Datatype defining rewrite systems for the ring of integers, and for natural and integer arithmetic in unary view

Bergstra, J.A.; Ponse, A.

Publication date
2016

Document Version
Final published version

Citation for published version (APA):
Bergstra, J. A., & Ponse, A. (2016). *Datatype defining rewrite systems for the ring of integers, and for natural and integer arithmetic in unary view*. (1 ed.) Informatics Institute, University of Amsterdam. https://arxiv.org/abs/1608.06212v1

Datatype defining rewrite systems for the ring of integers, and for natural and integer arithmetic in unary view

Jan A. Bergstra and Alban Ponse

Informatics Institute, section Theory of Computer Science
University of Amsterdam

https://staff.fnwi.uva.nl/j.a.bergstra/ https://staff.fnwi.uva.nl/a.ponse/

Abstract

A datatype defining rewrite system (DDRS) is a ground-complete term rewriting system, intended to be used for the specification of datatypes. As a follow-up of an earlier paper we define two concise DDRSes for the ring of integers, each comprising only twelve rewrite rules, and prove their ground-completeness. Then we introduce DDRSes for a concise specification of natural number arithmetic and integer arithmetic in unary view, that is, arithmetic based on unary append (a form of tallying) or on successor function. Finally, we relate one of the DDRSes for the ring of integers to the above-mentioned DDRSes for natural and integer arithmetic in unary view.

Keywords and phrases: Datatype defining rewrite system, Equational specification, Integer arithmetic, Natural number arithmetic

Contents

(1)	$x + (y + z) = (x + y) + z$	(5)	$(x \cdot y) \cdot z = x \cdot (y \cdot z)$
(2)	$x + y = y + x$	(6)	$x \cdot y = y \cdot x$
(3)	$x + 0 = x$	(7)	$1 \cdot x = x$
(4)	$x + (-x) = 0$	(8)	$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

Table 1: Axioms for commutative rings

1 Introduction

In Table 1 we recall the axioms for commutative rings over the signature $\Sigma_r = \{0, 1, -, +, \cdot\}$. Our point of departure is that these axioms characterize integer arithmetic, while leaving open how numbers are represented (apart from the constants $0, 1 \in \Sigma_r$). In this paper we introduce a few simple specifications for the concrete datatype determined by these axioms, and for natural number and integer arithmetic with numbers represented in unary view, that is, in a numeral system based on unary append (a form of tallying) or on successor function.

Given some signature, a datatype defining rewrite system (DDRS) consists of a number of equations over that signature that define a term rewriting system when interpreting the equations from left to right. A DDRS must be ground-complete, that is, strongly terminating and ground-confluent (for some general information on term rewriting systems see e.g. [5]). While equational specifications are used to determine abstract datatypes, a DDRS is meant to be used for the specification of a *concrete* datatype, which is a canonical term algebra: for each congruence class of closed terms, a unique representing term is chosen, and this set of representing closed terms — *canonical terms* or *normal forms* — is closed under taking subterms.

In [2], DDRSes for natural number arithmetic (defining addition and multiplication) and integer arithmetic (also defining the unary minus operator) were specified according to three views:

1. The *unary* view. In this view, number representation is based on the constant 0 and either the well-known successor function $S(x)$, or a unary append constructor (a form of tallying).

2. The *binary* view, in which number representation is based on constants for the two digits 0 and 1, and on two binary append constructors.

3. The *decimal* view, in which number representation is based on constants for the ten digits $0, 1, \ldots, 9$, and on ten binary append constructors.

Furthermore, each of these DDRSes contains equations for rewriting constructor terms in one of other views to a term in the DDRS'es view, e.g., the DDRSes for the unary view contain the equation $1 = S(0)$ for the constant 1 in binary view and in decimal view. The design of these DDRSes is geared towards obtaining comprehensible specifications of natural number and integer arithmetic in binary and decimal view. The successor function $S(x)$ and predecessor function $P(x)$ appeared to be instrumental auxiliary functions for this purpose, thereby justifying the incorporation of the unary view as a separate view.

This paper is a follow-up of [2]. In Section 2 we define two different DDRSes for the ring of integers that both contain only twelve equations, and prove their ground-completeness. In Section 3 we provide DDRSes for the unary view that are more concise than those in [2], and relate one of the DDRSes for the ring of integers to these DDRSes. We end the paper with some

[R1]	$x + 0 = x$	[R7]	$-0 = 0$
[R2]	$0 + x = x$	[R8]	$(-1) + 1 = 0$
[R3]	$x + (y + z) = (x + y) + z$	[R9]	$(-(x + 1)) + 1 = -x$
		[R10]	$-(-x) = x$
[R4]	$x \cdot 0 = 0$		
[R5]	$x \cdot 1 = x$	[R11]	$x + (-y) = -((-x) + y)$
[R6]	$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$	[R12]	$x \cdot (-y) = -(x \cdot y)$

Table 2: The DDRS D_1 for the datatype $\mathbb{Z}_r$ that specifies the ring integers

conclusions in Section 4. In [2] we wrote that we judge the DDRSes for the unary view to be "irrelevant as term rewriting systems from which an efficient implementation can be generated", and we now also draw some positive conclusions.

