Brief Summary of Research Activities

William DeMeo

1. Introduction

The last two decades have witnessed explosive growth in the number of applications of abstract mathematics, especially in computer science, and this trend continues on a steep upward trajectory. Mathematical fields like *universal algebra* and *category theory* have long had a substantial influence on the development of theoretical computer science, particularly in *domain theory*, *denotational semantics*, and *programming languages research* (e.g., [7, 8, 15, 16]). Dually, progress in theoretical computer science has informed and inspired a substantial amount of pure mathematics in the last half-century (e.g., [2, 14, 16]), just as physics and physical intuition motivated so many of the mathematical discoveries of the last two centuries.

Universal algebra has also been invigorated by a recently discovered connection to complexity theory, and this connection was the primary focus of my work from 2015 to 2017. The theory of finite algebras has turned out to be broadly applicable in obtaining deep and definitive results about the complexity of algorithmic problems in the broad class of *constraint satisfaction problems* (CSPs). We call this area *algebraic CSP theory.* The tools of universal algebra, combined with combinatorial reasoning about polymorphisms (multi-variable endomorphisms) acting on finite graphs and other relational structures, has produced deep new results concerning the complexity of CSPs, and has explained and united older results. Furthermore, almost all of the new results have been turned around and used to produce startling new algebraic results of a kind never seen before in universal algebra.

More recently my focus has been the mechanization of mathematics in functional programming languages and proof assistants such as Coq [17], Lean [1], and Scala [13] (see, for example, Section 2.1 below).

2. Some Recent Work

2.1. **Formal Foundations for Informal Mathematics Research.**

After a half century of progress in universal algebra, category theory, proof theory, and type theory, it is now possible to present the foundations of mathematics and computer science in a unified way. Every mathematician and theoretical computer scientist knows from experience that the choices one makes at the outset—about logical foundations, languages, notations—play a decisive role in the theory that develops. Indeed, every new theorem is a product of the language in which it is expressed, and a poor choice of language or foundations can severely limit what is possible. In my view this is among the most fascinating aspects of scientific research.

Universal algebra (my field of specialty) was developed by some of the most brilliant mathematicians of the 20th century. Nonetheless, it seems that some important parts of the subject were created with little or no regard for the computational content of the theory. With the advent of dependent type theory, functional programming, and proof assistants, this aspect of our subject makes universal algebra less relevant and less powerful than it has the potential to be. It is this conviction that motivated me to embark on an ambitious new project to reexamine the core foundations of universal algebra, redevelop them as necessary in a category theoretic and constructive language, and produce a library of formalized universal algebra in the language of dependent type theory, and codified in the

Lean proof assistant language. For more details about this research program, including some early progress, please see the document demeo_informal_foundations.pdf, which is the full proposal for the *Formal Foundations for Informal Mathematics Research* (FFIMR) project.

2.2. **New Characterizations of Bounded Lattices and Fiber Products.** About a year ago I began collaborating with Peter Mayr (CU Boulder) and Nik Ruskuc (University of St. Andrews) who were interested in knowing when a homomorphism $\varphi\colon \mathbf{F} \to \mathbf{L}$ from a finitely generated free lattice $\mathbf{F}$ onto a finite lattice $\mathbf{L}$ has a kernel $\ker \varphi$ that is a finitely generated sublattice of $\mathbf{F}^2$. We conjectured that this could be characterized by whether or not the homomorphism is *bounded*,[1] and I presented a proof of one direction of this conjecture at the Algebras and Lattices in Hawai'i conference earlier this year. Last month we proved the converse and thus confirmed our conjecture. Mayr and Ruskuc have had in mind an application for the fact that our new result is equivalent to a characterization of *fiber products* of lattices. With the proof of our new characterization theorem complete, we expect to have a manuscript ready for submission by the end of February 2019.

2.3. **Tractability of Deciding Existence of Special Terms.** Among my most recent research accomplishments was the discovery that a certain decision problem about algebraic structures that previously seemed out of reach is actually tractable. In collaborative project with Ralph Freese (University of Hawaii) and Matthew Valeriote (McMaster University), we considered the following practical question: Given a finite algebra $\mathbf{A}$ in a finite language, can we efficiently decide whether the variety generated by $\mathbf{A}$ has a so called *difference term*? In a paper that will soon appear in the *International Journal of Algebra and Computation* (IJAC), we answer this question (positively) in the idempotent case and then describe algorithms for constructing difference term operations [4].

