No-Go Theorems for Distributive Laws

Maaike Zwart and Dan Marsden

May 3, 2019

Monads have become a key tool in computer science, and broader applications of category theory. For example, they are used in the semantics of computational effects [17], structuring functional programs [19, 23], probabilistic modelling [9, 11, 18], natural language semantics [21], quantum foundations [1, 12], formal language theory [4], and even appear explicitly in the standard library of the Haskell programming language [10].

Monads are a categorical concept. A monad on a category $\mathcal{C}$ consists of an endofunctor T and two natural transformations $1 \Rightarrow T$ and $\mu : T \circ T \Rightarrow T$ satisfying certain axioms. Given two monads with underlying functors S and T, it is natural to ask if $T \circ S$ always carries the structure of a monad. This would, for example, provide a way to combine simple monads together to model more complex computational effects. Unfortunately, monads cannot generally be combined in this way, but Beck [2] has shown that a natural transformation of type $S \circ T \Rightarrow T \circ S$ satisfying coherence conditions, referred to as a *distributive law*, is sufficient for $T \circ S$ to form a monad.

General-purpose techniques have been developed for constructing distributive laws [5, 8, 15, 16]. These methods are highly valuable, for as stated in [5]: *"It can be rather difficult to prove the defining axioms of a distributive law."* In fact, it can be so difficult that on occasion a distributive law has been published which later turned out to be incorrect; see [14] for an overview of such cases involving the powerset monad. More commonly, when searching for a distributive law the "obvious" candidate does not always work out. The challenge is then to search for other possibilities, but such a search may be doomed to failure, wasting valuable research time.

The literature has tended to focus on positive results, either demonstrating specific distributive laws, or developing general-purpose techniques for constructing them. The key contribution of our work is to provide a family of no-go theorems establishing conditions under which no distributive law can exist [1]. There has been a relative paucity of such negative results. The most well-known result of this type appears in [22], where it is shown that there is no distributive law combining the powerset and finite probability distribution monads, via a proof credited to Plotkin. This result was strengthened to show that the composite functor carries no monad structure at all in [7]. Recently, the same proof technique was used to show that composing the covariant powerset functor with itself does not carry any monad structure [14], correcting an earlier error in the literature [15]. To the best of our knowledge, these are currently the only published impossibility results.

Contribution

Monads have deep connections with universal algebra. Every finitary monad is the free algebra monad for an algebraic theory. For example the list, multiset and finite powerset monads are the free algebra monads for the theories of monoids, commutative monoids and join semilattices respectively. Given this correspondence, one might anticipate that monads composed via distributive laws relate to some form of composite algebraic theory. This turns out to be a good intuition, and a description of such a correspondence appears in [20], giving explicit algebraic conditions for the existence of a distributive law, inspired by earlier work in the setting of Lawvere theories [6].

We take advantage of this algebraic viewpoint. Our theorems are phrased in terms of abstract properties of the algebraic theories corresponding to both monads, whereas all previous results have needed to fix at least one of the two monads. We typically require two axiomatic components:

1. Conventional equational axioms. For example, idempotence $x * x = x$ and unitality $x + 0 = x$ axioms are of central importance.
2. More delicate properties of the algebraic theories, restricting how the variables appearing in certain provable equalities can vary across each side of the equation.

We restrict our attention to monads on the category of sets and functions, as this is already an incredibly rich setting. Our contribution can be divided into two families of results:

[1] This short submission is based on the conference paper [24] Further technical details appear in [25]

- Firstly, we widely generalize Plotkin's method [22], leading to purely algebraic conditions under which a no-go result holds. This theorem recovers all the known negative results we are aware of, and many useful new results. We immediately see that in fact there is no distributive law for any of the four pairwise combinations involving the powerset and distribution monads. The key ingredients for this theorem are binary terms that are idempotent and commutative. In subsequent theorems we establish that the restriction to binary terms is unnecessary, and that requiring commutativity is also not essential.

- Secondly, we identify an entirely novel approach, distinct from Plotkin's method. This technique leads to four new theorems, covering combinations of list and tree like monads that were previously out of reach:

  - A theorem establishing conditions under which algebraic theories with more than one constant do not combine well with other theories. This theorem identified a previously unnoticed error claiming a distributive law $L \circ (- + E) \Rightarrow (- + E) \circ L$ for the list and exception monads [16, Example 4.12] pointing to an error in [16, Theorem 4.6].
  - A no-go theorem for monads that do not satisfy the so-called *abides* equation [3]:
    $$(w + x) * (y + z) = (w * y) + (x * z)$$
    One application is to resolve negatively the open question [15, 16] of whether the list monad distributes over itself, a source of a previous error [13] in the literature.
  - A third no-go theorem focuses on the combination of idempotence and units. This theorem yields further new results, for example there is no distributive law of type $PM \Rightarrow MP$ for the powerset monad over the multiset monad.
  - Finally, we present a theorem defining conditions under which at most one distributive law can exist. For example, the well-known distributive law for the multiset monad over itself is unique.

