Type Theory and Themes in Philosophical Logic

Greg Restall

Arché Research Centre, Philosophy Department,
The University of St Andrews · gr69@st-andrews.ac.uk

Type theorists share interests and concerns with philosophical logicians. This isn't surprising, since Martin-Löf is (among other things) a philosopher, and type theory was born in philosophy [46, 47, 48, 49]. Although type theory has come into its own in computer science—and, more recently, in mathematics with the rise of assistants—the connections between type theory and philosophical logic go beyond Martin-Löf's original motivations. There are many fresh points of contact with active research areas in philosophical logic. In the interest of fostering communication between these different traditions, I will sketch some of these connections.

Modal and Substructural Logics. Modal logics (extending propositional logic with modal operators, like $\Box$, for necessity, and $\Diamond$ for possibility) became a focus in philosophical logic in the second half of the 20th Century [32, 36], while its application to computer science took some time, with the development of dynamic logic [29, 63, 64]. Substructural logics, on the other hand, arose independently inside philosophy [2, 3, 72] with the study of relevant logics and entailment, linguistics [38, 39, 40, 54, 55] with the Lambek calculus, and computer science [27, 28, 80, 81], in linear logic. Recent work on modal [10, 30, 34] and substructural [45, 53] type theories provide natural areas of intersection with contemporary philosophical logic. The vast bulk of formal work in modal logic uses possible worlds semantics [11, 14, 15], as does philosophical work on substructural logics [43, 44, 59, 67]. These models have their use to represent *propositions* (types) and the entailment between them, but provide little, to no insight concerning the identity of the *proofs* or *constructions* that bear those types, so they can be of only limited use in modelling a properly rich type theory. The same goes for work in *algebraic* models.[1]

Modal and substructural *algebras*, on the other hand, are well understood structures [22, 25, 58] which should prove useful for type-theoretic considerations. One important insight has been the centrality of *residuation* (Galois connection, adjunction) as a unifying principle [18, 21, 22, 66]. A necessity modality $\Box$ gains its distinctive features in connection connected with dual, *possibility* modality $\blacklozenge$, for which we have $a \leq \Box b$ iff $\blacklozenge a \leq b$. Work on the proof theory of modal and substructural logics [7, 62] can provide a more natural point of connection with categorical semantics for type theories [41].

Intensionality and Identity. Homotopy type theory [70, 79] and cubical type theory [4, 5, 16] bring the logic of *identity* into focus, by raising the prospect of different grounds for an identity fact of the form $a = b$. Questions about how best to model identity in a type theory raises questions that philosophers ask using the terms *sense* and *reference*. If $a = b$ then the two different terms a and b must have the same *reference* (or value), but the possibility remains open that these two terms might have different *senses* (meanings). The *extensional* theory of identity is straightforward: everything is identical to itself, and not to anything else [42, p. 192]. Once we move from reference to *sense*, and consider non-extensional phenomena, such as *meaning*, *knowledge*, *proof*, or *construction*, matters are more nuanced [20, 24, 50]. We might

[1] These are partially ordered structures $a \leq b$ iff a entails b. This is a degenerate category in which there is at most one arrow between any two objects.

know that Clark Kent is Clark Kent, without knowing that Clark Kent is Superman, even though Clark Kent *is* Superman. The terms 'Clark Kent' and 'Superman' refer to the same item, but do so in different ways [56, 57]. Different accounts of the semantics of identity give us different options for understanding *how* terms might pick out their values, and, more generally the space of possible semantic values of identity claims between items of each type [9].

Classical Logic. Intuitionistic type theory is constructive, and this raises the question of the status of classical reasoning. There are different approaches to relating classical and constructive logic, such as embedding classical reasoning *inside* a constructive language by way of a translation [78, Sect. 2.3] (which can be interpreted by way of continuations [69, 77]), or extending the term vocabulary [60, 61] or by extending judgement forms to include positive and negative judgements [17, 19]. These approaches parallel considerations in philosophical logic. Some approaches to classical logic start with intuitionist natural deduction and add new inference forms [51, 52], others are *bilateral* [65], encoding proofs involving both positive and negative judgement forms [73, 74, 75]. Other approaches interpret the sequent calculus in terms of assertion and denial [71, 68]. Recent work on assertion and denial distinguishes two forms: *strongly* deny p is to rule p out; to *weakly* deny it is to withdraw its assertion and to keep open the option to strongly deny it [33]. Weak and strong denial both clash with assertion[2] and when treating assertion and denial, it is important to distinguish their strong and weak forms.

