Effective Continued Fractions

David Lester
Department of Computer Science, Manchester University,
Oxford Road, Manchester M13 9PL, UK.
dlester@cs.man.ac.uk

Abstract

Only the leading seven terms of a continued fraction are needed to perform on-line arithmetic, provided the continued fractions are of the correct form. This forms the basis of a proof that there is an effective *representation of the computable reals as continued fractions; we also demonstrate that the basic arithmetic operations are computable using this representation.*

1. Introduction

In this paper we show how to represent the computable reals as continued fractions *effectively*. Informally an effective continued fraction is one in which every finite initial sequence can be produced in a finite time. In practical terms this means that we have an on-line system of continued fraction arithmetic. An alternative approach to the one taken in this paper is that of Kornerup and Matula [6, 7]. Their papers present an approach that has a faster convergence to $\pm\infty$ than the one in presented in this paper, which is based on Vuillemin's work. The key idea in this paper is to show that only the leading seven terms of a continued fraction are needed to perform on-line arithmetic.

In Section 2 we define notation and review previous work on continued fractions. In Section 3 we show what goes wrong with a naïve implementation. Section 4 presents a new way to give intervals for continued fractions, but avoids the awkward problem of dealing with zeros within the representation; this defect is remedied in Section 5. We conclude with Section 6.

2. Preliminaries

2.1. Computable Reals and Effectiveness

For formal definitions of computable real numbers (denoted by $\mathbb{R}$ in this paper) and computable functions over them, see Pour-El and Richards [9]. Informally, we will say that a continued fraction representation of the computable reals is *effective*, if for all $x \in \mathbb{R}$, we can generate an arbitrary length initial subsequence of x and if we can perform basic arithmetic in an on-line manner.

Finally, we note that the floor operation is not computable.

Observation 2.1 *The function $\lfloor x \rfloor$ is not a computable function, because it is not effectively uniformly continuous when x is an integer.*

2.2. Intervals

Because the construction of continued fractions involves repeatedly taking reciprocals, it is convenient to extend the notation of rational open intervals to explicitly include $\pm\infty$, and to permit the easy calculation of the reciprocal of an interval.

Definition 2.2 *An interval $(i, s) \in \mathbb{I}$ if one of the following holds: $i, s \in \mathbb{Q}$ with $i < s$; $i \in \mathbb{Q}$ and $s = \infty$; $s \in \mathbb{Q}$ and $i = -\infty$; $i, s \in \mathbb{Q}$ with $s < i$, this is the 'interval' $(-\infty, s) \cup (i, \infty)$.*

We need to be able to perform the following operations on intervals; notice that the set $\mathbb{I}$ remains closed under the following three operations.

Definition 2.3 *With $\pm\infty + q = \pm\infty$, and $1/(\pm\infty) = 0$ we have:*

$$\begin{aligned} q + (i, s) &= (q+i, q+s) \\ -(i, s) &= (-s, -i) \\ 1/(i, s) &= (1/s, 1/i) \end{aligned}$$

There is an ordering of intervals based on their 'length'.

Definition 2.4 *We say that $(i_1, s_1) \leq (i_2, s_2)$ if, and only if*

$$\begin{aligned} & (i_1 \leq s_1 \wedge i_2 \leq s_2 \wedge s_1 - i_1 \leq s_2 - i_2) \\ \vee \quad & (i_1 > s_1 \wedge (i_2 \leq s_2 \vee i_1 - s_1 \geq i_2 - s_2)). \end{aligned}$$

0-7695-1150-3/01 $10.00 © 2001 IEEE

2.3. Continued Fractions

In [12] a continued fraction expansion for x is defined as a sequence of numbers $[x_0, x_1, \ldots, x_{n-1}, x_n, \ldots]$ with the following property:

$$x = \lim_{n\to\infty} \left(x_0 + \cfrac{1}{x_1 + \cfrac{1}{\ddots_{x_{n-1}+\frac{1}{x_n}}}} \right).$$

Each of the numbers x_i is referred to as a term. Following Hurwitz [4, 5], Gosper [2, 3], and Kornerup and Matula [6, 7] we restrict attention to continued fractions with integer terms, and use them as representations of the real numbers. The simplest are the $\mathcal{N}$-fractions. We can define $\mathcal{N}$-fractions by: characterizing their properties or specifying their construction.

Definition 2.5 *If $[x_0, x_1, \ldots, x_n, \ldots]$ is an $\mathcal{N}$-fraction then for all $i \geq 1$, we have $x_i \geq 1$. Furthermore, if the continued fraction is finite then the last term will not be* 1.

To construct the $\mathcal{N}$-fraction of a real number x, we invoke the recursive function $\mathcal{N}$ to calculate each term of the continued fraction.

Definition 2.6

$$\begin{array}{l} \mathcal{N} :: \mathbb{R} \to \mathtt{CF} \\ \mathcal{N}\, x \;=\; \mathtt{if}\ n = x\ \mathtt{then}\ [n]\ \mathtt{else}\ n : \mathcal{N}\,(1/(x-n)) \\ \qquad\quad \mathtt{where}\ n = \lfloor x \rfloor \end{array}$$

Definition 2.6 is HASKELL pseudo-code, which will be used for the algorithmic specification of the remainder of the paper. HASKELL has been used because it has streams (infinite lists) as a built-in language feature. We do have the slight conceptual problem that we would need an implementation of the computable reals ($\mathbb{R}$) to obtain a continued fraction. In [10, 11] we find a proof of the equivalence of Definitions 2.5 and 2.6. To provide examples for subsequent discussion, we now provide a few examples of $\mathcal{N}$-fractions for irrationals.