2 DDRSes for the ring of integers

In [1, 2] we defined a DDRS consisting of fifteen equations for the (concrete) datatype $\mathbb{Z}_r$ over the signature Σ_r of (commutative) rings. Normal forms are 0 for zero, the positive numerals 1 and $t + 1$ with t a positive numeral, and the negations of positive numerals, thus $-t$ for each positive numeral t.

In this section we consider two different DDRSes that define $\mathbb{Z}_r$ and both contain only twelve equations. In Section 2.1 we define a DDRS that is close to the one defined in [1, 2], and in Section 2.2 we consider an alternative DDRS that is deterministic with respect to rewriting a sum of two nonnegative normal forms.

2.1 A concise DDRS for $\mathbb{Z}_r$

The DDRS D_1 in Table 2 defines the datatype $\mathbb{Z}_r$. The difference between the DDRS D_1 in Table 2 and the DDRS for $\mathbb{Z}_r$ defined in [1, 2] is that equation [R11] replaces the four equations

[r5] $1 + (-1) = 0,$

[r6] $(x + 1) + (-1) = x,$

[r7] $x + (-(y + 1)) = (x + (-y)) + (-1),$

[r11] $(-x) + (-y) = -(x + y).$

In the remainder of this section we prove the ground-completeness of the DDRS D_1.

Lemma 2.1.1. *The DDRS D_1 for $\mathbb{Z}_r$ in Table 2 is strongly terminating.*

Proof. Define the following weight function $|x|$ on closed terms over Σ_r to $\mathbb{N}$:

$$|0| = 2, \qquad |-x| = |x| + 1,$$
$$|1| = 2, \qquad |x + y| = |x| + 3|y|,$$
$$|x \cdot y| = |x|^{|y|}.$$

[R1]	$x + 0 = x$	[R7]	$-0 = 0$
[R2′]	$0 + 1 = 1$	[R8]	$(-1) + 1 = 0$
[R3′]	$x + (y + 1) = (x + y) + 1$	[R9]	$(-(x + 1)) + 1 = -x$
		[R10]	$-(-x) = x$
[R4]	$x \cdot 0 = 0$		
[R5]	$x \cdot 1 = x$	[R11]	$x + (-y) = -((-x) + y)$
[R6′]	$x \cdot (y + 1) = (x \cdot y) + x$	[R12]	$x \cdot (-y) = -(x \cdot y)$

Table 3: The DDRS D_2 for the datatype $\mathbb{Z}_r$ that specifies the ring integers

Thus, $|t| \geq 2$ for each closed term t. For arbitrary closed terms u, r and s we find $3 \cdot |s| > |s| + 1$, thus

$$|u|^{3|s|} > |u|^{|s|+1} \geq 2 \cdot |u|^{|s|} > |u|^{|s|} + 1,$$

and thus $|u|^{3|s|} - 1 > |u|^{|s|}$. Also, $|u|^{|r|} > 3$, hence

$$|u|^{|r|} \cdot (|u|^{3|s|} - 1) > 3 \cdot |u|^{|s|}$$

and thus $|u|^{|r|} \cdot |u|^{3|s|} > |u|^{|r|} + 3 \cdot |u|^{|s|}$, and hence $|u \cdot (r + s)| > |(u \cdot r) + (u \cdot s)|$, which proves that application of equation [R6] reduces weight. Furthermore,

$$\begin{aligned} |u + (-r)| &= |u| + 3(|r| + 1) \\ &> (|u| + 1 + 3|r|) + 1 \\ &= |-((-u) + r)|, \end{aligned}$$

which proves that application of equation [R11] reduces weight. It easily follows that applications of all remaining equations on closed terms also imply weight reduction, which implies that the DDRS D_1 is strongly terminating. □

Theorem 2.1.2. *The DDRS D_1 for $\mathbb{Z}_r$ defined in Table 2 is ground-complete.*

Proof. See Corollary 2.2.2. □

2.2 An alternative DDRS for $\mathbb{Z}_r$

The DDRS D_1 can be "simplified" by instantiating and combining several equations. In Table 3 we provide the DDRS D_2 that also specifies the datatype $\mathbb{Z}_r$, where the differences with D_1 show up in the tags: equations [R2′], [R3′] and [R6′] replace [R1], [R2] and [R6], respectively.

Theorem 2.2.1. *The DDRS D_2 for $\mathbb{Z}_r$ defined in Table 3 is ground-complete.*

Proof. With the weight function defined in the proof of Lemma 2.1.1 it follows that D_2 is strongly terminating: equations [R2′] and [R3′] are instances of the associated D_1-equations [R2] and [R3], and equation [R6′] is the result of an instance of the associated equation [R6] and an application of [R5].

