Real Functions Incrementally Computable by Finite Automata

Michal Konečný

University of Birmingham, School of Computer Science, Edgbaston, Birmingham B15 2TT, UK

Abstract

This is an investigation into exact real number computation using the incremental approach of [1–4] where numbers are represented as infinite streams of digits, each of which is a Möbius transformation. The objective is to determine for each particular system of digits which functions $\mathbb{R} \to \mathbb{R}$ can be computed by a finite transducer and ultimately to search for the most finitely-expressible Möbius representations of real numbers. The main result is that locally such functions are either not continuously differentiable or equal to some Möbius transformation. This is proved using elementary properties of finite transition graphs and Möbius transformations. Applying the results to the standard signed-digit representations, we can classify functions that are finitely computable in such a representation and are continuously differentiable everywhere except for finitely many points. They are exactly those functions whose graph is a fractured line connecting finitely many points with rational coordinates.

Key words: real number computation, finite automaton, Möbius transformation, sub-self-similarity

1 Introduction

When computing a real number without knowing in advance how precise a result is needed, we generally cannot set a limit on the memory needed for the computation. Computations for which only a fixed amount of memory is required are important because the amount of fast memory in a real computer is always limited and the time cost of accessing additional memory is unbearably high. If there is no outside influence, numbers that can be computed with

Email address: M.Konecny@cs.bham.ac.uk (Michal Konečný).
URL: www.cs.bham.ac.uk/˜mik (Michal Konečný).

a fixed amount of memory are exactly those whose representation is periodic. We want to investigate a similar situation when there is one input real number and find out which functions $I \to I$, where I is the represented subset of real numbers, can be computed with a fixed amount of memory.

In this article we consider only *incremental* representations of real numbers by *streams of Möbius transformations* as studied in [1–4]. This class of representations includes the standard signed/unsigned binary and decimal digit representations.

Our main result is that a function computed by a finite automaton using such a representation is locally equal to some Möbius transformation wherever it is continuously differentiable. This complements the result by Raney [5] that using a particular simple Möbius representation of non-negative real numbers one can compute all (unary) Möbius transformations with natural coefficients.

The following two sections are very straightforward introductions to incremental representations of real numbers and finite transducers computing over them. The following Section 4 defines Möbius transformations and reviews their standard properties relevant to their role in real number representation.

In Section 5 the main result is proved for functions computable by a special class of automata in which it is possible to return to the initial state from any state by suitable input. The following Sections 6 and 7 use graph-theoretic reasoning to extend the theorem to arbitrary finite automata. Finally, in Section 8 we apply the result to the standard signed-digit representations.

2 Representation of Real Numbers

The present approach is based on representing real numbers by sequences of shrinking closed intervals that converge to a single point $x \in \mathbb{R}$:

$$I_0 \supseteq I_1 \supseteq I_2 \supseteq \ldots, \qquad \bigcap_{i \in \mathbb{N}} I_i = \{x\}.$$

We further specialise to such sequences which are representable by a sequence of functions: $f_1, f_2, f_3, \ldots$ by the rule $I_i := f_1 \circ \cdots \circ f_i(I_0)$ assuming that the largest (base) interval I_0 is fixed. These functions are picked from a certain (usually finite) set of functions that satisfy $f(J) \subseteq J$ for each $J \subseteq I_0$. I.e. they are increasing within the continuous domain of closed subintervals of I_0 ordered by reverse inclusion. Let us denote this domain by $\mathcal{R}(I_0)$.

Throughout this article we will consider only digits $f\colon \mathcal{R}(I) \to \mathcal{R}(I)$ that arise as point-wise extensions of some continuous $g\colon I \to I$: $f(J) = \{g(x) \mid x \in J\}$.

Thus we will usually define f as a (continuous) function $I \to I$ and implicitly assume that f is extended to intervals point-wise.

Definition 1 *A pair $(I, \mathcal{D})$ where I is a closed interval in $\mathbb{R}$ and $\mathcal{D}$ is a finite set of functions $D\colon I \to I$ whose extensions $D\colon \mathcal{R}(I) \to \mathcal{R}(I)$ are increasing, is called (finite)* incremental representation *of I whenever it satisfies the following:*

(1) $\bigcup_{D\in\mathcal{D}} D(I) = I$
(2) For any sequence $\{D_i\}_{i=0}^{\infty} \in \mathcal{D}^\omega$ the sequence of nested intervals $I, D_1(I), D_1 \circ D_2(I), \ldots$ limits to a singleton interval.

With each such a representation comes a function $\rho_{(I,\mathcal{D})}\colon \mathcal{D}^\omega \to I$ returning the point contained in the limit singleton interval of (2). The function is surjective because of (1).

The elements of $\mathcal{D}$ are called digits. *Often we say that $\mathcal{D}$ (instead of $(I, \mathcal{D})$) is a representation of I.*

In general, condition (2) is not easy to check, but it is not within the scope of this article to investigate which systems of digits form a representation. We will assume that we have one and prove some results about it.

All representations considered in this article use only digits that are *injective* functions. We will call such a representation injective.

Example 2 *(a) The system*

$$\{(x + d)/B \mid d = 0, 1, \ldots, B - 1\}$$

is the ordinary B-ary representation of $[0, 1]$. Each digit $(x + d)/B$ is a bijection $[0, 1] \to [d/B, (d + 1)/B]$.

(b) The system

$$\{(x + d)/B \mid d = -(B - 1), -(B - 2), \ldots, -1, 0, 1, \ldots, B - 1\}$$

is the signed-digit (also called redundant) B-ary representation of $[-1, 1]$. Here, a digit $(x + d)/B$ is a bijection $[-1, 1] \to [(d - 1)/B, (d + 1)/B]$. Each two consecutive digits have ranges overlapping by half of their size.

Not every representation of I in the sense of the previous definition gives rise to the same set of computable functions on I. Especially, the ordinary B-ary representation does not allow one to compute some of the most basic functions like addition or multiplication. On the other hand, the signed B-ary representations are equivalent to all other sensible notions of computability over $\mathbb{R}$ considered in the literature (see [6]).

Thus a realistic representation should be computationally equivalent to the signed-digit representations. Nevertheless, for the purposes of this article it is sufficient to distinguish representations which are redundant in the following sense (see [7]):

Definition 3 *An incremental representation* $(I, \mathcal{D})$ *is called* redundant *if for every* $x \in \operatorname{Int} I$ *there is a digit* D *with* $x \in \operatorname{Int} D(I)$*.*

The standard binary representation is not redundant because the point 1/2 is an endpoint of the ranges of both digits 0,1.

In a redundant representation every point can be represented by a sequence of intervals each of which contains the point in its interior and therefore:

Lemma 4 *Let* $(I, \mathcal{D})$ *be a redundant representation and* $J \subseteq I$ *an open neighbourhood of* $x \in \operatorname{Int} I$*. Then there is some finite input* $P \in \mathcal{D}^*$ *such that* $x \in \operatorname{Int} P(I) \subset J$*.*

3 Finite Transducers

Assuming that we have an incremental digit representation for real numbers, we go on to formalise computation of unary real functions with finite memory. We choose a deterministic, input-driven model for simplicity. This choice does not harm generality in the study of finite state computability (see e.g. [8]).

Definition 5 *A (deterministic, non-terminating) finite transducer* $\mathcal{A}$ *is a tuple* $(Q, \mathit{In}, \mathit{Out}, \tau, q_S)$ *where*

- Q *is a finite set of* states
- In*,* Out *are finite* input *and* output alphabets
- τ *is a one-step* transition and output function $\tau = (\tau_1, \tau_2)\colon Q \times \mathit{In} \to Q \times \mathit{Out}^*$
- $q_S \in Q$ *is the* initial state.

For $\mathcal{A}$ *we define functions* $\mathrm{state}_{\mathcal{A}}\colon \mathit{In}^* \to Q$ *and* $\mathrm{output}_{\mathcal{A}}\colon \mathit{In}^* \to \mathit{Out}^*$ *inductively as follows (using* ϵ *to denote the empty sequence):*

- $\mathrm{state}_{\mathcal{A}}(\epsilon) = q_S$, $\mathrm{state}_{\mathcal{A}}(P\alpha) = \tau_1(\mathrm{state}_{\mathcal{A}}(P), \alpha)$ *and*
- $\mathrm{output}_{\mathcal{A}}(\epsilon) = \epsilon$, $\mathrm{output}_{\mathcal{A}}(P\alpha) = \mathrm{output}_{\mathcal{A}}(P) \cdot \tau_2(\mathrm{state}_{\mathcal{A}}(P), \alpha)$ *for* $P \in \mathit{In}^*, \alpha \in \mathit{In}$*.*

The function $\mathrm{output}_{\mathcal{A}}$ *can also be uniquely continuously extended to infinite inputs/outputs:* $\mathrm{output}_{\mathcal{A}}\colon \mathit{In}^\infty \to \mathit{Out}^\infty$*. Note though that it is not true that infinite input must produce infinite output.*

Definition 6 *Let $\mathcal{A} = (Q, In, Out, \tau, q_S)$ be a finite transducer. The* underlying graph $\mathcal{G}_\mathcal{A}$ *of $\mathcal{A}$ is defined by*

$$\mathcal{G}_\mathcal{A} = \left(Q, \bigcup_{\alpha \in In} \{(x, \tau_1(x, \alpha)) \mid x \in Q\}\right)$$

i.e. the nodes are states and an edge between A and B means that there is some input symbol that switches the machine from state A to state B (the usual transition graph but unlabelled).

