Journal of Algebra 474 (2017) 180–212

Contents lists available at ScienceDirect

Journal of Algebra

www.elsevier.com/locate/jalgebra

Proper divisibility in computable rings ☆

Noam Greenberg, Alexander Melnikov *

ARTICLE INFO

Article history:
Received 7 August 2015
Available online 17 November 2016
Communicated by Michel Broué

Keywords:
Constructive (effective) algebra
Reverse mathematics
Integral domain theory

ABSTRACT

We study divisibility in computable integral domains. We develop a technique for coding Σ^0_2 binary trees into the divisibility relation of a computable integral domain. We then use this technique to prove two theorems about non-atomic integral domains.

In every atomic integral domain, the divisibility relation is well-founded. We show that this classical theorem is equivalent to $\mathtt{ACA}_0$ over $\mathtt{RCA}_0$.

In every computable non-atomic integral domain there is a Δ^0_3 infinite sequence of proper divisions. We show that this upper bound cannot be improved to Δ^0_2 in general.

1. Introduction

Throughout most of the 19th century algebra was algorithmic; existence proofs were given by explicit constructions. Kronecker's elimination theory stands in contrast with the later abstract development led by Dedekind, Kummer and Hilbert; however all of

☆ Greenberg was supported by a Rutherford Discovery Fellowship, by the Marsden Fund, and by a Turing Research Fellowship "Mind, Mechanism and Mathematics" from the Templeton Foundation. We thank Damir Dzhafarov and Dan Turetsky for some useful conversations. We thank Valentin Bura, whose Master's thesis [4] contains an outline of Theorem 1.1, for allowing us to include material from his thesis. Valentin Bura declined co-authorship of this paper. We also thank the anonymous referee for many useful comments and corrections.

* Corresponding author.
E-mail address: alexander.g.melnuikov@gmail.com (A. Melnikov).

http://dx.doi.org/10.1016/j.jalgebra.2016.11.010
0021-8693/

mathematics up to that point was constructive [22]. *Computable algebra* aims to unearth the effective content of algebraic objects and constructions, revealing that aspect of mathematics which is lost when using the axiomatic, set-theoretic approach.

There has been much work considering fields. Early work concerning splitting algorithms in fields (Herrmann [17], van der Waerden [32]) was later made precise (for example Fröhlich and Shepherdson [14]) using the tools of computability theory, developed by Gödel, Church, Turing and Kleene (e.g., [30,31]). These tools allow us to rigorously define, for example, what a computable field is, and what operations on fields are computable. For example, Rabin [23] showed that every computable field can be computably embedded into a computable algebraically closed field; however the image of the embedding of the field into its algebraic closure may not always be computable, indeed to identify it sometimes extra computational power is required, such as the halting problem. Rabin's construction is not identical to Steinitz's original construction of an algebraic closure [28]; in the absence of a splitting algorithm, an alternative approach is necessary, presenting the algebraic closure as the quotient of a polynomial ring by a computable ideal.

Computability theory allows us not only to differentiate the computable from that which is not, but also compare different noncomputable objects, using *relative computability*. This captures the intuitive concepts of relative complexity, or information content: what it means for one object to be more complicated than another, or one problem to be easier to solve than another. For example, Friedman, Simpson and Smith showed [13] that in general, constructing a maximal ideal in a commutative ring is more complicated than constructing a prime ideal, but neither can be done computably. A yardstick for measuring complexity is given by iterations of Turing's jump operator, defined by taking the relative halting problem. For example, the full power of the halting problem is required to construct maximal ideals in all rings, but is not necessary for building prime ideals.

There is a fundamental connection between computability and foundational questions formalised in second order arithmetic. The project of reverse mathematics attempts to pin-point the proof-theoretic power of mathematical facts and theorems. It finds the number-theoretic axioms required to prove these theorems. Those axioms are often formalised as set existence axioms. Examples for the connection are the system $\mathtt{RCA}_0$ of "recursive comprehension", which in terms of set existence corresponds to relative computability; and the system $\mathtt{ACA}_0$ of "arithmetic comprehension", which corresponds to the Turing jump. Using this correspondence, the effective results from [13] mentioned above yield a proof that the statement "every ring has a maximal ideal" is equivalent to $\mathtt{ACA}_0$, but that the statement "every ring has a prime ideal" is strictly weaker. We remark though that computable algebra and reverse mathematics are complementary approaches for measuring the complexity of objects and theorems. Unlike reverse mathematics, computable algebra does not give any information about the amount of induction required to prove a theorem. On the other hand, in terms of set existence, computable