Example 2.7

$$\begin{array}{rcl} \sqrt{2} & = & [1, 2, 2, \ldots 2, \ldots] \\ \sqrt{3} & = & [1, 1, 2, 1, 2, \ldots 1, 2 \ldots] \\ e & = & [2, 1, 2, 1, \ldots 1, 2n, 1 \ldots] \end{array}$$

There are three important properties that we require of a continued fraction representation of real numbers:

1. If the continued fraction representation of the real number x has leading term n, then the 'primitive bound' is the interval within which any continued fraction with leading term n must lie, *i.e.* $x \in \mathcal{B}_P(n)$. As we will see, we use $\mathcal{B}_P$ to output the next term of a continued fraction expansion.
2. For the continued fraction representation xs of the real number x we may determine an interval for x by considering a finite number of the leading terms. We refer to this as the 'bound' of a continued fraction, and we have $x \in \mathcal{B}(xs)$. As we will see, we use $\mathcal{B}$ to determine an interval for the continued fraction inputs to our arithmetic algorithms. We would certainly expect that $\mathcal{B}[x_0, \ldots] \subseteq \mathcal{B}_P(x_0)$.
3. For the continued fraction representation xs of the real number x we may determine an interval for x (by considering a finite number of the leading terms) after discounting an initial sequence of xs. We refer to this as the 'next bound' of a continued fraction, and we have $x \in \mathcal{B}'(xs)$. What we desire is that by discounting the initial sequence we do not inadvertently widen the interval within which we could find x.

To expand on the final point, we consider the various relations between $\mathcal{B}'(xs)$ and $\mathcal{B}(xs)$ that can arise for the continued fraction representation xs of the real number x.

$\mathcal{B}'(xs) \not\subset \mathcal{B}(xs)$ If this can happen then as we process a continued fraction (*i.e.* consider longer and longer initial sequences of it) we reach a stage where the interval for x starts to get larger.

$\mathcal{B}'(xs) = \mathcal{B}(xs)$ If this can happen then as we process a continued fraction we might reach a stage where the interval for x stops getting smaller, and no further progress is made.

$\mathcal{B}'(xs) \subset \mathcal{B}(xs)$ In this case as we process the continued fraction xs the interval for x reduces.

For $\mathcal{N}$-fractions, the functions $\mathcal{B}_P$, $\mathcal{B}$, and $\mathcal{B}'$ are defined as follows:

Definition 2.8 *For $\mathcal{N}$-fractions (and accepting that the intervals returned are semi-open, rather than the open intervals implied by the use of $\mathbb{I}$)*

$$\begin{array}{lcl} \mathcal{B}_P :: \mathbb{Z} \to \mathbb{I} & & \\ \mathcal{B}_P(x_0) & = & [x_0, x_0 + 1) \\ \mathcal{B} :: [\mathbb{Z}] \to \mathbb{I} & & \\ \mathcal{B}[x_0, x_1, \ldots] & = & [x_0, x_0 + 1) \\ & = & \mathcal{B}_P(x_0) \\ \mathcal{B}' :: [\mathbb{Z}] \to \mathbb{I} & & \\ \mathcal{B}'[x_0, x_1, \ldots] & = & (x_0 + \frac{1}{x_1 + 1}, x_0 + \frac{1}{x_1}] \\ & = & x_0 + \cfrac{1}{\mathcal{B}[x_1, \ldots]} \end{array}$$

Proposition 2.9 *For all $\mathcal{N}$-fractions xs of length* 2 *or more, we have $\mathcal{B}'(xs) \subset \mathcal{B}(xs)$.*

Whenever $\mathcal{B}'(xs) \subseteq \mathcal{B}(xs)$ we will have shown that we can safely absorb terms from a continued fraction. Furthermore,

provided that the inclusion is strict ($\subset$), we will have shown that progress is made, *i.e.* the intervals are shrinking as more and more of the continued fraction is considered.

Unfortunately Proposition 2.9 is not quite true, because when $x_1 = 1$ we have a problem with the upper bound of the interval:

$$\mathcal{B}'(xs) = (x_0 + \frac{1}{2},\ x_0 + 1] \not\subset \mathcal{B}(xs) = [x_0,\ x_0 + 1).$$

Despite this, for all practical purposes successive approximations to $\mathcal{N}$-fractions have their intervals nested in the sense of Proposition 2.9. (The remedy is to observe that if $x_1 = 1$ then the continued fraction must have at least another term: no $\mathcal{N}$-fraction ends with a 1. Therefore the correct bound is the open interval: $(x_0 + 0.5,\ x_0 + 1)$).

To perform arithmetic on continued fractions, we follow Gosper [2, 3] in defining two functions which he refers to as the algebraic algorithm and the quadratic algorithm. In outline they have the following effect:

$$\mathbf{aa} :: \mathbb{Z}_{2\times 2} \to \mathbb{R} \to \mathbb{R}$$
$$\mathbf{aa} \begin{pmatrix} n_0 & n_1 \\ d_0 & d_1 \end{pmatrix} x = \frac{n_0 x + n_1}{d_0 x + d_1} \quad \text{if } n_0 d_1 \neq n_1 d_0$$

$$\mathbf{qa} :: \mathbb{Z}_{4\times 2} \to \mathbb{R} \to \mathbb{R} \to \mathbb{R}$$
$$\mathbf{qa} \begin{pmatrix} n_0 & n_1 & n_2 & n_3 \\ d_0 & d_1 & d_2 & d_3 \end{pmatrix} x\, y = \frac{n_0 xy + n_1 y + n_2 x + n_3}{d_0 xy + d_1 y + d_2 x + d_3}$$

Observe that we can perform the basic arithmetic operations ($+$, $-$, $\times$, and $\div$) with correctly initialised states:

$$x + y = \mathbf{qa} \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x\, y$$
$$x - y = \mathbf{qa} \begin{pmatrix} 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x\, y$$
$$x \times y = \mathbf{qa} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x\, y$$
$$x \div y = \mathbf{qa} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} x\, y$$

The reason that these two algorithms are important is that we can process continued fractions term-by-term by manipulating the algorithms' states.

