On termination and confluence of rewriting with real numbers

Fer-Jan de Vries * ;3ED =gG72[†]p

Summary. We study a term rewriting system for positional arithmetic with real numbers (with finite decimal expansions) and $+$ and $\times$. The system is terminating, and can be extended to a rewrite system which modulo associativity and commutativity of $+$ and $\times$ is both terminating and confluent.

1 Introduction

In this paper we introduce a term rewriting system which closely mimicks positional arithmetic with positive real numbers and $+$ and $\times$. If one considers finite terms only this TRS represents arithmetic with positive real numbers with finite expansions. In a context with infinitary rewriting as in [KKSdV] it performs arithmetic for arbitrary real numbers. Using Zantema's recent semantic labeling technique we prove the termination of the system. The closed normal forms of the system correspond to real numbers. Closed terms have unique normal forms. Hence the system is ground confluent. The system is not locally confluent. However, there exists a natural extension of the system which is confluent modulo associativity and commutativity of $+$ and $\times$.

Rules for subtraction, as introduced by Walters in his TRS for integer arithmetic, can be added. However, then we are no longer able to prove termination[1]. Walters (cf. [Wal94]) proved termination for closed terms of integer arithmetic with $+$ and $-$. This proof is rather intricate, and does not seem to allow extension of the signature to multiplication.

The rewrite system for postive reals in this paper uses explicit concatenation symbols in order to capture arithmetic in the term rewriting formalism. We use $x : y$ meaning $10x + y$ to deal with digits in front of the decimal point and $x; y$ meaning $x + \frac{1}{10}y$ to cope with the digits in the decimal expansion.

For integer arithmetic with $+, -$ and $\times$, Cohen and Watson (cf. [CW91] and appendix A) seem to have been the first to experiment with such term rewriting systems. Termination of their system modulo associativity and commutativity of $+$ and $\times$ was left open (cf. [Kir93]).

Note: throughout this paper we will work with ternary number systems, this is inessential.

* NTT Communication Science Laboratories, Kyoto

† Junnosuke YAMADA, NTT %3%_%e%K%1!<%7%g%s2J3X8&5f=j

[1] In fact, Hans Zantema pointed out a rather embarrassing mistake in a proof attempt of us in an earlier draft of this paper, which we have not been able to not recover. He also claimed that he just had solved the termination problem for full integer arithmetic. We have not yet seen his proof.

$$\begin{aligned} x : (y : z) &\to (x + y) : z \\ x + y : z &\to y : (x + z) \\ x : y + z &\to x : (y + z) \\ 0 : x &\to x \end{aligned} \qquad \begin{aligned} 0 + 0 &\to 0 \\ 0 + 1 &\to 1 \\ 0 + 2 &\to 2 \\ 1 + 1 &\to 2 \\ 1 + 2 &\to 1 : 0 \\ 2 + 0 &\to 2 \\ 2 + 1 &\to 1 : 0 \\ 2 + 2 &\to 1 : 1 \end{aligned}$$

Table 1. The TRS NAT for addition of ternary natural numbers.

1.1 Intuition and an example

The idea behind the rewrite systems for numbers considered in this paper is quite simple: the number 123 is actually a concatenation $1 : 2 : 3$ of the digits 1, 2, and 3. The concatenation symbol is left implicit, just as well as the bracket convention that $1 : 2 : 3$ should be read as $(1 : 2) : 3$ because

$$1 : 2 : 3 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 = (1 \times 10^1 + 2 \times 10^0) \times 10^1 + 3 \times 10^0 = (1 : 2) : 3.$$

Apparently arbitrary numbers can be concatenated meaningfully, provide we interpret $x : y = 10x+y$. For example $1(23) = 1 : (2 : 3) = 10\times1+(10\times2+3) = 10\times(1+2)+3 = 33$. So, in order to rewrite $1(23)$ to the more preferably format 33 we need a rewrite rule

$$x(yz) \to (x + y)z$$

involving addition. Such considerations lead naturally to the concise term rewriting system NAT of Table 1 that performs addition on (ternary) natural numbers. The rule $0 : x \to x$ follows from the convention to write natural numbers without leading zeros, e.g., 00122 is not a natural number. We will be using base 3 in this paper. The following (decimal) example shows that addition follows the familiar pattern.

$$\begin{aligned} 2 : 5 + (1 : 7) : 6 &\to (1 : 7) : (2 : 5 + 6) \\ &\to (1 : 7) : (2 : (5 + 6)) \\ &\to (1 : 7) : (2 : (1 : 1)) \\ &\to (1 : 7) : ((2 + 1) : 1) \\ &\to (1 : 7) : (3 : 1) \\ &\to (1 : 7 + 3) : 1 \\ &\to (1 : (7 + 3)) : 1 \\ &\to (1 : (1 : 0)) : 1 \\ &\to (1 + 1) : 0) : 1 \\ &\to (2 : 0) : 1 \end{aligned}$$

The system is designed for easy understanding of its correctness. It is strong enough that all closed terms rewrite to the desired normal form format. The system consist of a table of ground rules, that explain how to add the digits, and structural rewrite rules that explain the interaction of addition with the basic concatenation operator.