It remains to be proven that D_2 is ground-confluent. Define the set NF of closed terms over Σ_r as follows:

$$\begin{aligned} NF &= \{0\} \cup NF^+ \cup NF^-, \\ NF^+ &= \{1\} \cup \{t+1 \mid t \in NF^+\}, \\ NF^- &= \{-t \mid t \in NF^+\}. \end{aligned}$$

It immediately follows that if $t \in NF$, then t is a normal form (no rewrite step applies). Furthermore, two distinct elements in NF have distinct values in $\mathbb{Z}$. Clearly, the equations in Table 2 are semantic consequences of the axioms for commutative rings (Table 1). In order to prove ground-confluence of this DDRS it suffices to show that for each closed term over Σ_r, either $t \in NF$ or t has a rewrite step, so that each normal form is in NF.

We prove this by structural induction on t. The base cases $t \in \{0, 1\}$ are trivial. For the induction step we have to consider three cases:

1. Case $t = -r$. Assume that $r \in NF$ and apply case distinction on r:
   - if $r = 0$, then $t \to 0$ by equation [R7],
   - if $r \in NF^+$, then $t \in NF$,
   - if $r \in NF^-$, then t has a rewrite step by equation [R10].
2. Case $t = u + r$. Assume that $u, r \in NF$ and apply case distinction on r:
   - if $r = 0$, then $t \to u$ by equation [R1],
   - if $r = 1$, then apply case distinction on u:
     - if $u = 0$, then $t \to 1$ by equation [R2′],
     - if $u \in NF^+$, then $t \in NF$,
     - if $u = -1$, then $t \to 0$ by equation [R8],
     - if $u = -(u' + 1)$, then t has a rewrite step by equation [R9],
   - if $r = r' + 1$, then $t \to (u + r') + 1$ by equation [R3′],
   - if $r = -r'$ with $r' \in NF^+$, then $t \to -((-u) + r')$ by equation [R11].
3. Case $t = u \cdot r$. Assume that $u, r \in NF$ and apply case distinction on r:
   - if $r = 0$, then $t \to 0$ by equation [R4],
   - if $r = 1$, then $t \to u$ by equation [R5],
   - if $r = r' + 1$, then $t \to (u \cdot r') + u$ by equation [R6′],
   - if $r = -r'$ with $r' \in NF^+$, then $t \to -(u \cdot r')$ by equation [R12].

□

Corollary 2.2.2. *The DDRS D_1 for $\mathbb{Z}_r$ defined in Table 2 is ground-complete.*

Proof. The equations of the DDRS D_1 are semantic consequences of the axioms for commutative rings (Table 1). It suffices to consider the proof of Theorem 2.2.1 and to observe that each rewrite step by one of the equations [R2′], [R3′], and [R6′] implies a rewrite step of the associated equation in D_1. With Lemma 2.1.1 this proves that D_1 is ground-complete. □

A particular property of D_2 concerns the addition of two nonnegative normal forms.

Proposition 2.2.3. *With respect to addition of two nonnegative normal forms, the DDRS D_2 in Table 3 is* ***deterministic****. That is, for nonnegative normal forms t, t', in each state of rewriting of $t + t'$ to its normal form, only one equation (rewrite rule) applies.*

Proof. By structural induction on t'. For $t' \in \{0, 1\}$ this is immediately clear.

If $t' = r + 1$, then the only possible rewrite step is $t + (r + 1) \xrightarrow{[R3']} (t + r) + 1$. If $r = 1$, this is a normal form, and if $r = r' + 1$, the only redex in $(t + (r' + 1)) + 1$ is in $t + (r' + 1)$, and by induction the latter rewrites deterministically to some normal form u not equal to 0.

Consider this reduction: $t + (r' + 1) \to ... \to u$. Then $(t + (r' + 1)) + 1 \to ... \to u + 1$, and in each state of this reduction the rightmost addition $.. + 1$ does not establish a new redex according to equations [R1], [R2′] and [R3′], and results in $u + 1$, which is also a normal form. □

It is clear that for example $1 + t'$ with t' a negative normal form also rewrites deterministically in D_2, and that this determinism is not preserved for $t + t'$ if $t = 0$, e.g.,

$$0 + (-(r + 1)) \xrightarrow{[R11]} -((-0) + (r + 1)) \left[\begin{array}{l} \xrightarrow{[R3']} -(((-0) + r) + 1) \xrightarrow{[R7]} \\ \xrightarrow{[R7]} \;\; -(0 + (r + 1)) \quad\;\; \xrightarrow{[R3']} \end{array} \right] -((0 + r) + 1),$$

or if t is a negative normal form, e.g.,

$$(-t) + (-(r + 1)) \xrightarrow{[R11]} -((-(-t)) + (r + 1)) \left[\begin{array}{l} \xrightarrow{[R3']} -(((-(-t)) + r) + 1) \xrightarrow{[R10]} \\ \xrightarrow{[R10]} \;\; -(t + (r + 1)) \quad\;\; \xrightarrow{[R3']} \end{array} \right] -((t + r) + 1).$$