We begin by presenting pseudo-HASKELL implementation of the algebraic algorithm.

Definition 2.10

```
aa :: Z2×2 → CF → CF
aa = aaX f where f n h xs = n : aa (aae h n) xs
```

The definitions of the auxiliary functions for `aa` are:

```
aaX :: (Z → Z2×2 → CF → CF) → Z2×2 → CF → CF
aaX f h xs =
    if ∃(n ∈ Z) • boundaa h (B(xs)) ⊆ BP(n)
    then f n h xs else aaa (aaX f) h xs
boundaa :: Z2×2 → I → I
boundaa ( n0 n1 ) x = (n0x + n1) / (d0x + d1)
        ( d0 d1 )
```

```
aae :: Z2×2 → Z → Z2×2
aae ( n0 n1 ) x = ( d0        d1      )
    ( d0 d1 )     ( n0 − d0x  n1 − d1x )
```

```
aaa :: (Z2×2 → CF → CF) → Z2×2 → CF → CF
aaa f h xs = f (foldl g h as) xs'
    where g ( n0 n1 ) x = ( n0x + n1  n0 )
            ( d0 d1 )     ( d0x + d1  d0 )
          (as, xs') = splitAt (A xs) xs
```

The HASKELL library functions `foldl` and `splitAt` are defined in Appendix A, and for $\mathcal{N}$-fractions $\mathcal{A}(xs) = 1$. The main technical difficulty is determining the transformed interval generated by

$$\mathtt{boundaa} \begin{pmatrix} n_0 & n_1 \\ d_0 & d_1 \end{pmatrix} (i,\ s).$$

Although the endpoints must be $i' = \frac{n_0 i + n_1}{d_0 i + d_1}$ and $s' = \frac{n_0 s + n_1}{d_0 s + d_1}$, the order of these endpoints is important, and this means that we must expend some effort to determine whether the interval is $(i',\ s')$ or $(s',\ i')$.

The definition of the quadratic algorithm is as follows:

Definition 2.11

```
qa :: Z4×2 → CF → CF → CF
qa h xs ys
=  if ∃(n ∈ Z)•
                bound h (B(xs)) (B(ys)) ⊆ BP(n)
   then n : qa (emit h n) xs ys
   else if select h xs ys
   then qa (absorbL h as) xs' ys
   else qa (absorbR h bs) xs ys'
   where (as, xs')  =  splitAt (A xs) xs
         (bs, ys')  =  splitAt (A ys) ys
```

Once more, for $\mathcal{N}$-fractions $\mathcal{A}(xs) = 1$. The definitions of the auxiliary functions for `qa` are:

$$\mathtt{bound} :: \mathbb{Z}_{4\times 2} \to \mathbb{I} \to \mathbb{I} \to \mathbb{I}$$
$$\mathtt{bound} \begin{pmatrix} n_0 & n_1 & n_2 & n_3 \\ d_0 & d_1 & d_2 & d_3 \end{pmatrix} x\, y = \frac{n_0 xy + n_1 y + n_2 x + n_3}{d_0 xy + d_1 y + d_2 x + d_3}$$

$$\texttt{select} :: \mathbb{Z}_{4\times 2} \to \mathbb{I} \to \mathbb{I} \to \texttt{Bool}$$

$$\texttt{select} \begin{pmatrix} n_0 & n_1 & n_2 & n_3 \\ d_0 & d_1 & d_2 & d_3 \end{pmatrix} (i_x, s_x)\ (i_y, s_y)$$

$$= I_x \geq I_y \text{ where}$$

$$I_x = \max(f\ i_y\ (i_x, s_x), f\ s_y\ (i_x, s_x))$$

$$I_y = \max(g\ i_x\ (i_y, s_y), g\ s_x\ (i_y, s_y))$$

$$f\ y = \texttt{bound}_{aa} \begin{pmatrix} n_0 y + n_2 & n_1 y + n_3 \\ d_0 y + d_2 & d_1 y + d_3 \end{pmatrix}$$

$$g\ x = \texttt{bound}_{aa} \begin{pmatrix} n_0 x + n_1 & n_2 x + n_3 \\ d_0 x + d_1 & d_2 x + d_3 \end{pmatrix}$$

$$\texttt{emit} :: \mathbb{Z}_{4\times 2} \to \mathbb{Z} \to \mathbb{Z}_{4\times 2}$$

$$\texttt{emit} \begin{pmatrix} n_0 & n_1 & n_2 & n_3 \\ d_0 & d_1 & d_2 & d_3 \end{pmatrix} x$$

$$= \begin{pmatrix} d_0 & d_1 & d_2 & d_3 \\ n_0 - d_0 x & n_1 - d_1 x & n_2 - d_2 x & n_3 - d_3 x \end{pmatrix}$$

$$\texttt{absorb}_L :: \mathbb{Z}_{4\times 2} \to [\mathbb{Z}] \to \mathbb{Z}_{4\times 2}$$

$$\texttt{absorb}_L = \texttt{foldl}\ f$$

$$\text{where } f \begin{pmatrix} n_0 & n_1 & n_2 & n_3 \\ d_0 & d_1 & d_2 & d_3 \end{pmatrix} x$$

$$= \begin{pmatrix} n_0 x + n_1 & n_0 & n_2 x + n_3 & n_2 \\ d_0 x + d_1 & d_0 & d_2 x + d_3 & d_2 \end{pmatrix}$$

$$\texttt{absorb}_R :: \mathbb{Z}_{4\times 2} \to [\mathbb{Z}] \to \mathbb{Z}_{4\times 2}$$

$$\texttt{absorb}_R = \texttt{foldl}\ f$$

$$\text{where } f \begin{pmatrix} n_0 & n_1 & n_2 & n_3 \\ d_0 & d_1 & d_2 & d_3 \end{pmatrix} y$$

$$= \begin{pmatrix} n_0 y + n_2 & n_1 y + n_3 & n_0 & n_1 \\ d_0 y + d_2 & d_1 y + d_3 & d_0 & d_1 \end{pmatrix}$$