In this paper we propose an extension of the integer representation to real numbers. We will use another concatenation symbol to deal with the digits in the decimal expansion.

For example:

$$-321.6789 \equiv -((3:2):1);(6;(7;(8;9)))$$

That is we use a new concatenation symbol ";" with intuitive interpretation $x;y = x + \frac{1}{10}y$. Under this interpretation it follows that the operator ; associates to the right. Using both concatenation symbols we no longer need a notation for the decimal point.

It is not easy to give a direct proof of the termination of such arithmetic systems based on concatenation (cf. the direct, but complicated, proof of Walters for the termination of integer arithmetic with $+$ and $-$ in [Wal94]). This is because a direct lexicographic path order argument does not exist: the first rule requires that $: > +$, whereas the second and third rule need the converse precedence $+ >:$. However, using a suitable semantics one can reduce the termination problem of NAT to the termination problem of the labeled version NAT^{lab} presented in Table 2 by the semantic labeling technique. NAT^{lab} terminates via a straightforward lexicographic path order argument with the following precendence on the function symbols:

$$\ldots > +_{i+1} > :_{i+1} > +_i > :_i > \ldots > +_0 > :_0 > 2 > 1 > 0.$$

$x :_{i+j+k+1} (y :_{j+k} z) \rightarrow (x +_{i+j+1} y) :_{i+j+k+1} z$
$x +_{i+j+k+2} (y :_{j+k+1} z) \rightarrow y :_{i+j+k+1} (x +_{i+k+1} z)$
$(x :_{i+j+1} y) +_{i+j+k+2} z \rightarrow x :_{i+j+k+1} (y +_{j+k+1} z)$
$0 :_{i+1} x \rightarrow x$

$0 +_1 0 \rightarrow 0$
$0 +_2 1 \rightarrow 1$
$0 +_3 2 \rightarrow 2$
$1 +_3 1 \rightarrow 2$
$1 +_4 2 \rightarrow 1 :_2 0$
$2 +_3 0 \rightarrow 2$
$2 +_4 1 \rightarrow 1 :_2 0$
$2 +_5 2 \rightarrow 1 :_2 1$

Table 2. Labeled version NAT^{lab} of the TRS NAT. $(i, j, k \in \mathbb{N})$.

2 Preliminaries

We assume the basic definitions and notations of term rewiting to be known, and refer to the overview papers by Dershowitz and Jouannaud [DJ89], and Klop [Klo92]. We will only sketch the tools we need for our termination and confluence (modulo E) proofs.

2.1 Lexicographic path order

Given a partially ordered (well-founded or just) finite signature, the lexicographic path ordering extends the given order to a well founded ordering on terms over the signature. See [DJ89] for details and references. Let $\succ$ denote such a partial order. Then the lexicographic path order $\succ_{lpo}$ is defined by induction. By $\succ_{lpo}^{=}$ we denote its reflexive closure.

DEFINITION 2.1 $s = f(s_1, \ldots, s_n) \succ_{lpo} t$ if one of the following mutally exclusive alternatives holds:

- $t = f(t_1, \ldots, t_n)$ and there is an $i \in \{1, \ldots, n\}$ such that

- $s_j = t_j$ for all $1 \le j < i$,
- $s_i \succ_{lpo} t =_i$,
- $s \succ_{lpo} t =_j$ for all $i < j \le n$

- $t = g(t_1, \ldots, t_m)$, $f \succ g$, and $s \succ_{lpo} t_i$ for all $1 \le i \le m$
- $s_i \succ_{lpo}^{=} t$ for some $1 \le i \le n$.

2.2 Semantic labeling

Zantema [Zan] has developed a useful technique to reduce the termination problem of a TRS to the termination problem of another TRS, which is a labeled version of the original TRS, based on some semantics for it. A TRS which can not be proven to be terminating by, say, the lexicographic path order can sometimes in this way be transformed with a well chosen semantic labeling into a TRS for which the lexicographic path order can prove termination. We give the definitions which lead to formulation of this result.

DEFINITION 2.2 Let R be a TRS over the signature $< F, X >$.

- Let $(M, >)$ and $(N, >)$ be partial orders. A function $f : M^n \to N$ is said to be weakly monotone if
$$f(a_1, \ldots, a_n) \ge f(b_1, \ldots, b_n)$$
for all $a_1, \ldots, a_n, b_1, \ldots, b_n \in M$ satisfying $a_i \ge b_i$ for all i.
- A (weakly monotone) F-algebra is a partial ordered set $(M, >)$ together with for each $f \in F$ a weakly monotone function $[\![f]\!] : M^{arity(f)} \to M$.
- A (quasi-)model for a TRS R is F-algebra M such that
$$[\sigma](l) \ge [\sigma](r)$$
for all rules $l \to r$ of R and for all valuations $\sigma : X \to M$.
- A labeling lab over an F-algebra M consists of a well-founded partial order $(L_f, >)$ and a weakly monotone function $lab_f : M^n \to L_f$ for each $f \in F$.
Labeling of terms is defined inductively by:
$$\begin{aligned} lab(x, \sigma) &= x, \\ lab(f(t_1, \ldots, t_n), \sigma) &= f_{lab_f([\sigma](t_1), \ldots, [\sigma](t_2))} \end{aligned}$$
for $x \in X, \sigma : X \to M, f \in F$ and $t_1, \ldots, t_n \in T(F, X)$.
- Given a labeling lab over an F-algebra M, define
$$F^{lab} = \{f_l \mid f \in F, l \in L_f\} \cup \{f \mid f \in F, L_f = \emptyset\},$$
define R^{lab} as the TRS with signature $< F^{lab}, X >$ and set of rules
$$\{lab(l, \sigma) \to lab(r, \sigma) \mid l \to r \in R, \sigma : X \to M\}$$
and define
$$Dec(F^{lab}) = \{f_{l_1} \to f_{l_2} \mid f \in F, l_1 > l_2 \in L_f\}.$$