Note that from each node of $\mathcal{G}_\mathcal{A}$ leads at least one edge.

Definition 7 *A finite transducer* uses representation $(I, \mathcal{D})$ *if both its input and output alphabets are equal to $\mathcal{D}$.*

Such a finite transducer $\mathcal{A} = (Q, \mathcal{D}, \mathcal{D}, \tau, q_S)$ computes *a function $f \colon I \to I$ if the following condition holds: For any infinite input $P \in \mathcal{D}^\omega$ also $O :=$* $\mathrm{output}_\mathcal{A}(P)$ *is infinite and $\rho_{(I,\mathcal{D})}(O) = f(\rho_{(I,\mathcal{D})}(P))$.*

In Figure 1 there are some examples of finite transducers using the redundant binary representation that compute functions $-x$, $x/2$, and $|x|$.

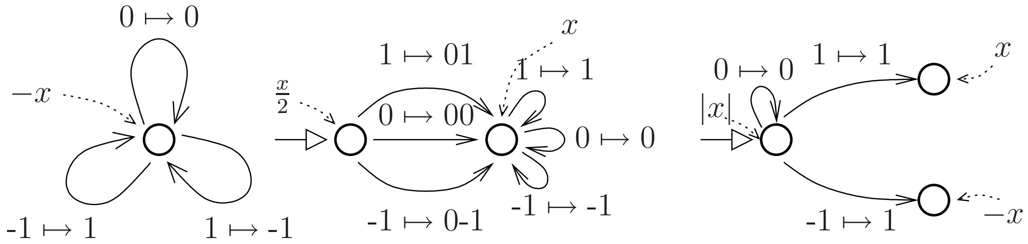


Fig. 1. Simple examples of finite transducers

These automata are drawn as transition graphs where each edge is labelled by a pair $a \mapsto b$ where a is an input symbol and b the corresponding output sequence. The first transducer changes each 1 into a -1 and each -1 into 1. The second one adds a 0 in front of the input digit sequence. The last automaton flips all -1 and 1 as the first one if the first non-0 digit is -1. It copies the input without any change otherwise.

Lemma 8 *For a finite transducer $\mathcal{A}$ the following are equivalent:*

(1) For any infinite input $P \in \mathcal{D}^\omega$ also the $\mathrm{output}_\mathcal{A}(P)$ *is infinite.*
(2) $\mathcal{A}$ does not contain a cycle with empty output. Formally:

$$(\forall P_1, P_2 \in \mathcal{D}^*) \\ \left(\mathrm{state}_\mathcal{A}(P_1) = \mathrm{state}_\mathcal{A}(P_1 P_2) \implies \mathrm{output}_\mathcal{A}(P_1) \neq \mathrm{output}_\mathcal{A}(P_1 P_2)\right)$$

PROOF. (1) $\implies$ (2): If there is an empty cycle, i.e. P_1, P_2 with $\mathrm{state}_{\mathcal{A}}(P_1) = \mathrm{state}_{\mathcal{A}}(P_1P_2)$ and $\mathrm{output}_{\mathcal{A}}(P_1) = \mathrm{output}_{\mathcal{A}}(P_1P_2)$, then on the infinite input $P_1P_2P_2\dots$ the transducer produces only finite output $\mathrm{output}_{\mathcal{A}}(P_1)$, because $\mathrm{output}_{\mathcal{A}}(P_1) = \mathrm{output}_{\mathcal{A}}(P_1P_2) = \mathrm{output}_{\mathcal{A}}(P_1P_2P_2)$, etc.

(2) $\implies$ (1): Let $P = D_{s_1}D_{s_2}\dots$ where each D_{s_i} is a digit. Denote $q_n := \mathrm{state}_{\mathcal{A}}(P_n)$ where $P_n := D_{s_1}\dots D_{s_n}$. There is a state q such that $q = q_n$ for infinitely many indices n. Pick an infinite increasing sequence m_i of such indices. Now, by (1), we have an infinite growing sequence $\mathrm{output}_{\mathcal{A}}(P_{m_1}) \neq \mathrm{output}_{\mathcal{A}}(P_{m_2}) \neq \dots$ of prefixes of $\mathrm{output}_{\mathcal{A}}(P)$, so $\mathrm{output}_{\mathcal{A}}(P)$ must be infinite. $\square$

As we said before, in all representations considered in this article the digits are point-wise extensions of *injective* functions $D\colon I \to I$. Such a function is also a bijection $I \to D(I)$ and has an inverse $D^{-1}\colon D(I) \to I$.

Lemma 9 *Let $\mathcal{D}$ be an injective representation. If $\mathcal{A} = (Q, \mathcal{D}, \mathcal{D}, \tau, q_S)$ computes function f, then for any reachable state $q = \mathrm{state}_{\mathcal{A}}(P)$, $P \in \mathcal{D}^*$ the automaton $\mathcal{A}_q := (Q, \mathcal{D}, \mathcal{D}, \tau, q)$ computes $f_q := \mathrm{output}_{\mathcal{A}}(P)^{-1} \circ f \circ P$.*

PROOF. Suppose $\mathcal{A}_q$ outputs V on input $U \in \mathcal{D}^\omega$. Thus $\mathcal{A}$ outputs OV on input PU, where $O = \mathrm{output}_{\mathcal{A}}(P)$. As $\mathcal{A}$ computes f, we have that OV is infinite and $f(\rho(PU)) = \rho(OV)$. From this it follows that V is infinite and $\rho(V) = O^{-1}(\rho(OV)) = O^{-1} \circ f(\rho(PU)) = (O^{-1} \circ f \circ P)(\rho(U))$ which means that $\mathcal{A}_q$ computes $O^{-1} \circ f \circ P$. $\square$

Justified by this lemma we will freely use the term "function computed by $\mathcal{A}$ from state q". The following is a kind-of opposite to the preceding lemma:

Lemma 10 *Let $\mathcal{A} = (Q, \mathcal{D}, \mathcal{D}, \tau, q_S)$ be a finite transducer using a representation $\mathcal{D}$ of I. If there is an assignment of functions f_q to the states of $\mathcal{A}$ which fulfils*

$$f_q \circ D = \tau_2(q, D) \circ f_{\tau_1(q,D)}$$

for every $q \in Q$, $D \in \mathcal{D}$, and $\mathcal{A}$ does not contain a cycle with empty output, then $\mathcal{A}$ computes f_{q_S}.

PROOF. Take any input $U = D_{s_1}D_{s_2}\dots \in \mathcal{D}^\omega$ and denote $q_n := \mathrm{state}_{\mathcal{A}}(P_n)$ where $P_n := D_{s_1}\dots D_{s_n}$. Let E_n be the output made during inputting n-th digit of the sequence: $\mathrm{output}_{\mathcal{A}}(P_n) = E_1 \dots E_n$ for all $n \in \mathbb{N}$. Now we have $f_{q_0} \circ D_{s_1} = E_1 \circ f_{q_1}$, as well as $f_{q_{n-1}} \circ D_{s_n} = E_n \circ f_{q_n}$ for all n. By induction it follows that $E_1 \circ \dots \circ E_n \circ f_{q_n} = f_{q_0} \circ P_n = f_{q_S} \circ P_n$, especially $\mathrm{output}_{\mathcal{A}}(P_n)(I) \supseteq$

$f_{q_S}(P_n(I))$. For $n \to \infty$ both $P_n(I)$ and $\text{output}_{\mathcal{A}}(P_n)(I)$ monotonously converge to singleton sets by the definition of incremental representation and previous lemma. Therefore $\rho_{\mathcal{D},I}(\text{output}_{\mathcal{A}}(U)) = f_{q_S}(\rho_{\mathcal{D},I}(U))$. $\square$

This lemma justifies the preceding examples and later will even serve as a basis to construct transducers for some more complicated functions.

A special case of Lemma 9 is when $q = q_S$ for a non-empty cycle P. Then, of course, $f_q = f$ and f must satisfy a very strong condition: $f = \text{output}_{\mathcal{A}}(P)^{-1} \circ f \circ P$. In case the digits are affine functions of the same shrinking factor, this equation means that a certain subsection of the graph of f is similar to the whole graph (see Figure 2).

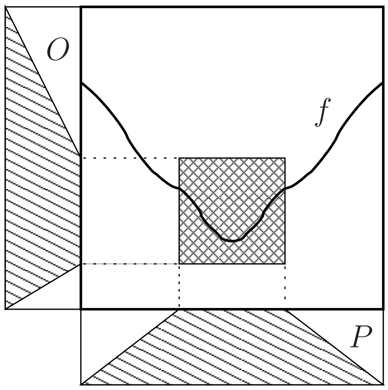


Fig. 2. Sub-self-similarity

Therefore, we refer to this situation as *sub-self-similarity*. (The prefix "sub-" is used to avoid confusion with the well-established self-similarity in fractal theory which would correspond to a situation where the graph is covered by several sub-self-similarities.)

Definition 11 *An automaton $\mathcal{A}$ is called strongly connected if its underlying graph $\mathcal{G}_{\mathcal{A}}$ is strongly connected (i.e. every two nodes are connected by some path).*

In strongly connected automata we can find a "dense" set of sub-self-similarities because after arbitrary input there is some possible input that returns the automaton to its initial state, thus getting a cycle P with $P(I) \subseteq J$ for an arbitrary open interval $J \subseteq I$.

