Division safe calculation in totalised fields

Bergstra, J.A.; Tucker, J.V.

DOI
10.1007/s00224-007-9035-4
Publication date
2008
Document Version
Final published version
Published in
Theory of Computing Systems

Theory Comput Syst (2008) 43: 410–424
DOI 10.1007/s00224-007-9035-4

Division Safe Calculation in Totalised Fields

J.A. Bergstra · J.V. Tucker

Published online: 26 July 2007

Abstract A 0-totalised field is a field in which division is a total operation with $0^{-1} = 0$. Equational reasoning in such fields is greatly simplified but in deriving a term one still wishes to know whether or not the calculation has invoked 0^{-1}. If it has not then we call the derivation *division safe*. We propose three methods of guaranteeing division safe calculations in 0-totalised fields.

Keywords Rational number · Meadow · Zero totalised field · Elementary algebraic specification

1 Introduction

The primary algebraic properties of the rational, real and complex numbers are captured by the operations and axioms of *fields*. The field axioms consist of the equations that define commutative rings and, in particular, two axioms, which are *not* equations, that define the inverse operator and the distinctness of the two constants. Traditionally, fields are *partial* algebras because the inverse operations are undefined at 0. The class of fields does *not* possess an equational axiomatisation.

However fields, especially the field of rational numbers and finite fields, are among the most important data types for computation. Rationals define measurements in the physical world and computer real arithmetic is based on a finite subset of the rational numbers. Computer integer arithmetic is based on finite rings and fields. All these fields are computable fields.

J.A. Bergstra (✉)
Informatics Institute, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
e-mail: j.a.bergstra@uva.nl

J.V. Tucker
Department of Computer Science, Swansea University, Singleton Park, Swansea, SA2 8PP, UK
e-mail: j.v.tucker@swansea.ac.uk

In [1, 7, 8], we have begun to investigate the field of rationals, and fields in general, using the elementary methods of abstract data type theory, especially equations, initial algebras and term rewriting. Calculations in fields are commonplace and the aim is to simplify algebraic reasoning and term rewriting for fields by removing the complications of partial functions and non-equational axioms.

A 0-*totalised field* is a field which has its inverse operator made total by imposing the equation

$$0^{-1} = 0.$$

If F is a field we denote the 0-totalised field by F_0; so for the fields $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$ of rational, real and complex numbers the 0-totalised fields are denoted $\mathbb{Q}_0$, $\mathbb{R}_0$ and $\mathbb{C}_0$, respectively.

Interestingly, the study of 0-totalised fields leads to new axioms and structures. For example, an new equational theory called "elementary number algebra" (*ENA*) has been identified in [8] (there built from three sets of equations and denoted $CR \cup SIP \cup Ril$) as a single sorted finite equational specification for the operations $+, -, \cdot, ^{-1}$ which has all 0-totalised fields among its models and, in addition, a large class of commutative rings with inverses and 0-divisors. A model of *ENA* has been baptized a *meadow* in [8] and a theory of meadows is emerging.

Equational specification, term rewriting and reasoning for totalised fields is *much* simpler than for fields with partial division. But in calculations one still wishes to know whether or not one has invoked 0^{-1}. Consider the derivation of a term:

$$\frac{1+1}{1+(-1)} + 1 = \frac{1+1}{0} + 1 = (1+1)\cdot 0^{-1} + 1 = (1+1)\cdot 0 + 1 = 1$$

in any 0-totalised field. The algebraic manipulation is simple but allows 0 in denominators and, moreover, makes use of the equation $0^{-1} = 0$. It is important to note that the outcome of the calculation is the valid term 1 and it is impossible to see from the outcome of the calculation alone that the derivation of the term involved these steps. We may wish to consider the derivation unsafe or exceptional in some way. Conversely, we will call a calculation *division safe* if it does not involve 0^{-1}.

The question to be discussed in this paper is this:

How do we formalise division safety for totalised fields? How do we detect and avoid unsafe divisions in calculations in 0*-totalised fields?*

We propose three methods of guaranteeing division safe calculations in 0-totalised fields, as follows:

1. *Proof system:* Once a proof of $t = r$ has been found, prove additional information that implies that $t = r$ was derived in a division safe way.
2. *Axioms:* Change the axioms of *ENA* to a weaker set that do not permit any division unsafe derivations.
3. *Algebra:* Modify a field to create a new algebra that satisfies all equations with division safe proofs but fails to satisfy other equations.