A *difference term* for a variety $\mathcal{V}$ is a ternary term d in the language of $\mathcal{V}$ that satisfies the following: if $\mathbf{A} = \langle A, \dots\rangle \in \mathcal{V}$, then for all $a, b \in A$ we have

$$d^{\mathbf{A}}(a, a, b) = b \quad \text{and} \quad d^{\mathbf{A}}(a, b, b)\ [\theta, \theta]\ a, \tag{1}$$

where θ is any congruence containing (a, b) and $[\cdot, \cdot]$ denotes the *commutator*. When the relations in (1) hold for a single algebra $\mathbf{A}$ and term d we call $d^{\mathbf{A}}$ a *difference term operation* for $\mathbf{A}$.

Difference terms are widely studied in the general algebra literature. (See, for example, [9, 10, 11, 12].) There are many reasons to study difference terms, but one obvious reason is the following: if we know that a variety has a difference term, this fact allows us to deduce that the algebras inhabiting this variety must satisfy certain interesting properties. Perhaps the most important property can be summarized in the following heuristic slogan: *varieties with a difference term have a commutator that behaves nicely.*

Computers have become invaluable as a research tool and have helped to broaden and deepen our understanding of algebraic structures and the varieties they inhabit. This would not be possible without the efforts of researchers who, over the last three decades, have found ingenious ways to coax computers into solving challenging abstract algebraic decision problems, and to do so very quickly. To give a couple of examples related to our own work, it is proved in [18] (respectively, [6]) that deciding whether a finite idempotent algebra generates a variety that is congruence-n-permutable (respectively, congruence-modular) is *tractable*. Our work continues this effort by presenting an efficient algorithm for deciding whether a finitely generated idempotent variety has a difference term.

[1] See [5] for the definition of a bounded lattice homomorphism.

2.4. **Algebras and Algorithms, Structure and Complexity.** There is now an extensive and growing research literature on algebraic CSP theory. Consequently, the field of universal algebra is more active now than at any other time in its brief eighty-five year history. In 2015, I joined a group of 8 other scientists to form a universal algebra research group and secure a 3-year NSF grant for a project called "Algebras and Algorithms, Structure and Complexity Theory." We focus primarily on classes of algebraic problems whose solutions yield definitive results about the complexity of CSPs. These include scheduling problems, resource allocation problems, and problems reducible to solving systems of linear equations. CSPs are theoretically solvable, but some are not solvable efficiently in practice. Our work provides procedures for deciding whether a given instance of a CSP is tractable or intractable, and we develop efficient algorithms for finding solutions in the tractable cases. A number of new results in algebraic CSP theory that I proved in the last two years are presented in a 50-page manuscript entitled "Universal Algebraic Methods for Constraint Satisfaction Problems," [3] which I authored with Cliff Bergman and which was recently accepted for publication in *Logical Methods in Computer Science* (LMCS).

References

[1] Jeremy Avigad, Mario Carneiro, Leonardo de Moura, Johannes Hölzl, and Sebastian Ullrich. The Lean theorem proving language, 2018. Carnegie Mellon University and Microsoft Research. URL: https://leanprover.github.io/.

[2] Andrej Bauer, Gordon D. Plotkin, and Dana S. Scott. Cartesian closed categories of separable Scott domains. *Theoret. Comput. Sci.*, 546:17–29, 2014. doi:10.1016/j.tcs.2014.02.042.

[3] Clifford Bergman and William DeMeo. Universal algebraic methods for constraint satisfaction problems. *CoRR: arXiv preprint*, abs/1611.02867, 2016. URL: https://arxiv.org/abs/1611.02867, arXiv:1611.02867.

[4] William DeMeo, Ralph Freese, and Matthew Valeriote. Polynomial-time tests for difference terms in idempotent varieties. *Intl. J. Algebra and Computation*, 2018. (to appear) preprint available at github.com/UniversalAlgebra/term-conditions/.