Once again, the main technical difficulty is determining the transformed interval generated by

$$\texttt{bound}\ [n_0, n_1, n_2, n_3, d_0, d_1, d_2, d_3]\ (i_x, s_x)\ (i_y, s_y).$$

There are four possible endpoints – which are easily determined – but selecting the correct pair, and placing them in the correct order is complicated. If the algorithm is unable to emit a term, we must absorb terms from one or other of the argument continued fractions. The `select` function returns `True` if the lefthand argument (x) is to be preferred to the righthand argument (y). We aim to absorb terms from the argument which will cause the greatest reduction in the size of the overall interval.

3. What's wrong with the usual continued fractions?

Let's consider the calculation of $\sqrt{2} \times \sqrt{2}$ using the quadratic algorithm on $\mathcal{N}$-fractions. We recall that this is:

$$\texttt{qa} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} [1, 2, 2, \ldots]\ [1, 2, 2, \ldots].$$

By considering the first term of the $\mathcal{N}$-fraction for $\sqrt{2}$ we know that $\sqrt{2} \in [1, 2)$, and hence that $\sqrt{2} \times \sqrt{2} \in [1, 4)$. This interval is not small enough to allow us to emit a term, so we must absorb a term from one of the argument $\mathcal{N}$-fractions (in fact this will be from the lefthand argument). The first four intervals for $\sqrt{2}$ and $\sqrt{2} \times \sqrt{2}$ are:

$\sqrt{2}$	$\sqrt{2} \times \sqrt{2}$
[1, 2)	[1.000, 4.000)
(4/3, 3/2]	(1.777, 2.250]
[7/5, 10/7)	[1.960, 2.041)
(24/17, 17/12]	(1.993, 2.007]

No matter how many terms we consider, all we will be able to say is that the result of the calculation $\sqrt{2} \times \sqrt{2}$ will be in a semi-open interval containing 2; furthermore 2 will not be one of the endpoints. We are permitted to emit a leading term of 1 provided that our result lies in the interval [1, 2) or alternatively a leading term of 2 provided that our result lies in the interval [2, 3). Therefore no matter how many terms of the argument $\mathcal{N}$-fractions are absorbed, we will not be able to emit the first term of the answer. This is because there is no way to determine (in a finite time) whether to emit a 1 or a 2 as the leading term of our answer. Notice that this problem does not occur with finite continued fractions (*i.e.* rationals), because we will eventually reach the end of the continued fraction and then be able to determine which leading term to emit.

An alternative analysis of the problem is to note that the floor operation on an arbitrary computable real is not a computable function (see final part of Section 2), and so it was unreasonable to expect it to prove satisfactory as part of an effective representation of the computable reals. We would also expect any effective representation to have redundancy. In order to effectively compute the continued fraction of any computable real number we must:

1. replace the flooring operation of $\mathcal{N}$-fractions,
2. determine intervals within which our new continued fractions must lie given some finite initial segment of the continued fraction, and
3. demonstrate that as the continued fraction is consumed the intervals remain nested.

The reason for the third condition is that as terms are absorbed by either the algebraic or quadratic algorithm, it is no longer possible to reconstruct the original intervals.

Vuillemin proposed a solution in [10, 11] which addressed *desiderata* i in the list above. Insufficient attention was paid to the remaining two points. Firstly, in order to make his system work Vuillemin introduced a "normalization phase" which forces all intermediate results to be rationals. Secondly, from a practical point of view, the convergence is very slow (and without the the normalization could not be guaranteed).

4. Effective continued fractions: making a start

We begin the presentation of our new representation with the replacement primitive bound function $\mathcal{B}_P$. It is based on a similar function found in Vuillemin's work, with some slight changes. We follow Vuillemin and prescribe an interval within which a continued fraction lies depending on the value of its initial term. We shall call this interval the primitive bound (or $\mathcal{B}_P$) of the continued fraction.

Definition 4.1 *The primitive bound of a continued fraction* $[x_0, \ldots]$ *is* $\mathcal{B}_P(x_0)$*, where* $\mathcal{B}_P$ *is defined as:*

$$\begin{array}{lcll} \mathcal{B}_P :: \mathbb{Z} \to \mathbb{I} & & & \\ \mathcal{B}_P(x) & = & (-x+0.5, \quad x+0.5) & \text{if } x \leq -2 \\ & & (x-0.6, \quad x+0.6) & \text{if } |x| = 1 \\ & & (-0.5, \quad 0.5) & \text{if } x = 0 \\ & & (x-0.5, \quad -x-0.5) & \text{if } x \geq 2 \end{array}$$

This contrasts with Vuillemin's formulation in which

$$\mathcal{B}_P(x) = \text{if } x = 0 \text{ then } (-2, 2) \text{ else } (|x| - \frac{1}{2}, |x| + \frac{1}{2}),$$

Vuillemin's definition gives slightly wider intervals in most cases, and much wider intervals for $|x| = 1$. Notice that in both definitions of $\mathcal{B}_P$, the intervals for adjacent integers overlap; this means that for an arbitrary computable real, we can effectively compute the leading term of the continued fraction.