Each of these methods has merit and works for fields in general. The key idea is this: over the signature of fields, for each term t we can construct a new *check term*

C_t such that

$$C_t = 1 \Longleftrightarrow \text{“}t \text{ can be evaluated in a division safe way”}.$$

The origin of our work is found in two sources: a contemplation of recent work by Larry Moss and the objective to proceed with previous works on the algebraic specification of computable and semi-computable data types (in particular Bergstra and Tucker [2–5]) in the context of data types relevant for the theory of computation over the real numbers.

Recently Moss found in [16] that there exists an equational specification of the ring of rationals (i.e., without division or inverse) with just *one* unary hidden function. He used a remarkable enumeration theorem for the rationals in Calkin and Wilf [9]. He also gave specifications of other rational arithmetics and asked if hidden functions were necessary.

In [8] we proved that there exists a finite equational specification under initial algebra semantics, *without* further hidden functions, but making use of an inverse operation, of the field of rational numbers. The existence of an equational specification using hidden functions follows from a result in [2], plus the observation that the rational number field is a computable algebra. The issue is to limit the use of hidden functions to useful and familiar operations. The fact that only a single hidden function is used depends upon special properties of the field of rational numbers. In [7] the specification found for the rational numbers was extended to the complex rationals with conjugation, and in [1] a specification was given of the algebra of rational functions with field and degree operations that are all total.

2 Elementary Algebraic Specifications

2.1 Elementary Algebraic Specifications and Totality

The theory of computable data types demonstrates that any computable system can be modelled using a finite set of equations or conditional equations under initial algebra semantics, possibly with the help of auxiliary or hidden functions.

In [7] we have discussed a very limited specification technique which we have termed *elementary algebraic specification* (EAS). In fact EAS limits the expressive power of specifications to the original minimum of features that were used when algebraic specification of abstract data types was developed as a topic in the 1970s. In EAS, each algebraic specification (Σ', E') of a *total* Σ algebra uses a set E' of equations, or conditional equations, and initial algebra semantics such that $I(\Sigma', E')|_\Sigma \cong A$. In particular, the elementary specifications *require* total functions, *allow* hidden functions and sorts, and may or may not be complete term rewriting systems. Clearly, there are plenty of restrictions in force in EAS as there are many properties ruled out—see [7] for a long list with arguments for their omission. The *EAS specification problem* is this: Given a Σ algebra A, can one find an elementary algebraic specification (Σ', E') such that $I(\Sigma', E')|_\Sigma \cong A$.

An EAS is ‘better’ if it is finite rather than infinite, contains equations rather than conditional equations, or features nice term rewriting properties such as confluence and termination.

To use these EAS methods, we need to make algebras total that are usually considered to contain partial operators. Unavoidably, totalisation introduces an element of arbitrariness or artificiality because values are added which are not based on the primary intuitions at hand.

Totalisation is not without problems when specifying a stack, as we have seen in our [6]. Totalisation is a matter of costs and benefits and in some cases the theory of a totalised data type, even when specified by means of a convincing EAS, may be harder to swallow than some of its non-elementary expositions, even including the required meta-theory for those non-elementary features. Stacks are a candidate of such a data type.

However, in the case of fields we have found totalisation and EAS methods convincing. For that we have four arguments:

(1) The EAS specification theory of totalised fields is rich and attractive.
(2) Totalisation of fields leads to a specification *ENA* which itself has a larger class of models, consisting of the so-called meadows and having remarkably natural properties.
(3) EAS provides a decoupling of syntax and semantics that is fundamental. All simple answers to the question why 0^{-1} fails to exist depend on the observation that this piece of syntax should not have been written down in the first place because it carries no intended meaning. Exactly this interplay between syntax and semantics is completely removed in the setting of EAS and totalised fields.
(4) The costs of totalisation, due to the introduction of a "fake" value for 0^{-1} and its impact on the theory of numbers are already compensated by the gains mentioned in (1) and (3) above.

2.2 Technical Preliminaries on Algebraic Specifications

We assume the reader is familiar with using equations and conditional equations and initial algebra semantics to specify data types. Some accounts of this are: ADJ [10], Kamin [13], Meseguer and Goguen [15], or Wirsing [21].

The theory of algebraic specifications is based on theories of universal algebras (e.g., Wechler [20], Meinke and Tucker [14]), computable algebras (Stoltenberg-Hansen and Tucker [17]), and term rewriting (Terese [19]). The theory of computable fields is surveyed in Stoltenberg-Hansen and Tucker [18].

