Formalizations of Gödel's Ontological Proof of God's Existence

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

[1] Theory and Logic Group, Vienna University of Technology, Austria
bruno@logic.at
[2] Dahlem Center for Intelligent Systems, Freie Universität Berlin, Germany
c.benzmueller@gmail.com

Attempts to prove the existence (or non-existence) of God by means of abstract ontological arguments are an old tradition in philosophy and theology, Gödel's proof is a modern culmination of this tradition, following particularly the footsteps of Leibniz. Gödel defines God as a being who possesses all *positive* properties. He does not extensively discuss what positive properties are, but instead he states a few reasonable but debatable axioms that they should satisfy. From these axioms, he claims (without detailed proof) that:

1. If it is possible that God exists, then it is necessary that God exists.
2. It is possible that God exists.

In the literature [**?**] slightly different versions of the axioms exist and derivations of the above mentioned claims are presented with various degrees of detail and formality. Here [**?**] (a version of) Gödel's proof is constructed for the first-time with the utmost degree of detail and formality. The following has been done:

- A detailed natural deduction proof.
- A formalization of the axioms and theorems in the TPTP THF format [**?**].
- Automatic verification of the consistency of the axioms and definitions with ToDo [**?**].
- Automatic demonstration of the theorems with the provers LEO-II [**?**] and Satallax [**?**].
- A step-by-step formalization using the Coq proof assistant [**?**].
- A formalization using the Isabelle proof assistant [**?**] partially automated with Sledgehammer [**?**] and Metis [**?**].

Gödel's proof is challenging to formalize and automatically verify because it requires very expressive logical languages with modal operators (*possible* and *necessary*) and higher-order quantifiers. The computer-assisted formalizations rely on an embedding of the modal logic S5 into classical higher-order logic with Henkin semantics [**?**] and employed recently developed interactive and automated deduction tools designed for this logic.

This work attests the maturity of contemporary interactive and automated deduction tools and opens new perspectives for a computer-assisted theoretical philosophy. The critical discussion of the underlying concepts, definitions and axioms remains a human responsibility, but the computer can assist in building and checking rigorously correct logical arguments. In case of logico-philosophical disputes, the computer can check the disputing arguments and partially fulfill Leibniz' dictum: Calculemus — Let us calculate!