[5] Ralph Freese, Jaroslav Ježek, and J. B. Nation. *Free lattices*, volume 42 of *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI, 1995. URL: http://dx.doi.org/10.1090/surv/042, doi:10.1090/surv/042.

[6] Ralph Freese and Matthew A. Valeriote. On the complexity of some Maltsev conditions. *Internat. J. Algebra Comput.*, 19(1):41–77, 2009. URL: http://dx.doi.org/10.1142/S0218196709004956, doi:10.1142/S0218196709004956.

[7] Martin Hyland and John Power. The category theoretic understanding of universal algebra: Lawvere theories and monads. In *Computation, meaning, and logic: articles dedicated to Gordon Plotkin*, volume 172 of *Electron. Notes Theor. Comput. Sci.*, pages 437–458. Elsevier Sci. B. V., Amsterdam, 2007. URL: https://doi-org.colorado.idm.oclc.org/10.1016/j.entcs.2007.02.019, doi:10.1016/j.entcs.2007.02.019.

[8] A. Jung. *Cartesian closed categories of domains*, volume 66 of *CWI Tract*. Stichting Mathematisch Centrum, Centrum voor Wiskunde en Informatica, Amsterdam, 1989.

[9] Keith Kearnes, Ágnes Szendrei, and Ross Willard. A finite basis theorem for difference-term varieties with a finite residual bound. *Trans. Amer. Math. Soc.*, 368(3):2115–2143, 2016. URL: http://dx.doi.org/10.1090/tran/6509, doi:10.1090/tran/6509.

[10] Keith A. Kearnes. Varieties with a difference term. *J. Algebra*, 177(3):926–960, 1995. URL: http://dx.doi.org/10.1006/jabr.1995.1334, doi:10.1006/jabr.1995.1334.

[11] Keith A. Kearnes and Ágnes Szendrei. The relationship between two commutators. *Internat. J. Algebra Comput.*, 8(4):497–531, 1998. URL: http://dx.doi.org/10.1142/S0218196798000247, doi:10.1142/S0218196798000247.

[12] Keith A. Kearnes, Ágnes Szendrei, Ross Willard, and Keith A. Kearnes. Simpler Maltsev conditions for (weak) difference terms in locally finite varieties. *Algebra Universalis*, 78(4):555–561, 2017. doi:10.1007/s00012-017-0475-7.

[13] Martin Odersky, Philippe Altherr, Vincent Cremet, Burak Emir, Sebastian Maneth, Stéphane Micheloud, Nikolay Mihaylov, Michel Schinz, Erik Stenman, and Matthias Zenger. An overview of the Scala programming language. Technical Report IC/2004/64, 2004. EPFL, Lausanne, Switzerland. URL: https://www.scala-lang.org/.

[14] Don Pigozzi and Antonino Salibra. Lambda abstraction algebras: representation theorems. *Theoret. Comput. Sci.*, 140(1):5–52, 1995. Selected papers of AMAST '93 (Enschede, 1993). URL: http://dx.doi.org/10.1016/0304-3975(94)00203-U, doi:10.1016/0304-3975(94)00203-U.

[15] Gordon Plotkin and John Power. Algebraic operations and generic effects. *Applied Categorical Structures*, 11(1):69–94, Feb 2003. URL: https://doi.org/10.1023/A:1023064908962, doi:10.1023/A:1023064908962.

[16] Dana S. Scott. A type-theoretical alternative to ISWIM, CUCH, OWHY. *Theoret. Comput. Sci.*, 121(1-2):411–440, 1993. A collection of contributions in honour of Corrado Böhm on the occasion of his 70th birthday. URL: http://dx.doi.org/10.1016/0304-3975(93)90095-B, doi:10.1016/0304-3975(93)90095-B.

[17] The Coq Development Team. The Coq proof assistant, version 8.9.0, Jan 2019. URL: https://doi.org/10.5281/zenodo.2554024, doi:10.5281/zenodo.2554024.

[18] M. Valeriote and R. Willard. Idempotent n-permutable varieties. *Bull. Lond. Math. Soc.*, 46(4):870–880, 2014. URL: http://dx.doi.org/10.1112/blms/bdu044, doi:10.1112/blms/bdu044.