However, our interest in deterministic reductions concerns nonnegative normal forms and we return to this point in the next section. Finally, note that with respect to multiplication of two nonnegative normal forms, the DDRS D_2 is not deterministic:

$$0 \cdot (1 + 1) \xrightarrow{[R6']} (0 \cdot 1) + 0 \left[\begin{array}{l} \xrightarrow{[R4]} 0 + 0 \xrightarrow{[R1]} \\ \xrightarrow{[R1]} \; 0 \cdot 1 \; \xrightarrow{[R4]} \end{array} \right] 0.$$

3 DDRSes for natural number and integer arithmetic in unary view

In this section we consider a simple form of number representation that is related to tallying and establishes a unary numeral system based on the constant 0. The *unary append* is the one-place (postfix) function

$$\_ :_u 1 : \mathbb{Z} \to \mathbb{Z}$$

and is an alternative notation for the successor function $S(x)$. In [2, App.C] the unary append is used to define (concrete) datatypes for natural number arithmetic and integer arithmetic. The signature we work in is $\Sigma_U = \{0, -, \ \_ :_u 1, +, \cdot\}$, where $-$ represents the unary minus function.

For natural numbers, normal forms are 0 for zero, and applications of the unary append function that define all successor values: each natural number n is represented by n applications

[U1]	$x+0=x$	[U3]	$x\cdot 0=0$
[U2]	$x+(y:_u 1)=(x:_u 1)+y$	[U4]	$x\cdot(y:_u 1)=x+(x\cdot y)$

Table 4: The DDRS Nat_1 for $\mathbb{N}_U$, natural numbers in unary view with unary append function

of the unary append to 0 and can be seen as representing a sequence of 1's of length n having 0 as a single prefix, e.g.

$$(0:_u 1):_u 1$$

is the normal form that represents 2 and can be abbreviated as 011. We name the resulting datatype $\mathbb{N}_U$. For integers, each minus instance $-t$ of a nonzero normal form t in $\mathbb{N}_U$ is a normal form over Σ_U, e.g.

$$-((0:_u 1):_u 1)$$

is the normal form that represents -2 and can be abbreviated as -011. We name the resulting datatype $\mathbb{Z}_U$.

In Section 3.1 we introduce concise DDRSes based on Σ_U. In Section 3.2 we investigate in what way the DDRS D_2 for the ring of integers is related, and in Section 3.3 we briefly discuss the use of the successor function as an alternative for unary append.

3.1 Concise DDRSes for $\mathbb{N}_U$ and $\mathbb{Z}_U$

In Table 4 we define the DDRS Nat_1 for the datatype $\mathbb{N}_U$ over the signature $\Sigma_U \setminus \{-\}$. This is an alternative for the DDRS for natural number arithmetic in unary view defined in [2, App.C], in particular equation [U4] replaces the equation

[u′4] $\quad x\cdot(y:_u 1)=(x\cdot y)+x.$

Below we prove that Nat_1 is deterministic with respect to addition and multiplication of two normal forms.

The transition to DDRSes for $\mathbb{Z}_U$ can be taken in different ways. As an alternative to the equations for negative numbers in unary view provided in [2, App.C], we provide in Table 5 the DDRS Int_1 that defines a smaller extension of Nat_1 to integer numbers (thus, to the datatype $\mathbb{Z}_U$): equation [U8] replaces the three equations

[u′8] $\quad 0+x=x,$

[u′9] $\quad (x:_u 1)+(-(y:_u 1))=x+(-y),$

[u′10] $\quad (-x)+(-y)=-(x+y).$

The following example shows that with respect to addition of negative normal forms, the DDRS Int_1 is not deterministic:

$$(-011)+01 \xrightarrow{[\mathrm{U2}]} ((-011)1)+0 \begin{bmatrix} \xrightarrow{[\mathrm{U6}]} (-01)+0 \xrightarrow{[\mathrm{U1}]} \\ \xrightarrow{[\mathrm{U1}]} (-011)1 \quad \xrightarrow{[\mathrm{U6}]} \end{bmatrix} -01. \tag{9}$$

[U1]	$x + 0 = x$	[U5]	$-0 = 0$
[U2]	$x + (y :_u 1) = (x :_u 1) + y$	[U6]	$(-(x :_u 1)) :_u 1 = -x$
[U3]	$x \cdot 0 = 0$	[U7]	$-(-x) = x$
[U4]	$x \cdot (y :_u 1) = x + (x \cdot y)$	[U8]	$x + (-y) = -((-x) + y)$
		[U9]	$x \cdot (-y) = -(x \cdot y)$

Table 5: The DDRS Int_1 for $\mathbb{Z}_U$ that specifies integer arithmetic over Σ_U