THEOREM 2.3 [ZANTEMA'S SEMANTIC LABELING THEOREM] *Let M be a model for a TRS R, and let lab be a labeling of F over M. Then R is terminating if and only if $R^{lab} \cup Dec(F^{lab})$ is terminating.[Zan]*

2.3 Rewriting modulo E

We will encounter rewriting systems which are confluent modulo a set of equations. Results by Huet and by Jouannaud and Muñoz are useful to establish this. We first give definitions of the necessary concepts to formulate their results.

Let R be a rewrite system and E a set of equations. Let $\rightarrow_R$ denote the rewrite relation of R and let $\leftrightarrow_E$ denote the smallest symmetric relation containing E and being closed under contexts and substitutions. The equational rewrite system R modulo E (notation R/E) is the rewrite relation $\rightarrow_R \circ \leftrightarrow_E^*$.

DEFINITION 2.4
- R is terminating modulo E, if R modulo E is terminating.
- R is *quasi E-commutative*, if $\leftrightarrow_E^* \circ \rightarrow_R \;\subseteq\; \rightarrow_R \circ \rightarrow_{R/E}^*$.
- R is *confluent modulo E*, if, $\leftarrow_R^* \circ \leftrightarrow_E^* \circ \rightarrow_R^* \;\subseteq\; \rightarrow_{R/E}^* \circ \leftrightarrow_E^* \circ \leftarrow_{R/E}^*$.
- R is *locally confluent modulo E*, if $\leftarrow_R \circ \rightarrow_R \;\subseteq\; \rightarrow_{R/E}^* \circ \leftrightarrow_E^* \circ \leftarrow_{R/E}^*$.
- R is *locally coherent modulo E*, if $\leftrightarrow_E \circ \rightarrow_R \;\subseteq\; \rightarrow_R^* \circ \leftrightarrow *_E \circ \leftarrow_R^*$.

THEOREM 2.5 (JOUANNAUD-MUÑOZ) *If R is terminating and quasi E-commutative then R is terminating modulo E.(Cf. [JM83])*

THEOREM 2.6 (HUET) *If R is terminating modulo E, locally confluent modulo E, and locally coherent modulo E then R is confluent modulo E. (Cf. [Hue80])*

For a nice proof of these theorems see the thesis of van Oostrom ([vO94]).

$0+0 \rightarrow 0$	$0 \times 0 \rightarrow 0$
$0+1 \rightarrow 1$	$0 \times 1 \rightarrow 0$
$0+2 \rightarrow 2$	$0 \times 2 \rightarrow 0$
$1+1 \rightarrow 2$	$1 \times 1 \rightarrow 1$
$1+2 \rightarrow 1:0$	$1 \times 2 \rightarrow 2$
$2+0 \rightarrow 2$	$2 \times 0 \rightarrow 0$
$2+1 \rightarrow 1:0$	$2 \times 1 \rightarrow 2$
$2+2 \rightarrow 1:1$	$2 \times 2 \rightarrow 1:1$

$x+(y:z) \rightarrow y:(x+z)$	$(x:y) \times z \rightarrow (x \times z):(y \times z)$
$(x:y)+z \rightarrow x:(y+z)$	$x \times (y:z) \rightarrow (x \times y):(x \times z)$
$x:(y:z) \rightarrow (x+y):z$	
$0:x \rightarrow x$	

Table 3. The rewrite system NAT for ternary natural numbers with + and ×

3 Natural numbers with + and ×

As rewrite system for (ternary) natural numbers with addition and multiplication we propose the system listed in Table 3.

The system consist of ground rules explaining how to add and multiply the digits, and structural rewrite rules that explain the interaction of addition and multiplication with the basic concatenation operator.

The standard model for this TRS is of course the set of natural numbers $\mathbb{N}$. Digits and addition as well as multiplication are interpreted as expected, and concatenation is interpreted by $x : y = 10x + y$. Proving correctness of this interpretation and establishing the format of the normal forms of the closed terms is easily done using induction.

