a philosophical argument that has fascinated philosophers and theologians for about 1000 years: the ontological argument for the existence of God [23]. We have thereby concentrated on Gödel's [17] modern version of this argument and on Scott's [22] modification, which employ a second-order modal logic (S5) for which, until now, no theorem provers were available. In our computer-assisted study of the argument, the HOL provers LEO-II [8], Satallax [11] and Nitpick [10] have made some interesting observations, some of which were unknown so far. This is a landmark result, with media repercussion in a global scale, and yet it is only a glimpse of what can be achieved by combining computer science, philosophy and theology.

We briefly summarize some of these observations: Nitpick confirms that Scott's axioms are consistent, while LEO-II and Satallax demonstrate that Gödel's

original, slightly different axioms are inconsistent. As far as we are aware, this is a new result. As experiments with LEO-II revealed, the problem lies in a subtle difference in the definitions of the predicate *essence* (characterizing the essential properties of an entity) between Gödel and Scott. In several philosophy papers, however, this difference has been commented as a simple cosmetic change. Moreover, instead of using modal logic S5, LEO-II and Satallax can prove the final theorem (that is, $\Box\exists x.G(x)$, necessarily there exists God) already for modal logic KB. This is highly relevant since several philosophers have criticized Gödel's argument for the use of logic S5. Axiom B (symmetry), however, cannot be dropped, which in turn is confirmed by Nitpick. LEO-II and Satallax can also show that Gödel's and Scott's axioms imply what is called the modal collapse: $\phi \supset \Box\phi$. This expresses that contingent truth implies necessary truth (which can even be interpreted as an argument against free will; cf. [23]) and is probably the most fundamental criticism put forward against Gödel's and Scott's versions of the argument. Other theorems that can be shown by LEO-II and Satallax include flawlessness of God and monotheism.

[margin: citation needed? weaken the "several" claim?]

[margin: citation needed? weaken the "several" claim?]

Ongoing and future work concentrates on the systematic study of Gödel's and Scott's proofs. We have also begun to study more recent variants of the argument [1, 2, 9, 15, 14, 18, 19], which claim to remedy some fundamental problems of Gödel's and Scott's proofs, especially the modal collapse. One interesting and very encouraging observation from these studies is, that the argumentation granularity typical of these philosophy papers is already within reach of the capabilities of our higher-order automated theorem provers. This provides good evidence for the potential relevance of the embedding approach (not only) w.r.t. other similar applications in metaphysics.

The long-term goal is to methodically determine the range of logical parameters (e.g., constant vs. varying domains, rigid vs. non-rigid terms, logics KB vs. S4 vs. S5, etc.) under which the proposed variants of the modern ontological argument hold or fail.

There has been little related work [20, 21], and this focused solely on the comparably simpler, original ontological argument by Anselm of Canterbury. These works do not achieve the close correspondence between the original formulations and the formal encodings that can be found in our approach and they also do not reach the same degree of proof automation.

Our work attests the maturity of contemporary interactive and automated deduction tools for HOL and demonstrates the elegance and practical relevance of the embeddings-based approach. Most importantly, our work opens new perspectives towards computational metaphysics.

References

1. C.A. Anderson. Some emendations of Gödel's ontological proof. *Faith and Philosophy*, 7(3), 1990.
2. C.A. Anderson and M. Gettings. Gödel ontological proof revisited. In *Gödel'96: Logical Foundations of Mathematics, Computer Science, and Physics: Lecture Notes in Logic 6*, pages 167–172. Springer, 1996.

3. C. Benzmüller. Automating quantified conditional logics in HOL. In F. Rossi, editor, Proc. of *IJCAI 2013*, pages 746–753, Beijing, China, 2013.

4. C. Benzmüller. A top-down approach to combining logics. In *Proc. of ICAART 2013*, pages 346–351, Barcelona, Spain, 2013. SciTePress Digital Library.

5. C. Benzmüller and B. Woltzenlogel Paleo. Gödel's God in Isabelle/HOL. *Archive of Formal Proofs*, 2013.

6. C. Benzmüller and B. Woltzenlogel Paleo. Automating Gödel's ontological proof of god's existence with higher-order automated theorem provers. In *ECAI 2014*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 163 – 168. IOS Press, 2014.

7. C. Benzmüller and L. Paulson. Quantified multimodal logics in simple type theory. *Logica Universalis (Special Issue on Multimodal Logics)*, 7(1):7–20, 2013.

8. C. Benzmüller, F. Theiss, L. Paulson, and A. Fietzke. LEO-II - a cooperative automatic theorem prover for higher-order logic (system description). In *Proc. of IJCAR 2008*, volume 5195 of *LNCS*, pages 162–170. Springer, 2008.

9. F. Bjørdal. Understanding gödel's ontological argument. In T. Childers, editor, *The Logica Yearbook 1998*. Filosofia, 1999.

10. J.C. Blanchette and T. Nipkow. Nitpick: A counterexample generator for higher-order logic based on a relational model finder. In *Proc. of ITP 2010*, volume 6172 of LNCS, pp. 131–146. Springer, (2010).

11. C.E. Brown. Satallax: An automated higher-order prover. In *Proc. of IJCAR 2012*, number 7364 in LNAI, pages 111 – 117. Springer, 2012.

12. A. Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5:56–68, 1940.

13. M. Fitting and R.L. Mendelsohn. *First-Order Modal Logic*, volume 277 of *Synthese Library*. Kluwer, 1998.

14. M. Fitting. *Types, Tableaus, and Gödel's God*. Kluwer, 2002.

15. A. Fuhrmann. Existenz und Notwendigkeit — Kurt Gödels axiomatische Theologie. In W. Spohn et al., editor, *Logik in der Philosophie*. Heidelberg (Synchron), 2005.

16. D.M. Gabbay. *Labelled Deductive Systems*. Clarendon Press, 1996.

17. K. Gödel. *Appx.A: Notes in Kurt Gödel's Hand*, pages 144–145. In [23], 2004.

18. P. Hajek. A new small emendation of gödel's ontological proof. *Studia Logica: An International Journal for Symbolic Logic*, 71(2):pp. 149-164, 2002.

19. P. Hajek. Ontological proofs of existence and non-existence. *Studia Logica: An International Journal for Symbolic Logic*, 90(2):pp. 257–262, 2008.

20. P.E. Oppenheimera and E.N. Zalta. A computationally-discovered simplification of the ontological argument. *Australasian J. of Philosophy*, 89(2):333–349, 2011.

21. J. Rushby. The ontological argument in PVS. In *Proc. of CAV Workshop "Fun With Formal Methods"*, St. Petersburg, Russia,, 2013.

22. D. Scott. *Appx.B: Notes in Dana Scott's Hand*, pages 145–146. In [23], 2004.

23. J.H. Sobel. *Logic and Theism: Arguments for and Against Beliefs in God*. Cambridge U. Press, 2004.

24. R.C. Stalnaker. A theory of conditionals. In *Studies in Logical Theory*, pages 98–112. Blackwell, 1968.

25. G. Sutcliffe and C. Benzmüller. Automated reasoning in higher-order logic using the TPTP THF infrastructure., *J. of Formalized Reasoning*, 3(1), 1–27, (2010).