The Nature of Logic and its Logics: Notes for Tel Aviv University Seminar on Logic and Formal Methods, June 1, 2011

Robert Constable
Cornell University

Abstract

Logic is one of the oldest intellectual disciplines; it provides the *deductive method* for the sciences and all rational discourse. It is central to the philosophy of knowledge and draws on understanding human cognition. After more than 2360 years it still inspires innovations and probes more deeply into its core concerns which include providing and investigating modes of thought used to organize the evidential structure of knowledge. In the 21st century this involves learning to create knowledge from vast amounts of *digital evidence* as well as providing the designs for computer systems that assist us in reasoning about "digital knowledge." At the core of this activity is the desire to construct convincing evidence for declarative statements (assertions). Logic raises questions such as "What assertions are logically equivalent because they are justified by the same evidence?", "What conclusions indisputably follow from given assumptions (hypotheses)?", "What utterances are meaningful as knowledge claims?" and so forth.

Natural language conveys evidence and knowledge, and the study of its grammars reveals universal means of structuring arguments. Aristotle's fundamental works on logic, the *Organon* (approx 350 BCE), are grounded in grammar and mark an ancient origin of logic as a discipline. Aristotle also introduced the notion of a particular logic (of syllogisms) with specific *rules for demonstrating* that an assertion follows from premises. Logic was a key component of the study of *rhetoric*.

The logical mode of thought is essential to mathematics and indeed is required to convey nearly all pure mathematical thought; thus logic can be seen as *universal pure mathematics*. Euclid's *Elements* is the canonical example of logic in service of mathematics, and the canonical example of *a logic*. Amazingly this logic (circa 306 BCE) remains to this day an active topic of logical investigation, as is clear from topics in this Advanced Seminar on Logic and Formal Methods. Logical systems are studied mathematically (at least since Leibnitz and especially with Boole), creating a subject called *mathematical logic*, now required of most computer science majors.

In this lecture, I will look at the development of logic after the critical year 1907 when Russell wrote his *The Principles of Mathematics* [8], introducing *type theory*, and Brouwer [5] proposed an intuitive computational basis for logic as an aspect of mathematical thought, an approach now called *intuitionism*. Russell worked to reduce mathematics to logic, and Brouwer tried to show that logic expressed general mathematical constructions. I will argue that each Russell and Brouwer were fundamentally and deeply right, although mainstream mathematical logic does not yet reflect this.[1] We see most clearly where both authors are right from certain results in logic that gave birth to computer science and from the subsequent deep interactions between logic and computer science – explaining the curricular requirement mentioned above.

Although many of Brouwer's radical insights are thoroughly studied and well understood in both logic and computer science, one of his "logical" insights has been soundly rejected, including by my colleagues and me. That insight is his belief that there is no fundamental separation between what are called *object logics* and *meta-logics*. Brouwer relied on this insight to prove his principle of *Bar Induction* [6]. I now think Brouwer was right on this point as well and hope to incorporate his approach more fully into the logic I have studied most deeply and helped create, *Computational Type Theory (CTT)* [3, 1].

[1] I wrote about this topic in a short paper presented at the the 100th anniversary celebration of the publication of *Principia Mathematica* of Whitehead and Russell [10], under the title The Triumph of Types: *Principia Mathematica*'s Impact on Computer Science.

This theory already incorporates proof terms into its logic as well as the evaluation of terms. However, CTT is also supported by an implemented meta-language (ML) that automates reasoning using *tactics*. I will also use this specific CTT logic to illustrate why there are comprehensive logics such as *Principia Mathematica* (PM) and its off spring, such as Higher-Order Logic (HOL) [4], CTT, and the *Calculus of Inductive Constructions* (CIC) [7]. I will explain why some of these including HOL, CIC, and CTT are *implemented as interactive theorem proving systems* (e.g. by HOL, Coq, and Nuprl respectively) and how Brouwer's insights could make these implemented logics even more useful.[2]

Why does mainstream logic adhere to the object logic vs meta-logic distinction? One of the key reason's is *Tarski's Theorem* [9] – that for all sufficiently expressive formal logics, certainly for all comprehensive logics, the notion of truth cannot be defined in the logic. Truth can be defined in a sufficiently strong meta-logic. For instance, the truth of first-order arithmetic cannot be defined in that theory, but it can be defined in higher-order logic or in set theory. Some logicians think that Tarski's Theorem is more consequential even than Gödel's Theorem. I will review this result and point out that it is sometimes very informative and sometimes not. It is not so informative for set theory and type theory and thus might be overvalued as a design issue.[3] I will point out that CTT has progressively incorporated more and more meta-logic and speculate about the limit of this trend.

If there is time, I will also speculate on the emerging connection between physics and logic in the study of *quantum computing* and the possible long term implications of this research on the subject of automated reasoning that supports the implemented logics mentioned above.

References

[1] Stuart Allen, Mark Bickford, Robert Constable, Richard Eaton, Christoph Kreitz, Lori Lorigo, and Evan Moran. Innovations in computational type theory using Nuprl. *Journal of Applied Logic*, 4(4):428–469, 2006.

[2] Stuart F. Allen. A Non-type-theoretic Definition of Martin-Löf's Types. In D. Gries, editor, *Proceedings of the 2^{nd} IEEE Symposium on Logic in Computer Science*, pages 215–224. IEEE Computer Society Press, June 1987.

[3] Robert L. Constable, Stuart F. Allen, H. M. Bromley, W. R. Cleaveland, J. F. Cremer, R. W. Harper, Douglas J. Howe, T. B. Knoblock, N. P. Mendler, P. Panangaden, James T. Sasaki, and Scott F. Smith. *Implementing Mathematics with the Nuprl Proof Development System.* Prentice-Hall, NJ, 1986.

[4] Michael Gordon and Tom Melham. *Introduction to HOL: A Theorem Proving Environment for Higher-Order Logic.* Cambridge University Press, Cambridge, 1993.

[5] A. Heyting, editor. *L. E. J. Brouwer. Collected Works*, volume 1. North-Holland, Amsterdam, 1975. (see On the foundations of mathematics 11-98.).

[6] S. C. Kleene and R. E. Vesley. *Foundations of Intuitionistic Mathematics.* North-Holland, 1965.

[7] Christine Paulin-Mohring. Inductive definitions in the system Coq; rules and properties. In J. F. Groote M. Bezem, editor, *Typed Lambda Calculi and Applications*, Lecture Notes in Computer Science. Springer-Verlag, 1993.

[8] Bertrand Russell. *The Principles of Mathematics.* Cambridge University Press, Cambridge, 1908.

[9] Alfred Tarski. *The Concept of Truth in Formalized Languages*, pages 152–278. Clarendon Press, Oxford, 1956. In *Logic, Semantics, Meta-Mathematics.*

[10] A.N. Whitehead and B. Russell. *Principia Mathematica*, volume 1, 2, 3. Cambridge University Press, 2nd edition, 1925–27.

[2] Brouwer's ideas are especially potent for Nuprl because it is also a *Logical Programming Environment* in the spirit of early Lisp machines, and Lisp embodies Brouwer's notions quite well.

[3] It is easy to model ZFC set theory in ZFC plus κ one inaccessible cardinal. In contrast, modeling CTT with one universe using *partial equivalence relations* (PER models) [2] rather than CTT with two universes is very informative.