Proposition 3.1 [Correctness] *If $t \to s$ in NAT then $t = s$ in $\mathbb{N}$ for terms t, s in NAT.* □

Proposition 3.2 *A closed term t is a normal form in NAT if and only if*

$$t \equiv ((\ldots(d_1 : d_2) : \ldots) : d_n),$$

where the d_i are digits and $d_1 \neq 0$ □

Theorem 3.3 [Termination] *The TRS NAT for natural numbers with addition and multiplication is terminating.*

Proof.

- As weakly monotone NAT-algebra we take a the set of natural numbers $\mathbb{N}$ with the following interpretion of the function symbols:

$$\begin{aligned} [\![0]\!] &= 0 \\ [\![1]\!] &= 1 \\ [\![2]\!] &= 2 \\ [\![:]\!](n,m) &= n+m+1 \\ [\![+]\!](n,m) &= n+m \\ [\![\times]\!](n,m) &= 5^{n+m+1} \end{aligned}$$

  It is not difficult to check that it makes a rewrite model for NAT: with respect to the multiplication rules, observe that

$$5^{2+i+j+k} \geq 1 + 5^k(\cdot 5^{i+1} + 5^{j+1}),$$

  for any $i, j, k \in \mathbb{N}$.

- For both $+$ and $:$ we chose $(\mathbb{N}, <)$ as well founded set of labels. As weakly monotone label functions we take:

$$\begin{aligned} \pi_:(n,m) &= n+m+1 \\ \pi_+(n,m) &= n+m+1 \end{aligned}$$

  The other symbols we label with the empty set, i.e., in effect we don't label them.

- According to the recipe of the semantic labeling technique we now construct a new TRS NAT^{lab}. Its signature consists of the constants $0, 1, \ldots, 9$ and infinitely many binary symbols $x_i, +_i, :_i$ for $i \in \mathbb{N}$.

$0 +_1 0 \to 0$	$0 \times 0 \to 0$
$0 +_2 1 \to 1$	$0 \times 1 \to 0$
$0 +_3 2 \to 2$	$0 \times 2 \to 0$
$1 +_3 1 \to 2$	$1 \times 1 \to 1$
$1 +_4 2 \to 1 :_3 0$	$1 \times 2 \to 2$
$2 +_3 0 \to 2$	$2 \times 0 \to 0$
$2 +_4 1 \to 1 :_3 0$	$2 \times 1 \to 2$
$2 +_5 2 \to 1 :_4 1$	$2 \times 2 \to 1 :_4 1$

$$
\begin{array}{rcl}
(x+y)\times z & \rightarrow & (x\times z)+(y\times z)\\
x\times(y+z) & \rightarrow & (x\times y)+(x\times z)\\
0+x & \rightarrow & x\\
0\times x & \rightarrow & 0\\
x\times 0 & \rightarrow & 0\\
1\times x & \rightarrow & x\\
x\times 1 & \rightarrow & x\\
x+x & \rightarrow & 2\times x\\
x+2\times x & \rightarrow & 3\times x\\
 & \vdots & \\
x+9\times x & \rightarrow & x:0
\end{array}
$$

Table 4. The set R_{NAT} needed for local confluence modulo AC$(+,\times)$ of NAT

$$
\begin{array}{rcl}
x :_{i+j+k+2} (y :_{j+k+1} z) & \rightarrow & (x +_{i+j+1} y) :_{i+j+k+1} z\\
x +_{i+j+k+2} (y :_{j+k+1} z) & \rightarrow & y :_{i+j+k+1} (x +_{i+k+1} z)\\
(x :_{i+j+1} y) +_{i+j+k+2} z & \rightarrow & x :_{i+j+k+1} (y +_{j+k+1} z)\\
0 :_{i+1} x & \rightarrow & x\\
\\
(x :_{2i+j+1} y) \times z & \rightarrow & (x \times z) :_{2\times 5^{i+k+1}+5^{j+k+1}+1} (y \times z)\\
x \times (y :_{2j+k+1} z) & \rightarrow & (x \times y) :_{2\times 5^{i+y}+5^{i+k}+1} (x \times z)
\end{array}
$$

- Using following lexicographic precedence order on the signature it is an easy verification that $l \succ^{lpo} r$ for each of the above rules $l \rightarrow r$:

$$\times > \ldots \succ +_{i+1} \succ :_{i+1} \succ +_i \succ :_i \succ \ldots \succ :_0 \succ 9 \succ \ldots \succ 0.$$

Hence the labeled TRS NAT^{lab} is terminating by the lexicographic path order (2.1). By Zantema's semantic labeling theorem (2.3) the original TRS NAT is terminating. □

Any closed term of NAT reduces to a normal form. Because of the correctness of the standard model, a closed term can have only one normal form. It follows that:

PROPOSITION 3.4 *NAT is ground CR.* □

The TRS NAT is not even locally confluent on open terms. Inspection of the normal forms of all critical pairs leads besides associativity and commutativity of +, AC(+), to the set of rules R_{NAT} of Table 4. All rules can be ordered lexicographically, when labeled as in the above termination proof. This implies that NAT can be extended by R_{NAT} preserving termination termination.