We use standard notations: typically, we let Σ be a many sorted signature and A a total Σ algebra. The class of all total Σ algebras is $Alg(\Sigma)$ and the class of all total Σ-algebras satisfying all the axioms in a theory T is $Alg(\Sigma, T)$. The word 'algebra' will mean total algebra.

3 Axioms for Number Algebras

The primary signature Σ is simply that of the *field*:

signature Σ
sorts *field*

operations
$0: \to \mathit{field}$;
$1: \to \mathit{field}$;
$+: \mathit{field} \times \mathit{field} \to \mathit{field}$;
$-: \mathit{field} \to \mathit{field}$;
$\cdot: \mathit{field} \times \mathit{field} \to \mathit{field}$;
$^{-1}: \mathit{field} \to \mathit{field}$
end

3.1 Commutative Rings and Fields

The signature Σ_{CR} consists of Σ minus the inverse operator $^{-1}$. The first set of axioms is that of a *commutative ring with* 1, which establishes the standard properties of $+$, $-$, and $\cdot$.

equations *CR*

$$(x + y) + z = x + (y + z)$$
$$x + y = y + x$$
$$x + 0 = x$$
$$x + (-x) = 0$$
$$(x \cdot y) \cdot z = x \cdot (y \cdot z)$$
$$x \cdot y = y \cdot x$$
$$x \cdot 1 = x$$
$$x \cdot (y + z) = x \cdot y + x \cdot z$$

end

These axioms generate a wealth of properties of $+$, $-$, $\cdot$ with which we will assume the reader is familiar.

At this point there are different ways to proceed with the introduction of division. The orthodoxy is to add the following two axioms for fields: let *Gil* (general inverse law) denote the axiom

$$x \neq 0 \Longrightarrow x \cdot x^{-1} = 1$$

and let *Sep* (the axiom of separation) denote

$$0 \neq 1.$$

Let $(\Sigma, T_{\text{field}})$ be the axiomatic specification of fields, where

$$T_{\text{field}} = CR \cup Gil \cup Sep.$$

3.2 Totalised Fields

In field theory the value of 0^{-1} is left undefined. However, in working with elementary specifications, operations are total. Thus, the class $Alg(\Sigma, T_{\text{field}})$ is the class of all possible *total* algebras satisfying the axioms in T_{field}; we refer to these algebras as *totalised fields*.

Now, for all totalised fields $A \in Alg(\Sigma, T_{\text{field}})$ and all $x \in A$, the inverse x^{-1} is defined. If 0_A is the zero element in A then, in particular, 0_A^{-1} is defined. The actual value 0_A^{-1} can be anything but it is convenient to set $0_A^{-1} = 0_A$ (see [8], and compare, e.g., Hodges [12], p. 695). A field A with $0_A^{-1} = 0_A$ is called 0-*totalised*. This choice gives us a nice equational specification to use, the zero inverse law *Zil*:

$$0^{-1} = 0.$$

With *ZTF* we specify zero totalised fields:

$$ZTF = CR \cup Gil \cup Sep \cup Zil.$$

Let $Alg(\Sigma, ZTF)$ be the class of all 0-totalised fields. One of the main Σ-algebras we are interested in is

$$\mathbb{Q}_0 = (\mathbb{Q}|0, 1, +, -, \cdot, ^{-1}) \in Alg(\Sigma, ZTF),$$

where the inverse is total $x^{-1} = 1/x$ if $x \neq 0$ and 0 if $x = 0$.

Following [8] one may replace the axioms *Gil* and *Sep* by other axioms for division, especially, the three equations in an unit called *SIP* for *strong inverse properties*. They are considered "strong" because they are equations involving $^{-1}$ *without any guards*, such as $x \neq 0$:

equations *SIP*

$$(-x)^{-1} = -(x^{-1})$$
$$(x \cdot y)^{-1} = x^{-1} \cdot y^{-1}$$
$$(x^{-1})^{-1} = x$$

end

In [8] we find that the following equations are provable:

Lemma 3.1 *$CR \cup SIP \vdash 0^{-1} = 0$ and $CR \cup SIP \vdash 0 \cdot x = 0$. Thus, $CR \cup SIP \vdash 0 \cdot 0^{-1} = 0$.*

In dealing with division it is helpful to have functions such as

$$Z(x) = 1 - x \cdot x^{-1} \quad \text{and} \quad N(x) = x \cdot x^{-1}.$$

Clearly, $Z(x) = 1 - N(x)$ and $Z(x) = 0 \Leftrightarrow x \cdot x^{-1} = 1$.