We can generalize the bounding operation to finding the interval associated with a continued fraction by considering its n-th primitive bound.

Definition 4.2

$$\begin{array}{l} \mathcal{B}_N :: ([\mathbb{Z}] \times \mathbb{N}) \to \mathbb{I} \\ \mathcal{B}_N([x_0, x_1, \ldots, x_{n-1}, x_n, \ldots], n) \\ \quad = x_0 + \cfrac{1}{x_1 + \cfrac{1}{\ddots_{x_{n-1} + \frac{1}{\mathcal{B}_P(x_n)}}}} \end{array}$$

We can now define our representation of continued fractions.

Definition 4.3 *An infinite sequence of integers xs is an* effective continued fraction*, written* $xs \in \mathbb{E}$*; if, and only if, there exists a computable real x such that for all* $n \in \mathbb{N}$, $x \in \mathcal{B}_N(xs, n)$*. If, in addition, all terms (except possibly the leading one) are non-zero, then xs is a* zero-free effective continued fraction*, written* $xs \in \mathbb{E}_0$.

The key part of our representation of effective continued fractions is contained in the definition of the $\mathcal{D}$ function given in Figure 1. Instead of dealing with an arbitrary number of leading terms in order to calculate a bound for the continued fraction we instead look at (at most) the first six terms of a continued fraction, and determine a bound based on these terms. To do this we use the $\mathcal{D}$ function, which returns two natural numbers, with the intended use:

1. The first component tells us how many terms we need to consider to obtain the lower bound on our interval.
2. The second component tells us how many terms we need to consider to obtain the upper bound on our interval.

Definition 4.4 *The* $\mathcal{D}$ *function is defined in Figure 1.*

We now define a bound function $\mathcal{B}$ for zero-free effective continued fractions.

Definition 4.5

$$\begin{array}{lcl} \mathcal{B} :: \mathbb{E}_0 \to \mathbb{I} & & \\ \mathcal{B}(xs) & = & (i_n, s_m) \texttt{ where } (n, m) = \mathcal{D}(xs) \\ & & \qquad\qquad\quad (i_j, s_j) = \mathcal{B}_N(xs, j) \end{array}$$

The reason for the rather complicated specification of $\mathcal{D}$ is that it captures the information about the interval formed by intersecting all of the intervals derived from the initial sequence of the continued fraction.

Lemma 4.6 *For all* $xs \in \mathbb{E}_0$*, with* $(n, m) = \mathcal{D}(xs)$

$$\mathcal{B}(xs) = \bigcap_{j=0}^{\max(n,m)} \mathcal{B}_N(xs, j)$$

Lemma 4.6 is true, by machine-assisted construction. The function $\mathcal{D}$ was constructed so that this would be true; furthermore, as perusal of Figure 1 shows, neither component returned by $\mathcal{D}$ is zero. Finally we note that, although there are sequences of integers for which $\mathcal{D}$ is not defined, the function $\mathcal{D}$ is *total* for $\mathbb{E}_0$.

Because neither component returned by $\mathcal{D}$ is zero, the leading term never contributes to the size of the interval generated by $\mathcal{B}$, and we can therefore always absorb at least the leading term.

Lemma 4.7 *For all* $xs \in \mathbb{E}_0$*, we have the strict subset property:* $\mathcal{B}(xs) \subset \mathcal{B}_N(xs, 0)$.

Notice that all of the cases of $\mathcal{D}$ the lower and upper bounds are determined by considering at least the second term x_1. This demonstrates that at least $\mathcal{B}(xs) \subseteq \mathcal{B}_N(xs, 0)$. Demonstrating strict inclusion is another laborious machine-assisted case analysis.

Definition 4.8 *For all* $xs \in \mathbb{E}_0$*, we are permitted to absorb* $\mathcal{A}(xs)$ *terms of xs, where*

$$\begin{array}{l} \mathcal{A} :: \mathbb{E}_0 \to \mathbb{N} \\ \mathcal{A}(xs) = \min(n,m) \texttt{ where } (n, m) = \mathcal{D}(xs) \end{array}$$

$$
\begin{array}{lcll}
\mathcal{D} : \mathbb{E}_0 \to (\mathbb{N} \times \mathbb{N}) & & & \\
\mathcal{D}[x_0,\ 2,\ x_2, \ldots] & = & (2,\ 2) & \text{if } |x_0| = 1 \wedge (x_2 \geq 3 \vee x_2 = 1 \vee x_2 \leq -4) \\
\mathcal{D}[x_0,\ 2,\ 2,\ x_3, \ldots] & = & (3,\ 3) & \text{if } |x_0| = 1 \wedge (x_3 \geq 1 \vee x_2 \leq -3) \\
\mathcal{D}[x_0,\ 2,\ 2,\ -2, \ldots] & = & (2,\ 3) & \text{if } |x_0| = 1 \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ x_5, \ldots] & = & (3,\ 5) & \text{if } |x_0| = 1 \wedge (x_5 = 2 \vee x_5 = -2 \vee x_5 = -3) \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ x_5, \ldots] & = & (5,\ 5) & \text{if } |x_0| = 1 \wedge (x_5 \geq 3 \vee x_5 = 1 \vee x_5 \leq -4) \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_4,\ -2, \ldots] & = & (4,\ 5) & \text{if } |x_0| = 1 \wedge x_4 \geq 3 \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_4,\ x_5, \ldots] & = & (5,\ 5) & \text{if } |x_0| = 1 \wedge x_4 \geq 3 \wedge (x_5 \geq 1 \vee x_5 \leq -3) \\
\mathcal{D}[x_0,\ 2,\ -3,\ -1, \ldots] & = & (3,\ 3) & \text{if } |x_0| = 1 \\
\mathcal{D}[x_0,\ 2,\ -3,\ x_3,\ 2, \ldots] & = & (3,\ 4) & \text{if } |x_0| = 1 \wedge x_3 \leq -2 \\
\mathcal{D}[x_0,\ 2,\ -3,\ x_3,\ x_4, \ldots] & = & (4,\ 4) & \text{if } |x_0| = 1 \wedge x_3 \leq -2 \wedge (x_4 \geq 3 \vee x_4 \leq -1) \\
\mathcal{D}[x_0,\ 2,\ 1, \ldots] & = & (2,\ 2) & \text{if } x_0 = 0 \vee x_0 \leq -2 \\
\mathcal{D}[x_0,\ 2,\ x_2,\ -2, \ldots] & = & (2,\ 3) & \text{if } (x_0 = 0 \vee x_0 \leq -2) \wedge x_2 \geq 2 \\
\mathcal{D}[x_0,\ 2,\ x_2,\ x_3, \ldots] & = & (3,\ 3) & \text{if } (x_0 = 0 \vee x_0 \leq -2) \wedge x_2 \geq 2 \wedge (x_3 \geq 1 \vee x_3 \leq -3) \\
\mathcal{D}[x_0,\ -2, \ldots,\ x_n, \ldots] & = & (j,\ i) & \text{where } (i,\ j) = \mathcal{D}[-x_0,\ 2, \ldots,\ -x_n, \ldots] \\
\mathcal{D}[x_0,\ x_1, \ldots] & = & (1,\ 1) & \text{if } (|x_0| \geq 2 \wedge |x_1| \in \{1,\ 2\} \wedge x_0 x_1 > 0) \vee |x_1| \geq 3
\end{array}
$$