When we tried to complete NAT $\cup$ R_{NAT} modulo AC(+) with help of Marché's implementation C*i*ME of his normalised completion algorithm [Mar94], we observed that we also needed AC($\times$) to make to program terminate. The analysis of the program showed that the following subsystem (from which NAT $\cup$ R_{NAT} clearly can be derived) in Table 5 is complete modulo AC$(+,\times)$:

THEOREM 3.5 *NAT$\cup R_{NAT}$ is terminating modulo AC$(+,\times)$ and confluent modulo AC$(_{+,\times})$.*

$$
\begin{aligned}
x + 0 &\rightarrow 0 \\
2 + 1 &\rightarrow 1 : 0 \\
x + x &\rightarrow 2 \times x \\
x + (y : z) &\rightarrow y : (x + z) \\
x : (y : z) &\rightarrow (x + y) : z \\
0 : x &\rightarrow x \\
x + (x \times 2) &\rightarrow x : 0 \\
x \times 0 &\rightarrow 0 \\
x \times 1 &\rightarrow x \\
2 \times 2 &\rightarrow 1 : 1 \\
x \times (y : z) &\rightarrow (x \times y) : (x \times z) \\
x \times (y + z) &\rightarrow (x \times y) + (x \times z)
\end{aligned}
$$

Table 5. A TRS for natural numbers arithmetic which is complete modulo AC$(+, \times)$

$$
\begin{aligned}
x; 0 &\rightarrow 0 \\
(x; y) + z &\rightarrow (x + z); y \\
x + (y; z) &\rightarrow (x + y); z \\
(x; y); z &\rightarrow x; (y + z) \\
\\
(x; y) : z &\rightarrow x : (y + z) \\
x; (y : z) &\rightarrow (x + y); z \\
x : (y; z) &\rightarrow (x : y); z \\
w; ((x : y); z) &\rightarrow (w + x); (y; z) \\
\\
(x; y) \times z &\rightarrow (x \times z); (y \times z) \\
x \times (y; z) &\rightarrow (x \times y); (x \times z)
\end{aligned}
$$

Table 6. The rules which extend NAT to the TRS posREALS

PROOF. Since NAT $\cup$ R_{NAT} is terminating, it suffices by Theorem 2.5 of Jouannoud and Munoz to verify that NAT $\cup$ R_{NAT} is quasi-AC$(+, \times)$ commuting in order to conclude that NAT $\cup$ R_{NAT} is terminating modulo AC$(+, \times)$. This verification is a straightforward inspection of all cases. In a similar way one may verify that NAT $\cup$ R_{NAT} is locally confluent modulo AC$(+, \times)$ (if one does not trust the verification by machine) and locally coherent modulo AC$(+, \times)$. By Huet's Theorem 2.6 it follows that NAT $\cup$ R_{NAT} is CR modulo AC$(+, \times)$. □

4 Positive real numbers with + and ×

We will represent reals with two concatenation symbols, as indicated by the example:

$$1.02 \equiv (1; (0; 2))$$

$$(x;_{i+j}y)+_{i+j+k+2}z \to (x+_{i+k+1}z);_{i+j+k}y$$
$$x+_{i+j+k+2}(y;_{j+k}z) \to (x+_{i+j+1}y);_{i+j+k}z$$
$$(x;_{i+j}y);_{i+j+k+1}z \to x;_{i+j+k}(y+_{j+k+1}z)$$

$$(x;_{i+j}y):_{i+j+k+3}z \to x:_{i+j+k+1}(y+_{j+k+1}z)$$
$$x;_{i+j+k+1}(y:_{j+k+1}z) \to (x+_{i+j+1}y);_{i+j+k}z$$
$$x:_{i+j+k+2}(y;_{j+k}z) \to (x:_{i+j+1}y);_{i+j+k+2}z$$
$$w;_{i+j+k+l+2}((x:_{j+k+1}y);_{j+k+l+1}z) \to (w+_{i+j+1}x);_{i+j+k+l+1}(y;_{k+l}z)$$

$$(x;_{i+j}y)\times z \to (x\times z);_{5^{i+k+1}+5^{j+k+1}}(y\times z)$$
$$x\times(y;_{j+k}z) \to (x\times y);_{5^{i+j+1}+5^{i+k+1}}(x\times z)$$

Table 7. The rules which extend NAT^{lab} to the TRS posREALS^{lab}

The interpretation is $x;y = x + \frac{1}{10}y$. Indeed,

$$1.02 = 1 + \frac{1}{10}(0 + \frac{1}{10}2).$$

We don't need notation for the decimal point. The interaction between the new concatenation symbol and the signature of NAT can easily be derived from this interpretation. The term rewrite system posREALS of real numbers with finite decimals expansion has as signature the signature of NAT extended with ; and as rules the rules of NAT extended with the rules of Table (6).