Lemma 3.1.1. *The DDRSes Nat_1 for $\mathbb{N}_U$ (Table 4), and Int_1 for $\mathbb{Z}_U$ (Table 5) are strongly terminating.*

Proof. Modify the weight function $|x|$ defined in the proof of Lemma 2.1.1 to closed terms over Σ_U by deleting the clause $|1| = 2$ and adding the clause

$$|x :_u 1| = |x| + 2.$$

Then $|t| \geq 2$ for each closed term t, and thus $|t| < |t|^{|u|}$ for each closed term u, so

$$\begin{aligned} |t \cdot (u :_u 1)| &= |t|^{|u|+2} \\ &\geq 4 \cdot |t|^{|u|} \\ &= |t|^{|u|} + 3 \cdot |t|^{|u|} \\ &> |t| + 3 \cdot |t|^{|u|} \\ &= |t + (t \cdot u)|, \end{aligned}$$

and hence each [U4]-rewrite step on a closed term reduces its weight. It easily follows that all remaining equations also reduce weight, hence this DDRS for $\mathbb{Z}_U$ is strongly terminating. As a consequence, the DDRS for $\mathbb{N}_U$ is also strongly terminating. □

Theorem 3.1.2. *The DDRSes for Nat_1 for $\mathbb{N}_U$ (Table 4), and Int_1 for $\mathbb{Z}_U$ (Table 5) are ground-complete.*

Proof. By Lemma 3.1.1 it remains to be shown that both these DDRSes are ground-confluent. We prove this for the DDRS for $\mathbb{Z}_U$, which implies ground-confluence of the DDRS for $\mathbb{N}_U$. Define the set N as follows:

$$\begin{aligned} N &= \{0\} \cup N^+ \cup N^-, \\ N^+ &= \{0 :_u 1\} \cup \{t :_u 1 \mid t \in N^+\}, \\ N^- &= \{-t \mid t \in N^+\}. \end{aligned}$$

It immediately follows that if $t \in N$, then t is a normal form (no rewrite rule applies). Furthermore, with the semantical equation

$$x :_u 1 = x + 1,$$

it follows that two distinct elements in N have distinct values in $\mathbb{Z}$, and that the equations in Table 5 are semantic consequences of the axioms for commutative rings (Table 1). In order to prove ground-confluence it suffices to show that for each closed term t over Σ_1, either $t \in N$ or t has a rewrite step, so that each normal form is in N.

We prove this by structural induction on t. The base case is simple: if $t = 0$, then $t \in N$.

For the induction step we have to distinguish four cases:

1. Case $t = -r$. Assume that $r \in N$ and apply case distinction on r:
   - if $r = 0$, then $t \to 0$ by equation [U5],
   - if $r = r' :_u 1$, then $t \in N$,
   - if $r = -(r' :_u 1)$, then $t \to r' :_u 1$ by equation [U7].
2. Case $t = r :_u 1$. Assume that $r \in N$ and apply case distinction on r:
   - if $r = 0$, then $t \in N$,
   - if $r = r' :_u 1$, then $t \in N$,
   - if $r = -(r' :_u 1)$, then $t \to -r'$ by equation [U6].
3. Case $t = u + r$. Assume that $u, r \in N$ and apply case distinction on r:
   - if $r = 0$, then $t \to u$ by equation [U1],
   - if $r = r' :_u 1$, then $t \to (u :_u 1) + r'$ by equation [U2],
   - if $r = -(r' :_u 1)$, then $t \to -((-u) + r' :_u 1)$ by equation [U8].
4. Case $t = u \cdot r$. Assume that $u, r \in N$ and apply case distinction on r:
   - if $r = 0$, then $t \to 0$ by equation [U3],
   - if $r = r' :_u 1$, then $t \to u + (u \cdot r')$ by equation [U4],
   - if $r = -(r' :_u 1)$, then $t \to -(u \cdot (r' :_u 1))$ by equation [U9].

□

A particular property of the DDRS Nat_1 for $\mathbb{N}_U$ is captured in the following proposition.

Proposition 3.1.3. *With respect to addition and multiplication of normal forms, the DDRS Nat_1 in Table 4 is* ***deterministic****. That is, for normal forms t, t', in each state of rewriting of $t + t'$ and $t \cdot t'$ to their normal form, only one equation (rewrite rule) applies.*

Proof. The case for addition is simple: applicability of equations [U1] or [U2] excludes the other.

The case for multiplication follows by structural induction on t'.

Case $t' = 0$. Equation [U3] defines the only possible rewrite step.

Case $t' = 0 :_u 1$. Equation [U4] defines the only possible rewrite step, resulting in $t + (t \cdot 0)$, which rewrites deterministically to $t + 0$ and then to t.

Case $t' = (r :_u 1) :_u 1$. Equation [U4] defines the only possible rewrite step, deterministically resulting in a repeated addition $t + (t + (... + (t + 0)..))$ with at least two occurrences of t. It is easily seen that the only redex in any such term is $t + 0$, thus it remains to be shown that each repeated addition $t + (... + (t + t)..)$ with at least two occurrences of t rewrites deterministically, which can be shown with induction to the length ℓ of this sequence.