Figure 1. Definition of the $\mathcal{D}$ function

We now define a function $\mathcal{B}'$ that gives the bound on the continued fraction *after* absorbing the $\mathcal{A}(xs)$ leading terms.

Definition 4.9

$$
\begin{array}{l}
\mathcal{B}' :: \mathbb{E}_0 \to \mathbb{I} \\
\mathcal{B}'[x_0,\ \ldots,\ x_a,\ \ldots] \ = \ x_0 + \cfrac{1}{\ddots_{x_{a-1}} + \cfrac{1}{\mathcal{B}\,[x_a,\ \ldots]}} \\
\qquad\qquad \text{where } a = \mathcal{A}\,[x_0,\ \ldots,\ x_a,\ \ldots]
\end{array}
$$

We are finally in a position to state our main result – Theorem 4.10 – which states that when we absorb terms, we always reduce the size of our intervals, *i.e.* as we absorb terms of a continued fraction we obtain better and better approximations to the value it represents.

Theorem 4.10 *For all $xs \in \mathbb{E}_0$, we have the strict subset property: $\mathcal{B}'(xs) \subset \mathcal{B}(xs)$.*

When the two components returned by $\mathcal{D}$ are the same, this result follows from Lemma 4.7. In the other cases, a more detailed proof is required; not surprisingly, this proof was made using machine-assistance.

5. Effective continued fractions: dealing with 0

We now address the issue we have so far avoided: how to handle continued fractions containing non-leading zeros. Following Vuillemin [10, 11], we use this to handle continued fractions that are converging to $\pm\infty$, which is needed whenever we generate a rational continued fraction from irrational continued fractions. For an alternative approach see Kornerup and Matula in [7], in which this convergence is handled in a bit-wise manner. Let us look at an example of a process that generates zeros.

Example 5.1 *Let $y = \sqrt{2}$, and assume the approximations to y are 1.4 ± 0.1 and 1.41 ± 0.01. We wish to calculate the $\mathbb{E}$-fraction for $x = y \times y$; the intervals for x are: $(1.69,\ 2.25)$, and $(1.96,\ 2.0164)$. The first approximation permits us to emit the two leading terms of an $\mathbb{E}$-fraction for x.*

$$
\begin{array}{rcll}
x & = & 2 + 1/r & r \in (4,\ \frac{100}{39}) \\
 & = & 2 + 1/(-3 + 1/r) & r \in (\frac{1}{7},\ \frac{-31}{7})
\end{array}
$$

The next interval for x allows further progress to be made, by reducing the size of the interval for r.

$$
\begin{array}{rcll}
x & = & 2 + 1/(-3 + 1/r) & r \in (\frac{-1}{22},\ \frac{164}{10492}) \\
 & = & 2 + 1/(-3 + 1/(0 + 1/r)) & r \in (-22,\ \frac{10492}{164}) \\
 & = & 2 + 1/(-3 + 1/(0 + 1/(-22 + 1/r))) & \\
 & & & r \in (\frac{164}{14100},\ \infty)
\end{array}
$$

To proceed further we would need a more accurate interval for y.

We see that the way that the terms of continued fractions of rational numbers are emitted involves the use of 0 to allow better and better approximations to be generated. In his original papers [10, 11], Vuillemin suggests that we transform continued fractions to eliminate zeros.

Lemma 5.2 *The number represented by the continued fraction $[\ldots,\ x_n,\ 0,\ x_{n+2},\ \ldots]$ is the same as that represented by $[\ldots,\ x_n + x_{n+2},\ \ldots]$.*

Unfortunately we have a problem with the intervals resulting from the use of the zero-elimination technique.

Example 5.3 *We have $[2,\ 0,\ 4,\ \ldots] \in (5.5,\ -2.5)$, whereas $[2 + 4,\ \ldots] \in (5.5,\ -6.5)$.*

As we can see, the transformed continued fraction does not have the same interval associated with it as the original one. One of the bounds is much too tightly defined, meaning that the interval for the transformed continued fraction is smaller than the original. Importantly, in Example 5.3 the original continued fraction might continue $[2, 0, 4, 0, -9, 4, \ldots]$, which lies in the interval $(\frac{-23}{7}, \frac{-19}{7})$, whereas $[6, 0, -9, 4, \ldots] \notin \mathbb{E}$. To prevent this behaviour, we restrict continued fractions so that these sequences are not permitted.