As standard model we take the set $\mathbb{R}$ of real numbers with the usual interpretion of $+$ and $\times$. Correctness of the system is straightforward. As interpretation of the two concatenation symbols we take:

$$[\![t:s]\!] = 10[\![t]\!] + [\![s]\!]$$
$$[\![t;s]\!] = [\![t]\!] + \frac{1}{10}[\![s]\!]$$

Proposition 4.1 Correctness. *For all finite terms t, s it holds that if posREALS $\models t \to s$ then $\mathbb{R} \models t = s$.* □

Proposition 4.2 *A closed term t is a normal form in posREALS if and only if*

$$t \equiv (\ldots((\ldots(d_1 : d_2) : \ldots) : d_n); e_1)\ldots); e_m),$$

where the d_i's and e_j's are digits, $n > 0$ and $m \geq 0$, and $d_1 \neq 0$. □

The TRS posREALS is an extension of NAT with one symbol and some rules: its termination proof will be a simple extension of the termination proof of NAT.

Theorem 4.3 Termination. *The TRS posREALS is terminating.*

Proof. Semantic labeling:

- We take the model we used in the termination proof of NAT and interpret the new concatenation symbol $x;y$ by

$$[\![;]\!](n,m) = n + m + 1.$$

It is easy to check that this gives a model for posREALS.

- We will label ; and give it the label set $(\mathbb{N}, >)$. As weakly monotone label function we take

$$\pi_;(n,m) = n + m$$

- We extend the precedence order as follows:

$$\times > \ldots > +_{i+1} > :_{i+1} > ;_{i+1} > +_i > :_i > ;_i > \ldots > ;_0 > 9 > \ldots > 0.$$

Remains to prove that this precedence order indeed orders all the labeled rules. The new labeling is identical to the old one on rules of NAT. So we have only to verify that $l \succ^{lpo} r$ for each of the labeled rules $l \to r$ of Table 7. Hence the labeled TRS REAL^{lab} is terminating by the lexicographic path order (2.1). By Zantema's semantic labeling theorem (2.3) the original TRS posREALS is terminating. Note that in the labeled version of $(x;y):z \to x:(y+z)$ we have to inspect the concatenation symbol : from left to right in the lexicographic order argument. □

As for NAT we have:

PROPOSITION 4.4 *posREALS is ground CR.* □

Note that posREALS is not locally confluent for the same reason that its subTRS NAT was not locally confluent. However the same set of extra rules R_{NAT} needed to make NAT locally confluent suffices to make posREALS locally confluent modulo AC$(+,\times)$. With C*i*ME we found the following completion modulo AC$(+,\times)$ of posREALS.

$$\begin{array}{rcl}
x+0 & \to & 0 \\
2+1 & \to & 1:0 \\
x+x & \to & 2\times x \\
x+(y:z) & \to & y:(x+z) \\
x:(y:z) & \to & (x+y):z \\
0:x & \to & x \\
x+(x\times 2) & \to & x:0 \\
x\times 0 & \to & 0 \\
x\times 1 & \to & x \\
2\times 2 & \to & 1:1 \\
x\times(y:z) & \to & (x\times y):(x\times z) \\
x\times(y+z) & \to & (x\times y)+(x\times z) \\
x;0 & \to & 0 \\
x+(y;z) & \to & (x+y);z \\
(x;y);z & \to & x;(y+z) \\
(x;y):z & \to & x:(y+z) \\
x;(y:z) & \to & (x+y);z \\
x:(y;z) & \to & (x:y);z \\
w;((x:y);z) & \to & (w+x);(y;z) \\
x\times(y;z) & \to & (x\times y);(x\times z)
\end{array}$$

THEOREM 4.5 *posREALS$\cup R_{NAT}$ is terminating modulo AC$(+,\times)$ and CR modulo AC$(+,\times)$.*

PROOF. The proof is analogous to the proof of the similar theorem 3.5 for NAT. □

5 Real numbers with $+, \times$ and $-$

In Appendix C we have listed our full system for real number arithmetic with addition, subtraction and multiplication. Binary subtraction can be defined from unary minus and addition, and needs no special treatment. If we would include it, Walters' system (see Appendix B, for a version on base 10) can be derived.

The standard model for this TRS are the integers $\mathbb{R}$ together with expected mappings. As before we can easily prove that:

PROPOSITION 5.1 [CORRECTNESS] *If $t \to s$ in REALS then $t = s$ in $\mathbb{R}$.* □

PROPOSITION 5.2 *A closed term t is a normal form in INT if and only if*

$$t \equiv (\ldots((\ldots(d_1 : d_2) : \ldots) : d_n); e_1) \ldots); e_m),$$

or where

$$t \equiv -(\ldots((\ldots(d_1 : d_2) : \ldots) : d_n); e_1) \ldots); e_m),$$

the d_i's and e_j's are digits, $n > 0$ and $m \geq 0$, and $d_1 \neq 0$. □

CONJECTURE 5.3 [TERMINATION] *The TRS REALS for arithmetic with (ternary) real numbers with addition, subtraction and multiplication is terminating.*

We can prove the whole system terminating with a variant of the above proof, but for 6 ground rules for addition that have two digits in the righthand side, like $1 + -2 \to 1 : 0$. Decomposition techniques (like hierarchical combinations) seem not be adequate for the present problem.

Also open is the determination set of equations E and the set of rules R such that REALS∪R is complete modulo E. Experiments with ORME and C*i*ME, as well as by hand suggest that besides associativity and commutativity of + and × E should contain the equations:

$$\begin{aligned} -x + -y &= -(x + y) \\ x \times -y &= -(x \times y) \end{aligned}$$

Both programs seem not suited for this particular problem, as we cannot give a precedence order for the TRS, but have to order the rules interactively.