[U1]	$x+0=x$	[U5]	$-0=0$
[U2′]	$x+(y:_u 1)=(x+y):_u 1$	[U6]	$(-(x:_u 1)):_u 1=-x$
		[U7]	$-(-x)=x$
[U3]	$x\cdot 0=0$		
[U4′]	$x\cdot(y:_u 1)=(x\cdot y)+x$	[U8]	$x+(-y)=-((-x)+y)$
		[U9]	$x\cdot(-y)=-(x\cdot y)$

Table 6: The DDRS Int_2 for $\mathbb{Z}_U$, where equations [U1] – [U4′] define the DDRS Nat_2 for natural number arithmetic

- Case $\ell = 2$. According to the case for addition, $t+t$ rewrites deterministically.
- Case $\ell > 2$. The only redex is in the right argument, say s. By induction, s rewrites deterministically to $u:_u 1$ for some normal form u (not equal to 0). Thus $t+s$ has a reduction to $t+u:_u 1$. In each intermediate step in this reduction, each expression has the form $t+(u_1+u_2)$ and has no redex in which $t+(..)$ occurs, thus this reduction is also deterministic and results in $t+(u:_u 1)$. According to the case for addition, $t+(u:_u 1)$ rewrites deterministically.

□

3.2 From the ring of integers to unary view

In this section we relate the DDRS D_2 for the ring of integers $\mathbb{Z}_r$ to integer arithmetic as defined in the previous section. If we use $t:_u 1$ as an alternative notation for $t+1$ in D_2 and then delete all equations that contain the constant 1, we obtain the DDRS Int_2 given in Table 6, which provides an alternative specification of integer arithmetic over the signature Σ_1 comparable to the DDRS Int_1 for $\mathbb{Z}_U$ defined in Table 5. Equations [U2′] (replacing [U2]) and [U4′] (replacing [U4]) are new. Of course, [U1] + [U2′] + [U3] + [U4′] define an alternative DDRS Nat_2 for natural number arithmetic.

Theorem 3.2.1. *The DDRSes Nat_2 for $\mathbb{N}_1$ and Int_2 for $\mathbb{Z}_1$ (Table 6) are ground-complete.*

Proof. Using the weight function defined in the proof of Lemma 3.1.1, it easily follows that these DDRSes are strongly terminating.

It remains to be shown that these DDRSes are ground-confluent and it suffices to prove this for the DDRS Int_2. Equations [U2′] and [U4′] are semantic consequences of the axioms for commutative rings. It immediately follows from the proof of Theorem 3.1.2 that [U2′] implies a rewrite step if [U2] does, and [U4′] implies a rewrite step if [U4] does. Hence the DDRS Int_2 is ground-complete. □

Furthermore, Proposition 2.2.3 implies that the DDRS Int_2 is deterministic with respect to rewriting a sum of two nonnegative normal forms. However, Int_2 is not deterministic with respect to multiplication of two nonnegative normal forms:

$$0\cdot(0:_u 1)\xrightarrow{[U4']}(0\cdot 0)+0\left[\begin{array}{l}\xrightarrow{[U1]}\ 0\cdot 0\xrightarrow{[U3]}\\ \xrightarrow{[U3]}\ 0+0\xrightarrow{[U1]}\end{array}\right]0.$$

[S1]	$x + 0 = x$	[S5]	$-0 = 0$
[S2]	$x + S(y) = S(x + y)$	[S6]	$S(-S(x)) = -x$
		[S7]	$-(-x) = x$
[S3]	$x \cdot 0 = 0$		
[S4]	$x \cdot S(y) = (x \cdot y) + x$	[S8]	$x + (-y) = -((-x) + y)$
		[S9]	$x \cdot (-y) = -(x \cdot y)$

Table 7: The DDRS Int_3 for integer arithmetic with the constant 0 and successor function, where the equations in the left column define the DDRS Nat_3 for natural number arithmetic

3.3 Successor function instead of unary append

Finally, we consider the adaptations of the DDRSes defined in Sections 3.2 and 3.1 (in that order) to the successor function $S(x)$. Replacing all $t :_u 1$-occurrences by $S(t)$ in the DDRSes Nat_2 and Int_2 defined in Table 6 results in the DDRSes Nat_3 and Int_3 in Table 7, which define alternative and concise DDRSes for natural number and integer arithmetic in unary view with successor function. The same replacement in the DDRSes Nat_1 (Table 4) and Int_1 (Table 5) results in the DDRSes Nat_4 and Int_4 defined in Table 8. Normal forms are $0, S(0), S(S(0)), \ldots$ and for integers also the negations of all nonzero normal forms.

Theorems 3.2.1 and 3.1.2 immediately imply the following corollary with respect to these alternative representations.