Definition 5.4 *An infinite sequence of integers xs is a* nested effective continued fraction, *written* $\mathbb{E}^+$*; if and only if,* $xs \in \mathbb{E}$*, and any subsequence:*

$$[\ldots, x_n, 0, x_{n+2}, 0, \ldots, x_{n+2m}, \ldots]$$

has the property

$$|x_n| < |x_n + x_{n+2}| < \cdots < |\sum_{i=0}^{m} x_{n+2i}|.$$

It is this controlled use of 0 in continued fractions that makes the system on-line. The convergence is potentially slow in comparison to that in Kornerup and Matulas' system [6, 7]. In their system extra bits are generated at each step; in this system the performance could be as slow as that display by:

$$[2, 0, -5, 0, 7, 0, -9, \ldots].$$

To generate the $\mathbb{E}^+$ fractions of Definition 5.4, we amend the definition of the `aa` algorithm.

Definition 5.5

$$\begin{aligned}
&\texttt{aa} = \texttt{aa}_{\texttt{X}}\ f_1 \text{ where } f_1\ n\ h\ xs = \\
&\quad \texttt{if } |n| \leq 1 \texttt{ then } n : \texttt{aa}\ (\texttt{aa}_{\texttt{e}}\ h\ n)\ xs \\
&\quad \texttt{else } n : \texttt{aa}_{\texttt{n}}\ n\ h\ xs \\
&\texttt{aa}_{\texttt{n}}\ n = \texttt{aa}_{\texttt{X}}\ f_2 \text{ where } f_2\ m\ h\ xs = \\
&\quad \texttt{if } m{=}0 \texttt{ then } m : \texttt{aa}_{\texttt{z}}\ n\ h\ xs \\
&\quad \texttt{else } m : \texttt{aa}\ (\texttt{aa}_{\texttt{e}}\ (\texttt{aa}_{\texttt{e}}\ h\ n)\ m)\ xs \\
&\texttt{aa}_{\texttt{z}}\ n = \texttt{aa}_{\texttt{X}}\ f_3 \text{ where } f_3\ m\ h\ xs = \\
&\quad \texttt{if } \exists (i \in \mathbb{Z}) \bullet 2 \leq |i| \leq |m| \wedge |n| < |n+i| \wedge \\
&\qquad \texttt{bound}_{\texttt{aa}}\ h\ (\mathcal{B}(xs)) \subseteq \mathcal{B}_P(n+i) \\
&\quad \texttt{then } i{-}n : \texttt{aa}_{\texttt{n}}\ (n{+}i)\ h\ xs \\
&\quad \texttt{else } \texttt{aa}_{\texttt{a}}\ (\texttt{aa}_{\texttt{z}}\ n)\ h\ xs
\end{aligned}$$

We also require a similar redefinition of the `qa` algorithm, with `qa` everywhere replacing `aa`. If the continued fractions are generated in this way, then we will always be able to eliminate zeros in an on-line fashion. In Example 5.3 we now emit $[2, 0, -5, 4, \ldots]$, instead of $[2, 0, 4, 0, -9, 4, \ldots]$. Unlike Vuillemin, we do not have a normalization phase that applies Lemma 5.2 indiscriminately, instead we modify the $\mathcal{D}$ function to selectively eliminate zeros.

Definition 5.6 *The* $\mathcal{D}$ *function is amended with extra clauses, as defined in Figure 2. The original clauses need to be modified so that they return the continued fraction unmodified.*

Once more we would like to show that our continued fractions have a nesting property, similar to Theorem 4.10. This necessitates a redefinition of $\mathcal{B}'$.

Definition 5.7

$$\begin{aligned}
&\mathcal{B}' :: \mathbb{E}^+ \to \mathbb{I} \\
&\mathcal{B}'[x_0, \ldots, x_a, \ldots] = \texttt{if } a{=}0 \texttt{ then } \mathcal{B}\ (xs') \texttt{ else} \\
&\qquad\qquad x_0 + \cfrac{1}{\ddots_{x_{a-1} + \cfrac{1}{\mathcal{B}_{[x_a, \ldots]}}}} \\
&\qquad \texttt{where } (n, m, xs') = \mathcal{D}[x_0, \ldots, x_a, \ldots] \\
&\qquad\qquad a = \min(n, m)
\end{aligned}$$

We state our intended result as Theorem 5.8.

Theorem 5.8 *For all* $xs \in \mathbb{E}^+$*, we have the strict subset property:* $\mathcal{B}'(xs) \subset \mathcal{B}(xs)$.

It will come as no surprise to discover this proof was machine checked.

6. Conclusion

The work presented in this paper is clearly heavily influenced by Gosper, Vuillemin, Kornerup and Matula. It is also motivated by Klaus Weierauch's challenge to demonstrate that there are computable or effective continued fractions for the reals. Personally, I feel that any implementation of exact arithmetic should be proved correct. Closely related practical work on Vuillemin's continued fractions is presented in Ménissier-Morain's thesis [8]; she also appears to have run into difficulties with this implementation of continued fractions.

There is a HASKELL implementation of exact arithmetic based on the effective continued fractions presented in this paper, which in addition includes simple transcendental functions. There are a number of practical problems with the system: it runs slowly because of all the interval calculations; it consumes a great deal of space in the homographies and bi-homographies of the `aa` and `qa` algorithms. On current architectures, the access to the continued fraction terms is far slower than to an equivalent bit-vector. For these reasons my entry to the exact arithmetic competition at CCA2000 [1], in which the only criterion was speed, was not based on continued fractions.