6 Future work

Beyond the scope of this paper, but within the context of infinite rewriting (cf. [KKSdV]) it is no problem to consider reals with infinite number of decimals. Correctness with respect to the standard model $\mathbb{R}$ is not a real problem:

PROPOSITION 6.1 CORRECTNESS. *For all finite and infinite terms t, s it holds that if $posREALS \models t \to^\infty s$ then $\mathbb{R} \models t = s$.*

With infinite terms computation cannot be expected to terminate, however, we conjecture the following

CONJECTURE 6.2 *Let $C[x_1, \ldots, x_n]$ be a finite term containing no occurences of : and ;. Then any instantiation $C[r_1, \ldots, r_n]$ with real numbers (ground normal form) is a strongly converging term.*

Acknowledgements Discussions with Aart Middeldorp, Vincent van Oostrom and Ken Mano are greatfully acknowledged. We thank Hans Zantema for pointing out an error in a previous draft, and we thank Pierre Lescanne and Claude Marché for providing information on their completion programs, ORME and CıME respectively.

References

[Boo91] R.V. Book, editor. *Rewriting Techniques and Applications*, number 488 in LNCS, Como, 1991. Springer-Verlag.

[CW91] D. Cohen and P. Watson. An efficient representation of aritmetic for term rewriting. In *[Boo91]*, pages 240–251, 1991.

[DJ89] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 15. North-Holland, 1989.

[Hue80] G. Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *JACM*, 27(4):797–821, 1980.

[JM83] J.P. Jouannaud and M. Munoz. Termination of a set of rules modulo a set of equations. In *Proc. 7th Conf. Automated Deduction*, pages 175–193. Napa Valley, CA, LNCS 170, Springer Verlag, 1983.

[Kir93] C. Kirchner, editor. *Rewriting Techniques and Applications*, number 690 in LNCS, Montreal, 1993. Springer-Verlag.

[KKSdV] J.R. Kennaway, J.W. Klop, M.R. Sleep, and F.J. de Vries. Transfinite reductions in orthogonal term rewriting systems. *Information and Computation*, ?? To appear.

[Klo92] J.W. Klop. Term rewriting systems. In S. Abramsky, D. Gabbay, and T. Maibaum, editors, *Handbook of Logic in Computer Science, Volume II.* Oxford University Press, 1992.

[Mar94] C. Marché. Normalised rewriting and normalised completion. In *Proc. LICS*, pages 394–403. Paris, France, IEEE Computer Society Press, 1994.

[vO94] V. van Oostrom. *Confluence for abstract and higher-order rewriting.* PhD thesis, Vrije Universiteit, Amsterdam, March 1994.

[Wal94] H.R. Walters. A complete term rewriting system for decimal integer arithmetic. Technical Report CS-9435, CWI, Amsterdam, 1994.

[Zan] H. Zantema. Termination of term rewriting by semantic labeling. *Fundamenta Informaticae*, ?? To appear.

Appendix A: The Cohen Watson system for integer arithmetic

Cohen and Watson proposed another system for integer aritmetic with place notation. They used base 4. We present it in our notation. It is modulo AC of all operators. Normal forms of the negative numbers are not as expected, e.g., normal form for -3 is $(-1) : 1$. The termination of the system is open.

$$
\begin{aligned}
0 \times x &\to 0 \\
1 \times x &\to x \\
2 \times 2 &\to 1 : 0 \\
3 \times x &\to x : (-1 \times x) \\
-1 \times -1 &\to 1 \\
2 \times -1 &\to -1 : 2 \\
(x : y) \times z &\to (x \times z) : (y \times z) \\
(x + y) \times z &\to (z \times x) + (z \times y) \\
0 + x &\to x \\
x + x &\to 2 \times x \\
1 + 2 &\to 3 \\
1 + (-1) &\to 0 \\
2 + (-1) &\to 1 \\
3 + x &\to 1 : (-1 + x) \\
(x : y) + z &\to x : (y + z) \\
(2 \times x) + x &\to 3 \times x \\
(-1 \times x) + x &\to 0 \\
(2 \times x) + (-1 \times x) &\to x \\
-1 : 3 &\to -1 \\
x : -1 &\to (-1 + x) : 3 \\
0 : x &\to x \\
x : (y : z) &\to (x + y) : z
\end{aligned}
$$

The following rules cause troubles, if we want to apply our semantic labelling.

$$
\begin{aligned}
x + x &\to 2 \times x \\
3 + x &\to 1 : (-1 + x) \\
(2 \times x) + x &\to 3 \times x \\
x : -1 &\to (-1 + x) : 3
\end{aligned}
$$