Corollary 3.3.1. *The DDRSes Nat_3 and Int_3 defined in Table 7 are ground-complete. The two DDRSes Nat_4 and Int_4 defined in Table 8 are ground-complete.*

The DDRS Nat_3 defined in Table 7 defines natural number arithmetic with 0 and successor function, and is very common (see, e.g. [4, 6, 8]). Furthermore, this DDRS is deterministic with respect to addition of two normal forms (cf. Proposition 2.2.3), but not with respect to their multiplication (cf. (9) in Section 3.2).

We note that the DDRS Int_3 for $\mathbb{Z}_1$ defined in Table 7 provides an alternative of smaller size for the datatype $\mathbb{Z}_{ubd}$ defined in [2][1] when in the latter the equations for predecessor and

[1] The name $\mathbb{Z}_{ubd}$ refers to: $\mathbb{Z}$ in unary view with conversion from binary and decimal notation (and view).

[S1]	$x + 0 = x$	[S5]	$-0 = 0$
[S2′]	$x + S(y) = S(x) + y$	[S6]	$S(-S(x)) = -x$
		[S7]	$-(-x) = x$
[S3]	$x \cdot 0 = 0$		
[S4′]	$x \cdot S(y) = x + (x \cdot y)$	[S8]	$x + (-y) = -((-x) + y)$
		[S9]	$x \cdot (-y) = -(x \cdot y)$

Table 8: The DDRS Int_4 for integer arithmetic with the constant 0 and successor function, where the equations in the left column define the DDRS Nat_4 for natural number arithmetic

conversion from binary and decimal notation are disregarded: equation [S8], used to complete the definition of addition on the integers, replaces equations [u8] – [u10] in our earlier paper [2], while the remaining eight equations are those defined in [2] for $\mathbb{Z}_{ubd}$.

Finally, the DDRS Nat_4 is deterministic with respect to addition and multiplication of two normal forms (cf. Proposition 3.1.3). However, equation [S2′] (as a rewrite rule) is less common for defining addition on this concrete datatype for natural number arithmetic, and neither is equation [S4′] for defining multiplication on the natural numbers. In [7] this specification (modulo symmetry) occurs as a running example.

4 Conclusions

In this paper we specify integer arithmetic for the ring of integers and for numbers represented in a unary numeral system by DDRSes that contain less equations than the associated DDRSes discussed in our paper [2]. In all cases, this is due to the equation

$$x + (-y) = -((-x) + y),$$

which is used to complete the definition of addition for integers for negative normal forms.

A general property of the DDRSes defined in this paper is that the recursion in the definitions of addition and multiplication takes place on the right argument of these operators (as is common), if necessary first replacing negation. Of course, we could have used recursion on the left argument instead, obtaining symmetric versions of these DDRSes (for natural number arithmetic with successor function, this is done in e.g. [3, 7, 9]).

In Section 2 we provided two DDRSes for the datatype $\mathbb{Z}_r$, the ring of integers with the set NF of normal forms defined by

$$\begin{aligned} NF &= \{0\} \cup NF^+ \cup NF^-, \\ NF^+ &= \{1\} \cup \{t + 1 \mid t \in NF^+\}, \\ NF^- &= \{-t \mid t \in NF^+\}. \end{aligned}$$

Each of these DDRSes consists of only twelve equations. Perhaps the DDRS D_2 is most attractive: it is comprehensible and deterministic with respect to addition of (two) nonnegative normal forms. We leave it as an open question whether $\mathbb{Z}_r$ can be specified by a DDRS with less equations (starting from the set NF of normal forms). Another open question is to find a DDRS for $\mathbb{Z}_r$ based on NF that is also deterministic with respect to rewriting $t \cdot t'$ for nonnegative normal forms t and t'.[2]

In Section 3 we provided two DDRSes for natural number and integer arithmetic in unary view, based on the constant 0 and unary append (instead of successor function). Both DDRSes for integer arithmetic contain only nine equations and we leave it as an open question whether

[2]Note that the alternative for equation [R6] suggested by the DDRS Int_2, that is, the equation

$$x \cdot (y + 1) = x + (x \cdot y)$$

does not solve this open question: although the reduction

$$1 \cdot ((((1 + 1) + 1) + 1) + 1) \twoheadrightarrow 1 + (1 + (1 + (1 + 1))) \xrightarrow{[R3']} 1 + (1 + ((1 + 1) + 1)) \xrightarrow{[R3']} 1 + ((1 + (1 + 1)) + 1)$$

is deterministic, the last term has a rewrite step to $(1 + (1 + (1 + 1))) + 1$ and to $1 + (((1 + 1) + 1) + 1)$.

the concrete datatype $\mathbb{Z}_1$ can be specified as a DDRS with less equations. Concerning their counterparts that define natural number arithmetic (Nat_1 and Nat_2, both containing four equations), the DDRS Nat_1 is deterministic with respect to addition and multiplication of normal forms (Proposition 3.1.3). Furthermore, the DDRSes Nat_1 and Nat_2 are attractive, if only from a didactical point of view:

1. Positive numbers are directly related to tallying and admit an easy representation and simplifying abbreviations for normal forms, such as 011 for $(0 :_u 1) :_u 1$, or even 11 or || when removal of the leading zero in positive numbers is adopted.