4 Linear Fractional Transformations

In this article we only investigate incremental representations, the digits of which are functions of the following kind. Informally, linear fractional transformations (shortly lft's, also called Möbius transformations) are unary functions of the form $(ax+c)/(bx+d)$ with one variable x and four coefficients a, b, c, d all from the same field. Most often it is the field of complex numbers, but for the sake of representing real numbers we will consider lft's over real numbers. Majority of the material in this section is standard, see e.g. [9]. For a wider study of lft's as digits see e.g. [1,10].

Lft's can be generalised to any number of variables as a fraction of two polynomials that are linear in each of the variables. Thus, nullary lft's are constants a/b, binary lft's are functions of the form $(axy+cy+ex+g)/(bxy+dy+fx+h)$, etc.

Any lft is invariant under scaling of its coefficients by a non-zero constant. Thus, as the form of nullary lft's suggests, it is convenient to think of nullary lft's as points in the projective line. Moreover, lft's of any arity n are n-ary linear maps over $\mathbb{R}^2$ with all domains and the codomain factored to the projective line:

$$\begin{array}{ccc} \mathbb{R}^* \times \cdots \times \mathbb{R}^* & \xrightarrow{\text{lft } L} & \mathbb{R}^* \\ \uparrow\!\!\uparrow & & \uparrow\!\!\uparrow \\ \mathbb{R}^2 \times \cdots \times \mathbb{R}^2 & \xrightarrow{\text{linear map } \phi} & \mathbb{R}^2 \end{array}$$

For this reason nullary lft's correspond to vectors and unary lft's to matrices. Analogously, lft's of bigger arity correspond to tensors of type $2, 2, \ldots, 2$ $(n\times)$ (as observed in [11,4,12]).

Computing with real numbers itself also requires to include infinity because division by zero cannot be avoided as there is no algorithm to decide whether an arbitrary real number is 0.

Definition 12 *Let $\mathbb{R}^* = \mathbb{R} \cup \{\infty\}$ be the projective line over $\mathbb{R}$ also called the* extended reals*.*

Addition, opposite, multiplication and inverse can be extended for $\mathbb{R}^*$ apart from $0 \cdot \infty$ and $\infty + \infty$ by

$$\begin{array}{l} r \neq \infty \implies r + \infty = \infty, \quad -\infty = \infty, \\ r \neq 0 \implies r \cdot \infty = \infty \quad \textit{and} \quad 0^{-1} = \infty \end{array}$$

Definition 13 *A* unary real lft *with coefficients $a, b, c, d \in \mathbb{R}$ ($\{b, d\} \neq \{0\}$) is the function $x \mapsto (ax+c)/(bx+d)$: $\mathbb{R}^* \to \mathbb{R}^*$ continuously extended to the (isolated) points where the expression might be undefined. Lft's with $b = 0$ are called* affine *transformations.*

Lft's are denoted by the matrix notation $\left\langle\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)\right\rangle$. *An lft* $\langle A\rangle$ *is called* regular *if* $\det A \neq 0$, *singular otherwise.*

The formula $(ax+c)/(bx+d)$ fails only if both its numerator and denominator are equal to ∞ or 0. The first case can happen only for $x=\infty$, but $\lim_{x\to\infty}(ax+c)/(bx+d)$ exists and is equal to a/b. The second case can happen only for a singular lft and a unique $x=-c/a=-d/b$. Such an lft is a constant function $x \mapsto a/b$.

Because of the previously mentioned fact that lft's are linear maps $\mathbb{R}^2 \to \mathbb{R}^2$ factored to $\mathbb{R}^* \to \mathbb{R}^*$, it holds $\langle A\rangle \circ \langle B\rangle = \langle A \cdot B\rangle$. Therefore, in the context of lft's we will write $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ instead of $\left\langle\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)\right\rangle$ and consider matrices equivalent up to scaling by a non-zero real constant.

Lemma 14 *Every unary lft* $\left\langle\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)\right\rangle : \mathbb{R}^* \to \mathbb{R}^*$ *is a continuous function with continuous derivative* $(ad-bc)/(bx+d)^2 \colon \mathbb{R}^* \to \mathbb{R} \cup \{+\infty, -\infty\}$ *on the whole of* $\mathbb{R}^*$. *Every regular lft is a bijection.*

PROOF. To check the formula in $\mathbb{R}$ we can apply the well-known derivative formulae. The derivative of $f(x)=(ax+c)/(bx+d)$ in ∞ is defined as the limit

$$\lim_{x\to\infty} \frac{f(x)}{x} = \lim_{x\to\infty} \frac{ax+c}{bx^2+dx} = 0$$

and coincides with the value of $(ad-bc)/(bx+d)^2$ in ∞. $\square$

Especially, every singular lft is a constant function and every regular lft is a strictly increasing or decreasing function depending on the sign of its determinant. (Although an lft doesn't have a unique determinant due to coefficient scaling, the sign of the determinant is scaling-independent.) The derivative of a regular non-affine lft is 0 only in ∞ and is $+\infty$ or $-\infty$ only in the point where the value itself is ∞. The derivative of a regular affine lft is constant and different from 0 and $\pm\infty$.

Given a closed interval I of real numbers we want to use certain lft's as digits to represent elements of I. Let us refer to the class of lft's that can be used as digits over I as $\mathrm{LFT}(I)$ and any incremental representation using exclusively lft's as digits as an *lft-representation.*

We will not define $\mathrm{LFT}(I)$ precisely as we do not construct representations in this article. We only expect to have an lft-representation and prove results about it. Nevertheless, we need to observe certain properties common to all $L \in \mathrm{LFT}(I)$. First of all, the sequence $I, L(I), L^2(I), \ldots$ converges to a singleton set, say $\{\ell\}$, and thus L has a fixpoint $\ell \in I$.

A digit L which is not singular has an inverse L^{-1} fulfilling $L^{-1}(\overline{\mathbb{R}^* \setminus I}) \subseteq \overline{\mathbb{R}^* \setminus I}$. By Brower's fixpoint theorem, L^{-1} has got a fixpoint $\ell' \in \overline{\mathbb{R}^* \setminus I}$. It is also a fixpoint of L.

Singular digits make the rest of the digit sequence irrelevant, thus allowing a countable subset of finitely representable real numbers. We will not consider singular digits in this article and assume that $\mathrm{LFT}(I)$ contains only regular lft's. The author believes that extending the results in this article for representations with singular digits is possible and in principle not difficult.

Fixpoints of lft's correspond to eigenvectors of their matrices, so an arbitrary non-identity lft cannot have more than two fixpoints. Thus a regular $L \in \mathrm{LFT}(I)$ has either two fixpoints, one in $\mathrm{Int}\, I$ and one in $\mathbb{R}^* \setminus I$, or one fixpoint coinciding with an endpoint of I. Moreover, the fixpoint $\ell \in I$ of L is attracting within I and the fixpoint $\ell' \in \overline{\mathbb{R}^* \setminus I}$ of L is repelling within $\overline{\mathbb{R}^* \setminus I}$.

Table 1 summarises types of lft's according to the number and type of their fixpoints. We can see there that only hyperbolic (affine/non-affine) and parabolic lft's can be used as regular digits in incremental representations.

Lemma 15 *A hyperbolic lft A and its attracting fixpoint u fulfil $|A'(u)| < 1$.*

PROOF. Let $A = \left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ scaled so that $a + d > 0$ so that

$$u = \left(a - d + \sqrt{(a+d)^2 - 4D}\right) / 2b.$$

Now

$$A'(u) = \frac{D}{\left(b\frac{a-d+\sqrt{(a+d)^2-4D}}{2b} + d\right)^2} = \frac{4D}{\left(a + d + \sqrt{(a+d)^2 - 4D}\right)^2}$$

Let us distinguish two cases (a) $D < 0$, (b) $D > 0$:

(a) We need to show $A'(u) > -1$ which is equivalent to

$$4D > -2(a+d)^2 + 4D - 2(a+d)\sqrt{(a+d)^2 - 4D}$$

and this to

$$(a+d)^2 + (a+d)\sqrt{(a+d)^2 - 4D} > 0$$

which is trivial because $a + d > 0$.

(b) Here it suffices to show $A'(u) < 1$. By equivalent transformations:

$$4D < 2(a+d)^2 - 4D + 2(a+d)\sqrt{(a+d)^2 - 4D}$$

Class	Coefficients	Fixpoints	Type
singular	$D=0$	a/b	attracting
identity	$b,c=0,\ a=d$	$\mathbb{R}^*$	non-attracting, non-repelling
square root of identity	$a+d=0$	0,1 or 2	non-attracting non-repelling
elliptic	$(a+d)^2-4D<0$	none	—
parabolic	$(a+d)^2=4D$	$(a-d)/2b$	
	$a+d>0$		left-repelling, right-attracting
	$a+d<0$		right-repelling, left-attracting
hyperbolic affine	$(a+d)^2-4D>0,\ b=0$	$\infty, c/(d-a)$	
	$a>d$		attracting, repelling
	$a<d$		repelling, attracting
hyperbolic	$(a+d)^2-4D>0,\ b\neq 0$	$\frac{a-d\pm\sqrt{(a+d)^2-4D}}{2b}$	
	$a+d>0$		+attracting, −repelling
	$a+d<0$		+repelling, −attracting

Table 1
Classification of lfts

and

$$(a+d)^2-4D+(a+d)\sqrt{(a+d)^2-4D}>0$$

is again trivial because $(a+d)^2-4D>0$ as well as $a+d>0$. $\square$

Lemma 16 *Let x,y,z be 3 different points in $\mathbb{R}^*$ and $x',y',z'\in\mathbb{R}^*$. Then there is a unique lft f with $f(x)=x'$, $f(y)=y'$, and $f(z)=z'$.*

Corollary 17 *For any two closed non-empty intervals I,I' there exists an increasing lft mapping I onto I'.*

Any such mapping translates between computations over I and computations over I':

Lemma 18 *Let I,I' be two closed intervals of non-zero length and L be some lft with $L(I')=I$ and let $\mathcal{A}$ be a finite transducer using an lft-representation $\mathcal{D}=\{D_1,\dots,D_n\}$ of I computing $f\colon I\to I$. Then*

$$\mathcal{D}_L := \{L^{-1}D_1L,\dots,L^{-1}D_nL\}$$

is a representation of I' and the finite transducer $\mathcal{A}_L$ which differs from $\mathcal{A}$ only by substituting each D_i with $L^{-1}D_iL$ computes the function $L^{-1}\circ f\circ L\colon I'\to I'$.