Speech Acts. A type theory is an account of *judgement*. One distinctive features of *dependent* type theory is that the rules governing different concepts can interleave the different judgement forms. Whether $B(a)$ counts as a type may depend on whether the term a inhabits another type A. To focus on propositions, whether B counts as a prop can depend on whether another proposition A is true.[3] Traditional grammars form the syntax of the language first, independently of truth conditions, but some contemporary theories of propositional content mirror this structure. 'The king of France is bald' expresses an assertion only if there is a King of France [76]: predication *presupposes* reference. Dependent type theories form a natural context in which presupposition phenomena like this can be modelled and studied.

However, we can do more with our language than form assertions, denials, suppositions and inferences [35]. In natural and in artificial languages, we find imperatives, promises, requests, etc., which differ from judgement forms in many ways [6, 8]. Philosophers and linguists have done a great deal of work on the function and logic different forms of speech acts [23, 26, 37], which may prove salient when exploring the semantics of languages with imperatives, and other non-assertoric forms.

Formal and Applied Theory. Beneath these points of contact, there is a deeper connection between *applied* computational type theory and philosophical logic. A formal type theory is a structural presentation of forms of judgement, which may be interpreted, as Martin-Löf showed us, as a theory of sets, of computation, of propositions, and in other ways besides. A properly *computational* type theory takes the inductive presentation of types and terms to stand atop a fundamental computational substrate [1, 31, 47]. Terms describe *computations*, which are classified by types. The distinction between a formal and an applied type theory parallels

[2] And dually, we have strong assertion (ruling p *in*) and weak assertion (which merely withdraws the strong denial of p and leaves the possibility of strongly asserting p open).

[3] A formation rule for the conditional $\supset$ states that $A \supset B$ is a proposition when A is a proposition and requires that B is a proposition *only given that A is true* [49].

Brandom's distinction between a formal and a material account of inference [12, 13]. A properly *material* theory of inference describes the inferential connections between the judgements implicit in our everyday practice. As I learn the concept 'square' I learn that squares are not circles, and that squares have four sides, of equal length. These are not learned as *facts* we articulate, but as capacities we exercise. To learn a language is to learn how different concepts bear upon each other and on the world. Rules for logical concepts enable us to *make explicit* these inferential relations between our more basic commitments, in that the rules governing those concepts tie them to the underlying and preexisting communicative practice. We take justification for a conditional claim $A \to B$ to be provided by the means to justify B in a context where we take A as given. The formal structure of a material theory inference bears a remarkable resemblance to the computational theory of types, where instead of the norms governing justifications and grounds for human judgements, we have computations and their classification into different types. So, it is not surprising that insights from one area can be applied to the other, since there is a single formal framework that describes both domains.

A fresh challenge for research, however, is to develop a properly *hybrid* type theory, encompassing both computation and human communicative practices, in order to better understand the possibilities for communication involving both human judgement and machine computation. After all, two domains of application for the discipline of type theory are in (a) the design of expressive and performant dependently typed functional programming languages, and (b) the design of modular, expressive and natural proof assistants. Both of these tasks involve taking both the *computational* and the *communicative* roles of the underlying type theory seriously, so it seems appropriate to adopt a framework that helps us keep both roles in view at once.[4]

References

[1] S. F. Allen, M. Bickford, R. L. Constable, R. Eaton, C. Kreitz, L. Lorigo, and E. Moran. Innovations in computational type theory using Nuprl. *Journal of Applied Logic*, 4(4):428–469, 2006.

[2] Alan R. Anderson and Nuel D. Belnap. *Entailment: The Logic of Relevance and Necessity*, volume 1. Princeton University Press, Princeton, 1975.

[3] Alan Ross Anderson, Nuel D. Belnap, and J. Michael Dunn. *Entailment: The Logic of Relevance and Necessity*, volume 2. Princeton University Press, Princeton, 1992.