2. Natural number arithmetic on small numbers can be represented in a comprehensible way that is fully independent of the learning of any positional system for number representation, although names of numbers (zero, one, two, and so on) might be very helpful.[3] Furthermore, notational abbreviations for units of five, like in

   ~~1111~~ ~~1111~~ 11 or $\cancel{||||}\ \cancel{||||}\ ||$ or 011111 11111 11

   can be helpful because 0111111111111 (thus twelve) is not very well readable or easily distinguishable from 011111111111 (thus eleven).

With respect to negative numbers, similar remarks can be made, but displaying computations according to the DDRSes Int_1 and Int_2 will be more complex and bracketing seems to be unavoidable. Consider for example

$$(-(\,||\,)) + \cancel{||||}\ \cancel{||||}\ || \;=\; (-(\,||\,)\,|\,) + \cancel{||||}\ \cancel{||||}\ | \;=\; (-(\,|\,)) + \cancel{||||}\ \cancel{||||}\ | \;=\; \ldots$$

Although it can be maintained that as a constructor, unary append is a more illustrative notation than successor function, it is of course only syntactic sugar for that function. In Section 3.3 we provided DDRSes with successor function, in order to emphasize their conciseness and comprehensibility for integer arithmetic.

Furthermore, if we add the predecessor function $P(x)$ to the DDRSes defined in Section 3 by the three equations

[P1]	$P(0) = -(0 :_u 1)$	respectively	$P(0) = -S(0)$
[P2]	$P(x :_u 1) = x$	respectively	$P(S(x)) = x$
[P3]	$P(-x) = -(x :_u 1)$	respectively	$P(-x) = -S(x)$

we find that the resulting DDRSes improve on those for unary view defined in [2] in terms of simplicity and number of equations.

Finally, if we also add the subtraction function $x - y$ by the single equation

[Sub] $\quad x - y = x + (-y)$

this also improves on the specification of integer arithmetic in unary view in [8], which does not employ the unary minus function and uses seventeen rules, and thus eighteen rules when adding the minus function by the rewrite rule

$$-x \;\to\; 0 - x.$$

However, we should mention that in [8], $P(0), P(P(0)), \ldots$ are used as normal forms for negative numbers, instead of $-(0 :_u 1), -((0 :_u 1) :_u 1), \ldots$ or $-S(0), -S(S(0)), \ldots$.

[3] In English, Dutch and German, this naming is up to twelve independent of decimal representation, and in French this is up to sixteen.

References

[1] Bergstra, J.A. and Ponse, A. (2016). Fracpairs and fractions over a reduced commutative ring. *Indagationes Mathematicae*, 27:727–748 (`doi.org/10.1016/j.indag.2016.01.007`). Preprint available at `arXiv/1411.4410v2` [math.RA] (2016, 22 Jan).

[2] Bergstra, J.A. and Ponse, A. (2016). Three datatype defining rewrite systems for datatypes of Integers each extending a datatype of Naturals. Available at arXiv:1406.3280v4 [cs.LO], 18 July 2016.

[3] Bouma, L.G. and Walters, H.R. (1989). Implementing algebraic specifications. In J.A. Bergstra, J. Heering, and P. Klint (Eds.): *Algebraic Specification* (Chapter 5), Addison-Wesley, pp. 199–282.

[4] Klop, J.W. and Vrijer, R.C. de (2003). *First-order term rewriting systems*. Chapter 2 in [5], pp. 24–59.

[5] Terese (2003). *Term Rewriting Systems*. Cambridge Tracts in Theoretical Computer Science, Vol. 55, Cambridge University Press.

[6] Urso, P. and Kounalis, E. (2003). "Term partition" for mathematical induction. In R. Nieuwenhuis (Ed.): *RTA 2003*, Lecture Notes in Computer Science, Vol. 2706, Springer, pp. 352–366. Extended version available at `https://hal.inria.fr/inria-00072023/document` (2006).

[7] Walters, H.R. (1990). Hybrid implementations of algebraic specifications. In H. Kirchner, W. Wechler (Eds.): *Algebraic and Logic Programming*, Lecture Notes in Computer Science, Vol. 463, Springer, pp. 40–54.

[8] Walters, H.R. and Zantema, H. (1995). Rewrite systems for integer arithmetic. In J. Hsiang (Ed.): *Rewriting Techniques and Applications (Proceedings 6th International Conference, RTA'95)*, Lecture Notes in Computer Science, Vol. 914, Springer, pp. 324–338. Preprint available at `http://oai.cwi.nl/oai/asset/4930/4930D.pdf`.

[9] Zantema H. (2003). *Termination*. Chapter 6 in [5], pp. 181–259.