5 Strongly Connected Automata

Recall that a cycle in an automaton induces a strong condition of sub-self-similarity on the functions computed from states on the cycle (cf. remarks following Lemma 10). The following proposition states the crucial property of functions sub-self-similar by Möbius transformations.

Proposition 19 *Let $f\colon I\to I$ be a continuous function on some closed interval $I\subseteq\mathbb{R}$, and $P,O\in\mathrm{LFT}(I)$ with fixpoints $p,o\in\mathrm{Int}\,I$ and $r,q\in\mathbb{R}^*\setminus I$, respectively, such that $f\circ P=O\circ f$.*

Then $f(p)=o$ and if $f'(p)$ exists and is different from 0 and $\pm\infty$, f is an lft on I. Moreover,

$$f=T_O^{-1}\begin{pmatrix} s\cdot f'(p) & 0\\ 0 & 1\end{pmatrix}T_P,$$

where $s=\mathrm{sign}(\mathrm{Det}(O)\cdot\mathrm{Det}(P))$ and T_P,T_O are instances of:

$$T_A=\begin{cases}\begin{pmatrix} u-v & -u(u-v)\\ 1 & -v\end{pmatrix} & \text{if } v\neq\infty\\ \begin{pmatrix} 1 & -u\\ 0 & 1\end{pmatrix} & \text{if } v=\infty\end{cases}$$

for any lft $A\in\mathrm{LFT}(I)$ with fixpoints $u\in\mathrm{Int}\,I$ and $v\in\mathbb{R}^*\setminus I$.

PROOF. Firstly, $O(f(p))=f(P(p))=f(p)$, so $f(p)$ is a fixpoint of O in I and thus must be equal to o.

To prove the other claim, let us evaluate the following expression that by definition of derivative converges to $f'(p)$ for any $x\neq p$, $f(x)\neq o$:

$$f'(p)=\lim_{n\to\infty}\frac{f(P^n(x))-o}{P^n(x)-p}=\lim_{n\to\infty}\frac{O^n(f(x))-o}{P^n(x)-p}$$

We proceed by expressing $A^n(x)-u$ for an arbitrary lft A with an attracting fixpoint $u\in I$ and a repelling fixpoint $v\in\overline{\mathbb{R}^*\setminus I}$, $v\neq u$. The function $A^n(x)-u$ is an lft L_n characterised by the following properties: $L_n(u)=0$, $L_n(v)=v-u$, $L_n'(u)=k^n$ where k is the derivative $A'(u)$. In terms of the coefficients a,b,c,d of L_n we get

(1) $au+c=0$ (2) $(av+c)/(bv+d)=v-u$ (3) $ad-bc=(bu+d)^2k^n$

Let us distinguish two cases: (a) $v\neq\infty$ and (b) $v=\infty$.

(a) Applying (1) to (2) we get: $a(v-u) = (v-u)(bv+d)$ i.e. $a = bv+d$. Using this and (1) on (3) gives us: $ad-bc = a(d+ub) = (bv+d)(bu+d) = (bu+d)^2k^n$ and thus $bv+d = (bu+d)k^n$ because $bu+d$ can't be 0 as $L^n(u) = 0$. Knowing the ratio of b and d we determine L_n completely:

$$L_n(x) = \frac{k^n(v-u)x - k^n(v-u)u}{(k^n-1)x + v - uk^n}$$

(b) In this case condition (2) becomes $b = 0$ and (3) becomes $a = dk^n$. Thus

$$L_n(x) = k^n(x-u).$$

Now we will apply this to O, P with their attracting fixpoints o, p and repelling fixpoints q, r, respectively, and $k_O = |O'(o)|$, $k_P = |P'(p)|$. Remember that $k_O, k_P < 1$ by Lemma 15. Thus we get:

$$s \cdot f'(p) =$$
$$\lim_{n\to\infty} \frac{k_O^n \dfrac{(q-o)f(x) - (q-o)o}{(k_O^n - 1)f(x) + q - ok_O^n}}{k_P^n \dfrac{(r-p)x - (r-p)p}{(k_P^n-1)x + r - pk_P^n}} = \left(\lim_{n\to\infty} \frac{k_O^n}{k_P^n}\right) \cdot \frac{\dfrac{(q-o)f(x) - (q-o)o}{-f(x)+q}}{\dfrac{(r-p)x - (r-p)p}{-x+r}}$$

for $r \neq \infty \neq q$. If $k_P \neq k_O$, the above expression is either equal to $0, \pm\infty$, or undefined which was excluded in the text of the lemma. Therefore, $k_P = k_O$ and $s \cdot f'(p) = T_O(f(x))/T_P(x)$.

If $r = \infty$ or $q = \infty$ or both, it can be checked routinely that again $s \cdot f'(p) = T_O(f(x))/T_P(x)$. This equation can be also rewritten to $\left(\begin{smallmatrix} s\cdot f'(p) & 0 \\ 0 & 1 \end{smallmatrix}\right) \circ T_P = T_O \circ f$ which is equivalent to $f = T_O^{-1} \circ \left(\begin{smallmatrix} s\cdot f'(p) & 0 \\ 0 & 1 \end{smallmatrix}\right) \circ T_P$. It holds for all points $x \neq p$ where $f(x) \neq o$. Since f is continuous and the right hand side is equal to o only for input value p, either $f(x) = o, \forall x \in I$ or the equation above holds for all $x \in I$. But $f'(p) \neq 0$ disables the former option. $\square$

We are now ready to prove the main theorem for strongly connected automata.

Theorem 20 *Let $\mathcal{D}$ be an lft-representation of I. Any function $f\colon I \to I$ continuously differentiable on some open subinterval $J \subseteq I$ and computed by a strongly connected finite transducer $\mathcal{A}$ over $\mathcal{D}$ is an lft.*

PROOF. Take some input $P_1 \in \mathcal{D}^*$ with $P_1(I) \subseteq J$. Because $\mathcal{G}_\mathcal{A}$ is strongly connected there is some possible further input $P_2 \in \mathcal{D}^*$ such that $\text{state}_\mathcal{A}(P_1P_2) = q_S$, the initial state. Denoting $P := P_1P_2$ and $O := \text{output}_\mathcal{A}(P)$ we have $f \circ P = O \circ f$.

Firstly, f has continuous derivative on the whole of I because existence and continuity of a derivative in $x \in I$ translates by sub-self-similarity from $P(x) \in J$:

$$f'(x) = f'(P(x))P'(x)/O'(f(x)).$$

(Neither of $P'(x), O'(f(x))$ is 0 or infinite because P and O are lft's mapping I to I and $\infty \notin I$.)

By Lemma 19 and existence of a derivative on I we get that either

(1) in any subinterval of non-zero length $J \subseteq I$ there is a point with derivative 0 or $\pm\infty$ (i.e. a fixpoint of sub-self-similarity), or
(2) that f is an lft on I.

Note that the sub-self-similarity lft's can be avoided to have just one fixpoint (to accommodate the conditions of Lemma 19) by choosing P_1 as above but with $P_1(I)$ not including an endpoint of I.

In the case (1) we get that there is a dense subset of I where the derivative is 0 or $\pm\infty$. By continuity of the derivative of f, f must be a constant function. So in any case, f is an lft on I. $\square$

Notice that in the course of proving Lemma 19 we established that the lft's P, O that form the sub-self-similarity must have the same absolute value of derivative in their attracting fixpoints. Otherwise, the derivative $f'(p)$ would have to be 0 or $\pm\infty$ provided that it exists. This observation can be supported by a naive reflection on the speed of convergence: Suppose that the sequence $O(I), O^2(I), \ldots$ converges to the singleton set significantly faster than the sequence $P(I), P^2(I), \ldots$ and it holds $O \circ f = f \circ P$. Then the graph of f is contained in the rectangle $P^n(I) \times O^n(I)$ for every n. But this rectangle gets flatter and flatter with increasing n therefore the derivative of f in p should be 0 which is illustrated in Figure 3. If this intuition is correct, we should not need to assume the existence of a derivative and prove the same result. In fact, we have to work a bit harder to make the most of it:

Lemma 21 *Let $f\colon I \to I$ be a continuous function on some closed interval $I \subseteq \mathbb{R}$, and $P, O \in \mathrm{LFT}(I)$ with fixpoints $p, o \in I$, respectively, such that $f \circ P = O \circ f$.*

(1) If $|P'(p)| > |O'(o)|$, then $f'(p) = 0$.
(2) If $|P'(p)| < |O'(o)|$, either
$f'(p) = 0$ (if f is a constant function: $f(x) = o, \forall x \in I$), or
$f'(p)$ does not exist (if $f(x) = o$ for some $x \neq p$ but also $f(x) \neq o$ for some x), or
$f'(p) \in \{+\infty, -\infty\}$ (if $f(x) = o \iff x = p$).