[4] Carlo Angiuli, Guillaume Brunerie, Thierry Coquand, Robert Harper, Kuen-Bang Hou (Favonia), and Daniel R. Licata. Syntax and models of Cartesian cubical type theory. *Mathematical Structures in Computer Science*, 31(4):424–468, 2021.

[5] Carlo Angiuli, Robert Harper, and Todd Wilson. Computational higher-dimensional type theory. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages*, POPL '17, pages 680–693, New York, NY, USA, 2017. Association for Computing Machinery.

[6] J. L. Austin. *How to do things with words*. Clarendon Press, Cambrige, 1962.

[7] Nuel Belnap. Display logic. *Journal of Philosophical Logic*, 11:375–417, 1982.

[8] Nuel Belnap. Declaratives are not enough. *Philosophical Studies*, 59(1):1–30, 1990.

[9] Nuel Belnap. Under Carnap's lamp: Flat pre-semantics. *Studia Logica*, 80(1):1–28, 2005.

[10] Lars Birkedal. Developing theories of types and computability via realizability. *Electronic Notes in Theoretical Computer Science*, 34:2, 2000.

[11] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, 2001.

[12] Robert B. Brandom. *Making It Explicit*. Harvard University Press, 1994.

[4] Thanks to the referees for helpful comments on the first draft of this abstract.

[13] Robert B. Brandom. *Articulating Reasons: an introduction to inferentialism*. Harvard University Press, 2000.

[14] Alexander Chagrov and Michael Zakharyaschev. *Modal Logic*, volume 35 of *Oxford Logic Guides*. Oxford University Press, 1997.

[15] Brian F. Chellas. *Modal Logic: An Introduction*. Cambridge University Press, Cambridge, 1980.

[16] Cyril Cohen, Thierry Coquand, Simon Huber, and Anders Mörtberg. Cubical type theory: A constructive interpretation of the univalence axiom. In *21st International Conference on Types for Proofs and Programs (TYPES 2015)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018.

[17] Pierre-Louis Curien and Hugo Herbelin. The duality of computation. In *ICFP '00: Proceedings of the fifth* ACM SIGPLAN *international conference on Functional programming*, pages 233–243, New York, NY, USA, 2000. ACM.

[18] David Darais and David Van Horn. Constructive Galois connections. *Journal of Functional Programming*, 29, 2019.

[19] Paul Downen and Zena M. Ariola. Compiling with classical connectives. *Logical Methods in Computer Science*, 16(3), 07 2019.

[20] Michael Dummett. Sense and reference from a constructivist standpoint. *The Bulletin of Symbolic Logic*, 27(4):485–500, 2021.

[21] J. Michael Dunn. Gaggle theory: An abstraction of Galois connections and residuation with applications to negation and various logical operations. In *Logics in AI, Proceedings European Workshop* JELIA *1990*, volume 478 of *Lecture Notes in Computer Science*. Springer-Verlag, 1991.

[22] J. Michael Dunn and Gary M. Hardegree. *Algebraic Methods in Philosophical Logic*. Clarendon Press, Oxford, 2001.

[23] Daniel Fogal, Daniel W. Harris, and Matt Moss, editors. *New work on speech acts*. Oxford University Press, Oxford, United Kingdom, 2018.

[24] Gottlob Frege. Sense and reference. *The Philosophical Review*, 57(3):209–230, 1948.

[25] Nikolaos Galatos. *Residuated Lattices: An Algebraic Glimpse at Substructural Logics*. Elsevier, Amsterdam; Boston, 2007.

[26] Bart Geurts. Communication as commitment sharing: speech acts, implicatures, common ground. *Theoretical Linguistics*, 45(1-2):1–30, 2019.

[27] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–101, 1987.

[28] Jean-Yves Girard, Yves Lafont, and Paul Taylor. *Proofs and Types*, volume 7 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1989.

[29] Robert I. Goldblatt. *Logics of Time and Computation*. Number 7. CSLI Publications, 1987.

[30] Daniel Gratzer, Jonathan Sterling, and Lars Birkedal. Implementing a modal dependent type theory. *Proceedings of the ACM on Programming Languages*, 3(ICFP):1–29, 2019.

