Programmatic Semantics

TOPOS INSTITUTE Benjamin Brast-McKie June 17, 2025

Abstract

This talk presents a *programmatic methodology* which uses the model-checker software that I developed to rapidly prototype semantic theories.

I will begin by presenting a standard methodology in philosophical logic to highlight a number of shortcomings which motivate the programmatic methodology. I will then introduce the model-checker which draws on the SMT solver Z3 to rule out finite countermodels of a user specified size, providing evidence that a logical consequences has no countermodels if in fact there are none. Implementing a programmatic semantics with the model-checker extends the standard methodology by easing the process of exploring and prototyping novel semantic theories.

In addition to facilitating the study of complex semantic theories, the model-checker provides resources for uploading semantic theories to the TheoryLib to facilitate collaboration. Programmatic semantic theories are also modular, making them easy to combine and compare, allowing users to survey the interactions in languages with many operators. Moreover, the computability of a semantic theory provides an objective measure that may be weighed alongside other theoretical virtues.

Although the model-checker is a general purpose utility for working in semantics, applications in hyperintensional semantics are particularly natural given the increased complexity of these semantic systems. Rather than a deficiency, I will characterize well-motivated forms of theoretical complexity as a sign of the maturity of semantics as a discipline. It is in support of both the future development and accessibility of semantics that the model-checker aims to make a contribution. The talk will conclude with a brief demonstration to make the workflow concrete.

References

Brast-McKie, Benjamin. 2025. "Counterfactual Worlds." *Journal of Philosophical Logic* 1–42. ISSN 1573-0433. doi: 10.1007/s10992-025-09793-8.