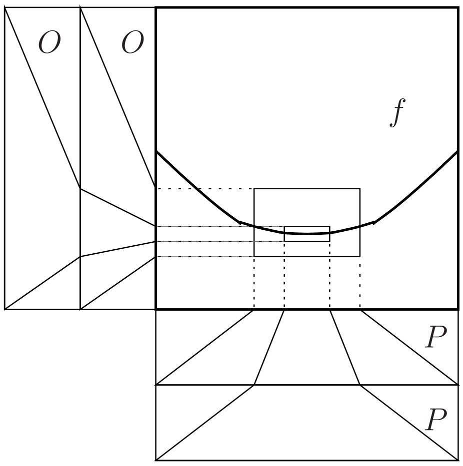


Fig. 3. Output converging faster than input

PROOF. For any $x \in I$ the sequence $P^n(x)$ converges to p. For $x \neq p$ the sequence t_{n+1}/t_n where $t_n = |P^n(x) - p|$ limits to $P'(p)$. Analogously, $O'(o)$ is approximated by u_{n+1}/u_n where $u_n = |O^n(f(x)) - o|$ for $f(x) \neq o$.

Assume now $|P'(p)| > |O'(o)|$. This implies that there are numbers $n' \in \mathbb{N}$ and $t, u \in [0,1]$ such that for any $n > n'$ and any $x \in I$, $x \neq p$, $f(x) \neq o$

$$\frac{|t_{n+1}|}{|t_n|} > t > u > \frac{|u_{n+1}|}{|u_n|}$$

As a consequence we have:

$$|t_n| > |t_{n'}| \cdot t^{n-n'} \qquad |u_n| < |u_{n'}| \cdot u^{n-n'}$$

Take an arbitrary sequence x_n with $x_n \in I \setminus \{p\}$, $\lim_{n\to\infty} x_n = p$ for the purpose of computing the derivative of f in p. For each n the number x_n lies in a set $P^m(I) \setminus P^{m+1}(I)$ for a unique m which we will denote $\pi(n)$. Because x_n converges to p, $\pi(n) \to \infty$ for $n \to \infty$. Choose $\tilde{n}$ s.t. $(\forall n > \tilde{n})(\pi(n) > n')$. Now for $n > \tilde{n}$

$$\frac{|f(x_n) - o|}{|x_n - p|} = \frac{|f(P^{\pi(n)}(y_n)) - o|}{|P^{\pi(n)}(y_n) - p|} = \frac{|O^{\pi(n)}(f(y_n)) - o|}{|P^{\pi(n)}(y_n) - p|}$$

(where $y_n \in I \setminus P(I)$) is a sequence which can be bound from above by

$$|u_{\pi(n)}|/|t_{\pi(n)}| = (|u_{n'}|/|t_{n'}|) \cdot (u/t)^{\pi(n)-n'}$$

which limits to 0. Thus $f'(p)$ is 0.

The proof of (2) is similar to the proof of (1): it is $u > t$ and π is defined in terms of $f(x_n)$ and O instead of x_n and P. If there is an $y \neq p$ with $f(y) = o$ then for each n, $f(P^n(y)) = O^n(f(y)) = O^n(o) = o$ and the sequence

$(f(P^n(y)) - o)/(P^n(y) - p)$ is constantly 0. On the other hand if f is not constant, there is some z with $f(z) \neq o$ and $|(f(P^n(y)) - o)/(P^n(y) - p)|$ limits to ∞. In the remaining case when $f(x) = o \iff x = p$ by an analogous procedure as in the proof of (1) we conclude $|f'(p)| = \infty$. $\square$

Definition 22 *A finite transducer $\mathcal{A} = (Q, \mathcal{D}, \mathcal{D}, \tau, q_S)$ using an lft-representation $(I, \mathcal{D})$ is* balanced *if for any $P, P', O \in \mathcal{D}^*$ such that* $\text{state}_{\mathcal{A}}(P') = \text{state}_{\mathcal{A}}(P'P)$ *and* $\text{output}_{\mathcal{A}}(P'P) = \text{output}_{\mathcal{A}}(P')O$ *(a cycle within $\mathcal{A}$ in which P is input and O is output) the lft's P and O fulfil $|P'(p)| = |O'(o)|$ where p, o are the fixpoints of P, O in I, respectively.*

The importance of Lemma 21 is in showing that any automaton that computes an interesting continuously differentiable function must be balanced:

Proposition 23 *Let $\mathcal{D}$ be an lft-representation of I and $f\colon I \to I$ a function computable by a strongly connected finite transducer $\mathcal{A}$ using $\mathcal{D}$. If $\mathcal{A}$ is not balanced and f is continuously differentiable, f is a constant function.*

PROOF. Suppose that there is some unbalanced cycle in $\mathcal{A}$ and for the time being also that q_S is in that cycle. Let P_c be the input leading through the cycle and O_c the output of that cycle. (I.e. $\text{state}_{\mathcal{A}}(P_c) = q_S$, $\text{output}_{\mathcal{A}}(P_c) = O_c$, $|P_c'(p_c)| \neq |O_c'(o_c)|$ for the attracting fixpoints p_c, o_c of P_c, O_c, respectively.) By Lemma 21 $f'(p_c) \in \{0, +\infty, -\infty\}$. Since $\infty \notin I$, the only lft's mapping I to I that have such a point are the constant ones. Thus f is constant.

If q_S is not on the unbalanced cycle, it can reach a state q' on the unbalanced cycle by some input P. Denote $f_{q'}$ the function computed by $\mathcal{A}$ in state q'. We get $\text{output}_{\mathcal{A}}(P')^{-1} \circ f \circ P' = f_{q'}$. As we have shown that $f_{q'}$ is constant and any digit lft's are bijections, also f must be constant on $P'(I)$. Since f is an lft on I by Theorem 20, it is constant on the whole of I. $\square$

6 Zones in Oriented Graphs

The following technical lemmas about oriented graphs lead to a certain grading of states of a finite transducer. The lowest grade are the strongly connected components of its underlying graph. For them the results of the previous section apply. The definition of higher grades allows us to prove a weaker version of the main result for functions computed by an arbitrary finite transducer.

Definition 24 *Let $\mathcal{G} = (V, E)$ be an oriented graph. A subset $U \subseteq V$ is called*

- edge-closed *if there is no edge leading from U outside U,*
- *an* SC-zone *if it is strongly connected and edge-closed, and*

- *a* final zone *if it is accessible from each node and edge-closed.*

Note that any two different SC-zones of $\mathcal{G}$ are disjoint.

Translated to the transducer language, an edge-closed zone is such a subset of states, from which it is not possible to get out by any possible input. An SC-zone is a subset of states, to which the transducer can be restricted and thus forms a strongly connected transducer. A final zone is a set of states, that can be reached from any state by some input and is again impossible to leave.

Lemma 25 *Let $\mathcal{G} = (V, E)$ be a finite oriented graph in which each vertex has at least one outgoing edge. Then from any vertex $v \in V$ a path leads to some SC-zone (i.e. the union of all SC-zones is a final zone).*

PROOF. By induction on the number of vertices. For a 1-vertex graph there is only one SC-zone consisting of all of the graph—the lemma is trivial. Suppose it holds for any graph of less than $|V|$ elements. Take the set U of all vertices accessible from v. If from every vertex $u \in U$ there is a path back to v, v is within an SC-zone already. Otherwise take a $u \in U$ from which there is no path to v. Now consider the full subgraph of $\mathcal{G}$ with exactly those vertices accessible from u. It has fewer elements than $\mathcal{G}$ and each vertex has the same outgoing edges as in $\mathcal{G}$, so it has an SC-zone accessible from u, which is accessible from v. $\square$

Lemma 26 *Let $U \subsetneqq V$ be a final zone of a graph $\mathcal{G} = (V, E)$. Put*

$$W := \{w \in V \mid (\forall v \in V)(w \to v \implies (v \to w \text{ or } v \in U))\}$$

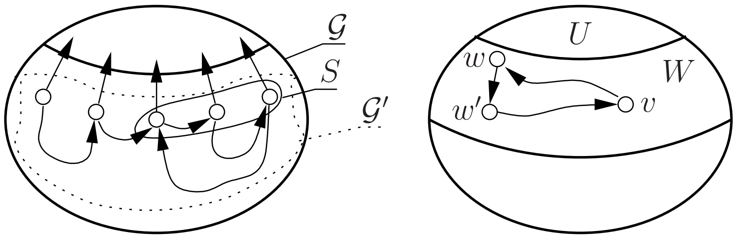
where $v \to w$, etc. means "there is a path from v to w". (Thinking of U as some generalised final SC-zone, $W \setminus U$ can be viewed as the largest kind-of SC-zone when ignoring paths leading to U.) The claim: $W \supsetneqq U$ and W is also a final zone.

PROOF. First of all $U \subseteq W$ and thus W is also accessible from each node. To show that $W \setminus U \neq \emptyset$ consider the graph

$$\begin{aligned}
&\mathcal{G}' = (V', E^*|_{V' \times V'}),\\
&V' := \{v \in V \setminus U \mid (\exists u \in U)((v, u) \in E)\},\\
&E^* = \Delta_V \cup E \cup E \circ E \cup \ldots
\end{aligned}$$

with vertices $v \in V \setminus U$ that have an edge to U and with edges corresponding to the accessibility relation in $\mathcal{G}$ (cf. figure A). V' is not empty because $U \subsetneqq V$ and U is a final zone. By Lemma 25 there exists a non-empty SC-zone S in $\mathcal{G}'$.