[31] Robert Harper. *Practical Foundations for Programming Languages*. Cambridge University Press, second edition, 2016.

[32] G. Hughes and M. Cresswell. *An Introduction to Modal Logic*. Methuen, London, 1968.

[33] Luca Incurvati and Julian J. Schlöder. *Reasoning With Attitude: Foundations and Applications of Inferential Expressivism*. Oxford University Press, New York, 2023.

[34] G. A. Kavvos and Daniel Gratzer. Under lock and key: A proof system for a multimodal logic. *The Bulletin of Symbolic Logic*, 29(2):264–293, 2023.

[35] John T. Kearns. Conditional assertion, denial, and supposition as illocutionary acts. *Linguistics and Philosophy*, 29(4):455–485, 2006.

[36] Saul A. Kripke. A completeness theorem in modal logic. *The Journal of Symbolic Logic*, 24(1):1–14, 1959.

[37] Rebecca Kukla and Mark Lance. *Yo! and Lo!: The Pragmatic Topography of the Space of Reasons*. Harvard University Press, 2009.

[38] Joachim Lambek. Deductive systems and categories I: Syntactic calculus and residuated categories. *Mathematical Systems Theory*, 2:287–318, 1968.

[39] Joachim Lambek. Deductive systems and categories II: Standard constructions and closed categories. In Peter Hilton, editor, *Category Theory, Homology Theory and their Applications*, number 86 in Lecture Notes in Mathematics, pages 76–122. Springer-Verlag, 1969.

[40] Joachim Lambek. Deductive systems and categories III: Cartesian closed categories, intuitionist propositional calculus, and combinatory logic. In Bill Lawvere, editor, *Toposes, Algebraic Geometry, and Logic*, number 274 in Lecture Notes in Mathematics, pages 57–82. Springer-Verlag, 1972.

[41] F. William Lawvere. Adjointness in foundations. *Dialectica*, 23(3-4):281–296, 1969.

[42] David K. Lewis. *On the Plurality of Worlds*. Blackwell, Oxford, 1986.

[43] Edwin Mares. Relevant logic and the theory of information. *Synthese*, 109(3):345–360, 1997.

[44] Edwin D. Mares. *Relevant Logic: A Philosophical Interpretation*. Cambridge University Press, 2004.

[45] Danielle Marshall and Dominic Orchard. Non-linear communication via graded modal session types. *Information and Computation*, 301:105234, 2024.

[46] Per Martin-Löf. *Notes on Constructive Mathematics*. Almqvist and Wiksell, Stockholm, 1970.

[47] Per Martin-Löf. Constructive mathematics and computer programming. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 312(1522):501–518, 1984.

[48] Per Martin-Löf. *Intuitionistic Type Theory: Notes by Giovanni Sambin of a Series of Lectures Given in Padua, June 1980*. Number 1 in Studies in Proof Theory. Bibliopolis, Naples, 1984.

[49] Per Martin-Löf. On the meaning of the logical constants and the justification of the logial laws. *Nordic Journal of Philosophical Logic*, 1:11–69, 1996.

[50] Per Martin-Löf. The sense/reference distinction in constructive semantics. *The Bulletin of Symbolic Logic*, 27(4):501–513, 2021.

[51] Peter Milne. Classical harmony: rules of inference and the meaning of the logical constants. *Synthese*, 100:49–94, 1994.

[52] Peter Milne. Harmony, purity, simplicity and a 'seemingly magical fact'. *Monist*, 85(4):498–534, 2002.

[53] Benjamin Moon, Harley Eades, and Dominic Orchard. Graded modal dependent type theory. In *Programming Languages and Systems: 30th European Symposium on Programming, Esop 2021*. Springer International Publishing, 2021.

[54] Michael Moortgat. *Categorial Investigations: Logical Aspects of the Lambek Calculus*. Foris, Dordrecht, 1988.

[55] Glyn Morrill. *Type Logical Grammar: Categorial Logic of Signs*. Kluwer, Dordrecht, 1994.