Thanks are due to Peter Kornerup and Warren Ferguson for their suggested improvements to this paper.

$$
\begin{array}{lcl}
\mathcal{D} : \mathbb{E}^+ \to (\mathbb{N} \times \mathbb{N} \times \mathbb{E}^+) & & \\
\mathcal{D}[x_0,\ 0,\ x_2, \ldots] & = & (0,\ 0,\ \ [x_0 + x_2,\ \ldots]) \\
\mathcal{D}[x_0,\ 2,\ 0,\ x_3,\ \ldots] & = & \mathcal{D}[x_0,\ x_3 + 2,\ \ldots] \\
\mathcal{D}[x_0,\ -2,\ 0,\ x_3,\ \ldots] & = & \mathcal{D}[x_0,\ x_3 - 2,\ \ldots] \\
\mathcal{D}[x_0,\ x_1,\ 0,\ x_3,\ \ldots] & = & (1,\ 0,\ \ [x_0,\ x_1 + x_3,\ \ldots]) \\
\mathcal{D}[x_0,\ 2,\ 2,\ 0,\ -5,\ \ldots] & = & \mathcal{D}[x_0,\ 2,\ -3,\ \ldots] \\
\mathcal{D}[x_0,\ 2,\ x_2,\ 0,\ x_4 \ldots] & = & (2,\ 0,\ \ [x_0,\ 2,\ x_2 + x_4,\ \ldots]) \\
\mathcal{D}[x_0,\ 2,\ x_2,\ -2,\ 0,\ x_5,\ \ldots] & = & \mathcal{D}[x_0,\ 2,\ x_2,\ x_5 - 2,\ \ldots] \\
\mathcal{D}[x_0,\ 2,\ -3,\ x_3,\ 0,\ x_5,\ \ldots] & = & (3,\ 0,\ \ [x_0,\ 2,\ -3,\ x_3 + x_5,\ \ldots]) \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ 0,\ x_6,\ \ldots] & = & \mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_6 + 2,\ \ldots] \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_4,\ 0,\ x_6,\ \ldots] & = & (4,\ 0,\ \ \mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_4 + x_6,\ \ldots]) \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ 2,\ 0,\ x_7 \ldots] & = & \mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ x_7 + 2,\ \ldots] \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_4,\ -2,\ 0,\ x_7 \ldots] & = & \mathcal{D}[x_0,\ 2,\ -2,\ -1,\ x_4,\ x_7 - 2,\ \ldots] \\
\mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ -3,\ 0,\ x_7 \ldots] & = & \mathcal{D}[x_0,\ 2,\ -2,\ -1,\ 2,\ x_7 - 3,\ \ldots]
\end{array}
$$

Figure 2. Definition of new clauses for the $\mathcal{D}$ function

References

[1] J. Blanck. Exact real arithmetic systems: results of competition. In Jens Blanck, Vasco Brattka, Peter Hertling, and Klaus Weihrauch, editors, *Computability and Complexity in Analysis 2000*, Berlin, 2001. Springer.

[2] R. Gosper. Continued fraction arithmetic. HAKMEM 101b, MIT, 1972.

[3] R. Gosper. Continued fraction arithmetic. Unpublished Draft Paper, 1977.

[4] A. Hurwitz. Über die Entwicklung Komplexer Grössen in Kettenbrüche. *Acta Mathematica*, 11:187–200, 1888.

[5] A. Hurwitz. Über eine besondere Art der Kettenbruch-Entwicklung reeller Grössen. *Acta Mathematica*, 12: 367–405, 1889.

[6] P. Kornerup and D. Matula. An on-line arithmetic unit for bit-pipelined rational arithmetic. *Journal of Parallel and Distributed Computing*, 5(3):310–330, 1988.

[7] P. Kornerup and D. Matula. An algorithm for redundant binary bit-pipelined rational arithmetic. *IEEE Transactions on Computers*, 39(8):1106–1115, 1990.

[8] V. Ménissier-Morain. *Arithmétique Exacte*. PhD thesis, L'Université Paris VII, Dec. 1994.

[9] M. Pour-El and J. Richards. *Computability in Analysis and Physics*. Perspectives in Mathematical Logic. Springer-Verlag, Heidelberg, 1989.

[10] J. Vuillemin. Arithmétic réelle exacte par les fractions continues. Technical Report 760, Instituit National de Recherche en Informatique et en Automatique, Domaine de Voluceau, Roquencourt, BP105, 78153 Le Chesnay Cedex, France, Nov. 1987.

[11] J. Vuillemin. Exact real computer arithmetic with continued fractions. *IEEE Transactions on Computers*, 39(8):1087–1105, Aug. 1990.

[12] H. Wall. *Analytic Theory of Continued Fractions*. Van Nostrand, Inc., 250 Fourth Avenue, New York 3, 1948.

A. HASKELL library functions

$$
\begin{array}{l}
\texttt{splitAt} :: \mathbb{N} \to [a] \to ([a],\ [a]) \\
\texttt{splitAt}\ n\ [x_0,\ x_1,\ \ldots,\ x_{n-1},\ x_n,\ \ldots] \\
\quad = \ ([x_0,\ x_1,\ \ldots,\ x_{n-1}],\ [x_n,\ \ldots])
\end{array}
$$

$$
\begin{array}{l}
\texttt{foldl} :: (a \to b \to a) \to a \to [b] \to a \\
\texttt{foldl}\ \oplus\ a\ [x_0,\ x_1,\ \ldots,\ x_n] \\
\quad = \ (\ldots((a \oplus x_0) \oplus x_1)\ldots) \oplus x_n
\end{array}
$$