Let us check $S \subseteq W$. For any $v \in V \setminus U$ accessible from $s \in S \subseteq V'$ there is a vertex $v' \in V'$ accessible from v (because U is a final zone) and it must be in S because S is edge-closed in $\mathcal{G}'$. As S is an SC-zone in $\mathcal{G}'$ there is a path $v' \to s$ in $\mathcal{G}$, implying altogether $v \to s$, which is all we needed to prove $s \in W$. It



(a) $W \setminus U$ not empty (b) W is a final zone

Fig. 4.

remains to show that W is edge-closed. Take any $w \in W$ and $(w, w') \in E$. If $w' \to v$ then $w \to v$ and thus either $v \to w$ or $v \in U$ (since $w \in W$). The former implies $v \to w'$ and thus we have $w' \in W$ (cf. Figure 4(b)). $\square$

As a result of this lemma, we can divide states of a finite transducer into layers as follows:

Definition 27 *Let $\mathcal{A} = (Q, I, O, \tau, q_S)$ be a finite transducer and $\mathcal{G}_\mathcal{A}$ its underlying graph. Set*

$$SC^0_\mathcal{A} := \{q \in Q \mid q \textit{ is in some SC-zone of } \mathcal{G}_\mathcal{A}\}$$

and

$$SC^{n+1}_\mathcal{A} := \{q \in Q \mid (\forall q' \in Q)(q \to q' \implies (q' \to q \textit{ or } q' \in SC^n_\mathcal{A}))\}$$

be the construction of Lemma 26 where paths are taken in $\mathcal{G}_\mathcal{A}$.

Directly from Lemmas 25 and 26 we can observe:

Lemma 28 *Under the conditions of the previous definition all $SC^n_\mathcal{A}$ are final zones of $\mathcal{G}_\mathcal{A}$. Also there is a unique number $n_\mathcal{A} \in \mathbb{N}$ such that*

$$SC^0_\mathcal{A} \subsetneqq SC^1_\mathcal{A} \subsetneqq \cdots \subsetneqq SC^{n_\mathcal{A}}_\mathcal{A} = Q.$$

This defines a canonical decomposition of Q.

7 Arbitrary Finite Automata

To generalise to an arbitrary finite transducer we need to assume redundancy of the representation. Without redundancy we could not rule out functions composed of many lft's linked so that the derivative would be continuous even in the linking points. There is no way to rule out the possibility of these points forming a set similarly complex to the Cantor set. Although we are losing something by imposing redundancy here, we can do it with a clean conscience because as was said before, redundancy is an essential requirement for a representation to be computationally complete.

It is interesting to note at this point that Raney's result from [5] mentioned in the Introduction uses a non-redundant representation and is therefore insufficient to deal with binary functions (e.g. addition). It is not known to the author whether some redundant Möbius representation has a similar power to compute unary functions with finite memory as Raney's non-redundant representation.

Theorem 29 *Let $\mathcal{D}$ be a redundant lft-representation of $I \in \mathcal{R}$. If $f\colon I \to I$ is a function computable by a finite transducer $\mathcal{A}$ using $\mathcal{D}$ and is continuously differentiable on an open subinterval $J \subset I$, it is an lft on J.*

PROOF. To prove the theorem for $\mathcal{A}$ we prove the same for all transducers $\mathcal{A}_q$ that differ from $\mathcal{A}$ only by having initial state q, computing functions f_q. We do so by induction for states from sets $SC^0_{\mathcal{A}}, \ldots, SC^{n_{\mathcal{A}}}_{\mathcal{A}}$.

For $q \in SC^0_{\mathcal{A}}$ we have that f_q is either an lft or is not continuously differentiable on any interval of non-zero length by Theorem 20. Thus the claim is proved for such states.

Suppose that present theorem is valid for all f_q, $q \in SC^k_{\mathcal{A}}$. Take some $q' \in SC^{k+1}_{\mathcal{A}}$. From the definition of $SC^{k+1}_{\mathcal{A}}$ it follows that after any input $P \in \mathcal{D}^*$ the automaton is in a state which is either in $SC^k_{\mathcal{A}}$ or there is a path back to q'. In the former case the theorem holds on $P(I)$, in the latter case if $P(I)$ does not contain an endpoint of I there is a sub-self-similarity of $f_{q'}$. This holds for any P.

Suppose there is an open subinterval $J \subset \operatorname{Int} I$ on which $f_{q'}$ is continuously differentiable. If there is an input P s.t. for the simultaneous output O it holds $O^{-1} \circ f_{q'} \circ P = f_{q'}$ and $P(I) \subset J$ then $f_{q'}$ is continuously differentiable on the whole of I. If even ${f_{q'}}'(p) \notin \{0, \pm\infty\}$ for the fixpoint of P then we can apply Lemma 19 to conclude that $f_{q'}$ is an lft on I.

Suppose this does not happen and thus for each P with $P(I) \subseteq J$ either $f|_{P(I)}$

is an lft (if P leads to $SC^k_{\mathcal{A}}$) or $\exists x_P \in P(I)$ with $f'(x_P) \in \{0, \pm\infty\}$. If the latter happens for every P then by continuity of the derivative of f on J the derivative is constantly equal to 0 and $f(x)$ is constant on J. On the other hand if there is such a P for which $f|_{P(I)} = L|_{P(I)}$ for some lft L, we consider the following numbers:

$$u := \inf\{z \in \overline{J}, z < x \mid f|_{[z,x]} = L|_{[z,x]}\}$$
$$v := \sup\{z \in \overline{J}, z > x \mid f|_{[x,z]} = L|_{[x,z]}\}$$

where x is some number in J. We aim to prove that $J =]u, v[$ by showing that $u, v \notin J$. Suppose on the contrary that $u \in J$. Let Ω be any neighbourhood of u such that $\Omega \subseteq J$. By Lemma 4 there is $P_\Omega \in \mathcal{D}^*$ with $u \in \mathrm{Int}(P_\Omega(I)) \subset \Omega$. If f would be an lft on $P_\Omega(I)$ then it would have to be equal to L because it would coincide with L on $[u, \infty] \cap \Omega$. So there must be $x_\Omega \in \Omega$ with $f'(x_\Omega) \in \{0, \pm\infty\}$. By choosing smaller and smaller Ω we get a sequence of points x_n limiting to u with $f'(x_n) \in \{0, \pm\infty\}$ for every $n \in \mathbb{N}$. By continuity of the derivative of f on J we get that also $f'(u) = L'(u) \in \{0, \pm\infty\}$ which is impossible for an lft L with $L(J) \subseteq I$, $\infty \notin I \supset J$. So by this contradiction we have proved that $u \notin J$. To prove the same for v is analogous. $\square$

Thus we have shown that a finitely computable function must be an lft wherever it is continuously differentiable. Next we will limit the choice of coefficients these lft's can have as well as the endpoints of continuously differentiable sections of finitely computable functions.

Definition 30 *Let $\mathcal{D}$ be an lft-representation of I. Denote by*

$$\mathrm{Fix}_{\mathcal{D}} := \{P_2(p) \mid P_1, P_2 \in \mathcal{D}^* \;\&\; p = P_1(p) \in I\}$$

the set of numbers finitely computable in $\mathcal{D}$ and by $\mathbb{Q}_{\mathcal{D}}$ the subfield of $\mathbb{R}$ generated by both $\mathrm{Fix}_{\mathcal{D}}$ and the coefficients of lft's in $\mathcal{D}$.

Theorem 31 *Let $\mathcal{D}$ be a redundant lft-representation of I and $f\colon I \to I$ a function finitely computable using $\mathcal{D}$. If f is continuously differentiable on an open interval $]u, v[\subseteq I$ but not on any interval $K \supsetneq]u, v[$, then*

(1) $u, v, f(u), f(v) \in \mathrm{Fix}_{\mathcal{D}}$
(2) $f|_{]u,v[}$ is an lft whose coefficients can be chosen from $\mathbb{Q}_{\mathcal{D}}$.

PROOF. Fix some finite transducer $\mathcal{A}$ that computes f using representation $\mathcal{D}$.

Let us start by proving $u \in \mathrm{Fix}_{\mathcal{D}}$. Let $\{D_{s_i}\}_{i=1}^{\infty} \in \mathcal{D}^\omega$ be a sequence of digits representing u by a sequence of intervals that contain u in its interior. For certain index m it holds $D_{s_1} \cdots D_{s_m}(I) \not\supset [u, v]$. The same holds for any $i > m$.

Denote by q_i the state of $\mathcal{A}$ in which the machine is after receiving the first i digits $D_{s_1} \dots D_{s_i}$. There must exist 3 different indices $j_1, j_2, j_3 \geq m$ such that $q_{j_1} = q_{j_2} = q_{j_3}$. Out of these 3 indices there must be two $j < k$ such that lft's $D_{s_1} \cdots D_{s_j}$ and $D_{s_1} \cdots D_{s_k}$ are both increasing or both decreasing. This means that $P := D_{s_{j+1}} \cdots D_{s_k}$ is increasing. Let us further denote $P_1 := D_{s_1} \cdots D_{s_j}$ and $O_1 := \mathrm{output}_{\mathcal{A}}(P_1)$, $O := O_1^{-1}\,\mathrm{output}_{\mathcal{A}}(P_1 P)$ be the outputs that are made during inputs P_1 and P, respectively.