In [8] (Theorem 3.5) an axiom L, based on Lagrange's Theorem, is used to give an equational specification of the rationals. Lagrange's Theorem states that every natural number can be represented as the sum of four squares. We define a special equation L (for Lagrange):

$$Z(1+x^2+y^2+z^2+u^2)=0.$$

L expresses that for a large collection of numbers, in particular those q which can be written as 1 plus the sum of four squares, $q \cdot q^{-1}$ equals 1.

Theorem 3.2 *There exists a finite elementary equational specification* $(\Sigma, CR \cup SIP \cup L)$, *without hidden functions, of* $\mathbb{Q}_0$ *under initial algebra semantics.*

3.3 ENA and Meadows

In [8] we also add to $CR \cup SIP$ the *restricted inverse law* (*Ril*):

$$x \cdot (x \cdot x^{-1}) = x$$

which, using commutativity and associativity, expresses that $x \cdot x^{-1}$ is 1 in the presence of x.

Definition 3.3 We define the specification elementary number algebra $ENA = CR \cup SIP \cup Ril$.

Following [8] a model of *ENA* is called a *meadow*. A meadow satisfying *Sep* is called *non-trivial*. We note the following immediate consequences of *Ril*:

Lemma 3.4 $Ril \vdash x \cdot x^{-1} = 0 \Longleftrightarrow x = 0$ *and* $Ril \vdash x \cdot x = 0 \Longleftrightarrow x = 0$.

All total fields are clearly non-trivial meadows but not conversely. In particular, the prime fields $\mathbb{Z}_p$ of prime characteristic are meadows. That the initial algebra of $CR \cup SIP \cup Ril$ is not a field follows from the fact that $(1+1) \cdot (1+1)^{-1} = 1$ cannot be derivable because it fails to hold in the prime field $\mathbb{Z}_2$ of characteristic 2 which is a model of these equations as well.

Yoram Hirschfeld [11] has noticed that equations SIP1 and SIP2 are derivable from SIP3 using $CR \cup Ril$.

4 Equational Proof Systems for Safe Division

First, we will introduce the technical idea of the check term and how it can be used to define division safety. Then we will give a simple proof system for verifying division safety.

4.1 Check Terms and Division Safety

Let Σ be the signature of fields and $T(\Sigma, X)$ be the algebra of all Σ-terms with variables from X.

Definition 4.1
To each closed term t over Σ we assign a check term C_t as follows:
$C_0 = 1$
$C_1 = 1$
$C_{t_1+t_2} = C_{t_1} \cdot C_{t_2}$
$C_{-t} = C_t$
$C_{t_1 \cdot t_2} = C_{t_1} \cdot C_{t_2}$
$C_{t^{-1}} = C_t \cdot t \cdot t^{-1}$

If we extend the idea from closed terms to open terms $t(x_1, \ldots, x_n)$ then we would like the check term $C_t(x_1, \ldots, x_n)$ to have the same variables as t. One way to do this is to add variable x to the base case in Definition 4.1 and define $C_x = 1 + 0 \cdot x$.

The idea of the construction of our check terms is that for a closed term t:

$$C_t = 1 \text{ in } F_0 \Longleftrightarrow \text{“inside-out evaluation of } t \text{ in } F \text{ can be done in a division safe way”.}$$

Consider some examples of check terms. In a non-safe derivation we can expect to see the term 0^{-1} and this is certified by the check terms as $C_{0^{-1}} = C_0 \cdot 0 \cdot 0^{-1} = 1 \cdot 0 \cdot 0^{-1} = 0$.

The value of the closed check term $C_{0^{-1}}$ is 0 in all 0-totalised fields. Here is a calculation of a check term with variables: $C_{(x+y)/(z+1)} = C_{x+y} \cdot C_{1/(z+1)} = C_x \cdot C_y \cdot C_{z+1} \cdot (z+1)/(z+1) = (1 + 0 \cdot x) \cdot (1 + 0 \cdot y) \cdot C_z \cdot C_1 \cdot (z+1)/(z+1) = 1 \cdot 1 \cdot C_z \cdot C_1 \cdot (z+1)/(z+1) = (1+0 \cdot z) \cdot 1 \cdot (z+1)/(z+1) = (1+0) \cdot (z+1)/(z+1) = (z+1)/(z+1)$.