Appendix B: Walters' system for integer arithmetic

$$
\begin{array}{rcl}
0x & \to & x \\
x(yz) & \to & (x+y)z \\
x(-y(yz)) & \to & -((y-x)z) \\
1(-1) & \to & 9 \\
 & \vdots & \\
9(-9) & \to & 81 \\
x0(-1) & \to & x(-1)9 \\
 & \vdots & \\
x9(-9) & \to & x81 \\
(-x)y & \to & -(x(-y)) \\
--x & \to & x \\
-0 & \to & 0 \\
0+x & \to & x \\
x+0 & \to & x \\
1+1 & \to & 2 \\
 & \vdots & \\
9+9 & \to & 18 \\
x+yz & \to & y(x+z) \\
xy+z & \to & x(y+z) \\
x+-y & \to & x-y \\
-x+y & \to & y-x \\
0-x & \to & -x \\
x-0 & \to & x \\
1-1 & \to & 0 \\
 & \vdots & \\
9-9 & \to & 0 \\
xy-z & \to & x(y-z) \\
x-yz & \to & -(y(z-x)) \\
x--y & \to & x+y \\
-x-y & \to & -(x+y)
\end{array}
$$

Appendix C: The system REALS for real number arithmetic

$$\begin{array}{l} 0+0 \to 0 \\ 0+1 \to 1 \\ 0+2 \to 2 \\ 1+1 \to 2 \\ 1+2 \to 1:0 \\ 2+0 \to 2 \\ 2+1 \to 1:0 \\ 2+2 \to 1:1 \end{array} \qquad \begin{array}{l} 0\times 0 \to 0 \\ 0\times 1 \to 0 \\ 0\times 2 \to 0 \\ 1\times 1 \to 1 \\ 1\times 2 \to 2 \\ 2\times 0 \to 0 \\ 2\times 1 \to 2 \\ 2\times 2 \to 1:1 \end{array}$$

$$\begin{array}{rcl} x+(y:z) &\to& y:(x+z) \\ (x:y)+z &\to& x:(y+z) \\ x:(y:z) &\to& (x+y):z \\ 0:x &\to& x \end{array} \qquad \begin{array}{rcl} (x:y)\times z &\to& (x\times z):(y\times z) \\ x\times(y:z) &\to& (x\times y):(x\times z) \end{array}$$

$$\begin{array}{l} 0+-1 \to -1 \\ 0+-2 \to -2 \\ 1+-1 \to 0 \\ 1+-2 \to -1 \\ 2+-1 \to 1 \\ 2+-2 \to 0 \end{array} \qquad \begin{array}{l} -1+1 \to 0 \\ -1+2 \to 1 \\ -2+0 \to -2 \\ -2+1 \to -1 \\ -2+2 \to 0 \end{array} \qquad \begin{array}{l} -1+-1 \to -2 \\ -1+-2 \to -(1:0) \\ -2+-1 \to -(1:0) \\ -2+-2 \to -(1:1) \end{array}$$

$$\begin{array}{rcl} 0+_1(-0) &\to& 0 \\ & \vdots & \\ 9+_1(-9) &\to& 0 \\ (-0)+_1 0 &\to& 0 \\ & \vdots & \\ (-9)+_1 9 &\to& 0 \\ (-0)+_1(-0) &\to& 0 \\ & \vdots & \\ (-9)+_1(-9) &\to& -(1:_1 8) \end{array} \qquad \begin{array}{rcl} -0 &\to& 0 \\ 1:-1 &\to& 2 \\ 2:-1 &\to& 1:2 \\ 1:-2 &\to& 1 \\ 2:-2 &\to& 1:1 \\ (x:0):-1 &\to& (x:-1):2 \\ (x:0):-2 &\to& (x:-1):1 \\ (x:1):-1 &\to& (x:0):2 \\ (x:1):-2 &\to& (x:0):-1 \\ (x:2):-1 &\to& (x:1):2 \\ (x:2):-2 &\to& (x:1):-1 \end{array}$$

$$\begin{array}{rcl} -(-x) &\to& x \\ (-x):y &\to& -(x:(-y)) \\ x:(-(y:z)) &\to& -(((-x)+y):z) \\ x+(-(y:z)) &\to& -(y:((-x)+z)) \\ (-(x:y))+z &\to& -(x:((-z)+y)) \\ x\times(-(y:z)) &\to& -((x\times y):(x\times z)) \\ (-(x:y))\times z &\to& -((x\times z):(y\times z)) \end{array}$$

$$\begin{array}{rcl} x;0 &\to& 0 \\ (x;y)+z &\to& (x+z);y \\ x+(y;z) &\to& (x+y);z \\ (x;y);z &\to& x;(y+z) \\ \\ (x;y):z &\to& x:(y+z) \\ x;(y:z) &\to& (x+y);z \\ x:(y;z) &\to& (x:y);z \\ w;((x:y);z) &\to& (w+x);(y;z) \end{array} \qquad \begin{array}{rcl} (x;y)\times z &\to& (x\times z);(y\times z) \\ x\times(y;z) &\to& (x\times y);(x\times z) \\ \\ x;(-y) &\to& -((-x);y) \\ (-(x;y);z) &\to& -(x;(y+(-z))) \\ x+(-(y;z)) &\to& -(((-x)+y);z) \\ (-(x;y))+z &\to& -((x+(-z));y) \end{array}$$