By Lemma 9 it holds $f_{q_j} = O_1^{-1} \circ f \circ P_1$ and also $f_{q_j} = f_{q_k} = O^{-1} \circ f_{q_j} \circ P$. The latter equation is a sub-self-similarity. Denote the endpoints of I as follows: $I = [g, h]$. The function f_{q_j} is continuously differentiable on one of the intervals $[g, u'[$ or $]u', h]$ where $u' := P_1^{-1}(u)$ (depending whether P_1 is increasing or decreasing) because $j > m$, and it is not continuously differentiable in u' itself. Our aim is to prove that u' is the fixpoint of P in I.

Without loss of generality suppose that f_{q_j} is continuously differentiable on $[g, u'[$. By the above sub-self-similarity f_{q_j} is also continuously differentiable on $P([g, u'[) = [P(g), P(u')[$ (recall that P is increasing). Now we also have $u' \in \operatorname{Int} P(I) =]P(g), P(h)[$ because $P_1 P$ is an approximation of u. This means that $[P(g), P(u')[$ and $[g, u'[$ have an non-empty intersection, so f_{q_j} is cont. differentiable on their union $[g, P(u')[$ which means that $[P(g), P(u')] \subseteq [g, u']$. Thus by Brower's fixpoint theorem there is a fixpoint p of P in $[g, u']$. If this fixpoint would be in $[g, u'[$ where f_{q_j} is continuously differentiable and also (by Theorem 29) an lft whose derivative is not in $\{0, \pm\infty\}$ on I, we could apply Lemma 19 to find out that f_{q_j} would be an lft on the whole of I which is a contradiction. Thus $p = u'$. Altogether $u = P_1(p)$ where $p = P(p)$, i.e. $u \in \mathrm{Fix}_{\mathcal{D}}$. Note that this means that u can be also represented by the periodic sequence $P_1 PPP \dots$.

Proof of $v \in \mathrm{Fix}_{\mathcal{D}}$ is analogous. That $f(u), f(v) \in \mathrm{Fix}_{\mathcal{D}}$ follows from elementary finite transducer theory [8] as any finite transducer outputs a periodic sequence whenever there is a periodic sequence on its input. This concludes the proof of (1).

Let us start proving (2) for a special case when $\mathcal{A}$ is strongly connected and therefore $[u, v] = I$ and f is an lft. Let $P, \overline{P} \in \mathcal{D}^*$ be some inputs leading back to the initial state such that their fixpoints $p, \overline{p} \in I$, respectively, are different. Similarly let $o, \overline{o} \in I$ be the fixpoints of the simultaneous outputs $O, \overline{O}$ of $\mathcal{A}$ on inputs $P, \overline{P}$, respectively. In the proof of Lemma 19 (3) it comes up that $f'(p) = T_O(f(x))/T_P(x)$ for any $x \in I$, $x \neq p$. (Notice that $f'(p) \notin \{0, \pm\infty\}$ because f is an lft and $\infty \notin I$.) Again by Lemma 19 we get $f(\overline{p}) = \overline{o}$, so altogether $f'(p) = T_O(\overline{o})/T_P(\overline{p}) \in \mathbb{Q}_{\mathcal{D}}$ and

$$f = T_O^{-1} \begin{pmatrix} \pm f'(p) & 0 \\ 0 & 1 \end{pmatrix} T_P$$

is an lft with coefficients from $\mathbb{Q}_{\mathcal{D}}$.

Now consider the general case when $\mathcal{A}$ does not need to be strongly connected. Select an input P_1 with $P_1(I) \subset]i,j[$. There must be a further input P_2 so that $q := \mathrm{state}_{\mathcal{A}}(P_1P_2) \in SC^0_{\mathcal{A}}$ by Lemma 25. In the previous part we proved that f_q is an lft with coefficients from $\mathbb{Q}_{\mathcal{D}}$. Denoting, as usual, O the simultaneous output of $\mathcal{A}$ on input $P := P_1P_2$, we get by Lemma 9:

$$f|_{]i,j[} = O f_q P^{-1}|_{]i,j[}$$

which is an lft with coefficients from $\mathbb{Q}_{\mathcal{D}}$. $\square$

8 The Standard Digit Systems

All digits used in the standard b-ary representations are affine lft's. This means that they all have fixpoint ∞. The following proposition shows that such systems with digits sharing a common fixpoint can compute only lft's with the same fixpoint or the constant lft's. It is a generalisation of Reinhold Heckmann's result [13] about the standard digit system translated to the interval $[0,\infty]$ considered by the group in Imperial College (see e.g. [2,1]). The translation from $[-1,1]$ to $[0,\infty]$ used there translates also fixpoints, and thus all digits in the system he considers have a fixpoint -1. Using an equivalent characterisation of the property "having fixpoint -1" he shows that regular lft's finitely computable in that system have fixpoint -1 as well.

Proposition 32 *Let $\mathcal{D}$ be an lft-representation of I and $\xi \in \mathbb{R}^*$, $\xi \notin I$ such that each $D \in \mathcal{D}$ has a fixpoint $\xi = D(\xi)$. If f is a function computed by a finite transducer over $\mathcal{D}$ and $f|_J = L|_J$ for some interval $J \subseteq I$ of non-zero length and a regular lft L, then $\xi = L(\xi)$.*

PROOF. Let us first prove the claim only for f computed by a strongly connected finite transducer $\mathcal{A}$ using $\mathcal{D}$. Take some input P that leads back to the initial state while producing output O. Because L is regular and $\infty \notin I$, it holds $L'(p) \notin \{0, \pm\infty\}$ and we can apply Lemma 19 (3) to get

$$f = T_O^{-1} \begin{pmatrix} \pm f'(p) & 0 \\ 0 & 1 \end{pmatrix} T_P.$$

The definition of T_A for $A \in \mathrm{LFT}(I)$ with fixpoints $u \in I$, $v \notin I$ is such that $T_A(v) = \infty$. Thus $T_P(\xi) = \infty$ and $T_O(\xi) = \infty$. Using the above equation we get $f(\xi) = \xi$.

To prove the general case, take some input $P_1 \in \mathcal{D}^*$ with $P_1(I) \subset J$ and a further input $P_2 \in \mathcal{D}^*$ such that P_1P_2 leads to a state q in an SC-zone, while producing O_1O_2. By Lemma 9 and previous proof that $f_q(\xi) = \xi$,

$$f = O_1O_2f_qP_2^{-1}P_1^{-1}$$

is again a function with $f(\xi) = \xi$. $\square$

Lemma 33 *Let $\mathcal{D}_b$ be the standard signed-digit b-ary representation of $[-1, 1]$. Then $\mathrm{Fix}_{\mathcal{D}_b} = [-1, 1] \cap \mathbb{Q}$ and $\mathbb{Q}_{\mathcal{D}_b} = \mathbb{Q}$.*

PROOF. Recall that the digits in $\mathcal{D}_b$ are the lft's $(x + r)/b$ for $r = -(b - 1), \ldots, b - 1$. Concatenation of n such digits are exactly lft's of the form $(x + r)/b^n$, $r \in \{-(b^n-1), \ldots, b^n-1\}$. (The choice of the digits corresponds to some b-ary representation of r.) The fixpoint of such an lft is equal to $r/(b^n - 1)$. Thus

$$\begin{aligned}
&\mathrm{Fix}_{\mathcal{D}_b} = \\
&\quad \left\{ \frac{\frac{r_1}{b^{n_1}-1} + r_2}{b^{n_2}} \mid n_i \in \mathbb{N}, r_i \in \{-(b^{n_i} - 1), \ldots, b^{n_i} - 1\}, \text{ for } i = 1, 2 \right\} = \\
&\quad \left\{ \frac{r}{b^{n_2}(b^{n_1} - 1)} \mid n_1, n_2 \in \mathbb{N}, r \in \{-b^{n_2}(b^{n_1} - 1), \ldots, b^{n_2}(b^{n_1} - 1)\} \right\} = \\
&\qquad \mathbb{Q} \cap [-1, 1].
\end{aligned}$$

The last equation is true because any number m with $\mathrm{GCD}(m, b) = 1$ divides $b^x - 1$ where x is a solution of $b^x \equiv 1 \pmod{m}$ (which exists thanks to the pigeon-hole principle: there must exist a $y < z$ such that $b^y \equiv b^z \pmod{m}$ and $x := z - y$ is a solution) and thus for any integer q we have n_1, n_2 such that q divides $b^{n_2}(b^{n_1} - 1)$.

It is now obvious that $\mathbb{Q}_{\mathcal{D}_b}$ which is generated by $\mathrm{Fix}_{\mathcal{D}_b}$ and the coefficients of $\mathcal{D}_b$ is $\mathbb{Q}$, the smallest field, itself. $\square$

Theorem 34 *For any function $f\colon [-1, 1] \to [-1, 1]$ which is finitely computable using $\mathcal{D}_b$ and continuously differentiable on $I \setminus \{s_1, \ldots, s_n\}$ for some $-1 = s_1 < \cdots < s_n = 1$ it holds:*

(1) $s_1, \ldots, s_n, f(s_1), \ldots, f(s_n) \in [-1, 1] \cap \mathbb{Q}$
(2) f *is affine on every* $[s_i, s_{i+1}]$.

Conversely, any such function f is finitely computable using $\mathcal{D}_b$.

PROOF. The first part of the theorem is a corollary to Theorems 29 and 31, Proposition 32 and Lemma 33 when we realise that affine regular lft's are

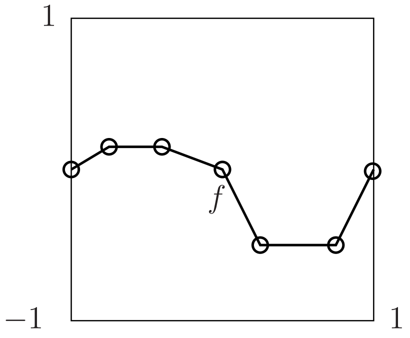


Fig. 5. A piecewise affine function

exactly those lft's with fixpoint ∞.