The value depends upon the value of z; in particular if $z = -1$ this value will be 0. In a field division is partial and for that reason many terms are undefined. The suitability of the check terms is confirmed by the following theorem:

Theorem 4.2 *Let F be a field and F_0 be its 0-totalised form of signature Σ. Then, for any closed term $t \in T(\Sigma)$, t is defined in $F \Longleftrightarrow C_t = 1$ in F_0.*

Proof By induction on the structure of closed terms. □

The purpose of the check term is to define forms of division safety, the first of which is this:

Definition 4.3 Let F_0 be a 0-totalised field. A closed equation $t = r$ is said to be division safe in F_0 if

(i) the equation is valid in F_0, i.e., $F_0 \models t = r$;
(ii) the check terms are safe in F_0, i.e., $F_0 \models C_t = 1$ and $F_0 \models C_r = 1$.

We write $F_0 \models_{ds} t = r$ if the equation is division safe. Thus:

$$F_0 \models_{ds} t = r \iff F_0 \models t = r \text{ and } F_0 \models C_t = 1 \wedge C_r = 1.$$

As with the check terms, division safety depends on the field. For example, the closed equation

$$\frac{1}{1+1} = \frac{1}{1+1}$$

is division safe in the field $\mathbb{Q}$ rational numbers but not division safe in the finite field $\mathbb{Z}_2$.

Definition 4.4 Let F_0 be a 0-totalised field. An open equation $t = r$ is said to be division safe in F_0 if every closed instance of the equation is division safe in F_0 according to Definition 5.2.

The open equation $x = x$ is not division safe as one substitution instance is $0^{-1} = 0^{-1}$, which is not division safe as $C_{0^{-1}} = 0$. Shortly, in Sect. 5, we will give a second, weaker form of division safety that is more plausible in this respect.

4.2 Equational Proof Systems

The proof system method to ensure division safety in a 0-totalised field F_0 is this: seek a set T of axioms and proof rules with relation $\vdash$ for F_0, i.e., $F_0 \in Alg(\Sigma, T)$, such that each proof $T \vdash t = r$ can be complemented by proofs that $T \vdash C_t = 1$ and $T \vdash C_r = 1$. Such a proof system for division safety will have the form:

Definition 4.5 Let t and r be closed terms over Σ. We write $(\Sigma, T) \vdash_{ds} t = r$ if

$$(\Sigma, T) \vdash t = r \quad \text{and} \quad (\Sigma, T) \vdash C_t = 1 \wedge C_r = 1.$$

Interestingly, we do not have far to look for one example: consider initial algebra specifications.

Theorem 4.6 *Let F_0 be any totalised field and (Σ, E) any equational specification such that $I(\Sigma, E) \cong F_0$. Then for any closed terms t, r we have*

$$(\Sigma, E) \vdash_{ds} t = r \iff F_0 \models_{ds} t = r.$$

Proof By initiality, equational reasoning is complete for closed identities relative to initial algebra specifications. By hypotheses, $I(\Sigma, E) \cong F_0$. In particular, for any *closed terms* t, r, $F_0 \models C_t = 1$ and $F_0 \models C_r = 1$ if, and only if, $(\Sigma, E) \vdash C_t = 1$ and $(\Sigma, E) \vdash C_r = 1$. □

Proving $(\Sigma, E) \vdash_{ds} t = r$ is a general approach to ensuring division safety; its practicality is dependent on the specification. Notice, in Theorem 5.4, that we have

no requirement on the equations in E to be division safe. Indeed we work with specifications containing equations that may be division unsafe; for example, the additive identity equation $x + 0 = x$ is not division safe. Now we will consider an approach that considers the safety of the specifications.

5 Equational Axioms for Weak Safe Division

5.1 Weak Safe Division in 0-Totalised Fields

We now consider a weaker notion of safety that has some interesting properties.

Definition 5.1 Let F_0 be a 0-totalised field. A closed equation $t = r$ is said to be weakly division safe in F_0 if

(i) the equation is valid in F_0, i.e., $F_0 \models t = r$;
(ii) the check terms are equal in F_0, i.e., $F_0 \models C_t = C_r$.

We write $F_0 \models_{wds} t = r$ if the equation is weakly division safe. Thus:

$$F_0 \models_{wds} t = r \iff F_0 \models t = r \text{ and } F_0 \models C_t = C_r.$$

Definition 5.2 Let F_0 be a 0-totalised field. An open equation $t = r$ is said to be weakly division safe in F_0 if every closed instance of the equation is weakly division safe in F_0 according to Definition 5.1.