Physicists emphasize the importance of *no-go theorems*, because they clearly identify theoretical directions that cannot succeed. We follow this example, and hope that by sharing our results, we prevent others from wasting time on forlorn searches for distributive laws that do not exist.

References

[1] Samson Abramsky, Rui Soares Barbosa, Nadish de Silva, and Octavio Zapata. The quantum monad on relational structures. In Kim G. Larsen, Hans L. Bodlaender, and Jean-François Raskin, editors, *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS 2017, August 21-25, 2017 - Aalborg, Denmark*, volume 83 of *LIPIcs*, pages 35:1–35:19. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[2] Jon Beck. Distributive laws. In B. Eckmann, editor, *Seminar on Triples and Categorical Homology Theory*, pages 119–140. Springer Berlin Heidelberg, 1969.

[3] Richard Bird. *Lectures on Constructive Functional Programming*. Oxford University Computing Laboratory Programming Research Group, 1988.

[4] Mikołaj Bojańczyk. Recognisable languages over monads. In *International Conference on Developments in Language Theory*, pages 1–13. Springer, 2015.

[5] Marcello M. Bonsangue, Helle Hvid Hansen, Alexander Kurz, and Jurriaan Rot. Presenting distributive laws. In Reiko Heckel and Stefan Milius, editors, *Algebra and Coalgebra in Computer Science*, pages 95–109, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[6] Eugenia Cheng. Distributive laws for Lawvere theories. *CoRR*, abs/1112.3076v2, 2011.

[7] Fredrik Dahlqvist and Renato Neves. Compositional semantics for new paradigms: Probabilistic, hybrid and beyond. *CoRR*, abs/1804.04145, 2018.

[8] Fredrik Dahlqvist, Louis Parlant, and Alexandra Silva. Layer by layer - combining monads. In Bernd Fischer and Tarmo Uustalu, editors, *Theoretical Aspects of Computing - ICTAC 2018 - 15th International Colloquium, Stellenbosch, South Africa, October 16-19, 2018, Proceedings*, volume 11187 of *Lecture Notes in Computer Science*, pages 153–172. Springer, 2018.

[9] Ernst-Erich Doberkat. *Stochastic relations: foundations for Markov transition systems*. Chapman and Hall/CRC, 2007.

[10] Paul Hudak, Simon Peyton Jones, Philip Wadler, Brian Boutel, Jon Fairbairn, Joseph Fasel, María M Guzmán, Kevin Hammond, John Hughes, Thomas Johnsson, et al. Report on the programming language Haskell: a non-strict, purely functional language version 1.2. *ACM SigPlan notices*, 27(5):1–164, 1992.

[11] Bart Jacobs. Measurable spaces and their effect logic. In *2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 83–92. IEEE, 2013.

[12] Bart Jacobs, Jorik Mandemaker, and Robert Furber. The expectation monad in quantum foundations. *Information and Computation*, 250:87–114, 2016.

[13] David J King and Philip Wadler. Combining monads. In *Functional Programming, Glasgow 1992*, pages 134–143. Springer, 1993.

[14] Bartek Klin and Julian Salamanca. Iterated covariant powerset is not a monad. In *Proceedings 34th Conference on the Mathematical Foundations of Programming Semantics, MFPS 2018*, 2018.

[15] Ernest Manes and Philip Mulry. Monad compositions I: general constructions and recursive distributive laws. *Theory and Applications of Categories*, 18:172–208, 04 2007.

[16] Ernie Manes and Philip Mulry. Monad compositions II: Kleisli strength. *Mathematical Structures in Computer Science*, 18(3):613âĂŞ643, 2008.

[17] Eugenio Moggi. Notions of computation and monads. *Information and Computation*, 93(1):55 – 92, 1991. Selections from 1989 IEEE Symposium on Logic in Computer Science.

[18] Prakash Panangaden. *Labelled Markov Processes*. World Scientific, 2009.

[19] Simon Peyton Jones. *Tackling the Awkward Squad: monadic input/output, concurrency, exceptions, and foreign-language calls in Haskell*, pages 47–96. IOS press, Amsterdam, 2001.

[20] Maciej Pirog and Sam Staton. Backtracking with cut via a distributive law and left-zero monoids. *Journal of Functional Programming*, 27, 2017.

[21] Chung-chieh Shan. Monads for natural language semantics. *arXiv preprint cs/0205026*, 2002.

[22] Daniele Varacca and Glynn Winskel. Distributing probability over non-determinism. *Mathematical Structures in Computer Science*, 16(1):87–113, 2006.

[23] Philip Wadler. Monads for functional programming. In Johan Jeuring and Erik Meijer, editors, *Advanced Functional Programming*, pages 24–52, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[24] M. Zwart and D. Marsden. No-go theorems for distributive laws. 2019. LiCS 2019, to appear.

[25] Maaike Zwart and Dan Marsden. Don't try this at home: No-go theorems for distributive laws. *CoRR*, abs/1811.06460, 2018.