[56] Yiannis N. Moschovakis. *Sense and denotation as algorithm and value*, volume Volume 2 of *Lecture Notes in Logic*, pages 210–249. Springer-Verlag, Berlin, 1993.

[57] Reinhard Muskens. Sense and the computation of reference. *Linguistics and Philosophy*, 28(4):473–504, 2005.

[58] Hiroakira Ono. *Proof Theory and Algebra in Logic*. Springer, Aug 2019.

[59] Francesco Paoli. *Substructural Logics: A Primer*. Springer, May 2002.

[60] Michel Parigot. $\lambda\mu$-calculus: An algorithmic interpretation of classical natural deduction. In Andrei Voronkov, editor, *International Conference on Logic for Programming Artificial Intelligence and Reasoning*, volume 624 of *Lecture Notes in Artificial Intelligence*, pages 190–201. Springer, 1992.

[61] Michel Parigot. Classical proofs as programs. In George Gottlob, Alexander Leitsch, and Daniele Mundici, editors, *Computational Logic and Proof Theory*, volume 713 of *Lecture Notes in Computer Science*, pages 263–276. Springer, 1993.

[62] Francesca Poggiolesi. *Gentzen Calculi for Modal Propositional Logic*. Trends in Logic. Springer, 2010.

[63] V. R. Pratt. Semantical considerations on Floyd–Hoare logic. In *Proceedings of the 17th IEEE Symposium on Foundations of Computer Science*, pages 109–121, 1976.

[64] V. R. Pratt. Dynamic algebras and the nature of induction. In *12th ACM Symposium on the Theory of Computation*, Los Angeles, 1980.

[65] Huw Price. Why 'not'? *Mind*, 99(394):222–238, 1990.

[66] Greg Restall. Display logic and gaggle theory. *Reports in Mathematical Logic*, 29:133–146, 1995.

[67] Greg Restall. *An Introduction to Substructural Logics*. Routledge, 2000.

[68] Greg Restall. Multiple conclusions. In Petr Hájek, Luis Valdés-Villanueva, and Dag Westerståhl, editors, *Logic, Methodology and Philosophy of Science: Proceedings of the Twelfth International Congress*, pages 189–205. KCL Publications, 2005.

[69] John C. Reynolds. The discoveries of continuations. LISP *and Symbolic Computation*, 6(3-4):233–247, 1993.

[70] Egbert Rijke. Introduction to homotopy type theory. ArXiV 2212.11082, 2022. Prepublication version. To be published by *Cambridge University Press*.

[71] David Ripley. Bilateralism, coherence, warrant. In Friederike Moltmann and Mark Textor, editors, *Act-Based Conceptions of Propositional Content*, pages 307–324. Oxford University Press, Oxford, UK, 2017.

[72] Richard Routley and Robert K. Meyer. Semantics of entailment. In Hugues Leblanc, editor, *Truth, Syntax and Modality*, pages 194–243. North Holland, 1973. Proceedings of the Temple University Conference on Alternative Semantics.

[73] Ian Rumfitt. "Yes" and "No". *Mind*, 109(436):781–823, 2000.

[74] Ian Rumfitt. *The Boundary Stones of Thought: An Essay in the Philosophy of Logic*. Oxford University Press, 2015.

[75] Timothy Smiley. Rejection. *Analysis*, 56:1–9, 1996.

[76] P. F. Strawson. On referring. *Mind*, 59(235):320–344, 1950.

[77] Th. Streicher and B. Reus. Classical logic, continuation semantics and abstract machines. *Journal of Functional Programming*, 8(6):543–572, 1998.

[78] A. S. Troelstra and H. Schwichtenberg. *Basic Proof Theory*, volume 43 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, Cambridge, second edition, 2000.

[79] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. https://homotopytypetheory.org/book, Institute for Advanced Study, 2013.

[80] Philip Wadler. Linear types can change the world! In M. Broy and C. Jones, editors, *Programming Concepts and Methods*, Sea of Galilee, Israel, April 1990. North Holland. IFIP TC 2 Working Conference.

[81] David Walker. Substructural type systems. In Benjamin C. Pierce, editor, *Advanced Topics in Types and Programming Languages*, chapter 1, pages 3–44. MIT Press, 2004.