Clearly, the idea of a weakly division safe equation is that either both sides of the equation are safe or unsafe. Compare the notion with division safety (in Definition 5.2). There are closed and open equations, such as $0^{-1} = 0^{-1}$ and $x = x$, that are weakly division safe but not necessarily division safe. Using equational specifications again:

Definition 5.3 We write $(\Sigma, T) \vdash_{wds} t = r$ if

$$(\Sigma, T) \vdash t = r \quad \text{and} \quad (\Sigma, T) \vdash C_t = C_r.$$

Again, by the completeness of initial algebra semantics for closed equations, we have:

Theorem 5.4 *Let F_0 be any totalised field and (Σ, E) any equational specification such that $I(\Sigma, E) \cong F_0$. Then for any closed terms t, r we have*

$$(\Sigma, E) \vdash_{wds} t = r \iff F_0 \models_{wds} t = r.$$

For many equations $t = r$ where r is the simplified or "calculated" result or normal form of t it will be obvious by inspection that $F_0 \models C_r = 1$. In this case we have:

Lemma 5.5 *Suppose that $F_0 \models C_r = 1$. Then $\vdash_{wds} t = r$ implies $\vdash_{ds} t = r$.*

Finally, we have this preservation property:

Theorem 5.6 *Let F_0 be a 0-totalised field and (Σ, E) be any specification true of F_0, i.e., $F_0 \models E$. Suppose every equation in E is weakly division safe for F_0. For every equation $t = r$ such that $(\Sigma, E) \vdash t = r$ then $t = r$ is weakly division safe.*

Proof By induction on the length of proofs made from closed instances of the equations. □

5.2 Meadows and the Rationals

In the case of meadows and the rationals, we are able to weaken the axioms *ENA* and *L* we have used in such a way that

(i) all closed division safe identities are provable; and
(ii) only weakly division safe open identities are provable.

In the light of Theorem 5.6, we start by checking the equations of our usual specification *ENA*. The following are the equations that are *not* weakly division safe.

(a) Additive Inverse: $x + (-x) = 0$ because it has $0^{-1} + (-0^{-1}) = 0$ as a substitution instance.
(b) $(x^{-1})^{-1} = x$ because it has $(0^{-1})^{-1} = 0$ as a substitution instance.
(c) *Ril*: $x \cdot x \cdot x^{-1} = x$ because it has $0 \cdot 0 \cdot 0^{-1} = 0$ as a substitution instance.

It is possible to replace each of these equations in *ENA* by weakly division safe alternates as follows:

In the set *CR* of commutative rings axioms we replace additive inverse by these three equations

$$x + (-x) = 0 \cdot x,$$
$$0 \cdot 0 = 0,$$
$$0 \cdot 1 = 0.$$

In the set *SIP* of inverse axioms the axiom $(x^{-1})^{-1} = x$ is replaced by:

$$(x^{-1})^{-1} = x \cdot x \cdot x^{-1}.$$

The axiom *Ril* is replaced by

$$x^{-1} \cdot x^{-1} \cdot x = x^{-1}.$$

Let *ENA*′ be the new set of axioms. Then we have:

Lemma 5.7 *For any 0-totalised field F_0 we have $F_0 \models ENA'$ and since ENA′ are weakly division safe all the equational consequences of ENA′ are division safe.*

Furthermore, in the special case of $\mathbb{Q}_0$ more can be shown. First, the Lagrange equation

$$L : Z(1 + x^2 + y^2 + z^2 + u^2) = 0$$

is not weakly division-safe as may be seen on substituting 0^{-1} for the variables x, y, z, u. But, the Lagrange axiom L can replaced by

$$Z(1 + x^2 + y^2 + z^2 + u^2) = 0 \cdot (x + y + z + u)$$

which is weakly division safe.

Lemma 5.8 *For any closed terms t, r*

$$\mathbb{Q}_0 \models_{ds} t = r \quad \textit{implies} \quad \textit{ENA}' \cup L' \vdash t = r.$$

Proof The proof is derived from the proof that $\mathbb{Q}_0 \cong I(\Sigma, \textit{ENA} \cup L)$ from Bergstra and Tucker [8]. The proof of weak division safe identities between closed terms does not depend on non-division safe identities. □

Thus, the axioms of $\textit{ENA}' \cup L'$ is a reasonable specification of $\mathbb{Q}_0$ since it is a complete proof system for division safe ground identities, and proves only weakly division safe identities as well, though not all weakly division safe identities.