We need to construct a finite transducer over $\mathcal{D}_b$ that computes any f of the form stated in the theorem. We can identify the states of the automaton-to-be with the functions computed from the states. Thus, we find a finite set of functions F and a transition and output function τ such that $\tau_1(f, D) = \tau_2(f, D)^{-1} \circ f \circ D$. By Lemma 10 such a structure is a finite transducer computing the functions in its states.

First of all, we find such an n that $f \circ P$ for any $P \in \mathcal{D}^n$ has at most one point without derivative. It is possible because $|P([-1,1])| = 2^{-n+1}$ limits to 0 for $n \to \infty$. While inputting the first n digits, the automaton will not produce any output. Now it remains to find the states and transition functions for functions of the form: $f(x) = (a_1 x + c_1)/d$ for $x \in [-1, u/d]$ and $f(x) = (a_2 x + c_2)/d$ for $x \in [u/d, 1]$ where $a_1, c_1, a_2, c_2 \in \mathbb{Z}$, $u, d \in \mathbb{N}$ with

$$(-a_1 + c_1)/d, (a_1(u/d) + c_1)/d = (-a_2(u/d) + c_2)/d, (a_2 + c_2)/d \in [-1, 1].$$

Furthermore, we can assume by taking sufficiently large n that $|a_1/d|, |a_2/d| \leq 1/2$. (Each input of a digit divides the coefficient of x by b.)

Consider the following finite set:

$$T := \Big\{ f_{v,w}(x) \Big| v, w \in \mathbb{Z}, |v| \leq d, |w| \leq d^2, \\ |a_1(-1 - v/d) + w| \leq d, |a_2(1 - v/d) + w| \leq d \Big\}$$

where

$$f_{v,w}(x) = \begin{cases} (a_1(x - v/d) + w/d)/d & \text{for } x \in [-1, v/d] \\ (a_2(x - v/d) + w/d)/d & \text{for } x \in [v/d, 1] \end{cases}$$

are endofunctions on $[-1, 1]$. The set T contains $f = f_{u, a_1 u + c_1 d}$.

On inputting a digit $(x+r)/b$ in state $f_{v,w}$ we want to output one digit $(x+s)/b$, so that $g := (\lambda x.(x+s)/b)^{-1} \circ f_{v,w} \circ (\lambda x.(x+r)/b)$ is equal to some $f_{v',w'}$ in T. This equation can be visualised by the picture in Figure 6 where the outer

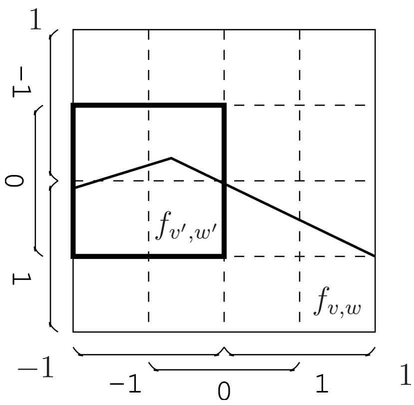


Fig. 6.

square contains the graph of $f_{v,w}$ and the input and output binary digits (here `-1` and `0`) cut a smaller square out of the big square in such a way that the section of the graph of $f_{v,w}$ clipped to this smaller square is a graph of some $f_{v',w'}$ within this smaller square.

For any basis b in the sequence of squares corresponding to the input digit r and all possible output digits, the neighbouring squares overlap by their halves. Therefore the conditions $|a_1|, |a_2| \leq d/2$ guarantee that there is at least one square among them that contains the whole of the graph of $f_{v,w}$ restricted to the interval $[(r-1)/b, (r+1)/b]$. We choose the output digit s corresponding to such a square. It remains to check that $g = f_{v',w'}$ for some v', w'. We don't need to check the inequality constraints on v', w' any more.

If $v/d \notin \mathrm{Int}[(r-1)/b, (r+1)/b]$, it is either

$$\begin{aligned} f'(x) = {} & \frac{b(a_1((x+r)/b - v/d) + w/d)}{d} - s = \\ & \frac{a_1 x + a_1 r - a_1 vb/d + wb/d - sd}{d} = \\ & \frac{a_1(x-1) + (a_1(rd+d-vb) + wb - sd^2)/d}{d} = f_{d, a_1(rd+d-vb)+wb-sd^2} \end{aligned}$$

or

$$\begin{aligned} f'(x) = {} & \frac{a_2 x + a_2 r - a_2 vb/d + wb - sd}{d} = \\ & \frac{a_2(x+1) + (a_2(rd-d-vb) + wb - sd^2)/d}{d} = f_{-d, a_2(rd-d-vb)+wb-sd^2}, \end{aligned}$$

thus being a member of T. If $v/d \in \mathrm{Int}[(r-1)/b, (r+1)/b]$, we set $v' = bv - rd$ and using similar computation as above, we get

$$f'(x) = \frac{a_1(x - v'/d) + (wb - sd^2)/d}{d} \quad \text{for } x \in [-1, v'/d]$$

and

$$f'(x) = \frac{a_2(x - v'/d) + (wb - sd^2)/d}{d} \quad \text{for } x \in [v'/d, 1],$$

so $f' = f_{v',wb-sd^2} \in T$. $\square$

We cannot compute affine functions on the whole of $\mathbb{R}$ using affine digits because in an interval that contains ∞ there can be at most one affine digit. But on intervals that do not contain ∞ we can compute by translated version of $\mathcal{D}_b$ to obtain an analogous theorem for that interval.

9 Concluding Remarks

It is not clear to the author whether there is an lft-representation which would allow one to finitely compute all lft's with coefficients from a dense subset of $\mathbb{R}$ [1]. But it is clear that the set of finitely computable functions is considerably limited.

Another by-product of this research which is not included for lack of space and time is an easy criterion to recognise whether an arbitrary strongly connected transducer using some lft-representation $\mathcal{D}$ computes an lft or not. It might lead to a further finer study on finite computability.

The results would not hold without the condition of continuous differentiability or a similar smoothness condition. In the author's PhD thesis [14, Subsection 4.2.3] there is a 10 state transducer over the signed binary representation which computes a strictly increasing function $f_w \colon [-1, 1] \to [-1, 0]$ such that there are 2 sets $A, B \subset [-1, 1]$ dense in $[-1, 1]$ for which it holds: $a \in A \implies f'_w(a) = 0$, $b \in B \implies f'_w(b) = \infty$.

10 Acknowledgements

This research was carried out sponsored by an ORS award of the Committee of Vice-Chancellors and Principals of the Universities of the United Kingdom and by the School of Computer Science, University of Birmingham.

The author is deeply indebted to his supervisor Prof. Achim Jung for inspiring and supporting the research and giving much valuable feedback.

[1] The mentioned result of Raney [5] gives some hope towards a possible existence of such a representation.

References

[1] P. J. Potts, Computable real arithmetic using linear fractional transformations, Ph.D. thesis, Imperial College, Department of Computing (July 1998).

[2] A. Edalat, P. J. Potts, A new representation for exact real numbers, in: S. Brookes, M. Mislove (Eds.), Electronic Notes in Theoretical Computer Science, Vol. 6, Elsevier Science Publishers, 2000.

[3] A. M. Nielsen, P. Kornerup, MSB-first digit serial arithmetic, Journal of Universal Computer Science 1 (7) (1995) 527–547.

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

[5] G. N. Raney, On continued fraction and finite automata, Mathematische Annalen 206 (1973) 265–283.

[6] A. Edalat, P. Sünderhauf, A domain theoretic approach to computability on the real line, Theoretical Computer Science 210 (1999) 73–98.

[7] C. Mazenc, On the redundancy of real number representation systems, Research Report 93-16, LIP, Ecole Normale Supérieure de Lyon (1993).

[8] B. A. Trakhtenbrot, Y. M. Barzdin, Finite Automata—Behaviour and Synthesis, Vol. 1 of Fundamental Studies in Computer Science, North Holland, Amsterdam-London, 1973.

[9] W. Rudin, Real and Complex Analysis, 3rd Edition, McGraw-Hill, New York, NY, 1986.

[10] R. Heckmann, Contractivity of linear fractional transformations, in: J.-M. Chesneaux, F. Jézéquel, J.-L. Lamotte, J. Vignes (Eds.), Third Real Numbers and Computers Conference, Université Pierre et Marie Curie, Paris, 1998, pp. 45–59, paris, France, April 27-29, 1998.

[11] R. W. Gosper, Continued fraction arithmetic, Technical Report HAKMEM Item 101B, MIT AI MEMO 239, MIT, on 12th June, 2000, available from: `ftp://ftp.netcom.com/pub/hb/hbaker/hakmem` (February 1972).

[12] P. Potts, A. Edalat, M. Escardó, Semantics of exact real number computation, in: Proceedings of the Twelveth Annual IEEE Symposium on Logic In Computer Science, Warsaw, Poland, 1997.

[13] R. Heckmann, The appearance of big integers in exact real arithmetic based on linear fractional transformations, in: M. Nivat (Ed.), Foundations of Software Science and Computation Structures, Vol. 1378 of Lecture Notes in Computer Science, Springer Verlag, 1998, pp. 172–188.

[14] M. Konečný, Many-valued real functions computable by finite transducers using IFS-representations, Ph.D. thesis, School of Computer Science, The University of Birmingham (October 2000).