6 Algebras for Safe Division

The third approach seeks a form of error algebra for fields, which are no longer 0-totalised fields. These specific error algebras are called *twin fields* in spite of the fact that they are strictly speaking not fields. (Similarly non-commutative skew fields cannot be fields either.) Then the idea is that $\textit{ENA}'$ and $\textit{ENA}' \cup L'$ might be part of specifications for such algebras.

Given a field F of signature Σ we define a new Σ algebra F_{twin} such that for closed t and r:

$$F_{\text{twin}} \models t = r \Longleftrightarrow F_0 \vdash_{wds} t = r.$$

For each element $a \in F$ we make a copy $\hat{a} \in F_{\text{twin}}$ which represents the same value but in a division unsafe form. We may write $\hat{a} = a + 0^{-1}$. In a 0-totalised field we have $\hat{a} = a$, of course.

Twin fields are defined as follows. Let F be a field. Let F_0 be the 0-totalised form of F. Let $B = \{t, f\}$ be the Booleans.

Definition 6.1 The twin field extension of F is defined to be a Σ algebra with carrier $B \times F$; the constants 0, 1 are

$$(t, 0_F) \quad \text{and} \quad (t, 1_F).$$

The operations are

$(b, x) +_{F_{\text{twin}}} (c, y) = (b \wedge c, x +_F y)$,
$(b, x) \cdot_{F_{\text{twin}}} (c, y) = (b \vee c, x \cdot_F y)$,
$(b, 0)^{-1} = (f, 0)$,
$(b, x)^{-1} = (b, y)$ where $x \neq_F 0$ and $x \cdot y =_F 1$.

Thus, F_{twin} contains an isomorphic copy of F, namely $\{t\} \times F$ and an isomorphic copy of F_0, namely $\{f\} \times F$. The inverse on the copy of F is made by: $(t, 0)^{-1} = (f, 0)$. Once an element lands in the error part of the twin field the operations keep it there. Notice that a twin field is not a field because

$$0 \cdot 0^{-1} \neq 0 \quad \text{and so} \quad 0 \cdot x = 0 \quad \text{fails in } F_{\text{twin}}.$$

Lemma 6.2 *Let F be a field, F_0 be its 0-totalised form and F_{twin} its twin field. For any terms t, r, if $F_{\text{twin}} \models t = r$ then $F \models t = r$ and the equation is weakly division safe in F_0.*

Given this definition of F_{twin} we give a set of equations that can play a role similar to *ENA*:

$$ENA_{\text{twin}} = ENA' \cup \{0^{-1} \cdot x = 0^{-1}, (0^{-1} + x)^{-1} = 0^{-1} + x^{-1}, 0 \cdot x + 0^{-1} = 0^{-1}\}.$$

Using a proof similar to that of Theorem 3.2 in Bergstra and Tucker [8] we have:

Theorem 6.3 $\mathbb{Q}_{\text{twin}} \cong I(\Sigma, ENA_{\text{twin}} \cup L')$.

7 Concluding Remarks

Our work on the rationals and other fields can be viewed as a case study in abstract data types. "Number algebra" specifications are to be compared with "process algebra" specifications: they are elementary algebraic specifications designed to capture mechanisms found in the theory of computers and computation.

In this number algebra one takes the liberty to depart from the algebraist's orthodoxy (fields with their partial operations) and adapt the design of algebras of numbers to meet the requirements of the computational modeling technique used, namely elementary algebraic specifications (EAS). Thus, one can view this topic as a theory of arithmetics, including fields, shaped according to one of many general modelling techniques that have been developed in computer science: algebraic specifications where equational reasoning is extremely important. Given its origins, the focus is on questions that one might pose from the computer science perspective: questions on specification, verification, prototyping, decidability and expressiveness.

However, the topic is also an attempt to answer the mathematical question: What *can* one accomplish with the rationals and other fields *using simple equational reasoning only*? The theory of meadows is not without interest in pure algebra.

Assuming that one wants to view fields as total algebras, two strategies are feasible. First, use 0-totalised fields which possess nice equational specifications but alone which provide no protection against weak division unsafe conclusions. In this case,

the use of additional proof obligations can protect against division unsafe results. An alternate is to use weaker equations.

Secondly, use dedicated error algebras customised to the setting of fields, such as twin fields. Each twin field contains a 0-totalised field as a substructure. Twin fields admit a specification theory similar to that of 0-totalised fields though require more complex equations. Twin fields guarantee that only weakly division safe conclusions are derived.

A check term is a term that tests a property by means of its value. The idea is independent of this division problem. The technique of designing check terms for a property and using the equational proof system for closed terms (based on the completeness of equational specifications and their initial algebra semantics) is general and may have other applications.

References

1. Bergstra, J.A.: Elementary algebraic specifications of the rational function field. In: Beckmann, A., et al. (eds.) Logical Approaches to Computational Barriers. Proceedings of Computability in Europe 2006. Lecture Notes in Computer Science, vol. 3988, pp. 40–54. Springer, New York (2006)
2. Bergstra, J.A., Tucker, J.V.: The completeness of the algebraic specification methods for data types. Inf. Control **54**, 186–200 (1982)
3. Bergstra, J.A., Tucker, J.V.: Initial and final algebra semantics for data type specifications: two characterisation theorems. SIAM J. Comput. **12**, 366–387 (1983)
4. Bergstra, J.A., Tucker, J.V.: Algebraic specifications of computable and semicomputable data types. Theor. Comput. Sci. **50**, 137–181 (1987)
5. Bergstra, J.A., Tucker, J.V.: Equational specifications, complete term rewriting systems, and computable and semicomputable algebras. J. ACM **42**, 1194–1230 (1995)
6. Bergstra, J.A., Tucker, J.V.: The data type variety of stack algebras. Ann. Pure Appl. Log. **73**, 11–36 (1995)
7. Bergstra, J.A., Tucker, J.V.: Elementary algebraic specifications of the rational complex numbers. In: Futatsugi, K., et al. (eds.) Algebra, Meaning and Computation. Goguen Festschrift. Lecture Notes in Computer Science, vol. 4060, pp. 459–475. Springer, New York (2006)
8. Bergstra, J.A., Tucker, J.V.: The rational numbers as an abstract data type. J. ACM **54**(2), Article 7 (April 2007), 25 pages
9. Calkin, N., Wilf, H.S.: Recounting the rationals. Am. Math. Mon. **107**, 360–363 (2000)
10. Goguen, J.A., Thatcher, J.W., Wagner, E.G.: An initial algebra approach to the specification, correctness and implementation of abstract data types. In: Yeh, R.T. (ed.) Current Trends in Programming Methodology. IV. Data Structuring, pp. 80–149. Prentice-Hall, Englewood Cliffs (1978)
11. Hirschfeld, Y.: Personal communication (August 2006)
12. Hodges, W.: Model Theory. Cambridge University Press, Cambridge (1993)
13. Kamin, S.: Some definitions for algebraic data type specifications. SIGLAN Not. **14**(3), 28 (1979)
14. Meinke, K., Tucker, J.V.: Universal algebra. In: Abramsky, S., Gabbay, D., Maibaum, T. (eds.) Handbook of Logic in Computer Science. Mathematical Structures, vol. I, pp. 189–411. Oxford University Press, Oxford (1992)
15. Meseguer, J., Goguen, J.A.: Initiality, induction, and computability. In: Nivat, M. (ed.) Algebraic Methods in Semantics, pp. 459–541. Cambridge University Press, Cambridge (1986)
16. Moss, L.: Simple equational specifications of rational arithmetic. Discret. Math. Theor. Comput. Sci. **4**, 291–300 (2001)
17. Stoltenberg-Hansen, V., Tucker, J.V.: Effective algebras. In: Abramsky, S., Gabbay, D., Maibaum, T. (eds.) Handbook of Logic in Computer Science. Semantic Modelling, vol. IV, pp. 357–526. Oxford University Press, Oxford (1995)
18. Stoltenberg-Hansen, V., Tucker, J.V.: Computable rings and fields. In: Griffor, E. (ed.) Handbook of Computability Theory, pp. 363–447. Elsevier, Amsterdam (1999)
19. Terese, K.: Term Rewriting Systems. Cambridge Tracts in Theoretical Computer Science, vol. 55. Cambridge University Press, Cambridge (2003)

20. Wechler, W.: Universal Algebra for Computer Scientists. EATCS Monographs in Computer Science. Springer, New York (1992)
21. Wirsing, M.: Algebraic specifications. In: van Leeuwen, J. (ed.) Handbook of Theoretical Computer Science. Formal Models and Semantics, vol. B, pp. 675–788. North-Holland, Amsterdam (1990)